Numerical Linear Algebra WS 26/27 Notes

Igor Dimitrov

2026-10-21

Table of contents

Preface **3**

1 Overview **4**
1.1 Numerical Linear Algebra Overview 4
1.1.1 Main computational problems 4
1.1.2 The special importance of large and sparse matrices 4
1.2 Course Overview 4
1.2.1 The five main chapters 6
1.2.2 Relationship Between Chapters 6
1.3 Linear Algebra and Numerics Prerequisites and Review 6
1.3.1 Essential linear algebra topics 6
1.3.2 Prerequisites from Numerics 0 10
1.3.3 The connection to the Chebyshev-filtered subspace iteration seminar . . 12
1.4 Revision Strategy 14
1.4.1 Phase 1 - linear algebra foundation 14
1.4.2 Phase 2 - numerical-analysis foundation 14
1.4.3 Phase 3 - Eigenvalues 14
1.4.4 Targeted Revision Plan 15
1.5 Structural Linear Algebra Review 18
1.5.1 What mastery should mean 19

References **20**

I Review I **21**

2 Session 1 — Vector Spaces and Linear Maps **22**
2.1 Subspaces and span 22
2.1.1 Subspaces 22
2.1.2 Span 23
2.1.3 Worked example: generators and constraints 23
2.2 Independence, bases, and dimension 23
2.2.1 Linear independence 24
2.2.2 Bases and coordinates 24
2.2.3 Dimension 25

2.2.4 A basis stored as a matrix . . . 26
2.3 Linear maps and their matrix representations . . . 26
2.3.1 Constructing the representing matrix . . . 26
2.3.2 Worked example: differentiation . . . 27
2.4 Kernel, image, and solvability . . . 27
2.4.1 Solvability and all solutions . . . 28
2.4.2 Injectivity and surjectivity . . . 28
2.4.3 Worked example: a complete rectangular system . . . 29
2.4.4 Why use the original pivot columns? . . . 30
2.5 The rank–nullity theorem . . . 31
2.5.1 Proof by extending a kernel basis . . . 31
2.5.2 Interpretation and consequences . . . 32
2.6 Change of basis . . . 32
2.6.1 Changing a map's matrix . . . 33
2.6.2 Worked example: coordinates reveal a simpler action . . . 33
2.7 Connections to numerical linear algebra . . . 34
2.8 Independent exercises . . . 35
2.8.1 Exercise 1 — Subspaces and coordinates . . . 36
2.8.2 Exercise 2 — A rectangular system . . . 36
2.8.3 Exercise 3 — Changing coordinates . . . 36
2.8.4 Exercise 4 — Losing information in two stages . . . 36
2.8.5 Exercise 5 — A map on polynomials . . . 37
2.9 Solutions — consult after attempting the exercises . . . 37
2.9.1 Solution 1 . . . 37
2.9.2 Solution 2 . . . 38
2.9.3 Solution 3 . . . 39
2.9.4 Solution 4 . . . 39
2.9.5 Solution 5 . . . 40
2.10 End-of-session mastery checklist . . . 41

3 Session 2 — Inner Products and Orthogonality 42
3.1 From coordinates to geometry . . . 42
3.1.1 One subtraction gives Cauchy–Schwarz . . . 43
3.2 Orthogonal complements turn span questions into kernel questions . . . 43
3.3 Orthonormal coordinates and orthogonal decomposition . . . 44
3.4 Gram–Schmidt from the residual equation . . . 45
3.4.1 Worked construction in the running plane . . . 46
3.5 What changes in numerical algorithms? . . . 47
3.6 Independent exercises . . . 48
3.7 Worked solutions . . . 48
3.7.1 Choosing a geometry . . . 48
3.7.2 Recovering a normal . . . 49
3.7.3 Decomposing a vector . . . 49

3.7.4 Orthogonalization by hand . . . 49
3.7.5 Complex coordinates . . . 49
3.7.6 A filtered block . . . 49
3.8 End-of-session mastery checklist . . . 50

4 Session 3 — Orthogonal Projections 51
4.1 The component selected by the inner product . . . 51
4.2 Projection onto a line . . . 52
4.3 From orthonormal coordinates to a projection matrix . . . 52
4.3.1 The plane from Sessions 1 and 2 . . . 53
4.4 Which matrices are orthogonal projectors? . . . 53
4.5 When the supplied basis is not orthonormal . . . 55
4.6 Why the projection solves least squares . . . 56
4.7 From one projection to projection methods . . . 57
4.8 Independent exercises . . . 58
4.9 Worked solutions . . . 58
4.9.1 A complex line . . . 58
4.9.2 A projector from two columns . . . 58
4.9.3 One image, many projectors . . . 59
4.9.4 A change of subspace coordinates . . . 59
4.9.5 A fit and its uniqueness . . . 59
4.9.6 Projecting an operator action . . . 60
4.10 End-of-session mastery checklist . . . 60

5 Session 4 — QR Factorization 61
5.1 One space, two coordinate systems . . . 61
5.2 Gram–Schmidt records its work in R . . . 62
5.2.1 Worked example — a plane in three dimensions . . . 62
5.2.2 Classical and modified Gram–Schmidt . . . 63
5.3 Householder QR by changing the ambient coordinates . . . 64
5.3.1 Deriving a reflector . . . 64
5.3.2 Worked example — two eliminations . . . 65
5.4 What should a numerical QR test measure? . . . 65
5.4.1 Experiment — making columns approach a common direction . . . 66
5.5 Least squares reduces to a triangular system . . . 68
5.5.1 Worked example — fitting a point to the plane . . . 69
5.5.2 Why avoid explicitly forming the normal equations? . . . 69
5.6 From QR to the seminar . . . 70
5.7 Independent exercises . . . 70
5.8 Solutions — consult after attempting the exercises . . . 71
5.9 End-of-session mastery checklist . . . 72

6 Session 5 — Eigenvalues and the Spectral Theorem **73**
6.1 Invariant lines and the shifted kernel . . . 73
6.2 Counting roots and counting independent directions . . . 74
6.3 Diagonalization as a change of coordinates . . . 75
6.4 Why symmetry gives orthonormal eigencoordinates . . . 75
6.5 Spectral projectors and matrix powers . . . 76
6.6 Positive definiteness and the geometry of energy . . . 77
6.7 Numerical experiment . . . 79
6.8 What the seminar algorithms exploit . . . 79
6.9 Independent exercises . . . 80
6.10 Worked solutions . . . 81
6.11 End-of-session mastery checklist . . . 82

7 Session 6 — Matrix Norms and Conditioning **83**
7.1 Measuring amplification . . . 83
7.1.1 From vector size to operator size . . . 83
7.2 One application versus repeated applications . . . 85
7.3 Conditioning a linear solve . . . 85
7.3.1 What changes when the data change? . . . 85
7.3.2 Stretching factors and inversion . . . 86
7.3.3 If the matrix changes too . . . 86
7.4 The error equation . . . 87
7.4.1 One residual, two directions . . . 87
7.4.2 Experiment in two dimensions . . . 88
7.5 Backward error and stability . . . 89
7.6 Central Synthesis — Conditioning, Backward Error, and Forward Error . . . 90
7.6.1 Connections to the lecture and seminar . . . 91
7.7 Independent exercises . . . 92
7.8 Worked solutions . . . 92
7.9 End-of-session mastery checklist . . . 93

8 Session 7 — Direct Numerical Methods **95**
8.1 Elimination changes the equations . . . 95
8.2 Triangular systems have a natural order . . . 96
8.3 LU remembers the elimination . . . 96
8.4 Pivot choice and element growth . . . 97
8.4.1 Experiment: hold conditioning nearly fixed and change the pivot . . . 98
8.5 Cholesky uses positive definiteness . . . 100
8.6 Choosing and assessing a solver . . . 101
8.7 Where the factors reappear . . . 103
8.8 Central Synthesis — From Elimination to Reliable Numerical Solvers . . . 103
8.9 Exercises . . . 104

8.10 Worked solutions . . . 105
8.10.1 End-of-session mastery checklist . . . 106

9 Session 8 — Basic Iterative Methods 107
9.1 From a linear system to a fixed point . . . 107
9.1.1 Build the update by choosing an easy solve . . . 107
9.2 Two choices of splitting, one system . . . 108
9.2.1 Jacobi uses a snapshot . . . 108
9.2.2 Gauss–Seidel reuses a corrected component . . . 108
9.2.3 Worked example — when does the second unknown first change? . . . 109
9.3 The error obeys a homogeneous equation . . . 110
9.4 When do all matrix powers decay? . . . 110
9.5 Two guarantees that avoid computing eigenvalues . . . 111
9.6 What can we measure without the solution? . . . 112
9.7 Computational experiment — predict the slopes . . . 113
9.8 From sparse solves to spectral filtering . . . 116
9.9 Central Synthesis — Iterative Solvers as Spectral Error Dynamics . . . 117
9.10 Exercises . . . 117
9.11 Worked solutions . . . 118
9.12 End-of-session mastery checklist . . . 119

10 Session 9 — Power and Subspace Iteration 121
10.1 From a linear solve to a spectral question . . . 121
10.2 What survives repeated multiplication? . . . 121
10.2.1 Normalized power iteration . . . 122
10.2.2 The direction error and its rate . . . 122
10.2.3 Worked example 1 — rotate into eigencoordinates . . . 123
10.2.4 Worked example 2 — where selection fails . . . 123
10.3 Estimating and checking an eigenpair . . . 124
10.4 Which object should a block approximate? . . . 124
10.4.1 Why separate column normalizations are insufficient . . . 124
10.4.2 Orthogonal subspace iteration . . . 125
10.4.3 Worked example 3 — a plane moves, QR represents it . . . 125
10.5 The block convergence estimate . . . 126
10.6 How can we tell whether a computed plane is good? . . . 126
10.7 Experiment — one matrix, three iterations . . . 127
10.8 Central Synthesis — From Spectral Amplification to Invariant Subspace Computation . . . 130
10.9 Exercises . . . 131
10.10 Worked solutions . . . 131
10.11 End-of-session mastery checklist . . . 132

11 Session 10 — Matrix Polynomials and Chebyshev Filtering **134**
11.1 From powers to designed spectral weights . . . 134
11.2 What makes a filter effective? . . . 135
11.3 A polynomial bounded on an entire interval . . . 136
11.3.1 Why this polynomial? A normalized minimax problem . . . 136
11.4 From spectral bounds to a filter . . . 137
11.5 Applying the filter with matrix products . . . 139
11.5.1 What changes in floating point? . . . 139
11.6 How much does a filtered sweep improve a subspace? . . . 140
11.6.1 Experiment: a prediction at equal work . . . 140
11.7 How the seminar algorithm fits together . . . 143
11.8 Central Synthesis — From Polynomial Approximation to Spectral Subspace Selection . . . 144
11.9 Independent practice (30 minutes, including checks) . . . 144
11.9.1 Solution checks . . . 144

12 Session 11 — Rayleigh–Ritz and Integration **146**
12.1 What equation can a trial space satisfy? . . . 146
12.1.1 What has been projected? . . . 147
12.1.2 A variational interpretation . . . 147
12.2 How much of the action escapes the space? . . . 148
12.2.1 Worked example 1 — Extracting two directions from the Session 9 plane 148
12.2.2 Worked example 2 — Tilting an exact invariant plane . . . 149
12.3 What does a small residual certify? . . . 149
12.3.1 Worked example 3 — A very small residual at a fixed angle . . . 150
12.4 A complete filtered eigensolver . . . 150
12.4.1 Decisions that affect the result . . . 151
12.5 Connections to the lecture and seminar . . . 152
12.5.1 Optional extension — The mass matrix changes the geometry . . . 152
12.6 Central Synthesis — From Linear Algebra Foundations to a Complete Filtered Eigensolver . . . 152
12.7 Optional computational laboratory — Follow a plane through extraction . . . 153
12.8 Independent exercises (28 minutes) . . . 156
12.9 Worked solutions . . . 158
12.10Final mastery checklist . . . 159
12.11Final revision-program assessment . . . 159

II Review II **160**

13 Module A — Structural Linear Algebra **161**
13.1 Introduction: The central idea . . . 161
13.2 The six perspectives . . . 162

13.3 Roadmap . . . 162
13.3.1 Sessions . . . 163
13.4 Session 1 — The structure of a linear transformation . . . 163
13.4.1 From matrices to linear maps . . . 163
13.4.2 The kernel: indistinguishable inputs . . . 164
13.4.3 The image: attainable outputs . . . 166
13.4.4 Rank–nullity . . . 167
13.5 Session 2 — RREF as a method for discovering structure . . . 169
13.5.1 Running example . . . 169
13.5.2 Why row reduction preserves the null space . . . 170
13.5.3 Constructing a null-space basis . . . 170
13.5.4 The free-variable parameterization is an isomorphism . . . 171
13.5.5 The general RREF structure . . . 172
13.5.6 Column dependencies and the image . . . 174
13.6 Session 3 — Subspace sums, direct sums, and complements . . . 175
13.6.1 The sum of two subspaces . . . 175
13.6.2 When is a sum direct? . . . 176
13.6.3 Internal and external direct sums . . . 177
13.6.4 The dimension formula for subspace sums . . . 178
13.6.5 Complements . . . 179
13.6.6 A complement of the kernel . . . 180
13.6.7 Our running matrix: an explicit decomposition . . . 181
13.7 Session 4 — Quotient spaces: identifying vectors modulo a subspace . . . 182
13.7.1 Defining the equivalence relation . . . 183
13.7.2 Geometric example . . . 184
13.7.3 Making the quotient into a vector space . . . 184
13.7.4 The canonical quotient map . . . 186
13.7.5 Dimension of a quotient space . . . 186
13.7.6 Quotients and complements . . . 188
13.8 Session 5 — The first isomorphism theorem . . . 189
13.8.1 Motivation: turning a noninjective map into an isomorphism . . . 189
13.8.2 Statement . . . 190
13.8.3 Proof . . . 190
13.8.4 The canonical factorization of a linear map . . . 192
13.8.5 Rank–nullity follows from the isomorphism theorem . . . 193
13.8.6 The theorem applied to our running matrix . . . 194
13.8.7 Connection to nonuniqueness of linear systems . . . 195
13.9 Session 6 — Universal properties, exact sequences, and projections . . . 196
13.9.1 What is a universal property? . . . 196
13.9.2 The universal property of the quotient . . . 196
13.9.3 Direct sums as universal constructions . . . 198
13.9.4 Exact sequences . . . 199
13.9.5 Why exact sequences matter beyond vector spaces . . . 200

13.9.6 Projections correspond to direct-sum decompositions . . . 201
13.9.7 Our running example: constructing a projection . . . 203
13.10 Session 7 — Applications: when the abstract structure becomes useful . . . 204
13.10.1 Application I: rank-deficient linear systems . . . 204
13.10.2 Application II: graphs, flows, and discrete differential operators . . . 207
13.10.3 Application III: differential operators and function spaces . . . 212
13.11 Session 8 — Invariant subspaces and the bridge to numerical eigenvalue methods 216
13.11.1 Eigenspaces and invariant subspaces . . . 216
13.11.2 Invariant subspaces and block matrices . . . 217
13.11.3 A symmetric matrix example . . . 218
13.11.4 The spectral theorem: a preview . . . 219
13.11.5 Polynomial functions of matrices . . . 220
13.11.6 Connection to subspace iteration . . . 221
13.11.7 The Rayleigh–Ritz connection . . . 222
13.12 The conceptual synthesis of Module A . . . 224
13.13 Exercises — Test your understanding . . . 225
13.13.1 Part I — Linear algebraic foundations . . . 225
13.13.2 Part II — Quotients and structural theorems . . . 226
13.13.3 Part III — Applications . . . 227
13.14 Worked solutions . . . 229
13.14.1 Solution 1 . . . 229
13.14.2 Solution 2 . . . 230
13.14.3 Solution 3 . . . 231
13.14.4 Solution 4 . . . 231
13.14.5 Solution 5 . . . 232
13.14.6 Solution 6 . . . 232
13.14.7 Solution 7 . . . 233
13.14.8 Solution 8 . . . 234
13.14.9 Solution 9 . . . 235
13.14.10 Solution 10 . . . 236
13.15 Computational laboratory — MATLAB . . . 237
13.15.1 Questions to investigate experimentally . . . 239
13.16 Final mastery assessment . . . 239
13.16.1 How I recommend studying the module . . . 240
13.16.2 Transition to Module B . . . 241

14 Module B — Duality and Linear Measurements 242
14.1 Introduction — A new perspective on linear algebra . . . 242
14.2 How Module B fits into the curriculum . . . 243
14.3 Six dimensions of understanding . . . 244
14.4 Study roadmap . . . 244
14.5 Session 1 — Linear functionals and the geometry of dual spaces . . . 245
14.5.1 What is a linear functional? . . . 245

14.5.2 Why do linear functionals deserve their own space? . . . 246
14.5.3 The natural pairing . . . 247
14.5.4 Geometric meaning: a covector as a family of parallel level sets . . . 248
14.5.5 Is a covector just an ordinary vector? . . . 250
14.5.6 Why the evaluation pairing is nondegenerate . . . 251
14.6 Session 2 — Dual bases and coordinate representations . . . 252
14.6.1 Every basis defines coordinate measurements . . . 252
14.6.2 The dual basis theorem . . . 253
14.6.3 A nonstandard basis in $\mathbb{R}^2$ . . . 255
14.6.4 How to calculate a dual basis using matrix inversion . . . 256
14.6.5 Why vectors and covectors transform differently . . . 257
14.7 Session 3 — Dual maps and the meaning of transpose . . . 260
14.7.1 Pulling a measurement backward . . . 260
14.7.2 Deriving the transpose matrix . . . 261
14.7.3 A concrete example . . . 262
14.7.4 The dual map of our running matrix . . . 263
14.7.5 Composition reverses order . . . 264
14.8 Session 4 — Annihilators and the four fundamental subspaces . . . 266
14.8.1 The annihilator of a subspace . . . 266
14.8.2 The dimension of an annihilator . . . 267
14.8.3 Fundamental properties of annihilators . . . 269
14.8.4 First fundamental theorem of duality . . . 270
14.8.5 Second fundamental theorem of duality . . . 272
14.8.6 The four fundamental subspaces without inner products . . . 273
14.8.7 Worked example using our running matrix . . . 274
14.8.8 A useful rank consequence . . . 276
14.9 Session 5 — Quotient duality and exact sequences . . . 277
14.9.1 A functional on a quotient . . . 277
14.9.2 The quotient duality theorem . . . 278
14.9.3 Geometric meaning of quotient duality . . . 279
14.9.4 Example in three dimensions . . . 280
14.9.5 Duality and subspaces . . . 281
14.9.6 Exact sequences under duality . . . 282
14.9.7 Duality and the first isomorphism theorem . . . 283
14.10 Session 6 — Double duals and the universal structure of duality . . . 284
14.10.1 What is the double dual? . . . 284
14.10.2 Every vector defines an evaluation functional . . . 285
14.10.3 The double-dual theorem . . . 285
14.10.4 Why is $V \cong V^{**}$ different from $V \cong V^*$? . . . 287
14.10.5 Naturality of the double-dual identification . . . 287
14.10.6 Double annihilators . . . 288
14.10.7 Duality reverses sums and intersections . . . 290
14.10.8 Duality and invariant subspaces . . . 291

14.10.9 Invariant subspaces also define quotient operators . . . 292
14.11 Session 7 — Applications of duality . . . 293
14.11.1 Application I — Compatibility conditions for linear systems . . . 293
14.11.2 Application II — Graph gradients, edge measurements, and discrete divergence . . . 295
14.11.3 Application III — Variational formulations and finite elements . . . 298
14.11.4 Application IV — Dual bases and spectral decompositions . . . 303
14.11.5 Left eigenfunctionals . . . 304
14.11.6 Spectral projectors from dual bases . . . 304
14.11.7 Polynomial filtering revisited . . . 306
14.12 Session 8 — Consolidation, exercises, and proof reconstruction . . . 308
14.12.1 Exercise 1 — Construct a dual basis . . . 308
14.12.2 Exercise 2 — Change of coordinates for covectors . . . 309
14.12.3 Exercise 3 — Calculate a dual map . . . 309
14.12.4 Exercise 4 — Find an annihilator . . . 309
14.12.5 Exercise 5 — Verify the fundamental duality identities . . . 310
14.12.6 Exercise 6 — Dual of a quotient . . . 310
14.12.7 Exercise 7 — Dual exact sequences . . . 310
14.12.8 Exercise 8 — The double-dual map . . . 311
14.12.9 Exercise 9 — Graph duality . . . 311
14.12.10 Exercise 10 — Left eigenvectors and spectral projections . . . 312
14.13 Worked solutions . . . 312
14.13.1 Solution 1 — Dual basis . . . 312
14.13.2 Solution 2 — Coordinate transformations . . . 314
14.13.3 Solution 3 — Dual map . . . 314
14.13.4 Solution 4 — Annihilator . . . 315
14.13.5 Solution 5 — Fundamental duality identities . . . 316
14.13.6 Solution 6 — Quotient duality . . . 317
14.13.7 Solution 7 — Dual exact sequence . . . 318
14.13.8 Solution 8 — Double dual . . . 319
14.13.9 Solution 9 — Graph duality . . . 320
14.13.10 Solution 10 — Spectral projectors . . . 320
14.14 Computational laboratory — MATLAB . . . 322
14.14.1 Interpret the numerical results . . . 324
14.15 Final conceptual synthesis . . . 325
14.15.1 The essential relationships . . . 326
14.15.2 Module B mastery checklist . . . 326
14.16 Transition to Module C — Inner-product geometry . . . 327

15 Module C — Inner-Product Geometry 329
15.1 Introduction — From algebraic structure to geometry . . . 329
15.2 Learning outcomes . . . 330
15.3 Study roadmap . . . 331

15.4 Session 1 — Inner products: introducing geometry into vector spaces 332
15.4.1 Why introduce an inner product? 332
15.4.2 Definition of an inner product 333
15.4.3 From inner products to lengths and angles 334
15.4.4 The Cauchy–Schwarz inequality 334
15.4.5 Orthogonality and the Pythagorean theorem 336
15.4.6 Positive definite matrices define inner products 337
15.4.7 Different inner products give different geometries 338
15.4.8 Gram matrices of nonorthonormal bases 339
15.5 Session 2 — The Riesz representation theorem 341
15.5.1 Every vector defines a linear functional 342
15.5.2 Finite-dimensional Riesz representation theorem 342
15.5.3 Proof using an orthonormal basis 343
15.5.4 Geometric interpretation 345
15.5.5 Connection to dual bases 346
15.5.6 Riesz representation in a weighted inner product 346
15.5.7 Gradients are Riesz representatives 348
15.5.8 The infinite-dimensional Riesz theorem: a preview 350
15.6 Session 3 — Adjoint operators and the geometry of linear maps 352
15.6.1 Defining the adjoint 352
15.6.2 Deriving the adjoint from Riesz representation 353
15.6.3 The adjoint and the algebraic dual map 354
15.6.4 Matrix representation in orthonormal coordinates 355
15.6.5 Geometric meaning: pulling measurements backward 356
15.6.6 Adjoints under nonstandard inner products 357
15.6.7 Algebraic properties of adjoints 359
15.6.8 Fundamental orthogonality relationships 360
15.7 Session 4 — Orthogonal projections and closest-point approximation 362
15.7.1 Orthogonal complements 363
15.7.2 Constructing the orthogonal decomposition 363
15.7.3 Why orthogonal projection gives the closest point 364
15.7.4 Matrix of an orthogonal projection 366
15.7.5 Projection using a nonorthonormal basis 367
15.7.6 Orthogonal projection onto the kernel of our running matrix 368
15.7.7 Orthogonal decompositions and quotient spaces 370
15.8 Session 5 — Self-adjoint operators, positive definiteness, and spectral geometry 371
15.8.1 Self-adjoint operators 371
15.8.2 Example: a nonsymmetric matrix representing a self-adjoint operator . 372
15.8.3 Orthogonality of eigenvectors 373
15.8.4 Why self-adjoint eigenvalues are real 374
15.8.5 Invariant subspaces and their orthogonal complements 374
15.8.6 The spectral theorem 376
15.8.7 Positive definite operators 377

15.8.8 Geometry of positive definiteness . . . 379
15.8.9 The Rayleigh quotient and eigenvalue bounds . . . 380
15.8.10 The special operators T^*T and TT^* . . . 381
15.9 Session 6 — Least squares, pseudoinverses, and numerical stability . . . 382
15.9.1 Formulating the least-squares problem . . . 383
15.9.2 Deriving the normal equations geometrically . . . 383
15.9.3 Alternative derivation using optimization . . . 384
15.9.4 Worked least-squares example . . . 385
15.9.5 When is the least-squares solution unique? . . . 387
15.9.6 Minimum-norm solutions and the pseudoinverse . . . 388
15.9.7 Why numerical methods avoid forming normal equations unnecessarily . 390
15.9.8 Weighted least squares . . . 391
15.10 Session 7 — Functional analysis, variational methods, and numerical PDEs . . 392
15.10.1 From finite-dimensional vectors to functions . . . 392
15.10.2 Continuous dual spaces . . . 393
15.10.3 A boundary-value problem . . . 394
15.10.4 The correct function-space setting . . . 395
15.10.5 Riesz representation as a PDE solvability principle . . . 396
15.10.6 The operator maps into the dual space . . . 397
15.10.7 Finite-dimensional Galerkin approximation . . . 397
15.10.8 Why the stiffness matrix is SPD . . . 399
15.10.9 A concrete finite-element example . . . 400
15.10.10 Galerkin orthogonality . . . 401
15.10.11 When the variational form is only semidefinite . . . 403
15.11 Session 8 — Generalized eigenvalue problems and numerical spectral methods . 404
15.11.1 Why generalized eigenvalue problems arise . . . 404
15.11.2 Transforming the generalized problem . . . 405
15.11.3 Orthogonality of generalized eigenvectors . . . 406
15.11.4 The generalized Rayleigh quotient . . . 407
15.11.5 Rayleigh–Ritz approximation . . . 408
15.11.6 Generalized Rayleigh–Ritz approximation . . . 409
15.11.7 Polynomial filtering and self-adjoint spectral theory . . . 410
15.11.8 Why Chebyshev polynomials are useful . . . 411
15.11.9 Connection to HASC and hardware-aware computing . . . 412
15.12 Exercises — Module C . . . 414
15.12.1 Exercise 1 — Inner products and Gram matrices . . . 414
15.12.2 Exercise 2 — Riesz representation in a weighted geometry . . . 414
15.12.3 Exercise 3 — A weighted adjoint . . . 415
15.12.4 Exercise 4 — Orthogonal and oblique projections . . . 415
15.12.5 Exercise 5 — Self-adjointness and positive definiteness . . . 416
15.12.6 Exercise 6 — Rank-deficient least squares . . . 416
15.12.7 Exercise 7 — A finite-element stiffness matrix . . . 417
15.12.8 Exercise 8 — Generalized eigenvalue geometry . . . 417

15.12.9 Exercise 9 — Infinite-dimensional orthogonality . . . 418
15.12.10 Exercise 10 — Structural synthesis . . . 418
15.13 Worked solutions . . . 419
15.13.1 Solution 1 — Weighted geometry . . . 419
15.13.2 Solution 2 — Weighted Riesz representative . . . 420
15.13.3 Solution 3 — Weighted adjoint . . . 421
15.13.4 Solution 4 — Orthogonal versus oblique projections . . . 422
15.13.5 Solution 5 — SPD matrix and Rayleigh quotient . . . 424
15.13.6 Solution 6 — Rank-deficient least squares . . . 425
15.13.7 Solution 7 — Finite-element stiffness matrix . . . 426
15.13.8 Solution 8 — Generalized eigenvalues . . . 427
15.13.9 Solution 9 — Dense but nonclosed range . . . 429
15.13.10 Solution 10 — Structural synthesis . . . 430
15.14 Computational laboratory — MATLAB . . . 432
15.14.1 Interpretation of the laboratory . . . 437
15.15 Final synthesis — What Module C establishes . . . 438
15.15.1 The conceptual chain . . . 438
15.15.2 Important relationships to retain . . . 439
15.15.3 Final mastery checklist . . . 439
15.16 Transition to Module D — Spectral theory . . . 440

16 Module D — Spectral Theory 442
16.1 Introduction — Understanding an operator through its spectrum . . . 442
16.2 Our six perspectives . . . 443
16.3 Study roadmap . . . 444
16.4 Session 1 — Eigenvalues, eigenspaces, and invariant subspaces . . . 444
16.4.1 From arbitrary transformations to operators . . . 444
16.4.2 Eigenvalues as failures of invertibility . . . 445
16.4.3 Real versus complex eigenvalues . . . 446
16.4.4 Algebraic and geometric multiplicity . . . 447
16.4.5 Invariant subspaces . . . 449
16.4.6 Adapted bases and block matrices . . . 450
16.4.7 Diagonalizability as an invariant direct sum . . . 451
16.5 Session 2 — The spectral theorem and spectral projections . . . 452
16.5.1 Self-adjoint and normal operators . . . 453
16.5.2 The finite-dimensional self-adjoint spectral theorem . . . 454
16.5.3 Spectral decomposition into eigenspaces . . . 456
16.5.4 Spectral projections . . . 457
16.5.5 A running spectral example . . . 458
16.5.6 Matrix functions and polynomial functional calculus . . . 460
16.5.7 Spectral projections as polynomials . . . 461
16.5.8 Matrix functions beyond polynomials . . . 462
16.5.9 Structural connection: projectors and algebra . . . 463

16.6 Session 3 — The geometry of spectral subspaces . . . 464
16.6.1 A subspace is more fundamental than a chosen basis . . . 464
16.6.2 Measuring the angle between two lines . . . 465
16.6.3 Principal angles between subspaces . . . 466
16.6.4 An illustrative two-dimensional example . . . 467
16.6.5 What is an approximately invariant subspace? . . . 469
16.6.6 Why spectral gaps matter . . . 470
16.7 Session 4 — Rayleigh quotients, variational principles, and Rayleigh–Ritz . . . 471
16.7.1 The Rayleigh quotient . . . 472
16.7.2 The min–max principle . . . 473
16.7.3 Rayleigh–Ritz approximation . . . 475
16.7.4 Deriving the Ritz equations from residual orthogonality . . . 476
16.7.5 Ritz value interlacing . . . 477
16.7.6 Worked Rayleigh–Ritz example . . . 478
16.7.7 Residuals give eigenvalue inclusion bounds . . . 479
16.7.8 Residuals and eigenvector accuracy . . . 480
16.7.9 Why eigenvalue errors can be smaller than eigenvector errors . . . 482
16.8 Session 5 — Singular value decomposition: spectral geometry for rectangular matrices . . . 484
16.8.1 From A to the self-adjoint matrix A^*A . . . 484
16.8.2 Constructing the left singular vectors . . . 485
16.8.3 Statement of the SVD theorem . . . 486
16.8.4 Geometric meaning: orthogonal directions and stretching factors . . . 487
16.8.5 The four fundamental subspaces from the SVD . . . 488
16.8.6 The SVD and the pseudoinverse . . . 489
16.8.7 Singular values and the condition number . . . 490
16.8.8 Optimal low-rank approximation . . . 490
16.8.9 A rectangular example . . . 493
16.9 Session 6 — From spectral theory to eigenvalue algorithms . . . 494
16.9.1 Why not simply solve the characteristic polynomial? . . . 494
16.9.2 Power iteration . . . 494
16.9.3 Shift-and-invert iteration . . . 496
16.9.4 Rayleigh quotient iteration . . . 497
16.9.5 Subspace iteration . . . 498
16.9.6 Rayleigh–Ritz extraction during subspace iteration . . . 499
16.9.7 Orthogonal iteration and the QR algorithm . . . 500
16.9.8 The role of orthogonality in numerical stability . . . 501
16.10 Session 7 — Polynomial filtering and Chebyshev-filtered subspace iteration . . . 501
16.10.1 Polynomial filtering revisited . . . 502
16.10.2 Why use a polynomial at all? . . . 503
16.10.3 Chebyshev polynomials . . . 503
16.10.4 The minimax property . . . 505
16.10.5 Mapping an unwanted spectral interval to $[-1, 1]$ . . . 506

16.10.6 Explicit example: filtering the spectrum $\{1, 3, 9\}$ . . . 507
16.10.7 Applying the Chebyshev recurrence to a matrix . . . 510
16.10.8 A rigorous subspace convergence estimate . . . 511
16.10.9 Finite-precision and hardware-aware considerations . . . 514
16.11 Session 8 — Spectral theory beyond matrices . . . 516
16.11.1 Spectrum in infinite-dimensional spaces . . . 516
16.11.2 A self-adjoint operator with no eigenvalues . . . 516
16.11.3 Compact self-adjoint operators . . . 517
16.11.4 Differential operators and PDE eigenvalue problems . . . 518
16.11.5 Variational formulation of the PDE eigenproblem . . . 520
16.11.6 The min–max principle in PDE approximation . . . 521
16.11.7 Spectral theory as algebra: a module viewpoint . . . 522
16.12 Exercises — Module D . . . 525
16.12.1 Exercise 1 — Diagonalizability and field dependence . . . 525
16.12.2 Exercise 2 — An invariant subspace without an invariant complement . . . 526
16.12.3 Exercise 3 — Spectral projectors . . . 526
16.12.4 Exercise 4 — Principal angles . . . 527
16.12.5 Exercise 5 — Rayleigh–Ritz and residuals . . . 527
16.12.6 Exercise 6 — Eigenvalue versus eigenvector accuracy . . . 528
16.12.7 Exercise 7 — SVD and minimum-norm least squares . . . 528
16.12.8 Exercise 8 — Power iteration . . . 529
16.12.9 Exercise 9 — A Chebyshev filter . . . 529
16.12.10 Exercise 10 — A residual-based subspace estimate . . . 530
16.12.11 Exercise 11 — Spectral theory of a differential operator . . . 530
16.12.12 Exercise 12 — Universal structure . . . 531
16.13 Worked solutions . . . 531
16.13.1 Solution 1 — Diagonalizability . . . 531
16.13.2 Solution 2 — No invariant complement . . . 532
16.13.3 Solution 3 — Spectral projectors . . . 533
16.13.4 Solution 4 — Principal angles . . . 534
16.13.5 Solution 5 — Rayleigh–Ritz . . . 535
16.13.6 Solution 6 — Quadratic eigenvalue error . . . 536
16.13.7 Solution 7 — SVD and pseudoinverse . . . 537
16.13.8 Solution 8 — Power iteration . . . 538
16.13.9 Solution 9 — Chebyshev filter . . . 539
16.13.10 Solution 10 — Residual and angular error . . . 540
16.13.11 Solution 11 — Dirichlet Laplacian . . . 541
16.13.12 Solution 12 — Operator modules . . . 542
16.14 Computational laboratory — MATLAB . . . 543
16.14.1 What to investigate . . . 549
16.15 Synthesis — How the concepts fit together . . . 550
16.15.1 Relationships worth retaining . . . 551

16.16 Supplementary proof — Why normal matrices are unitarily diagonalizable . . . 552
16.16.1 Step 1: Schur triangularization . . . 552
16.16.2 Step 2: A normal upper-triangular matrix is diagonal . . . 553
16.17 Module D — Final mastery assessment . . . 555
16.17.1 Recommended consolidation exercise . . . 556
16.18 Transition to Module E — Numerical methods and stability . . . 556

17 Module E — Numerical Eigenvalue Methods, Conditioning, and Stability 558
17.1 From a spectral decomposition to a numerical claim . . . 558
17.2 Session 1 — Floating-point arithmetic, conditioning, and stability . . . 559
17.2.1 What one rounding step changes . . . 559
17.2.2 From dot products to a matrix product . . . 560
17.2.3 Problem sensitivity versus the error of a method . . . 560
17.3 Session 2 — Eigenpair residuals and perturbations . . . 561
17.3.1 A residual as the price of exactness . . . 561
17.3.2 Locating an eigenvalue and locating a direction . . . 561
17.3.3 Weyl and subspace separation . . . 562
17.3.4 Why nonnormality changes the certificate . . . 562
17.4 Session 3 — Vector iterations as spectral transformations . . . 563
17.4.1 Power iteration and its rate . . . 563
17.4.2 Inverse iteration and Rayleigh quotient iteration . . . 563
17.5 Session 4 — Block iteration, principal angles, and extraction . . . 564
17.5.1 Why a basis needs attention . . . 564
17.5.2 Rayleigh–Ritz and finite precision . . . 564
17.6 Session 5 — Krylov spaces and projected eigenproblems . . . 565
17.6.1 Keeping the history . . . 565
17.6.2 Arnoldi, Lanczos, and the residual shortcut . . . 565
17.7 Session 6 — Designing and evaluating a Chebyshev filter . . . 566
17.7.1 The design problem . . . 566
17.7.2 Subspace convergence and degree choice . . . 567
17.7.3 Evaluating a polynomial without forming powers . . . 567
17.7.4 What a local error becomes . . . 568
17.7.5 Turning application error into angle error . . . 569
17.7.6 Bounds that miss part of the spectrum . . . 569
17.8 Session 7 — Mixed precision and hardware-aware computation . . . 569
17.8.1 An error budget by operation . . . 569
17.8.2 Cost of reaching an accuracy target . . . 571
17.9 Session 8 — A controlled experiment from prediction to verification . . . 571
17.9.1 Construct a problem with known answers . . . 571
17.9.2 A measured history . . . 572
17.9.3 Transfer to a variational PDE problem . . . 572
17.10 Exercises and worked solutions . . . 573
17.10.1 1. Which data are allowed to move? . . . 573

17.10.2 2. A Rayleigh quotient can conceal an inaccurate vector . . . 573
17.10.3 3. A backward error with symmetry . . . 574
17.10.4 4. Fixed versus adaptive inverse iteration . . . 574
17.10.5 5. A block that cannot recover its target . . . 574
17.10.6 6. From an approximate Arnoldi relation to a trustworthy residual . . . 574
17.10.7 7. Designing a degree . . . 575
17.10.8 8. Endpoint accumulation . . . 575
17.10.9 9. Verification with a mass matrix . . . 575
17.10.10 10. Distinguishing a measured floor from a theorem . . . 575
17.10.11 11. A filter written in terms of its residual . . . 576
17.10.12 12. Design an equal-work comparison . . . 576
17.11 Computational laboratory — MATLAB . . . 576
17.12 Synthesis — What makes the computation reliable? . . . 580
17.12.1 Mastery questions . . . 581
17.12.2 Where to go next . . . 581

III Review III **583**

18 Module A — Structural Linear Algebra: Linear Maps, Decomposition, and Quotient Structures **584**
18.1 Session 1 — When can a list of numbers describe a polynomial? . . . 584
18.1.1 From linear combinations to coordinates . . . 584
18.1.2 Why do all bases have the same size? . . . 585
18.2 Session 2 — How do we transport a calculation between bases? . . . 586
18.2.1 Change the encoding, then differentiate . . . 586
18.3 Session 3 — What can one reading determine? . . . 587
18.4 Session 4 — What does elimination preserve? . . . 588
18.4.1 Recovering the kernel and the embedded image . . . 589
18.5 Session 5 — When is a decomposition a coordinate system? . . . 590
18.5.1 A complement chooses one point per fiber . . . 590
18.6 Session 6 — Can we recover exactly what a measurement knows? . . . 593
18.7 Session 7 — Which maps survive a quotient, and which maps split it? . . . 595
18.7.1 What specifies a map into or out of a direct sum? . . . 595
18.7.2 Exactness and splitting . . . 595
18.8 Session 8 — Which part of a physical or numerical solution is identifiable? . . . 596
18.8.1 Compatible readings, inconsistent data, and an inverse on classes . . . 596
18.8.2 Edge differences, cycles and conservation . . . 597
18.8.3 Which boundary condition removes an integration constant? . . . 597
18.8.4 What does a subspace algorithm actually carry between iterations? . . . 598
18.9 Computational laboratory — From an inverse problem to a factorization . . . 599
18.9.1 A. What can row reduction certify? . . . 599
18.9.2 B. Which two components sum to the input? . . . 600

18.9.3 C. Can a new representative alter the quotient coordinates? 600
18.9.4 D. Where is the invertible part of a singular problem? 601
18.9.5 E. Which edge readings come from potentials? 601
18.9.6 F. Which basis represents the trial plane? 602
18.10 Exercises — Reconstruct the arguments 602
18.10.1 Build an encoding 602
18.10.2 Diagnose solvability and decomposition 602
18.10.3 Descend, factor, and split 603
18.10.4 Apply the structure 603
18.11 Worked solutions 604
18.12 Session 9 — Which information survives a boundary condition? 606
18.12.1 Integrated mastery problem 606
18.12.2 Worked reconstruction 606
18.12.3 How the constructions fit together 608
18.12.4 Mastery checklist 610
18.12.5 Transition to Module B — What can be measured? 610

19 Module B — Duality and Linear Measurements: From Covectors to Universal Structure 611

19.1 Session 1 — How much of a vector does a measurement describe? 611
19.1.1 The geometry a single reading gives you 612
19.1.2 Can all linear measurements distinguish two different vectors? 612
19.2 Session 2 — How does an oblique grid measure coordinates? 613
19.2.1 A grid whose axes meet at an oblique angle 613
19.2.2 Which quantities change oppositely? 615
19.3 Session 3 — How does an output instrument become an input instrument? . . 615
19.3.1 Deriving the transpose 616
19.4 Session 4 — Which questions about an unknown input can the data answer? . 616
19.4.1 Computing all four spaces in the sensor problem 617
19.4.2 A compatibility test that is also sufficient 618
19.5 Session 5 — Which instruments survive identification, and which survive restriction? 618
19.5.1 See both statements in the same three coordinates 619
19.5.2 What happens to a short exact sequence? 619
19.6 Session 6 — Can a vector be recovered from all its possible readings? 620
19.6.1 Can invisible directions be recovered from invisible measurements? . . . 621
19.7 Session 7 — Who reads the coefficient of an eigenvector? 622
19.7.1 A basis of modes brings its own instruments 622
19.7.2 Two eigenlines close together 623
19.8 Session 8 — Where does a transpose arise without a dot product? 624
19.8.1 A graph trades edge instruments for vertex instruments 624
19.8.2 A weak equation produces a functional before it produces a vector . . . 624
19.8.3 A derivative is a measurement of a perturbation 625

19.8.4 Where the finite-dimensional argument stops . . . 625
19.9 Computational laboratory — Seven experiments in measuring and reconstructing 626
19.9.1 A. Calibrate three instruments on a nonorthogonal basis . . . 626
19.9.2 B. Change coordinates while preserving every evaluation . . . 627
19.9.3 C. Pull a three-channel instrument back to four settings . . . 627
19.9.4 D. Test both annihilator equalities as equalities of spaces . . . 627
19.9.5 E. Calibrate a quotient instrument and change the representative . . . 628
19.9.6 F. Detect a circulation and reconstruct a vertex measurement . . . 629
19.9.7 G. Compare an exact projector with sensitive spectral coordinates . . . 629
19.10 Exercises — Measurements, obstructions and representations . . . 631
19.10.1 Construct and transport instruments . . . 631
19.10.2 Reconstruct the structural proofs . . . 632
19.10.3 Use measurements in an unfamiliar problem . . . 632
19.11 Worked solutions . . . 633
19.12 Session 9 — What can three curvature readings tell us about a cubic? . . . 635
19.12.1 Integrated mastery problem . . . 635
19.12.2 Worked reconstruction . . . 635
19.12.3 Conceptual map . . . 637
19.12.4 What geometry adds to duality . . . 638
19.12.5 Common wrong turns . . . 639
19.12.6 Mastery checklist . . . 640

20 Module C — Inner-Product Geometry: From Duality to Orthogonality, Approximation, and Variational Methods 641
20.1 Session 1 — How much change is one unit? . . . 641
20.1.1 Change the measuring instrument . . . 642
20.2 Session 2 — How do coordinates remember the geometry? . . . 643
20.2.1 Constructing coordinates in which lengths are simple . . . 644
20.3 Session 3 — Which vector represents a fixed functional? . . . 644
20.4 Session 4 — How should an output direction be pulled back? . . . 646
20.5 Session 5 — What makes one point in a subspace the closest? . . . 647
20.5.1 Compute the projection without first orthogonalizing . . . 648
20.5.2 Four subspaces and one closure . . . 648
20.6 Session 6 — Which transformations respect the geometry? . . . 649
20.6.1 Two sources of positive matrices . . . 650
20.7 Session 7 — When the equations disagree, what exactly should be minimized? . . . 651
20.7.1 A complete inverse problem with two projections . . . 651
20.7.2 Why the equations we derive can differ from the algorithm we run . . . 652
20.7.3 What if different sensors have different costs? . . . 652
20.8 Session 8 — How can a differential equation ask for a closest point? . . . 653
20.8.1 A mesh with two unknowns . . . 653
20.9 Session 9 — How does a mass matrix enter an eigenvalue problem? . . . 655
20.9.1 A self-adjoint operator behind a Hermitian pair . . . 655

20.9.2 Rayleigh–Ritz asks fewer questions than the full eigenproblem 655
20.10 Computational laboratory — Check the geometry as well as the equations . . . 656
20.10.1 A. Weighted unit directions 656
20.10.2 B. Re-encode the same geometry 657
20.10.3 C. Represent one complex measurement in two metrics 657
20.10.4 D. Pull back through different domain and codomain geometries 658
20.10.5 E. Two projections onto the same plane 658
20.10.6 F. Compare normal equations, QR and SVD 660
20.10.7 G. Select the minimum-norm input for incompatible data 661
20.10.8 H. Assemble the Poisson equation and measure the slope error 661
20.10.9 I. Compute generalized eigenpairs and test a proper trial subspace . . . 663
20.11 Exercises — Reconstruct, calculate, transfer 663
20.11.1 Geometry from measurements 663
20.11.2 Approximation and operator structure 664
20.11.3 From tests to modes 664
20.12 Worked solutions 665
20.13 Integrated mastery problem — One instrument, two geometries, one reduced model 667
20.13.1 Worked reconstruction 667
20.14 Synthesis — Which structure made each conclusion possible? 669
20.14.1 Mastery checklist 671

21 Module D — Spectral Theory 673
21.1 Session 1 — Finding structure that survives an application of the operator . . . 674
21.1.1 What the off-diagonal block measures 674
21.2 Session 2 — Reading spectral components with the dual basis 675
21.3 Session 3 — When can the modes be chosen orthogonal? 676
21.4 Session 4 — Turning a scalar function into an operator 677
21.4.1 An exact mode selector 677
21.5 Session 5 — Measuring an error when the answer is a space 678
21.5.1 Deriving the principal directions 678
21.5.2 A diagnostic when the target is unknown 679
21.6 Session 6 — Finding eigenvalues by optimizing a scalar measurement 679
21.7 Session 7 — What spectral information can a trial space certify? 680
21.8 Session 8 — Spectral geometry for maps between different spaces 682
21.8.1 Which inverse solves inconsistent data? 682
21.8.2 What can any rank budget achieve? 683
21.9 Session 9 — Designing a polynomial that improves a trial space 683
21.9.1 First follow a line 683
21.9.2 Now follow a space 684
21.9.3 From a spectral window to a Chebyshev filter 687
21.10 Session 10 — What survives beyond finite matrices? 688
21.10.1 Spectral values without eigenvectors 688

21.10.2 From a vibrating string to a generalized eigenproblem 688
21.10.3 When polynomials act like scalars 689
21.11 Exercises 690
21.12 Worked solutions 691
21.13 Computational laboratory 693
21.13.1 A. Count modes before computing directions 693
21.13.2 B. Reassemble a Hermitian operator from orthogonal projectors 694
21.13.3 C. Compare polynomial evaluation routes 694
21.13.4 D. Rotate one direction in a plane 695
21.13.5 E. Follow the energy around a circle 696
21.13.6 F. Solve the reduced problem and test it in the full space 697
21.13.7 G. Keep only the strongest channel of a rectangular map 698
21.13.8 H. Compare vector and block convergence with predicted factors 698
21.13.9 I. Change the degree and inspect the response 700
21.13.10 J. Filter, orthonormalize, extract and diagnose 702
21.13.11 K. Discretize a differential eigenproblem and refine the mesh 703
21.14 Synthesis — Which structure does the computation use? 704
21.14.1 Integrated problem — Certify a filtered mode without its coordinates . 706
21.14.2 Mastery assessment 706

22 Module E — Numerical Eigenvalue Methods: Conditioning, Stability, Polynomial Filtering, and Mixed Precision 708
22.1 From a spectral decomposition to a numerical claim 708
22.2 Session 1 — How accurately does a computer apply a matrix? 709
22.2.1 What one rounding step changes 709
22.2.2 From dot products to a matrix product 709
22.3 Session 2 — Is the problem sensitive or the algorithm unstable? 710
22.3.1 Problem sensitivity versus the error of a method 710
22.3.2 Which entries may change? 711
22.4 Session 3 — What does an eigenpair residual certify? 711
22.4.1 A residual as the price of exactness 711
22.4.2 Locating an eigenvalue and locating a direction 712
22.4.3 Weyl and subspace separation 712
22.4.4 Why nonnormality changes the certificate 713
22.5 Session 4 — Vector iterations as spectral transformations 713
22.5.1 Power iteration and its rate 713
22.5.2 Inverse iteration and Rayleigh quotient iteration 714
22.6 Session 5 — Block iteration, principal angles, and extraction 715
22.6.1 Why a basis needs attention 715
22.6.2 Rayleigh–Ritz and finite precision 716
22.7 Session 6 — Krylov spaces and projected eigenproblems 716
22.7.1 Keeping the history 716
22.7.2 Arnoldi, Lanczos, and the residual shortcut 717

22.8 Session 7 — Designing and evaluating a Chebyshev filter . . . 718
22.8.1 The design problem . . . 718
22.8.2 Subspace convergence and degree choice . . . 718
22.8.3 Evaluating a polynomial without forming powers . . . 719
22.9 Session 8 — What happens to an error made inside a filter? . . . 719
22.9.1 What a local error becomes . . . 719
22.9.2 Initialization, storage and one local product . . . 720
22.9.3 Turning application error into angle error . . . 720
22.9.4 Bounds that miss part of the spectrum . . . 721
22.10 Session 9 — Mixed precision and hardware-aware computation . . . 721
22.10.1 An error budget by operation . . . 721
22.10.2 Cost of reaching an accuracy target . . . 723
22.11 Session 10 — From a contraction factor to a verified eigenspace . . . 723
22.11.1 Construct a problem with known answers . . . 723
22.11.2 Predict before running . . . 724
22.11.3 Transfer to a variational PDE problem . . . 724
22.12 Exercises and worked solutions . . . 725
22.12.1 1. Which data are allowed to move? . . . 725
22.12.2 2. A Rayleigh quotient can conceal an inaccurate vector . . . 725
22.12.3 3. A backward error with symmetry . . . 725
22.12.4 4. Fixed versus adaptive inverse iteration . . . 726
22.12.5 5. A block that cannot recover its target . . . 726
22.12.6 6. From an approximate Arnoldi relation to a trustworthy residual . . . 726
22.12.7 7. Designing a degree . . . 726
22.12.8 8. Endpoint accumulation . . . 727
22.12.9 9. Verification with a mass matrix . . . 727
22.12.10 10. Distinguishing a measured floor from a theorem . . . 727
22.12.11 11. A filter written in terms of its residual . . . 727
22.12.12 12. Design an equal-work comparison . . . 728
22.12.13 13. Reconstruct the four proof steps behind a filtered solver . . . 728
22.13 Computational laboratory — From one rounded sum to a verified cluster . . . 728
22.13.1 A. Which digits did cancellation expose? . . . 729
22.13.2 B. Can we move a Hermitian eigenvalue farther than the perturbation? . . . 730
22.13.3 C. Which changes more, an eigenvector or its cluster? . . . 731
22.13.4 D. What changes when the shift follows the vector? . . . 732
22.13.5 E. Does block iteration attain the graph bound? . . . 733
22.13.6 F. When can the final Krylov coefficient be trusted? . . . 734
22.13.7 G. What if the interval misses an unwanted mode? . . . 735
22.13.8 H. Where does a local filter defect go? . . . 737
22.13.9 I. Which precision limits the final residual? . . . 738
22.13.10 J. Which method reaches the same verified cluster with the least operator work? . . . 740

22.14 Synthesis — What makes the computation reliable? . . . 741
22.14.1 Choosing an algorithm from the spectral information available . . . 742
22.14.2 Reading the measured evidence . . . 744
22.14.3 Integrated problem — Write a numerical acceptance report . . . 745
22.14.4 Mastery checklist . . . 745
22.14.5 Where to go next . . . 746

23 Module F1 — Functional Analysis: Infinite-Dimensional Geometry and Operator Theory 748
23.1 Session 1 — What is a limit of vectors? . . . 748
23.1.1 Distance adds structure to a vector space . . . 748
23.1.2 Why the choice of norm used to be harmless . . . 749
23.1.3 A dictionary of spaces . . . 749
23.1.4 Completing an unfinished space . . . 750
23.2 Session 2 — Which measurements survive approximation? . . . 751
23.2.1 A global bound from continuity at one point . . . 751
23.2.2 Two notions of duality . . . 751
23.2.3 Quotients and annihilators revisited . . . 752
23.3 Session 3 — Geometry that survives an infinite sum . . . 753
23.3.1 Existence of a nearest vector . . . 753
23.3.2 Orthonormal coordinates and their limits . . . 754
23.3.3 The representing vector and weak convergence . . . 754
23.4 Session 4 — Solvability, adjoints and the missing closure . . . 755
23.4.1 Compatibility and stability ask different questions . . . 755
23.5 Session 5 — From eigenvalues to spectra . . . 756
23.5.1 Compactness as uniform finite-dimensional approximation . . . 756
23.5.2 An inverse-based definition of spectrum . . . 757
23.5.3 What additional structure restores eigenvectors? . . . 757
23.6 Session 6 — Solving an equation by testing it . . . 758
23.6.1 From infinitely many probes to a linear system . . . 759
23.7 Session 7 — Which space should a differential equation live in? . . . 760
23.7.1 One derivative in the weak sense . . . 760
23.7.2 One boundary-value problem, two operators . . . 761
23.7.3 Assembly on an interval . . . 761
23.8 Session 8 — From a compact inverse to computed eigenpairs . . . 762
23.8.1 The variational characterization and its discretization . . . 762
23.9 Exercises . . . 763
23.9.1 Worked solutions . . . 764
23.10 Computational laboratory — What does refinement actually measure? . . . 766
23.10.1 Experiment 1 — A tail, a unit ball and amplified noise . . . 766
23.10.2 Experiment 2 — A Galerkin solution with zero nodal error . . . 768
23.10.3 Experiment 3 — Eigenvalue convergence and solver residual . . . 769

23.11 Synthesis — What did the topology change? . . . 771
23.11.1 Theorem map . . . 772
23.11.2 Mastery assessment . . . 774

24 Module F2 — Structural Algebra: From Linear Algebra to Universal Constructions and Category Theory 775
24.1 Session 1 — Which maps preserve the problem? . . . 775
24.1.1 From changing coordinates to changing categories . . . 775
24.1.2 What a diagram asserts . . . 776
24.2 Session 2 — Three problems that determine their own objects . . . 777
24.2.1 Identifying inputs while preserving all admissible measurements . . . 777
24.2.2 Assembling outputs and prescribing inputs . . . 778
24.2.3 Adding subspaces in an ambient space . . . 778
24.3 Session 3 — Kernels, obstructions and splitting . . . 779
24.4 Session 4 — Pulling measurements back . . . 780
24.4.1 Evaluation and its limitations . . . 780
24.4.2 Exactness seen by measurements . . . 781
24.5 Session 5 — When constructions respect changes of data . . . 782
24.5.1 Why a mapping problem can have an object as its answer . . . 782
24.6 Session 6 — A linear operator is a polynomial action . . . 783
24.6.1 Why enlarge the scalars? . . . 783
24.6.2 Cyclic modules, Krylov spaces and annihilators . . . 783
24.6.3 A two-dimensional extension that refuses to split . . . 784
24.6.4 Coprime annihilators give genuine projectors . . . 784
24.6.5 Why Jordan blocks arise, and what the minimal polynomial omits . . . 785
24.7 Session 7 — Turning bilinear rules into linear maps . . . 786
24.8 Session 8 — When zero composition leaves a hole . . . 787
24.8.1 From chains to differential operators . . . 788
24.9 Session 9 — Four problems seen through several constructions . . . 788
24.9.1 A. Recovering a potential from measurements . . . 788
24.9.2 B. What a Krylov process can recover . . . 789
24.9.3 C. A mesh with one remaining circulation . . . 789
24.9.4 D. From a quotient to an operator equation . . . 790
24.9.5 Bridge to F1 — Where topology changes the answer . . . 790
24.10 Exercises . . . 791
24.11 Worked solutions . . . 792
24.12 Computational laboratory — Exact identities in MATLAB . . . 793
24.12.1 1. Compute quotient coordinates and factor a map . . . 794
24.12.2 2. Separate primary spaces and measure Krylov growth . . . 794
24.12.3 3. Identify tensor coordinates . . . 795
24.12.4 4. Count classes before and after attaching faces . . . 795
24.13 Final synthesis — Choosing the object that answers the question . . . 796
24.14 Mastery assessment . . . 797

Preface

1 Overview

1.1 Numerical Linear Algebra Overview

1.1.1 Main computational problems

Problem	Mathematical formulation	Typical numerical methods
Linear systems	$Ax = b$	LU, Cholesky, CG, GMRES
Eigenvalue problems	$Av = \lambda v$	Power iteration, QR iteration, Lanczos, Arnoldi
Least-squares problems	$\min_x \|Ax - b\|_2$	QR, SVD
Matrix factorizations	$A = QR,\ A = LU,\ A = U\Sigma V^*$	Householder, Gaussian elimination, SVD algorithms
Low-rank approximation	$A \approx UV^*$	Truncated SVD, randomized algorithms
Matrix functions	$f(A)v$, e.g. $e^A v$	Polynomial and Krylov approximations

1.1.2 The special importance of large and sparse matrices

large and sparse matrices arise naturally in many settings, e.g. determining the stationary temperature distribution in a two-dimenstional plate.

Iterative methods and subspaces exploit sparsity. This is more effective than simply extending Gaussian elimination to bigger matrices.

1.2 Course Overview

The central focus is computing solutions of linear systems and eigenvalue problems for dense and large sparse matrices.



Figure 1.1: Sparsity pattern of a tridiagonal matrix.

1.2.1 The five main chapters

Chapter	Subject	Central ideas
1. Introduction	Applications and motivation	Discretized differential equations, least squares, sparse matrices, PageRank
2. Orthogonalization and QR Factorization	Constructing orthonormal bases	Gram–Schmidt, Householder transformations, Givens rotations
3. Dense Algebraic Eigenvalue Problems	Computing eigenvalues of dense matrices	Schur decomposition, eigenvalue conditioning, vector and subspace iterations, QR iteration
4. Large Sparse Linear Systems	Solving very large $Ax = b$ problems	Jacobi, Gauss–Seidel, projection methods, Krylov subspaces, CG, GMRES, preconditioning
5. Large Sparse Eigenvalue Problems	Finding selected eigenvalues of huge matrices	Rayleigh–Ritz, Lanczos, Arnoldi, implicitly restarted Arnoldi, interior eigenvalues

1.2.2 Relationship Between Chapters

orthogonalization and projection are the unifying ideas behind much of the course.

Orthogonalization provides suitable bases for subspaces. Projection reduces large computational problems to smaller ones.

A Krylov subspace has the form:

$$\mathcal{K}_m(A, v) = \text{span}\{v, Av, A^2v, \dots, A^{m-1}v\}.$$

This same construction appears in both large sparse linear-system solvers an dlarge sparse eigensovles. The same Arnoldi or Lanczos machinery can support both kinds of computation.

1.3 Linear Algebra and Numerics Prerequisites and Review

1.3.1 Essential linear algebra topics

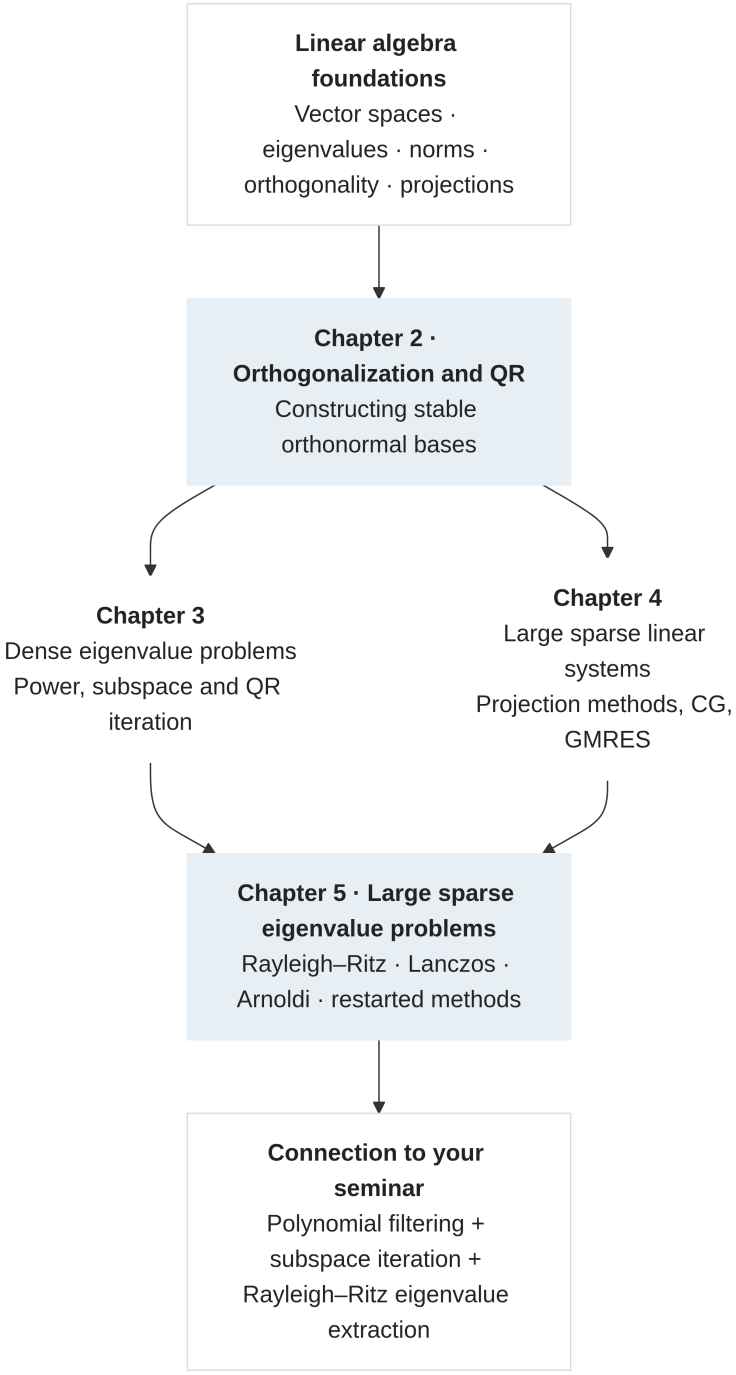


Figure 1.2: Conceptual relationships between the main chapters.

Topic	What you should be able to do	Why it matters for NLA
Vector spaces, bases and dimension	Work with span, linear independence, rank, kernel, image and basis changes	Almost every iterative algorithm constructs or manipulates subspaces
Inner products and orthogonality	Calculate inner products, orthogonal complements, orthonormal bases	Gram–Schmidt, QR, Arnoldi, Lanczos
Orthogonal projections	Understand projections onto subspaces and their matrix representation	Least squares, Galerkin and Rayleigh–Ritz methods
Eigenvalues and eigenvectors	Compute simple eigenpairs, understand diagonalization and eigenspaces	Chapters 3 and 5, and your seminar
Symmetric and Hermitian matrices	Understand the spectral theorem and orthogonal diagonalization	CG, Lanczos, eigenvalue conditioning
Positive-definite matrices	Recognize SPD matrices and understand quadratic forms	Cholesky, CG, energy minimization
Similarity transformations	Interpret $V^{-1}AV$ as a basis change	Schur decomposition, QR eigenvalue iteration
Matrix norms and spectral radius	Understand operator norms, eigenvalue magnitudes and basic inequalities	Error estimates, convergence, conditioning

Especially three concepts deserve extra attention:

Orthogonality, projection and subspaces

Suppose $Q \in \mathbb{R}^{n\times m}$ has orthonormal columns:

$$Q^TQ = I_m.$$

Then the matrix

$$P = QQ^T$$

is the orthogonal projector onto the subspace spanned by the columns of Q.

Eigenvalues, diagonalization and the spectral theorem

For a general diagonalizable matrix,

$$A = V\Lambda V^{-1}.$$

For a real symmetric matrix, the much stronger result

$$A = Q\Lambda Q^T, \qquad Q^TQ = I.$$

Is the spectral theorem. It t ells us that a symmetric matrix has real eigenvalues and admits an orthonormal basis or eigenvectors.

Consider an inital vector

$$v = \sum_{i=1}^{n} \alpha_i q_i,$$

where q_i are the orthonormal eigenvectors of A. Then

$$A^k v = \sum_{i=1}^{n} \alpha_i \lambda_i^k q_i.$$

Applying the matrix repeatedly changes the relative importance of the eigenvector accordidng to the magnitued of their eigenvalues. This explains the fundamental mechanism of **power iteration** and, after generalization, subspace iteration and polynomial filtering.

Symmetric positive definiteness and quadratic forms

A real symmetric matrix A is positive definite when

$$x^TAx > 0 \qquad \text{for every } x \neq 0.$$

Equivalent characterizations include the positivity of all eigenvalues and the existence of a Cholesky factorization.

The relationship between SPD matrices and minimization problems is especially useful. For example, solving $Ax = b$ for an SPD matrix is equivalnet to minimizing

$$f(x) = \frac{1}{2}x^TAx - b^Tx.$$

This provides a natural connection between linear systems, quadratic optimization and the conjugate gradient method.

1.3.2 Prerequisites from Numerics 0

Subject	Bastian's `num0.pdf`	Scheichl–Zech's `Num0_notes_ss25.pdf`	Priority
Floating-point arithmetic and rounding errors	§§2–3	§§2.3–2.4	High
Conditioning, matrix norms and stability	§§3, 5–6	§§2.1–2.2, 3.1–3.3	Very high
Direct methods: Gaussian elimination, LU, Cholesky	§§7–10	§§3.4–3.7	Medium–high
Least squares, QR and Householder	§§11–12	§§3.8–3.9	Very high
Basic iterative linear solvers	§§13–14	§6.3	Very high
Polynomial approximation, especially Chebyshev	§§17–19, 23 for broader background	§§4.9–4.10	High for seminar

Conditioning versus stability

Conditioning is a property of a mathematical problem. Stability is a property of an algorithm used to solve that problem.

For instance, suppose $Ax = b$ and we perturb the right-hand side by Δb. The solution changes according to

$$\Delta x = A^{-1}\Delta b$$

The condition number

$$\kappa(A) = \|A\|\|A^{-1}\|$$

measures the potential amplification of relative errors. But even if a problem is well-conditioned, a poor algorithm can introduce unnecessarily large computational errors.

It is alos essential to distinguish the residual

$$r = b - A\tilde{x}$$

from the actual solution error

$$e = x - \tilde{x}.$$

They are related by

$$Ae = r, \qquad e = A^{-1}r.$$

Thus a small residual does not automatically guarantee a small forward error when the matrix is ill-conditioned.

These ideas reappear throghout NLA, including in the conditioning of eigenvalues and eigenvectors.

Matrix factorizations and orthogonal transformations

Distinction between QR and LU:

- LU decomposes a matrix into triangular factors and supports the direct solution of linear systems.
- QR constructs orthonormal bases and triangular factors, which are especially useful least squares and eigenvalue algorithms.
- Cholesky exploits symmetric positive definiteness.

For NLA **QR is more urgent than detailed LU pivoting analysis**, because Ch 2 of NLA directly develops Gram-Schmidt, Householder and Givens transformations.

Basic iterative methods and convergence

Consider an iteration

$$x^{(k+1)} = Bx^{(k)} + c.$$

If x^* is its fixed point, then the error satisfies

$$e^{(k+1)} = Be^{(k)}.$$

Consequently,

$$e^{(k)} = B^k e^{(0)}.$$

The fundamental convergence criterion is

$$\boxed{\rho(B) < 1}$$

where $\rho(B)$ is the spectral radius.

This connects eigenvalues directly to the convergence of numerical algorithms. Jacobi and Gauss-Seidel should be revised, but with a focus on understanding how their iteration matrices arise and how their convergence can be analyzed.

Chevyshev Polynomials

The NLA script has Appendix B.2: Chebyshev polynomials, beginning on printed page 153.

It introduces the recurrence

$$T_0(x) = 1, \qquad T_1(x) = x,$$

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x),$$

and properties of Chebyshev polynomials on and outside $[-1, 1]$

1.3.3 The connection to the Chebyshev-filtered subspace iteration seminar

Ordinary subspace iteration

Suppose A is symmetric and we wish to compute several of its dominant eigenpairs.

We begin with a matrix Q_0 whose columns are orthonormal and repeatedly compute

$$Y_{k+1} = AQ_k,$$

followed by orthonormalization,

$$Q_{k+1} = \text{orth}(Y_{k+1}).$$

The successive subspaces approach an invariant subspace corresponding to dominant eigenvalues, subject to appropriate initial-subspace and spectral-gap assumptions.

Polynomial-filtered iteration

Instead of multiplying by A, we apply a polynomial in A:

$$Y_{k+1} = p_m(A)Q_k.$$

Why does this help?

Because for an eigenvector q_i,

$$Aq_i = \lambda_i q_i \quad \Longrightarrow \quad p_m(A)q_i = p_m(\lambda_i)q_i.$$

We can therefore choose a polynomial that makes the desired eigenvalue components comparatively large and suppresses the unwanted components.

A Chebyshev polynomial, after suitable shifting and scaling, is an effective choice because of its controlled magnitude on a specified interval and rapid growth outside it.

The important point is that we can apply such a polynomial using a recurrence involving matrix–vector products, without constructing the generally dense matrix $p_m(A)$.

What is needed to understand the complete algorithm

Component	Mathematical prerequisite	NLA connection
Spectral transformation $p(A)$	Diagonalization, matrix polynomials	Appendix A.1.2, §3.3
Desired invariant subspace	Subspaces, eigenspaces, spectral gaps	§§3.2–3.3
Chebyshev filter	Polynomial recurrence and approximation	Appendix B.2
Orthonormalization	Inner products, Gram–Schmidt, QR	Chapter 2
Subspace update	Orthogonal subspace iteration	§3.3.2
Approximate eigenpairs	Rayleigh quotient, orthogonal projection	§5.1
Competing eigensolvers	Krylov spaces, Lanczos, Arnoldi	Chapters 4–5

The script even develops polynomial filtering in §3.3. There is an example using

$$A = \mathrm{diag}(1, 2, 3)$$

and the polynomial

$$p(t) = (t-1)(t-3).$$

Here the unwanted eigenvalues 1 and 3 are mapped to zero, while the component associated with eigenvalue 2 is retained.

Chapter 5 then combines projection, Rayleigh–Ritz approximation and Krylov-based eigenvalue methods. This is a natural starting point for comparing your seminar algorithm with Lanczos and Arnoldi methods.

One subtle but important distinction is that the seminar's polynomial-filtered subspace method need not itself be a Krylov-subspace algorithm in the same sense as Arnoldi. Both make essential use of matrix polynomials and subspaces, but organize their iterations differently.

1.4 Revision Strategy

1.4.1 Phase 1 - linear algebra foundation

Vector spaces and bases, innder products, projections, eigenvalues, diagonalization, and SPD matrices.

Useful readiness test:

> can you explain, and then prove why QQ^T is an orthogonal projector when Q has orthonormal columns

1.4.2 Phase 2 - numerical-analysis foundation

Conditioning, residuals and error estimates, floating-point arithmetic, QR factorization and convergence of iterative methods.

readiness test:

1. why does conditioning matter?
2. why can a small residual be misleading?
3. why does the spectral radius control stationary iterations?

1.4.3 Phase 3 - Eigenvalues

Power method, Rayleigh quitient, subspace iteration and polynonial filtering. Chebyshev polynomials and Rayleigh-Ritz.

1.4.4 Targeted Revision Plan

11 focused sessions.

Week 1 - Rebuild the linear algebra foundations

Goal: Recover fluency with vectors, subspaces, projections, and orthogonalization so that Chapter 2 becomes manageable.

Session	Material to revise	Mastery check
1. Vector spaces and linear maps · 60 min	Span, linear independence, bases, dimension, rank, kernel, image, change of basis	Given a matrix, determine bases for its kernel and image; explain rank–nullity
2. Inner products and orthogonality · 75 min	Inner products, orthogonal complements, orthonormal bases, Gram–Schmidt	Construct an orthonormal basis from linearly independent vectors
3. Orthogonal projections · 75 min	Projection onto a line/subspace, $P = QQ^T$, least-squares geometry	Derive the projection matrix and prove $P^2 = P$
4. QR factorization · 90 min	Gram–Schmidt, Householder reflections, QR, least-squares solution	Perform QR on a small matrix and explain why orthogonality is useful numerically

Reading sources: NLA Appendix A.1–A.3, NLA Chapter 2, Bastian's §5 and §§11–12, and Scheichl–Zech's §§3.8–3.9.

Week 2 - Recover spectral theory and numerical analysis

Goal: Understand how matrix structure, conditioning and numerical algorithms interact.

Session	Material to revise	Mastery check
5. Eigenvalues and the spectral theorem · 90 min	Eigenvectors, eigenspaces, diagonalization, symmetric/Hermitian matrices, SPD	Diagonalize a symmetric matrix and use the decomposition to compute $A^k v$
6. Matrix norms and conditioning · 75 min	Induced norms, spectral radius, condition numbers, residual versus error, stability	Derive an error estimate using $e = A^{-1}r$ and explain why small residuals can mislead

Session	Material to revise	Mastery check
7. Direct numerical methods · 60 min	Gaussian elimination, LU, pivoting, triangular systems, Cholesky	Solve a system via LU and explain when Cholesky applies
8. Basic iterative methods · 75 min	Fixed-point iteration, Jacobi, Gauss–Seidel, iteration matrices, convergence	Derive $e^{(k+1)} = Be^{(k)}$ and explain the role of $\rho(B)$

Reading sources: NLA §§3.1–3.2, Appendix B.1 and §4.2; Bastian's §§3, 5–10 and 13–14; Scheichl–Zech's §§2–3 and 6.3.

Week 3 - Connect the foundations to eigensolvers and your seminar

Goal: Build a conceptual understanding of polynomial-filtered subspace iteration.

Session	Material to revise	Mastery check
9. Power and subspace iteration · 90 min	Power method, spectral gaps, invariant subspaces, orthogonal subspace iteration	Derive why the power method favors a dominant eigenvector and explain why subspace iteration requires orthonormalization
10. Matrix polynomials and Chebyshev filtering · 90 min	$p(A)$, spectral mapping for polynomials, Chebyshev recurrence, interval scaling	Explain why applying a polynomial to A can suppress unwanted eigenvector components
11. Rayleigh–Ritz and integration · 75 min	Q^TAQ, Ritz values/vectors, residual orthogonality, connection to filtered subspace iteration	Derive the Rayleigh–Ritz projection and describe a complete filtered eigensolver at a high level

Reading sources: NLA §§3.3, 5.1 and Appendix B.2; Scheichl–Zech's §4.10 for deeper Chebyshev approximation background.

The NLA script explicitly introduces polynomial filtering within vector iteration and later Rayleigh–Ritz approximation for large sparse eigenvalue problems.

Specific exercises to verify your pogress

Rather than completing a large number of routine exercises, a small nunmber that requires to combing computation with explanation

For examle five representative checkpoint problems

- **Checkpoint A - Linear algebra**

Given

$$A = \begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix},$$

determine $\ker(A)$, $\mathrm{im}(A)$, rank and nullity. Relate your results to invertibility.

- **Checkpoint B - Orthogonal projections**

Let

$$q = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Compute $P = qq^T$, prove that P is an orthogonal projector, and use it to project $(1, 0)^T$.

- **Checkpoint C - Spectral theorem**

Let

$$A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}.$$

Determine its eigenvalues and an orthonormal eigenbasis. Express A^k in terms of its eigendecomposition.

- **Checkpoint D - Numerical Conditioning**

Consider

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 10^{-4} \end{pmatrix}.$$

Calculate $\kappa_2(A)$. Explain how small perturbations in the second component of b can affect the solution of $Ax = b$.

- **Checkpoint D - Numerical Conditioning**

Consider

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 10^{-4} \end{pmatrix}.$$

Calculate $\kappa_2(A)$. Explain how small perturbations in the second component of b can affect the solution of $Ax = b$.

- **Checkpoint E - Polynomial Filtering**

Let

$$A = \mathrm{diag}(1, 2, 3), \qquad p(t) = (t-1)(t-3).$$

Determine $p(A)$. Explain how repeated application of $p(A)$ affects each eigenspace.

The first four checkpoints test prerequisites. The last one connects directly to the polynomial-filtering example in NLA §3.3.

1.5 Structural Linear Algebra Review

Reviewing linear algebra from a higher, structural level:

- **Module A - Structure of linear maps**: Vector spaces, bases, kernels, images, rank-nullity, direct sums, quotients, isomorphism theorems
- **Module B - Duality and linear measurements**: Dual bases, dual maps, anhiliators, transpose, quotient duality, exact sequences
- **Module C - Inner-product Geometry**: Riesz representation, adjoints, orthogonal projections, SPD matrices, Gram matrices, least squares
- **Module D - Spectral theory**: Invariant subspaces, spectral theorem, Rayleigh quotients, Rayleigh-Ritz, singular values, spectral projections
- **Module E - Numerical Methods**: Power iterations, subspace iteration, Chebyshev filtering ,residulas, conditioning, numerical stability
- **Module F1 - Functional Analysis**: normed and hilbert spaces, bounded operators, infinite dimensional spectral theory
- **Module F2 - Structural Algebra**: Universal properties, exact sequences, functors, categorical duality

1.5.1 What mastery should mean

For each major concept, I would require five kinds of understanding. Level Question you should be able to answer Definition What is the object, precisely? Structure and proof Why does it have the stated properties? Concrete computation How do I construct or calculate it for an example? Geometry and intuition What does it mean, and what does it preserve or forget? Application Where does this structure actually appear in an algorithm or mathematical model?

Take the adjoint as an example.

At the definitional level, you know that

$$\langle Tx, y\rangle = \langle x, T^*y\rangle.$$

At the structural level, you derive its existence from Riesz representation.

At the computational level, you calculate A^T or, under weighted inner products, $G_V^{-1}A^TG_W$.

At the geometric level, you understand that the adjoint pulls output measurements backward.

At the application level, you understand why A^* appears in the normal equations, matrix-free eigenvalue algorithms, and variational problems.

I would consider a concept well understood only when you can move between these perspectives without relying exclusively on a memorized formula.

This approach is especially appropriate for the kind of questions you've been asking.

References

Part I

Review I

2 Session 1 — Vector Spaces and Linear Maps

The central idea of this session is that **a linear map acts on vectors, while a matrix describes that action in chosen coordinates**. Bases connect these two viewpoints. Kernel, image, rank, and nullity describe what the map loses and what it can produce.

Allow approximately **35–40 minutes for the notes and worked examples**, followed by **20–25 minutes for the exercises**. The aim is to reconstruct the reasoning, not simply read the formulas.

Throughout, vector spaces are finite-dimensional over $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$, and coordinate vectors are columns.

2.1 Subspaces and span

A **vector space** is a set whose elements can be added and multiplied by scalars, obeying the usual distributive and associative laws, with a zero vector and additive inverses. Examples include $\mathbb{R}^n$, $\mathbb{C}^n$, and the space

$$\mathcal{P}_2 = \{a + bt + ct^2 : a, b, c \in \mathbb{F}\}$$

of polynomials of degree at most two.

The important point is that a "vector" need not be an arrow: it can be a polynomial, a function, or a matrix.

2.1.1 Subspaces

A subset $U \subseteq V$ is a **subspace** if it is itself a vector space under the operations inherited from V.

To check this, it suffices to verify

$$0 \in U, \qquad \alpha u + \beta v \in U \quad \text{for all } u, v \in U,\ \alpha, \beta \in \mathbb{F}.$$

Thus a subspace is closed under linear combinations. Over $\mathbb{R}$, lines and planes through the origin are familiar examples.

2.1.2 Span

The **span** of vectors $v_1, \dots, v_k$ is

$$\operatorname{span}\{v_1, \dots, v_k\} = \left\{ \sum_{j=1}^{k} c_j v_j : c_j \in \mathbb{F} \right\}.$$

It consists of all vectors obtainable by combining the given vectors. It is the smallest subspace containing them: any subspace containing the generators must also contain all their linear combinations.

2.1.3 Worked example: generators and constraints

Let

$$u = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \qquad v = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, \qquad w = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}.$$

Because $w = u + v$,

$$\operatorname{span}\{u, v, w\} = \operatorname{span}\{u, v\}.$$

Moreover,

$$\alpha u + \beta v = \begin{pmatrix} \alpha \\ \beta \\ \alpha + \beta \end{pmatrix},$$

so this span is exactly

$$U = \left\{ x \in \mathbb{R}^3 : x_3 = x_1 + x_2 \right\}.$$

We have described the same plane in two ways:

- **By generators:** $U = \operatorname{span}\{u, v\}$.
- **By a homogeneous constraint:** $x_3 - x_1 - x_2 = 0$.

Replacing the constraint by $x_3 - x_1 - x_2 = 1$ gives a translated plane. It is not a subspace because it does not contain zero.

2.2 Independence, bases, and dimension

Span answers: **Which vectors can we represent?**

Linear independence answers: **Is the representation unique?**

2.2.1 Linear independence

A family $v_1, \dots, v_k$ is **linearly independent** if

$$\sum_{j=1}^{k} c_j v_j = 0 \quad \Longrightarrow \quad c_1 = \cdots = c_k = 0.$$

Otherwise it is **linearly dependent**.

A nontrivial dependence relation allows at least one vector to be expressed using the others. Indeed, if $c_\ell \neq 0$, then

$$v_\ell = -\sum_{j \neq \ell} \frac{c_j}{c_\ell} v_j.$$

Thus dependence means redundancy. It does not require two of the vectors to be parallel.

2.2.2 Bases and coordinates

An ordered family

$$\mathcal{B} = (b_1, \dots, b_n)$$

is a **basis** of V if it is independent and spans V.

Every $x \in V$ then has a unique representation

$$x = \xi_1 b_1 + \cdots + \xi_n b_n.$$

Its coordinate vector relative to $\mathcal{B}$ is

$$[x]_{\mathcal{B}} = \begin{pmatrix} \xi_1 \\ \vdots \\ \xi_n \end{pmatrix}.$$

Why unique? If two coefficient lists represent x, subtracting the representations gives a zero linear combination of the basis vectors. Independence forces the corresponding coefficients to agree.

Consequently:

> **Spanning guarantees existence of coordinates; independence guarantees uniqueness.**

In the preceding example, u, v are independent because

$$\alpha u + \beta v = 0$$

forces $\alpha = \beta = 0$ from the first two components. Hence (u, v) is a basis of U, and

$$\left[\begin{pmatrix} a \\ b \\ a+b \end{pmatrix} \right]_{(u,v)} = \begin{pmatrix} a \\ b \end{pmatrix}.$$

The vector has three ambient components but only two coordinates within the plane.

2.2.3 Dimension

Theorem. Every basis of a finite-dimensional vector space has the same number of elements. This number is its **dimension**.

Two related facts are particularly useful:

1. Every independent family can be extended to a basis.
2. Every finite spanning family contains a basis.

For the first, repeatedly append a vector outside the current span. For the second, repeatedly remove a redundant vector. The underlying counting principle is that an independent family cannot contain more vectors than a spanning family; applying this to two bases in both directions proves that their lengths agree.

Therefore, in a space of dimension n:

- Any n independent vectors form a basis.
- Any n spanning vectors form a basis.
- More than n vectors are dependent.
- Fewer than n vectors cannot span.

The zero space $\{0\}$ has dimension zero and the empty basis. The family consisting of the zero vector is not independent.

2.2.4 A basis stored as a matrix

For vectors $b_j \in \mathbb{F}^m$, write

$$B = \begin{pmatrix} b_1 & \cdots & b_k \end{pmatrix} \in \mathbb{F}^{m \times k}.$$

Then

$$Bc = \sum_{j=1}^{k} c_j b_j.$$

Thus matrix–vector multiplication constructs a vector from its coefficients.

If the columns are independent, they form a basis of their column span, and

$$x = B[x]_{\mathcal{B}} \qquad (x \in \operatorname{span}\{b_1, \dots, b_k\}).$$

A basis matrix for a subspace can be rectangular. It need not possess a two-sided inverse.

2.3 Linear maps and their matrix representations

A map $T : V \to W$ is **linear** if

$$T(\alpha x + \beta y) = \alpha T(x) + \beta T(y).$$

In particular, $T(0) = 0$.

Linearity implies that a map is completely determined by its values on a basis:

$$T\left(\sum_j \xi_j b_j\right) = \sum_j \xi_j T(b_j).$$

2.3.1 Constructing the representing matrix

Choose bases

$$\mathcal{B} = (b_1, \dots, b_n) \quad \text{of } V, \qquad \mathcal{C} = (c_1, \dots, c_m) \quad \text{of } W.$$

The matrix

$$A = [T]_{\mathcal{C} \leftarrow \mathcal{B}}$$

is defined by taking the coordinates of $T(b_j)$ as its j-th column:

$$A = \begin{pmatrix} [T(b_1)]_{\mathcal{C}} & \cdots & [T(b_n)]_{\mathcal{C}} \end{pmatrix}.$$

Then

$$\boxed{[T(x)]_{\mathcal{C}} = A[x]_{\mathcal{B}}.}$$

This is the fundamental relationship:

The map takes vectors to vectors; its matrix takes input coordinates to output coordinates.

In standard bases, Ae_j is simply the j-th column of A.

2.3.2 Worked example: differentiation

Consider

$$D : \mathcal{P}_2 \longrightarrow \mathcal{P}_1, \qquad D(p) = p'.$$

Choose

$$\mathcal{B} = (1, t, t^2), \qquad \mathcal{C} = (1, t).$$

Since

$$D(1) = 0, \qquad D(t) = 1, \qquad D(t^2) = 2t,$$

we obtain

$$[D]_{\mathcal{C} \leftarrow \mathcal{B}} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}.$$

Indeed,

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} b \\ 2c \end{pmatrix},$$

which represents the derivative of $a + bt + ct^2$.

The matrix has three columns because the domain has dimension three, and two rows because the codomain has dimension two.

2.4 Kernel, image, and solvability

For a linear map $T : V \to W$, define

$$\ker T = \{x \in V : T(x) = 0\},$$

and

$$\operatorname{im} T = \operatorname{ran} T = \{T(x) : x \in V\}.$$

The **kernel** consists of input directions erased by the map. The **image**, also called the **range**, consists of attainable outputs.

Both are subspaces. For example, if $u, v \in \ker T$, then

$$T(\alpha u + \beta v) = \alpha T(u) + \beta T(v) = 0.$$

For the image,

$$\alpha T(u) + \beta T(v) = T(\alpha u + \beta v)$$

is again an attainable output.

Define

$$\operatorname{nullity} T = \dim \ker T, \qquad \operatorname{rank} T = \dim \operatorname{im} T.$$

For $A \in \mathbb{F}^{m\times n}$, regarded as the map $x \mapsto Ax$,

$$\ker A \subseteq \mathbb{F}^n, \qquad \operatorname{im} A \subseteq \mathbb{F}^m.$$

Since $Ax = \sum_j x_j a_j$,

$$\operatorname{im} A = \operatorname{span}\{a_1, \dots, a_n\}.$$

2.4.1 Solvability and all solutions

The equation $T(x) = b$ is solvable exactly when

$$b \in \operatorname{im} T.$$

If x_p is one solution, then

$$\boxed{\{x : T(x) = b\} = x_p + \ker T.}$$

Proof.

$$T(x) = b = T(x_p) \iff T(x - x_p) = 0 \iff x - x_p \in \ker T.$$

The kernel therefore describes the ambiguity in reconstructing an input from an output. A nonempty solution set is a translate of the kernel; for $b \neq 0$, it is not a subspace.

2.4.2 Injectivity and surjectivity

A map is **injective** if equal outputs imply equal inputs. Because

$$T(x) = T(y) \iff x - y \in \ker T,$$

we have

$$\boxed{T \text{ injective} \iff \ker T = \{0\}.}$$

A map is **surjective** if every vector in its codomain is attained:

$$\boxed{T \text{ surjective} \iff \operatorname{im} T = W.}$$

For differentiation,

$$\ker D = \operatorname{span}\{1\}, \qquad \operatorname{im} D = \mathcal{P}_1.$$

Thus $D : \mathcal{P}_2 \to \mathcal{P}_1$ is surjective but not injective: differentiation loses the constant term.

For instance,

$$p' = 2 + 6t \quad \Longleftrightarrow \quad p(t) = 2t + 3t^2 + c.$$

2.4.3 Worked example: a complete rectangular system

Consider

$$A = \begin{pmatrix} 1 & 2 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 2 & 1 & 2 \end{pmatrix}, \qquad A : \mathbb{R}^4 \to \mathbb{R}^3.$$

Subtracting rows 1 and 2 from row 3 gives

$$R = \begin{pmatrix} 1 & 2 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

The pivots lie in columns 1 and 3. Variables x_2, x_4 are free.

Kernel. Set $x_2 = s$, $x_4 = t$. The homogeneous equations give

$$x_1 = -2s - t, \qquad x_3 = -t.$$

Hence

$$x = s \begin{pmatrix} -2 \\ 1 \\ 0 \\ 0 \end{pmatrix} + t \begin{pmatrix} -1 \\ 0 \\ -1 \\ 1 \end{pmatrix}.$$

These vectors span the kernel and are independent: their second and fourth components recover s, t. Therefore

$$\ker A = \operatorname{span} \left\{ \begin{pmatrix} -2 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \\ -1 \\ 1 \end{pmatrix} \right\}, \qquad \text{nullity } A = 2.$$

Image. Take the pivot columns of the **original matrix**:

$$a_1 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \qquad a_3 = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}.$$

They are independent, and

$$a_2 = 2a_1, \qquad a_4 = a_1 + a_3.$$

Thus

$$\operatorname{im} A = \operatorname{span}\{a_1, a_3\} = \{b \in \mathbb{R}^3 : b_3 = b_1 + b_2\}, \qquad \operatorname{rank} A = 2.$$

Solvability. Applying the row operation to the augmented matrix gives

$$\left(\begin{array}{cccc|c} 1 & 2 & 0 & 1 & b_1 \\ 0 & 0 & 1 & 1 & b_2 \\ 0 & 0 & 0 & 0 & b_3 - b_1 - b_2 \end{array}\right).$$

Therefore,

$$Ax = b \text{ is solvable } \iff b_3 = b_1 + b_2.$$

For $b = (3, 2, 5)^T$, one solution is $x_p = (3, 0, 2, 0)^T$, and every solution is

$$x = \begin{pmatrix} 3 \\ 0 \\ 2 \\ 0 \end{pmatrix} + s \begin{pmatrix} -2 \\ 1 \\ 0 \\ 0 \end{pmatrix} + t \begin{pmatrix} -1 \\ 0 \\ -1 \\ 1 \end{pmatrix}.$$

For $b = (3, 2, 6)^T$, no solution exists.

2.4.4 Why use the original pivot columns?

Row operations produce $R = EA$, where E is invertible. Consequently,

$$Rx = 0 \iff Ax = 0,$$

so the kernel is unchanged.

They also preserve relations among columns:

$$\sum_j c_j E a_j = 0 \iff \sum_j c_j a_j = 0.$$

Thus pivot-column indices identify independent spanning columns in the original matrix.

However,

$$\operatorname{im} R = E(\operatorname{im} A),$$

which is generally a different subspace. Here $\operatorname{im} R$ is the plane $b_3 = 0$, while $\operatorname{im} A$ is the plane $b_3 = b_1 + b_2$.

2.5 The rank–nullity theorem

Theorem. For a linear map $T : V \to W$,

$$\boxed{\dim \ker T + \dim \operatorname{im} T = \dim V.}$$

2.5.1 Proof by extending a kernel basis

Let $k_1, \dots, k_q$ be a basis of $\ker T$. Extend it to a basis of V:

$$(k_1, \dots, k_q, v_1, \dots, v_{n-q}), \qquad n = \dim V.$$

We show that

$$T(v_1), \dots, T(v_{n-q})$$

form a basis of the image.

They span the image. Every input can be expanded in the extended basis. Applying T annihilates all kernel terms, leaving a combination of the $T(v_j)$.

They are independent. Suppose

$$\sum_j \alpha_j T(v_j) = 0.$$

Then

$$\sum_j \alpha_j v_j \in \ker T,$$

so it equals some $\sum_i \beta_i k_i$. Therefore

$$\sum_j \alpha_j v_j - \sum_i \beta_i k_i = 0.$$

Independence of the extended basis forces every coefficient to vanish.

Hence

$$\dim \operatorname{im} T = n - q,$$

which proves the theorem.

2.5.2 Interpretation and consequences

There are q independent input directions that the map erases. The remaining $n - q$ basis directions produce independent outputs.

This is a dimension statement. It does not say that kernel and image belong to the same ambient space or are orthogonal.

For the preceding 3×4 example,

$$\operatorname{rank} A + \operatorname{nullity} A = 2 + 2 = 4,$$

the number of **columns**, which is the domain dimension.

For $A \in \mathbb{F}^{m \times n}$, with rank r:

Property	Equivalent condition	Meaning for $Ax = b$
Injective	$r = n$, full column rank	At most one solution for each b
Surjective onto $\mathbb{F}^m$	$r = m$, full row rank	At least one solution for every b
Bijective	$m = n = r$	Exactly one solution for every b

Since $r \leq \min(m, n)$, a wide matrix cannot be injective, and a tall matrix cannot be surjective onto its full codomain.

For a square matrix,

$$\text{injective} \iff \text{surjective} \iff \text{invertible}.$$

A consistent system has $n - r$ free parameters.

2.6 Change of basis

The vector stays the same when we change its basis; its coordinate vector changes.

Let $\mathcal{B}$ and $\widetilde{\mathcal{B}}$ be bases of the same space. Define

$$S = \begin{pmatrix} [\tilde{b}_1]_{\mathcal{B}} & \cdots & [\tilde{b}_n]_{\mathcal{B}} \end{pmatrix}.$$

Expanding a vector in the new basis and then expressing those basis vectors in the old basis gives

$$\boxed{[x]_{\mathcal{B}} = S[x]_{\widetilde{\mathcal{B}}}.}$$

Thus:

- S converts **new coordinates to old coordinates**.
- S^{-1} converts **old coordinates to new coordinates**.

Its columns are new basis vectors expressed in old coordinates.

2.6.1 Changing a map's matrix

Suppose

$$A = [T]_{\mathcal{C} \leftarrow \mathcal{B}},$$

and the new input and output bases satisfy

$$[x]_{\mathcal{B}} = S[x]_{\widetilde{\mathcal{B}}}, \qquad [y]_{\mathcal{C}} = P[y]_{\widetilde{\mathcal{C}}}.$$

Then

$$[T(x)]_{\widetilde{\mathcal{C}}} = P^{-1}[T(x)]_{\mathcal{C}} = P^{-1}AS[x]_{\widetilde{\mathcal{B}}}.$$

Therefore,

$$\boxed{\widetilde{A} = P^{-1}AS.}$$

Read the product from right to left:

1. Convert input coordinates to the old basis.
2. Apply the old representing matrix.
3. Convert output coordinates to the new basis.

For an endomorphism $T : V \to V$, if we use the same basis change for input and output, $P = S$, giving

$$\boxed{\widetilde{A} = S^{-1}AS.}$$

This is a **similarity transformation**.

2.6.2 Worked example: coordinates reveal a simpler action

In standard coordinates, let

$$T(x) = Ax, \qquad A = \begin{pmatrix} 2 & 1 \\ 0 & 1 \end{pmatrix}.$$

Choose

$$b_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad b_2 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}.$$

The basis matrix and its inverse are

$$S = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}, \qquad S^{-1} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

Hence

$$S^{-1}AS = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}.$$

We can see this directly:

$$T(b_1) = 2b_1, \qquad T(b_2) = b_2.$$

These images have new coordinates $(2,0)^T$ and $(0,1)^T$, respectively.

For $x = (3,1)^T$,

$$[x]_{\mathcal{B}} = S^{-1}x = \begin{pmatrix} 4 \\ 1 \end{pmatrix}.$$

Applying the new matrix gives

$$[T(x)]_{\mathcal{B}} = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 4 \\ 1 \end{pmatrix} = \begin{pmatrix} 8 \\ 1 \end{pmatrix}.$$

Converting back,

$$T(x) = S \begin{pmatrix} 8 \\ 1 \end{pmatrix} = \begin{pmatrix} 7 \\ 1 \end{pmatrix} = Ax.$$

The map has not changed; its action is simpler to describe in the new basis.

Invertible coordinate changes preserve rank and nullity. The coordinate descriptions of kernel and image transform according to

$$\ker(P^{-1}AS) = S^{-1}(\ker A), \qquad \operatorname{im}(P^{-1}AS) = P^{-1}(\operatorname{im} A).$$

2.7 Connections to numerical linear algebra

Let $V \in \mathbb{F}^{n\times k}$ have independent columns spanning a subspace U. If $M \in \mathbb{F}^{k\times k}$ is invertible, then

$$\operatorname{im}(VM) = \operatorname{im} V,$$

because Mc ranges over all of $\mathbb{F}^k$.

This distinction between a **subspace** and a **basis representing it** is fundamental.

Orthogonalization and QR. A thin QR factorization

$$V = QR$$

replaces independent columns of V by orthonormal columns of Q. Since R is invertible, the span is preserved. Moreover,

$$x = Vc = Q(Rc),$$

so R converts V-basis coordinates into Q-basis coordinates. Sessions 2 and 4 develop the geometry and algorithms.

Krylov subspaces.

$$\mathcal{K}_k(A, v) = \text{span}\{v, Av, \dots, A^{k-1}v\}.$$

There are k generators, but their span may have smaller dimension if they are dependent. Counting generated vectors is not the same as counting independent directions.

Subspace iteration and filtering. Applying A, or a polynomial

$$p(A) = \sum_{j=0}^{d} \alpha_j A^j,$$

to a basis matrix produces the image of its subspace under that map. This may change the subspace and lower its dimension. Subsequent orthogonalization preserves the resulting span; it cannot recover directions erased by the map.

Invariant subspaces and projection methods. A subspace U is invariant under A if

$$AU \subseteq U.$$

If V is a basis matrix of U, this is equivalent to

$$AV = VH$$

for a unique $k \times k$ matrix H. Its columns contain the coordinates of the vectors Av_j within U. Thus H represents the restriction of A to U. If U is not invariant, projection methods instead represent a projected action, studied later.

Finally, our calculations used **exact rank**. In floating-point arithmetic, nearly dependent columns can be difficult to distinguish from dependent ones. For example,

$$\begin{pmatrix} 1 & 0 \\ 0 & \varepsilon \end{pmatrix}$$

has exact rank two whenever $\varepsilon \neq 0$, however small. Numerical rank requires a tolerance and attention to scale.

2.8 Independent exercises

Attempt these before opening the solutions below.

2.8.1 Exercise 1 — Subspaces and coordinates

Let

$$U = \{(a, b, a+b)^T : a, b \in \mathbb{R}\}, \qquad F = \{(a, b, a+b+1)^T : a, b \in \mathbb{R}\}.$$

Justify that U is a subspace and F is not. Show that

$$\mathcal{B} = ((1, 1, 2)^T, (1, -1, 0)^T)$$

is a basis of U, and find $[(3, 1, 4)^T]_{\mathcal{B}}$.

2.8.2 Exercise 2 — A rectangular system

For

$$B = \begin{pmatrix} 1 & 1 & 0 & 2 \\ 0 & 1 & 1 & 1 \\ 1 & 2 & 1 & 3 \end{pmatrix},$$

find bases of its kernel and image, compute rank and nullity, and verify rank–nullity.

Characterize all $b \in \mathbb{R}^3$ for which $Bx = b$ is solvable, and find every solution for $b = (2, 1, 3)^T$.

Is this map injective or surjective? Explain why columns of its row-reduced form cannot automatically serve as an image basis of B.

2.8.3 Exercise 3 — Changing coordinates

Let

$$T(x_1, x_2)^T = (x_1 + 2x_2, 3x_2)^T.$$

Use the basis

$$\mathcal{B} = ((1, 0)^T, (1, 1)^T)$$

for both domain and codomain.

Compute the representing matrix both directly from basis images and by similarity. Verify your answer on $x = (0, 1)^T$.

2.8.4 Exercise 4 — Losing information in two stages

Let $T : V \to W$ and $L : W \to Z$ be linear. Prove

$$\ker T \subseteq \ker(LT), \qquad \operatorname{rank}(LT) \leq \min(\operatorname{rank} T, \operatorname{rank} L).$$

Explain how composition can lose directions, and give an example where $T, L \neq 0$ but $LT = 0$.

2.8.5 Exercise 5 — A map on polynomials

Define

$$E : \mathcal{P}_2 \to \mathbb{R}^2, \qquad E(p) = (p(0), p(1))^T.$$

Find its matrix in the input basis $(1, t, t^2)$ and the standard output basis. Determine its kernel, image, rank, and nullity, and describe every polynomial satisfying $p(0) = a$, $p(1) = b$.

Finally, use the input basis

$$(1 - t,\ t,\ t(t-1)).$$

Find the new matrix directly and explain why this basis is adapted to the interpolation problem.

2.9 Solutions — consult after attempting the exercises

2.9.1 Solution 1

The zero vector belongs to U, and linear combinations preserve the condition $x_3 = x_1 + x_2$. Thus U is a subspace. The zero vector does not belong to F, so F is not.

For $b_1 = (1, 1, 2)^T$, $b_2 = (1, -1, 0)^T$,

$$c_1 b_1 + c_2 b_2 = (c_1 + c_2, c_1 - c_2, 2c_1)^T.$$

Every $(a, b, a + b)^T \in U$ has the unique coefficients

$$c_1 = \frac{a+b}{2}, \qquad c_2 = \frac{a-b}{2}.$$

Therefore (b_1, b_2) is a basis, and

$$[(3, 1, 4)^T]_{\mathcal{B}} = (2, 1)^T.$$

2.9.2 Solution 2

Row reduction gives

$$B \longrightarrow R = \begin{pmatrix} 1 & 0 & -1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

The homogeneous equations are

$$x_1 = x_3 - x_4, \qquad x_2 = -x_3 - x_4.$$

Thus a kernel basis is

$$\left\{ \begin{pmatrix} 1 \\ -1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \\ 0 \\ 1 \end{pmatrix} \right\}.$$

The third and fourth components establish independence.

The original pivot columns

$$c_1 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \qquad c_2 = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}$$

form an image basis: they are independent, and the other columns satisfy

$$c_3 = c_2 - c_1, \qquad c_4 = c_1 + c_2.$$

Hence rank and nullity are both two, and their sum is four.

The image is

$$\{b : b_3 = b_1 + b_2\}.$$

Indeed,

$$(b_1, b_2, b_1 + b_2)^T = (b_1 - b_2)c_1 + b_2 c_2.$$

For $b = (2, 1, 3)^T$, all solutions are

$$x = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} + s \begin{pmatrix} 1 \\ -1 \\ 1 \\ 0 \end{pmatrix} + t \begin{pmatrix} -1 \\ -1 \\ 0 \\ 1 \end{pmatrix}.$$

The map is neither injective nor surjective onto $\mathbb{R}^3$. Its kernel is nontrivial, and its image has dimension two.

The reduced matrix has image $\{y : y_3 = 0\}$, different from the original image. Row operations preserve column dependence relations, but generally move the column space.

2.9.3 Solution 3

Directly,

$$T(1,0)^T = (1,0)^T = b_1, \qquad T(1,1)^T = (3,3)^T = 3b_2.$$

Therefore

$$[T]_{\mathcal{B}\leftarrow\mathcal{B}} = \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix}.$$

Alternatively,

$$A = \begin{pmatrix} 1 & 2 \\ 0 & 3 \end{pmatrix}, \quad S = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \quad S^{-1} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix},$$

and multiplication gives

$$S^{-1}AS = \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix}.$$

For $x = (0,1)^T$,

$$[x]_{\mathcal{B}} = (-1,1)^T, \qquad [T(x)]_{\mathcal{B}} = (-1,3)^T.$$

Converting back yields

$$S(-1,3)^T = (2,3)^T = Ax.$$

2.9.4 Solution 4

If $x \in \ker T$, then $L(Tx) = L(0) = 0$. Thus

$$\ker T \subseteq \ker(LT).$$

Also,

$$\operatorname{im}(LT) = L(\operatorname{im} T) \subseteq \operatorname{im} L,$$

so $\operatorname{rank}(LT) \leq \operatorname{rank} L$.

If $w_1, \ldots, w_r$ is an image basis of T, then

$$L(w_1), \ldots, L(w_r)$$

span the image of LT. Its dimension is therefore at most $r = \operatorname{rank} T$.

For example,

$$T = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \qquad L = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

are both nonzero, but $LT = 0$. The first map's entire image lies in the second map's kernel.

2.9.5 Solution 5

The basis images are

$$E(1) = (1,1)^T, \qquad E(t) = (0,1)^T, \qquad E(t^2) = (0,1)^T,$$

so

$$[E] = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}.$$

For $p = c_0 + c_1 t + c_2 t^2$, the kernel conditions give

$$c_0 = 0, \qquad c_1 + c_2 = 0.$$

Hence

$$\ker E = \operatorname{span}\{t^2 - t\}.$$

The first two matrix columns are independent, so

$$\operatorname{im} E = \mathbb{R}^2, \qquad \operatorname{rank} E = 2, \qquad \operatorname{nullity} E = 1.$$

One polynomial with the prescribed values is $a(1-t) + bt$. Every solution is therefore

$$\boxed{p(t) = a(1-t) + bt + c\,t(t-1), \qquad c \in \mathbb{R}.}$$

The new input basis has images

$$E(1-t) = (1,0)^T, \quad E(t) = (0,1)^T, \quad E(t(t-1)) = 0,$$

giving

$$[E]_{\text{standard} \leftarrow \widetilde{\mathcal{B}}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}.$$

To check that it is a basis, evaluate a vanishing linear combination at 0 and 1, forcing its first two coefficients to vanish; the third then vanishes because $t(t-1)$ is a nonzero polynomial.

This basis separates the two prescribed values from the one undetermined direction. Only the input basis changed, so this is not a similarity transformation.

2.10 End-of-session mastery checklist

Without consulting the notes, you should be able to:

- State the subspace test and distinguish a subspace from a translated solution set.
- Explain how spanning and independence provide existence and uniqueness of coordinates.
- Distinguish a vector, its coordinate vector, and a basis matrix.
- Construct a map's matrix from the images of basis vectors.
- Find a kernel basis using free variables and an image basis using original pivot columns.
- Characterize solvability by $b \in \operatorname{im} A$ and all solutions by $x_p + \ker A$.
- Prove rank–nullity by extending a kernel basis.
- Connect rank and nullity to injectivity, surjectivity, and invertibility.
- Derive $\widetilde{A} = P^{-1}AS$ and explain when it becomes similarity.
- Explain why changing a basis preserves a subspace, while applying a linear map may change its dimension.

Ready for Session 2: you can complete Exercises 2 and 3 independently, reconstruct the rank–nullity proof, and explain why a basis need not be orthogonal. Session 2 will add lengths, angles, and orthogonality to this algebraic foundation.

3 Session 2 — Inner Products and Orthogonality

In Session 1, the plane $U = \{(s, t, s + t)^T : s, t \in \mathbb{R}\}$ had a basis, and each $x \in U$ had unique coordinates. But how much of x lies along a given basis vector? With an arbitrary basis, a coordinate need not answer that question. An inner product lets us measure components. Choosing an orthonormal basis makes those measurements *equal* to the coordinates.

Allow **45–50 minutes** for the text and worked examples and **25–30 minutes** for the exercises and recall. The mastery task is to start with independent columns, build an orthonormal basis of their span by hand, and justify each subtraction and division. Throughout, spaces are finite-dimensional over $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$, and coordinate vectors are columns.

3.1 From coordinates to geometry

A vector space specifies which linear combinations can be formed. It takes additional structure to measure their lengths or say that two directions meet at a right angle. Following Kanschat and Scheichl, Appendix A.2, we take the inner product **linear in the first argument**. For all $x, y, z \in V$ and $\alpha, \beta \in \mathbb{F}$, its axioms are

$$\begin{aligned} \langle \alpha x + \beta y, z\rangle &= \alpha\langle x, z\rangle + \beta\langle y, z\rangle, \\ \langle x, y\rangle &= \overline{\langle y, x\rangle}, \\ \langle x, x\rangle &\geq 0, \qquad \langle x, x\rangle = 0 \iff x = 0. \end{aligned}$$

Conjugate symmetry gives conjugate linearity in the second argument and ensures $\langle x, x\rangle$ is real. In real spaces conjugation disappears. The standard Euclidean and Hermitian products are respectively $y^T x$ and $y^* x = \sum_j x_j \overline{y_j}$, where $y^* = \overline{y}^T$. For instance, $x = (1, i)^T$ has $x^* x = 2$, even though $x^T x = 0$.

The induced length is $\|x\| = \sqrt{\langle x, x\rangle}$. Two vectors are **orthogonal**, written $x \perp y$, if $\langle x, y\rangle = 0$. Geometry depends on the chosen product. In $\mathbb{R}^2$, $u = (1, 1)^T$ and $v = (1, -1)^T$ are Euclidean-orthogonal. With $W = \operatorname{diag}(4, 1)$, the product $\langle x, y\rangle_W = y^T W x$ gives $\langle u, v\rangle_W = 3$. Its unit circle has equation $4x_1^2 + x_2^2 = 1$ because displacement along the first axis costs twice as much in length. Orthogonality has changed while spans and linear independence are exactly as before.

3.1.1 One subtraction gives Cauchy–Schwarz

For $y \neq 0$, seek a scalar c making $r = x - cy$ perpendicular to y. Linearity in the first slot gives

$$0 = \langle r, y\rangle = \langle x, y\rangle - c\|y\|^2, \qquad c = \frac{\langle x, y\rangle}{\|y\|^2}.$$

Since $x = cy + r$ and both cross terms vanish, expansion yields

$$\|x\|^2 = |c|^2\|y\|^2 + \|r\|^2 \geq \frac{|\langle x, y\rangle|^2}{\|y\|^2}.$$

Taking square roots proves **Cauchy–Schwarz**, $|\langle x, y\rangle| \leq \|x\|\,\|y\|$. Equality holds exactly when x, y are linearly dependent (including the case $y = 0$). Remember the residual r: Gram–Schmidt repeats this construction.

For real nonzero vectors the quotient $\langle x, y\rangle/(\|x\|\|y\|)$ thus lies in $[-1, 1]$ and defines the cosine of their angle. In complex space the quotient may have a phase. Its absolute value gives the cosine of the acute or right angle between the corresponding complex lines, allowing angle zero for coincident lines.

In either field,

$$\|x + y\|^2 = \|x\|^2 + 2\,\mathrm{Re}\langle x, y\rangle + \|y\|^2.$$

Cauchy–Schwarz bounds the right side by $(\|x\| + \|y\|)^2$, giving the triangle inequality. Together with positivity and $\|\alpha x\| = |\alpha|\|x\|$, this verifies that the induced length is a norm. For orthogonal vectors the middle term vanishes, giving **Pythagoras**. Over $\mathbb{R}$ the identity also implies orthogonality. Over $\mathbb{C}$ it only forces the real part to vanish, as 1 and i illustrate.

3.2 Orthogonal complements turn span questions into kernel questions

An **orthogonal family** has $u_i \perp u_j$ for $i \neq j$. An **orthonormal family** also has $\|u_i\| = 1$, so $\langle u_i, u_j\rangle = \delta_{ij}$ (one for equal indices and zero otherwise). Any orthogonal family of nonzero vectors is independent: taking the inner product of $\sum_i c_i u_i = 0$ with u_j gives $c_j\|u_j\|^2 = 0$ for every j.

For a subspace $U \subseteq V$, define

$$U^{\perp} = \{z \in V : \langle z, u\rangle = 0 \text{ for every } u \in U\}.$$

Linearity makes this a subspace. Testing a spanning family suffices because the inner product with any linear combination of its members also vanishes. In the standard geometry, if $U =$

$\operatorname{im} B$, this turns the geometric definition into the Session 1 operation of solving a homogeneous system:

$$z \in U^\perp \iff B^* z = 0.$$

Indeed, each entry of $B^* z$ tests one column. For a map $A : \mathbb{F}^n \to \mathbb{F}^m$ the two corresponding identities are

$$(\operatorname{im} A)^\perp = \ker A^* \subseteq \mathbb{F}^m, \qquad (\operatorname{im} A^*)^\perp = \ker A \subseteq \mathbb{F}^n.$$

The second follows by applying the first to A^*. Abstractly, the adjoint satisfies $\langle Ax, y\rangle = \langle x, A^* y\rangle$. The conjugate-transpose formula assumes standard products in coordinates (or orthonormal bases). For example, with ambient product $y^* W x$, testing the columns of B gives $B^* W z = 0$.

The plane from Session 1. Take its natural spanning columns

$$a = (1, 0, 1)^T, \qquad b = (0, 1, 1)^T, \qquad B = [\, a \; b \,].$$

They are independent but $a^T b = 1$. A vector z perpendicular to the plane satisfies $z_1 + z_3 = z_2 + z_3 = 0$, hence

$$U^\perp = \operatorname{span}\{n\}, \qquad n = (-1, -1, 1)^T.$$

The equation of the plane itself is $n^T x = 0$. Thus the same plane is an image when built from spanning directions and a kernel when specified by its normal. In $\mathbb{R}^2$ the analogous calculation sends the line through $(s, t)^T \neq 0$ to the line through $(-t, s)^T$.

3.3 Orthonormal coordinates and orthogonal decomposition

Suppose $q_1, \ldots, q_k$ is an orthonormal basis of U. For $x = \sum_i c_i q_i$ in U, pairing with q_j isolates c_j. In standard coordinates, writing $Q = [\, q_1 \; \cdots \; q_k \,]$, we obtain

$$c_j = \langle x, q_j\rangle = q_j^* x, \qquad c = Q^* x, \qquad x = Qc.$$

Session 1 used B to assemble a vector from its coordinates. For an orthonormal basis, the adjoint reads those coordinates back: the (i, j) entry of $Q^* Q$ is $q_i^* q_j = \delta_{ij}$. Therefore $Q^* Q = I_k$ and $\|Qc\|_2^2 = c^* Q^* Q c = \|c\|_2^2$. A rectangular Q is an isometry from coordinate space onto U. A square Q additionally satisfies $QQ^* = I$ and is called **unitary**, or **orthogonal** over the reals. For $k < n$, a nonzero normal to U is sent to zero by Q^*, so Q^* cannot be a two-sided inverse.

What if x is anywhere in V? Form

$$u = \sum_{j=1}^{k} \langle x, q_j\rangle q_j, \qquad z = x - u.$$

Testing z against each q_i gives zero, so $u \in U$ and $z \in U^\perp$. The decomposition is unique: two such decompositions would give a difference in $U \cap U^\perp$, where a vector is orthogonal to itself and hence zero. We have proved

$$V = U \oplus U^\perp, \qquad \dim V = \dim U + \dim U^\perp.$$

The dimension count follows by concatenating bases of the two summands. The construction in the next section supplies the orthonormal basis used in this proof for every nonzero subspace (the zero subspace uses the empty basis).

Check in the running plane. For $x = (0, 0, 3)^T$, the coefficient along its normal n is $n^T x/(n^T n) = 1$. Thus $x = (1, 1, 2)^T + n$. The first term equals $a + b$ and is perpendicular to n. The squared lengths are $9 = 6 + 3$.

3.4 Gram–Schmidt from the residual equation

Given independent $a_1, \dots, a_k$, we want each new basis vector to contribute the direction still missing from the preceding span. Suppose $q_1, \dots, q_{j-1}$ already form an orthonormal basis of that span. Write $w_j = a_j - \sum_{i<j} c_i q_i$. Requiring $\langle w_j, q_\ell \rangle = 0$ gives $c_\ell = \langle a_j, q_\ell \rangle$. This determines **classical Gram–Schmidt**:

$$w_j = a_j - \sum_{i=1}^{j-1} \langle a_j, q_i \rangle q_i, \qquad q_j = \frac{w_j}{\|w_j\|}, \qquad j = 1, \dots, k.$$

The first sum is empty, so $w_1 = a_1$. Subtraction **orthogonalizes** the residual against the earlier vectors. Division **normalizes** its length. Division alone leaves any nonzero pairwise correlation in place.

Why the algorithm works. Each w_j belongs to the first j input vectors' span. It is nonzero, since $w_j = 0$ would express a_j in the preceding span. Testing against the earlier q_i proves orthogonality, and normalization yields unit length. Conversely, $a_j = \|w_j\| q_j + \sum_{i<j} \langle a_j, q_i \rangle q_i$. Starting with q_1, induction now gives

$$\operatorname{span}\{q_1, \dots, q_j\} = \operatorname{span}\{a_1, \dots, a_j\} \quad \text{at every step.}$$

For a dependent input column a zero residual signals that its entire direction is already represented. In exact arithmetic one can skip that column when constructing a basis of the image.

3.4.1 Worked construction in the running plane

Start with $a_1 = a = (1, 0, 1)^T$ and $a_2 = b = (0, 1, 1)^T$. The first direction and the coefficient of the second vector along it are

$$q_1 = \frac{(1, 0, 1)^T}{\sqrt{2}}, \qquad \langle b, q_1 \rangle = \frac{1}{\sqrt{2}}.$$

Subtracting the parallel component gives

$$w_2 = b - \tfrac{1}{2}(1, 0, 1)^T = \tfrac{1}{2}(-1, 2, 1)^T, \qquad \|w_2\| = \frac{\sqrt{6}}{2}, \qquad q_2 = \frac{(-1, 2, 1)^T}{\sqrt{6}}.$$

Viewed *inside the plane*, b is the diagonal of a right triangle. Its two legs are the amount already represented and the residual that supplies the new direction.

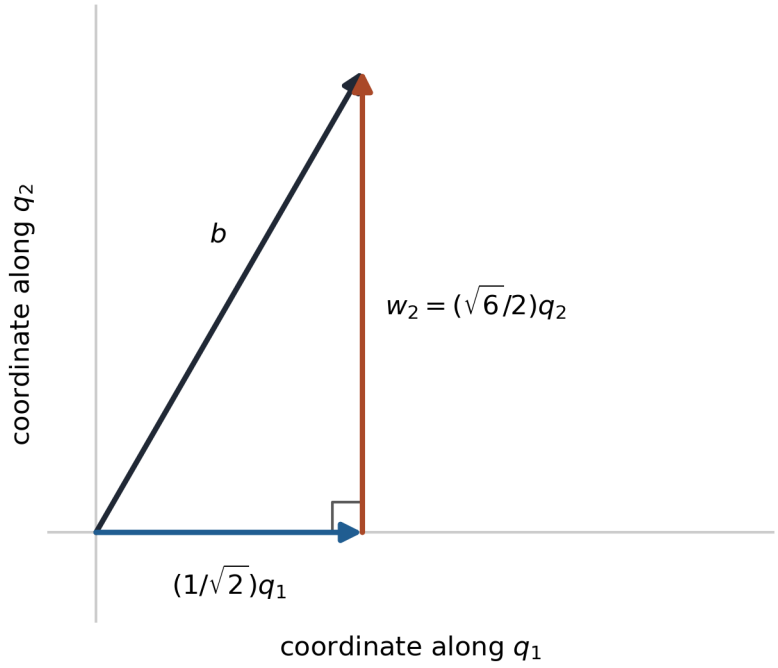


Figure 3.1: The running example in orthonormal coordinates within its plane. Subtracting the horizontal component of b leaves the vertical residual. Translating that residual to the origin gives the direction of the second basis vector.

The matrix check uses just three dot products:

$$Q = \begin{pmatrix} 1/\sqrt{2} & -1/\sqrt{6} \\ 0 & 2/\sqrt{6} \\ 1/\sqrt{2} & 1/\sqrt{6} \end{pmatrix}, \qquad Q^T Q = \begin{pmatrix} (1+1)/2 & (-1+1)/\sqrt{12} \\ (-1+1)/\sqrt{12} & (1+4+1)/6 \end{pmatrix} = I_2.$$

For $x = (3, 1, 4)^T = 3a + b$ from Session 1, $Q^T x = (7/\sqrt{2}, 3/\sqrt{6})^T$. The squared coordinate length is $49/2 + 9/6 = 26 = \|x\|_2^2$. The coefficients $(3, 1)^T$ in the original basis describe the same vector, but their squared length is only 10.

3.5 What changes in numerical algorithms?

Classical and modified Gram–Schmidt. For classical Gram–Schmidt each coefficient uses the original a_j. Modified Gram–Schmidt (MGS) initializes $w = a_j$ and, for each earlier q_i in order, computes $h = \langle w, q_i \rangle$ and updates $w \leftarrow w - hq_i$. It then normalizes the final w. In exact arithmetic, previously subtracted directions have zero inner product with q_i, so both algorithms give the same coefficients and residual. In floating point, cancellation can leave a residual whose rounding error is large relative to its length. Normalizing magnifies that error. MGS measures what is present in the *updated* vector and generally retains orthogonality better. Severe near-dependence can still require a second orthogonalization pass or a numerical rank decision. Monitoring $\|Q^*Q - I\|$ tests the property one intended to construct.

Why basis quality matters. Consider $B_\varepsilon = [\, e_1,\ e_1 + \varepsilon e_2 \,]$ for real $\varepsilon > 0$. It spans $\mathbb{R}^2$, yet e_2 requires coordinates $(-1/\varepsilon, 1/\varepsilon)^T$. Two large contributions nearly cancel. With an orthonormal Q, $\|Q\,\delta c\|_2 = \|\delta c\|_2$: every singular value of Q is one. This is a statement about the basis representation. A nearly dependent input may still make its inferred span sensitive to perturbations. Recording the coefficients that express each input in the successive q_i gives the triangular factor in QR (Session 4).

Why the seminar alternates filtering and orthonormalization. Power iteration normalizes one nonzero vector after multiplying by A. Subspace iteration applies A to a block and orthonormalizes its columns. Normalizing columns individually controls lengths but lets several columns align with the same dominant direction. For a Chebyshev filter p, an eigenvector v with eigenvalue λ satisfies $p(A)v = p(\lambda)v$. Thus each eigencomponent is rescaled and the trial subspace can move:

$$Y = p(A)Q, \qquad \operatorname{im} Y = p(A)(\operatorname{im} Q).$$

In contrast, if Y has independent columns, $Q_{\text{new}} = \operatorname{orth}(Y)$ has exactly the same image as Y in exact arithmetic. Filtering changes the balance of spectral components. Orthonormalization changes the coordinates used to represent the resulting subspace. If the filter annihilates a direction and reduces rank, orthogonalization alone cannot restore it.

Where the same residual appears. Arnoldi takes Aq_j, removes components along the current Krylov basis, and normalizes the remainder if nonzero. For Hermitian A, Lanczos uses symmetry to obtain a short recurrence in exact arithmetic. Both methods need accurate orthogonality to interpret the resulting small matrix. Session 3 develops the projection underlying the residual. Rayleigh–Ritz later uses Q^*AQ to represent the action seen within a trial subspace and asks for a residual perpendicular to that subspace.

Reading connections. Kanschat and Scheichl, Appendix A.2 and §2.1, give the convention and two Gram–Schmidt algorithms used here. Bastian, §5 and Algorithm 11.2, connects inner-product geometry to orthogonalization. Scheichl and Zech, §§3.8–3.9, give the subsequent least-squares and QR context.

3.6 Independent exercises

Allow about 25 minutes. Keep the solutions covered until you have a derivation or a calculation to check.

1. **Choosing a geometry.** On $\mathbb{C}^2$ set $\langle x, y\rangle_W = 2x_1\overline{y_1} + x_2\overline{y_2}$. Verify the axioms. Find a nonzero vector perpendicular to $(1, i)^T$ in this product. Check your answer against the standard product.
2. **Recovering a normal.** For a, b from the running plane, find $(\operatorname{im} B)^\perp$ by row equations. Which d make $Bc = d$ solvable? Explain why the same vector answers both questions.
3. **Decomposing a vector.** Decompose e_1 into a vector in the running plane and a vector in its complement. Verify membership and Pythagoras. Explain why a second decomposition is impossible.
4. **Orthogonalization by hand.** Apply classical Gram–Schmidt in order to $(1, 1, 0)^T$, $(1, 0, 1)^T$, $(0, 1, 1)^T$. Give all three unit vectors. Verify orthogonality and express the last input in your new basis.
5. **Complex coordinates.** Let $q = (1, i)^T/\sqrt{2}$ and $x = (i, 0)^T$. Find the scalar c such that $x - cq \perp q$. Check what happens if you reverse the two arguments when computing c. Prove for any orthonormal family that $\sum_j |\langle x, q_j\rangle|^2 \le \|x\|^2$, and characterize equality.
6. **A filtered block.** Let $Q = [\,(e_1 + e_3)/\sqrt{2},\ e_2\,]$ in $\mathbb{R}^3$, and suppose $p(A) = \operatorname{diag}(4, 2, 1)$. Find $Y = p(A)Q$ and an orthonormal basis of its image. Decide whether filtering and then orthonormalizing each changed the span in this example. Would normalizing columns separately suffice for a general filtered block? Justify your answer.

3.7 Worked solutions

3.7.1 Choosing a geometry

Each summand is linear in x, and swapping x, y conjugates the sum because the weights are real. The value on (x, x) is $2|x_1|^2 + |x_2|^2$, strictly positive for $x \neq 0$. Orthogonality to $(1, i)^T$ requires $2z_1 - iz_2 = 0$. Choose $z = (1, -2i)^T$: the weighted test is $2 + (-i)(-2i) = 0$, whereas the standard test gives $1 - 2 = -1$.

3.7.2 Recovering a normal

The two tests are $z_1 + z_3 = 0$ and $z_2 + z_3 = 0$. Putting $z_3 = t$ gives $z = t(-1, -1, 1)^T = tn$. Also $Bc = (c_1, c_2, c_1+c_2)^T$. Thus solvability requires and suffices that $d_3 = d_1+d_2$, or $n^T d = 0$. The normal annihilates both columns, and its perpendicular plane has the same dimension as the column span.

3.7.3 Decomposing a vector

Write $e_1 = u+tn$. Pairing with n gives $t = -1/3$. Hence the perpendicular part is $(1, 1, -1)^T/3$ and $u = (2, -1, 1)^T/3$ lies in the plane since $2 - 1 = 1$. The squared lengths are $3/9$ and $6/9$, which sum to one. Two decompositions would put their difference in both the plane and its perpendicular line, whose intersection is zero.

3.7.4 Orthogonalization by hand

The first unit vector is $q_1 = (1, 1, 0)^T/\sqrt{2}$. Removing $(1, 1, 0)^T/2$ from the second input leaves $(1, -1, 2)^T/2$, so $q_2 = (1, -1, 2)^T/\sqrt{6}$. For $a_3 = (0, 1, 1)^T$, the first two coefficients are $1/\sqrt{2}$ and $1/\sqrt{6}$. The remaining vector is

$$w_3 = (0, 1, 1)^T - \tfrac{1}{2}(1, 1, 0)^T - \tfrac{1}{6}(1, -1, 2)^T = \tfrac{2}{3}(-1, 1, 1)^T.$$

Thus $q_3 = (-1, 1, 1)^T/\sqrt{3}$. The unscaled dot products are $1 - 1 = 0$, $-1 + 1 = 0$, and $-1 - 1 + 2 = 0$; the denominators are precisely the lengths. Reconstruction gives $a_3 = q_1/\sqrt{2} + q_2/\sqrt{6} + (2/\sqrt{3})q_3$.

3.7.5 Complex coordinates

$c = q^*x = i/\sqrt{2}$ gives $x - cq = (i, 1)^T/2$ and $q^*(x - cq) = 0$. The reversed product gives $-i/\sqrt{2}$, leaving coefficient $q^*x - (-i/\sqrt{2}) = \sqrt{2}\, i$ along q. For the general claim, let $u = \sum_j \langle x, q_j\rangle q_j$. Orthonormality gives $\|u\|^2 = \sum_j |\langle x, q_j\rangle|^2$, while $x - u \perp u$. Thus $\|x\|^2 = \|u\|^2 + \|x - u\|^2$. Equality holds precisely when x lies in the span of the family.

3.7.6 A filtered block

The columns of Y are $(4, 0, 1)^T/\sqrt{2}$ and $2e_2$. They are perpendicular, so dividing each by its length gives $(4, 0, 1)^T/\sqrt{17}$ and e_2. Originally the plane obeyed $x_1 = x_3$. Its filtered image obeys $x_1 = 4x_3$, so filtering changed the plane. Orthonormalization just rescaled these particular columns. In general, $\langle p(A)q_i, p(A)q_j\rangle = q_j^* p(A)^* p(A) q_i$ can be nonzero even if $q_j^* q_i = 0$. Dividing by lengths cannot remove that component.

3.8 End-of-session mastery checklist

Without consulting the notes, you should be able to:

- State the inner-product axioms, identify the linear slot, and compute a complex coefficient with the correct conjugation.
- Derive Cauchy–Schwarz by constructing a perpendicular residual and identify when equality holds.
- Compute a complement as a kernel and explain which ambient space contains it.
- Construct $x = u + z$ with $u \in U$, $z \in U^{\perp}$, and prove uniqueness and the dimension identity.
- Recover coordinates with Q^*, verify $Q^*Q = I$, and explain why orthonormal coordinates preserve length.
- Perform Gram–Schmidt on independent columns and prove nonbreakdown, orthonormality, and preservation of every successive span.
- Identify which vector classical and modified Gram–Schmidt use to compute each coefficient, and explain the risk when a residual is tiny.
- Distinguish the subspace change caused by a polynomial filter from the basis change caused by orthonormalization.

Ready for Session 3: reconstruct Exercise 4 on blank paper and explain why the parallel component you subtracted was uniquely determined. That uniqueness is the starting point for orthogonal projection.

4 Session 3 — Orthogonal Projections

Given a vector x and a subspace U, which vector in U should we call the component of x along U? Session 1 supplied basis matrices for describing points already in a subspace. Session 2 supplied a new test: subtract a candidate component and ask whether the remainder is perpendicular to every direction in U. This session turns that test into a linear operator and shows why its output is the closest point in U.

Allow **45–50 minutes** for the derivations and worked examples, then **25–30 minutes** for the exercises. To test recall during the reading, cover each projector formula and reconstruct it from the residual condition.

Throughout, V is a finite-dimensional inner-product space over $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. As in Session 2, the inner product is linear in the first argument. Matrix formulas use standard orthonormal ambient coordinates, so $\langle x, y\rangle = y^*x$, where * denotes conjugate transpose (transpose over $\mathbb{R}$). Write $\|\cdot\|$ for the induced norm.

4.1 The component selected by the inner product

A direct sum $V = U \oplus W$ gives every x a unique expression $x = u + w$. Its component u depends on the choice of W. In the plane, for instance, one can reach the horizontal axis from x by moving vertically or by moving along a slanted line. The two paths generally reach different points. Once an inner product is fixed, it singles out the complement

$$U^\perp = \{z : \langle z, v\rangle = 0 \text{ for all } v \in U\}.$$

Existence. Choose an orthonormal basis $q_1, \dots, q_k$ of U, available by Gram–Schmidt. Set $u = \sum_{j=1}^{k} \langle x, q_j\rangle q_j$. For every i, $\langle x - u, q_i\rangle = \langle x, q_i\rangle - \langle x, q_i\rangle = 0$. Hence $x - u$ is perpendicular to their entire span. This includes $U = \{0\}$ by taking the empty sum.

Uniqueness. If $x = u + r = u' + r'$ with $u, u' \in U$ and $r, r' \in U^\perp$, then $u - u' = r' - r$ belongs to both spaces. A vector in their intersection is perpendicular to itself and therefore zero. Thus $V = U \oplus U^\perp$. The assignment $P_U x = u$ is the **orthogonal projection onto** U. It is linear by the coefficient formula above.

Choosing a basis of U tells us how to encode u *after* selecting it. Choosing $U^\perp$ tells us which u to select, even for an x outside U. Any basis of the same U must lead to the same selected vector if the inner product is fixed.

4.2 Projection onto a line

Let $U = \operatorname{span}\{v\}$ with $v \neq 0$. Every candidate has the form cv. To select one, impose the single residual equation

$$v^*(x - cv) = 0.$$

Since $v^*v = \|v\|^2 > 0$, it follows that

$$c = \frac{v^*x}{v^*v}, \qquad P_U x = \frac{v^*x}{v^*v}\, v, \qquad P_U = \frac{vv^*}{v^*v}.$$

The denominator converts a measurement against a possibly long vector into its basis coefficient. For the unit vector $q = v/\|v\|$, the formula becomes $P_U x = q(q^*x)$. Here q^*x is a scalar and qq^* an operator. Rescaling v by any nonzero complex scalar leaves the operator unchanged because numerator and denominator acquire the same factor.

Example. Take $v = (1,1)^T$ and $x = (3,1)^T$. Then $v^Tx = 4$, $v^Tv = 2$, and

$$P_U = \tfrac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \qquad P_U x = (2,2)^T, \qquad r = x - P_U x = (1,-1)^T.$$

The coordinates of $P_U x$ are equal, so it lies on the target line. The residual passes the independent test $v^Tr = 0$. A vertical path from x also reaches the line, at $(3,3)^T$. Its length is 2, whereas the perpendicular path has length $\sqrt{2}$.

4.3 From orthonormal coordinates to a projection matrix

Store an orthonormal basis of U as the columns of $Q \in \mathbb{F}^{n\times k}$. In the language of Session 1, Q assembles a point in U from k coefficients. We need the coefficients c for which $x - Qc$ is perpendicular to every column. The k tests form one equation:

$$Q^*(x - Qc) = 0 \quad \Longleftrightarrow \quad Q^*Q\, c = Q^*x.$$

By Session 2, $Q^*Q = I_k$. Thus $c = Q^*x$ and

$$P_U x = Q(Q^*x) = \sum_{j=1}^{k} q_j(q_j^*x), \qquad P_U = QQ^*.$$

Expression	Input and output	Meaning
Q^*x	$\mathbb{F}^n \to \mathbb{F}^k$	Measure the coordinates of the component in U.

Expression	Input and output	Meaning
Qc	$\mathbb{F}^k \to \mathbb{F}^n$	Assemble the ambient vector with those coordinates.
Q^*Q	$\mathbb{F}^k \to \mathbb{F}^k$	Assemble, then recover every coordinate exactly.
QQ^*	$\mathbb{F}^n \to \mathbb{F}^n$	Keep the component in U and discard the perpendicular component.

For $k < n$ there is a nonzero $z \in U^\perp$. Then $Q^*z = 0$ and $QQ^*z = 0 \neq z$, so QQ^* cannot equal I_n. In computation one can project by applying Q^* and then Q without storing an $n \times n$ matrix.

4.3.1 The plane from Sessions 1 and 2

Let $U = \{x \in \mathbb{R}^3 : x_3 = x_1 + x_2\}$. Its normal is $n = (-1, -1, 1)^T$. Session 2 orthonormalized the spanning columns $(1, 0, 1)^T$ and $(0, 1, 1)^T$ to obtain

$$q_1 = \frac{(1, 0, 1)^T}{\sqrt{2}}, \qquad q_2 = \frac{(-1, 2, 1)^T}{\sqrt{6}}.$$

Adding the two line projectors gives

$$\begin{aligned} P_U &= \tfrac{1}{2} \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} + \tfrac{1}{6} \begin{pmatrix} 1 & -2 & -1 \\ -2 & 4 & 2 \\ -1 & 2 & 1 \end{pmatrix} \\ &= \tfrac{1}{3} \begin{pmatrix} 2 & -1 & 1 \\ -1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}. \end{aligned}$$

For $b = (3, 2, 2)^T$ the two orthonormal coordinates are $Q^T b = (5/\sqrt{2}, 3/\sqrt{6})^T$. Reconstruction gives $P_U b = (2, 1, 3)^T$. Check that $3 = 2 + 1$ and that $b - P_U b = (1, 1, -1)^T = -n$. These two checks establish the result even without inspecting the matrix multiplication.

4.4 Which matrices are orthogonal projectors?

The two defining algebraic properties follow directly from the factorization:

$$P_U^2 = Q(Q^*Q)Q^* = QQ^* = P_U, \qquad P_U^* = (QQ^*)^* = QQ^* = P_U.$$

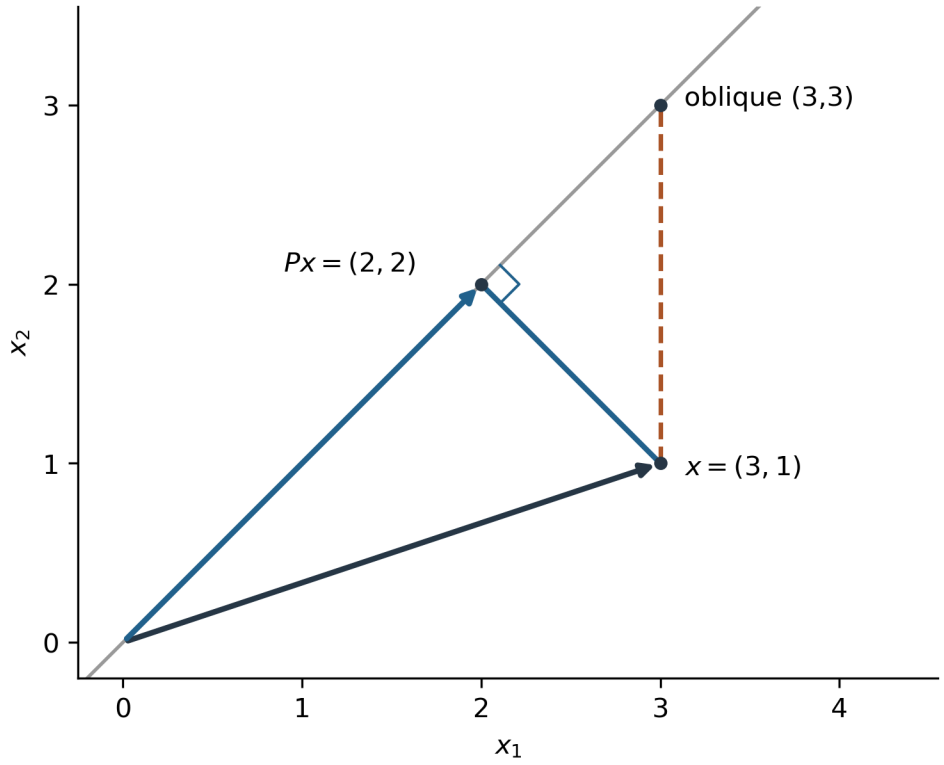


Figure 4.1: Two ways to project the same point onto the diagonal. The orthogonal residual meets the line at a right angle. Vertical travel reaches a different point on the same line.

The first identity is **idempotence**. Once a point has arrived in U, a second projection fixes it. Indeed, $P_U(Qc) = Qc$. Since every output belongs to U, this proves $\operatorname{im} P_U = U$. Also $P_U x = 0$ exactly when $Q^* x = 0$ (apply Q^* to $P_U x$), hence $\ker P_U = U^\perp$. The second identity is **self-adjointness**. It is what makes the directions discarded by an idempotent perpendicular to the ones retained.

Characterization. A square matrix P is an orthogonal projector exactly when $P^2 = P$ and $P^* = P$.

Proof of the converse. For any idempotent, $x = Px + (I-P)x$ with $Px \in \operatorname{im} P$ and $P(I-P)x = 0$. It fixes its image, so the intersection of image and kernel is zero. If P is also self-adjoint, then for $u = Py$ and $z \in \ker P$ we have $\langle u, z\rangle = \langle y, P^* z\rangle = 0$. The displayed decomposition is therefore the unique orthogonal decomposition for the subspace $\operatorname{im} P$. This proves the converse.

Complement and eigenvalues. The map $I - P_U$ returns the residual. It fixes $U^\perp$, annihilates U, and is the orthogonal projector onto $U^\perp$. For the running plane this gives the independent check $I - P_U = nn^T/3$. For any idempotent, an eigenpair $Px = \lambda x$, $x \neq 0$, implies $\lambda^2 = \lambda$. The eigenspace for 1 is the image and that for 0 the kernel whenever these spaces are nonzero. A projection onto a proper nonzero subspace has both eigenvalues. The zero and identity operators each have just one in positive dimension.

Oblique example. The vertical path in the line diagram is the rule $S(s,t)^T = (s,s)^T$. Its matrix and adjoint are

$$S = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}, \qquad S^T = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}.$$

It fixes the diagonal, so $S^2 = S$. It kills the vertical axis, which is not perpendicular to the diagonal. Consequently $S^T \neq S$. Specifying the image alone determines an *orthogonal* projector; specifying a general projector also requires the direction of projection, its kernel.

4.5 When the supplied basis is not orthonormal

Let $B \in \mathbb{F}^{n\times k}$ have independent columns spanning U. Seek the projected vector as Bc. The residual test still reads $B^*(x - Bc) = 0$, but the coefficient matrix is now the **Gram matrix** $G = B^*B$. Its entries record the lengths and mutual inner products of the basis vectors. For $d \neq 0$,

$$d^* G d = \|Bd\|^2 > 0$$

by column independence. Thus G is positive definite and invertible. Solving for c and reconstructing gives

$$c = (B^*B)^{-1} B^* x, \qquad P_U = B(B^*B)^{-1}B^*.$$

The factor G^{-1} converts the measurements B^*x to basis coefficients. For $B = Q$ it becomes I_k. Any other basis of U yields the same operator because its result satisfies the same two

conditions: membership in U and residual in $U^\perp$. Uniqueness of the decomposition settles basis independence.

The same plane in its original basis. Write

$$B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{pmatrix}, \quad G = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}, \quad G^{-1} = \tfrac{1}{3}\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}.$$

Multiplying in two stages makes the roles of the factors visible:

$$G^{-1}B^T = \tfrac{1}{3}\begin{pmatrix} 2 & -1 & 1 \\ -1 & 2 & 1 \end{pmatrix}, \qquad BG^{-1}B^T = \tfrac{1}{3}\begin{pmatrix} 2 & -1 & 1 \\ -1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}.$$

The last row is the sum of the first two, as required for a map into this plane. For $b = (3, 2, 2)^T$, $B^T b = (5, 4)^T$, whereas the basis coefficients are $G^{-1}(5, 4)^T = (2, 1)^T$. In the orthonormal basis they were $(5/\sqrt{2}, 3/\sqrt{6})^T$. Both coordinate columns assemble to the same ambient point $(2, 1, 3)^T$.

4.6 Why the projection solves least squares

Best-approximation theorem. For each $x \in V$, $P_U x$ is the unique minimizer of $\|x - v\|$ over $v \in U$.

Proof. For any candidate $v \in U$, split its error at the projected point:

$$x - v = \underbrace{x - P_U x}_{\in U^\perp} + \underbrace{P_U x - v}_{\in U}.$$

Pythagoras gives the exact identity

$$\|x - v\|^2 = \|x - P_U x\|^2 + \|P_U x - v\|^2.$$

The first term is independent of v; the second is nonnegative and vanishes exactly at $v = P_U x$. Thus the squared distance increases by precisely $\|P_U x - v\|^2$ when one moves from the projected point to another point of U.

For a matrix B, the least-squares problem asks for coefficients minimizing $\|Bc - b\|_2^2$. As c ranges over its domain, Bc ranges over $\operatorname{im} B$. The theorem therefore determines the **fitted vector** $Bc = P_{\operatorname{im} B} b$ uniquely. In coefficient space, the same condition reads

$$b - Bc \perp \operatorname{im} B \quad \Longleftrightarrow \quad B^*(Bc - b) = 0 \quad \Longleftrightarrow \quad B^*Bc = B^*b.$$

These are the **normal equations**. Full column rank gives a unique coefficient vector as well. With dependent columns, if c_0 is one solution, all solutions are $c_0 + \ker B$ and give the same fit.

Worked least-squares problem. Suppose three measurements suggest $s = 3$, $t = 2$, and $s + t = 2$, with equal weights. The first two would force $s + t = 5$, so the three equations are inconsistent. Their matrix and data are precisely the running B and b. The normal equations are $2s + t = 5$ and $s + 2t = 4$, giving $(s, t) = (2, 1)$. The fitted measurements are $(2, 1, 3)^T$, with squared error $1^2 + 1^2 + (-1)^2 = 3$. The theorem supplies more than this numerical value: any change (h, k) increases the squared error by $\|B(h, k)^T\|^2 = h^2 + k^2 + (h + k)^2$.

The inverse in the projector formula expresses the mathematics. In numerical work, forming B^*B squares the spectral condition number for a full-column-rank matrix, since its eigenvalues are the squared singular values of B. QR or SVD can treat the original matrix directly. Session 4 develops the QR approach.

4.7 From one projection to projection methods

- **Gram–Schmidt and QR.** If Q spans the previously accepted directions, the next candidate a contributes $(I - QQ^*)a$. Normalizing a nonzero residual adds a direction perpendicular to that span. Projecting a vector selects its component in a prescribed space. Orthonormalizing independent columns chooses a new basis of their existing space.
- **Galerkin and Krylov methods.** A Galerkin approximation to $Ax = b$ chooses a trial vector and requires its residual to be perpendicular to the test space, which in standard Galerkin is the trial subspace. Arnoldi builds an orthonormal Krylov basis by removing previously represented components of Aq_j. Lanczos uses Hermitian symmetry to shorten this recurrence in exact arithmetic. GMRES chooses a correction in a Krylov space K minimizing the residual norm. With initial residual r_0, it approximates r_0 by a vector in AK, so the minimizing residual is perpendicular to AK.
- **Rayleigh–Ritz.** For $x = Qy$, AQy may leave the trial subspace. Its projected component is $QQ^*AQy = QHy$ with $H = Q^*AQ$. Thus H represents the projected action in the trial basis. If $Hy = \theta y$, the ambient residual $AQy - \theta Qy$ is perpendicular to the trial space, since multiplying it by Q^* gives zero. This is the projection principle behind Session 11.
- **Subspace iteration and the seminar.** Multiplication by A, or by a Chebyshev filter $p(A)$, can move a block into a different subspace. Orthonormalization preserves the resulting span in exact arithmetic and supplies its basis Q. Rayleigh–Ritz can then extract approximate eigenpairs in that span. Filtering, basis construction, and projection answer three different questions: which directions to emphasize, how to represent them, and what part of the operator can be seen within them.

Course reading. Kanschat and Scheichl, Appendices A.2–A.3 and Chapter 2, connect the projector characterization to orthogonalization. For the normal equations and the transition to QR, see Scheichl and Zech, §§3.8–3.9, and Bastian, §11.2.

4.8 Independent exercises

1. **A complex line.** For $v = (1, i)^T$ and $x = (2, 0)^T$, derive the line projector from the residual equation. Compute Px and verify orthogonality of the residual.
2. **A projector from two columns.** Use $q_1 = (1, 1, 0)^T/\sqrt{2}$ and $q_2 = (0, 0, 1)^T$ to construct P. Identify its image and kernel and compute $(I-P)(3, 1, 2)^T$. Verify $P^2 = P$ and $P^* = P$ using its factors.
3. **One image, many projectors.** Find every real 2×2 idempotent with image the horizontal axis. Identify its kernel and determine exactly when it is an orthogonal projector.
4. **A change of subspace coordinates.** Suppose B has full column rank and $C = BT$ for an invertible $k \times k$ matrix T. Derive the projector from the residual test with C, prove algebraically that it equals the one from B, and relate the two coefficient vectors for the fit.
5. **A fit and its uniqueness.** Fit a line $f(t) = \alpha + \beta t$ in unweighted least squares to the data $(-1, 1), (0, 0), (1, 3)$. Give the design matrix, coefficients, fitted data, and residual. Prove optimality using orthogonality. If the model becomes $f(t) = \alpha+\beta t+\gamma t$, determine which quantities remain unique.
6. **Projecting an operator action.** Take $A = \operatorname{diag}(1, 3, 5)$ and the two-column Q of Exercise 2. Find $H = Q^*AQ$. For $y = (1, 0)^T$, compare AQy with QHy and identify the residual geometrically. Decide whether $\operatorname{im} Q$ is invariant under A.

4.9 Worked solutions

4.9.1 A complex line

Writing the candidate as cv, the condition $v^*(x - cv) = 0$ yields $c = v^*x/(v^*v) = 2/2 = 1$. For arbitrary input the matrix is

$$P = \tfrac{1}{2}\begin{pmatrix} 1 & -i \\ i & 1 \end{pmatrix}.$$

Hence $Px = (1, i)^T$, and $r = (1, -i)^T$ obeys $v^*r = 1 + (-i)(-i) = 0$. The conjugation in the test is essential.

4.9.2 A projector from two columns

Summing the outer products gives

$$P = \begin{pmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 1/2 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

It averages the first two coordinates and keeps the third. Its image is the plane $x_1 = x_2$ and its kernel is the line spanned by $(1, -1, 0)^T$. Applying it to $(3, 1, 2)^T$ gives $(2, 2, 2)^T$, so the residual is $(1, -1, 0)^T$. The columns have unit length and zero inner product, hence $Q^*Q = I_2$. Now $(QQ^*)^2 = QI_2Q^* = QQ^*$ and $(QQ^*)^* = QQ^*$.

4.9.3 One image, many projectors

The second row must vanish. An idempotent fixes its image, so its first column must be e_1. Thus every candidate has the form

$$S_t = \begin{pmatrix} 1 & t \\ 0 & 0 \end{pmatrix}, \qquad t \in \mathbb{R}.$$

Each such matrix squares to itself and has the required image. Its kernel consists of $(a, b)^T$ with $a = -tb$, the line spanned by $(-t, 1)^T$. Only $t = 0$ makes this line vertical, equivalently $S_t^T = S_t$.

4.9.4 A change of subspace coordinates

With projected point Cd, the residual condition $C^*(x - Cd) = 0$ gives $d = (C^*C)^{-1}C^*x$ because C has independent columns. Put $G = B^*B$. Then $C^*C = T^*GT$, whose inverse is $T^{-1}G^{-1}(T^*)^{-1}$. Inserting this into the projector cancels the two basis changes:

$$C(C^*C)^{-1}C^* = BT\,T^{-1}G^{-1}(T^*)^{-1}T^*B^* = BG^{-1}B^*.$$

If c denotes the B-coordinates, $Bc = Cd = BTd$. Injectivity of B gives $c = Td$, or $d = T^{-1}c$.

4.9.5 A fit and its uniqueness

The observations and the design matrix, ordered by increasing t, are

$$b = (1, 0, 3)^T, \qquad B = \begin{pmatrix} 1 & -1 \\ 1 & 0 \\ 1 & 1 \end{pmatrix}.$$

The columns are perpendicular, with squared lengths 3 and 2. Thus $B^TB = \mathrm{diag}(3, 2)$ and $B^Tb = (4, 2)^T$, giving $\alpha = 4/3$ and $\beta = 1$. The fitted vector is $(1/3, 4/3, 7/3)^T$, and the residual is $(2, -4, 2)^T/3$. Its entries sum to zero and its first and last entries agree, so both column tests vanish. Changing the coefficient vector by d adds exactly $\|Bd\|^2$ to the minimum squared error. This is positive for nonzero d because the original columns are independent.

Duplicating the slope column preserves the column space, the fit, and the residual. Only $\alpha = 4/3$ and the combined slope $\beta + \gamma = 1$ are determined. Every triple $(4/3, 1 - s, s)$ gives that fit.

4.9.6 Projecting an operator action

The vectors $Aq_1 = (1, 3, 0)^T/\sqrt{2}$ and $Aq_2 = 5q_2$ give $H = \mathrm{diag}(2, 5)$ by taking their inner products against q_1, q_2. With $y = e_1$, $QHy = 2q_1 = (2, 2, 0)^T/\sqrt{2}$. The residual is $(-1, 1, 0)^T/\sqrt{2}$, a nonzero vector in the kernel of the trial-space projector. It is exactly the part of Aq_1 lost in compression. Since Aq_1 leaves the plane, that plane is not invariant.

4.10 End-of-session mastery checklist

Without consulting the notes, you should be able to:

- Derive the line projector and $P = QQ^*$ from the requirement that the residual be orthogonal to the target space.
- Distinguish Q^*Q and QQ^* by their dimensions and actions.
- Prove $P^2 = P$ and $P^* = P$, explain their geometric meanings, and construct an oblique counterexample to idempotence alone being sufficient.
- Identify image and kernel, describe $I - P$, and explain where the two possible eigenvalues occur.
- Derive $P = B(B^*B)^{-1}B^*$, justify the inverse, and explain why changing the basis preserves the projector.
- Prove uniqueness of the closest point by splitting the squared error into two perpendicular contributions.
- Derive the normal equations and distinguish uniqueness of fitted data from uniqueness of coefficients.

Ready for Session 4. On a blank page, recover the projector for the running plane from either basis and check it on a vector outside the plane. Then reconstruct the best-approximation proof and the normal equations without differentiating. With those steps in hand, the next question is how QR computes the same fit reliably.

5 Session 4 — QR Factorization

Suppose two independent columns point almost in the same direction. They still form a basis of their plane, but recovering the small difference between their directions requires subtracting nearly equal vectors. Can we keep the plane, choose better axes in it, and still recover coordinates in the original basis? The two factors in $A = QR$ answer these questions separately.

Study route (90 minutes). Spend 20 minutes on coordinates and Gram–Schmidt, 20 on reflectors, 15 on floating-point behavior, 15 on least squares and the seminar connection, and 20 on the exercises. In the worked examples, cover the next line and predict either a coefficient, a zero, or the dimension of a factor before continuing.

5.1 One space, two coordinate systems

Let $A = [a_1, \dots, a_n] \in \mathbb{F}^{m\times n}$, where $\mathbb{F}$ is $\mathbb{R}$ or $\mathbb{C}$, $m \geq n$, and the columns are independent. A star denotes the conjugate transpose, which reduces to transpose over $\mathbb{R}$. Write $U = \operatorname{im} A$. From Session 1, $x \mapsto Ax$ takes coordinates in the ordered column basis to vectors in U. Choose an orthonormal basis $q_1, \dots, q_n$ of U as in Session 2 and collect it in Q. The coordinates of $y \in U$ in this basis are Q^*y, since $Q^*Q = I_n$. For arbitrary $b \in \mathbb{F}^m$, Session 3 gives the projection $P_U b = QQ^*b$.

What makes the coordinate matrix **triangular**? Choose the new basis to preserve every initial span,

$$\operatorname{span}(q_1, \dots, q_j) = \operatorname{span}(a_1, \dots, a_j).$$

Then a_j uses only $q_1, \dots, q_j$. Writing its coordinates in column j of R places zeros below the diagonal. Thus

$$A = QR, \qquad Ax = Q(Rx), \qquad Q^*A = R.$$

The middle identity fixes the direction of the coordinate change. Given orthonormal coordinates c, the original coordinates solve $Rx = c$ by back substitution, starting with $x_n = c_n/r_{nn}$. Invertibility of R implies $\operatorname{im} A = \operatorname{im} Q$.

Form	Orthonormal factor	Triangular factor
Reduced (thin)	Q is $m \times n$, $Q^*Q = I_n$	R is $n \times n$, upper triangular

Complete (full)	$U_f = [Q\ Q_\perp]$ is $m \times m$, unitary	$U_f^* A = \begin{bmatrix} R \\ 0 \end{bmatrix}$ is $m \times n$

The zero block has $m-n$ rows. For square A the two forms coincide. Thin Q is an isometry from coefficient space into U, since $\|Qc\|_2^2 = c^*Q^*Qc = c^*c$. A square unitary matrix preserves the norm of **every** vector in the ambient space. Consequently, transforming existing perturbations by a unitary matrix leaves their Euclidean lengths unchanged. Computing the transformation in floating point still introduces rounding errors.

Existence and uniqueness. Gram–Schmidt below constructs thin QR with $r_{jj} > 0$ real. This convention makes it unique: the first column fixes $q_1 = a_1/\|a_1\|_2$, and at each later step the previous columns determine a unique residual whose normalization fixes q_j. Without the convention, QD and D^*R also work for any diagonal D with entries of modulus one. An orthonormal completion $Q_\perp$ admits additional freedom. A QR factorization also exists for dependent columns (for example by Householder elimination), but then R is singular and the positive-diagonal construction breaks down.

5.2 Gram–Schmidt records its work in R

Suppose $Q_{j-1} = [q_1, \dots, q_{j-1}]$ already spans the first $j-1$ columns. Remove the projection of a_j onto this space, then normalize what remains:

$$\begin{aligned} r_{ij} &= q_i^* a_j && (i < j), \\ w_j &= a_j - \sum_{i<j} q_i r_{ij} = (I - Q_{j-1}Q_{j-1}^*)a_j, \\ r_{jj} &= \|w_j\|_2, \qquad q_j = w_j / r_{jj}. \end{aligned}$$

Here $q_i^* w_j = 0$ for $i < j$, and independence gives $w_j \neq 0$. Rearranging yields $a_j = \sum_{i=1}^{j} q_i r_{ij}$. This is column j of QR and proves the initial-span property by induction. Geometrically, the diagonal r_{jj} is the distance of a_j from the preceding span.

5.2.1 Worked example — a plane in three dimensions

Use the matrix

$$A = \begin{bmatrix} 3 & -2 \\ 4 & 4 \\ 0 & 3 \end{bmatrix}.$$

The first column gives $r_{11} = 5$ and $q_1 = (3, 4, 0)^T/5$. The component of a_2 along q_1 is $r_{12} = (-6 + 16)/5 = 2$. Removing it gives

$$w_2 = a_2 - 2q_1 = \tfrac{1}{5}(-16, 12, 15)^T, \qquad \|w_2\|_2 = 5.$$

Hence

$$Q = \tfrac{1}{25}\begin{bmatrix} 15 & -16 \\ 20 & 12 \\ 0 & 15 \end{bmatrix}, \qquad R = \begin{bmatrix} 5 & 2 \\ 0 & 5 \end{bmatrix}.$$

Verify both identities. The squared lengths of the integer columns in $25Q$ are 625 and 625, and their dot product is $-240 + 240 = 0$. Thus $Q^*Q = I_2$. In QR the first column is $5q_1 = (3, 4, 0)^T$ and the second is $2q_1 + 5q_2 = (-2, 4, 3)^T$. In particular the second original basis vector has coordinates $(2, 5)^T$ in the new basis.

5.2.2 Classical and modified Gram–Schmidt

Classical Gram–Schmidt (CGS) computes all coefficients against the **original** column, then subtracts its projection. Modified Gram–Schmidt (MGS) starts with $w = a_j$ and for $i = 1, \dots, j-1$ computes $r_{ij} = q_i^* w$ and immediately updates $w \leftarrow w - q_i r_{ij}$. Why does the changed inner product give the same coefficient? In exact arithmetic,

$$q_i^* \left(a_j - \sum_{k<i} q_k r_{kj} \right) = q_i^* a_j - \sum_{k<i} (q_i^* q_k) r_{kj} = q_i^* a_j.$$

Floating point changes this argument. A computed q_i may still contain a small component along an earlier direction. With CGS that component is tested against the entire incoming column. MGS first removes the large earlier components, which often makes the next coefficient more accurate. When w is very small, normalizing it magnifies any contamination it still carries. Even MGS can lose substantial orthogonality.

The following single-pass implementations expose the one operation that changes. They assume independent columns. An exact zero check catches division by zero; deciding numerical rank requires a scale and a tolerance.

```
import numpy as np
import matplotlib.pyplot as plt

def gram_schmidt(A, *, modified):
    A = np.asarray(A, dtype=np.result_type(A, np.float64))
    m, n = A.shape
    Q = np.zeros((m, n), dtype=A.dtype)
    R = np.zeros((n, n), dtype=A.dtype)
    for j in range(n):
        w = A[:, j].copy()
        if modified:
            for i in range(j):
                R[i, j] = np.vdot(Q[:, i], w)
```

```
                w -= Q[:, i] * R[i, j]
        else:
            basis = Q[:, :j]
            R[:j, j] = basis.conj().T @ w
            w -= basis @ R[:j, j]
        R[j, j] = np.linalg.norm(w)
        if R[j, j] == 0:
            raise ValueError("Zero residual during orthogonalization")
        Q[:, j] = w / R[j, j]
    return Q, R
```

Reorthogonalization. Before normalizing, project the residual w against the previous basis again. For a block projection compute $h = Q_{j-1}^* w$, subtract $Q_{j-1}h$ from w, and **add** h to the already stored coefficients. This last update keeps the reconstruction of a_j consistent. A second pass often restores near-working-precision orthogonality when the incoming directions are numerically resolvable. It costs additional inner products and vector updates.

5.3 Householder QR by changing the ambient coordinates

Gram–Schmidt builds one basis vector at a time. Householder QR instead transforms the whole matrix until the columns align with successive coordinate subspaces. The first transformation sends a_1 to the first axis. The next acts only below the first row, preserving the zeros already obtained.

5.3.1 Deriving a reflector

For $v \neq 0$ let $P_v = vv^*/(v^*v)$. Decompose $x = (I - P_v)x + P_v x$. Fixing the first component and reversing the second gives

$$Hx = (I - P_v)x - P_v x, \qquad H = I - 2P_v.$$

Since $P_v^* = P_v$ and $P_v^2 = P_v$, we get $H^* = H$ and $H^*H = I - 4P_v + 4P_v^2 = I$. Over $\mathbb{R}$, H reflects across the hyperplane $v^\perp$. Over $\mathbb{C}$, the same orthogonal decomposition into complex subspaces applies: H acts as $+1$ on $v^\perp$ and -1 on the complex line spanned by v. This is a complex-linear unitary map.

How to choose the normal. For $x \neq 0$, set $\rho = \|x\|_2$, $s = x_1/|x_1|$ when $x_1 \neq 0$, and $s = 1$ otherwise. Choose $\alpha = -s\rho$ and $v = x - \alpha e_1$. Reflection should subtract precisely v to take x to αe_1. Indeed,

$$v^*x = \rho^2 + \rho|x_1|, \qquad v^*v = 2(\rho^2 + \rho|x_1|).$$

Thus $Hx = x - v = \alpha e_1$. The opposite-phase target gives $|v_1| = |x_1| + \rho$, avoiding the cancellation that the other sign could cause when x already lies close to the axis. For a zero active column use the identity.

5.3.2 Worked example — two eliminations

Return to the plane example. For $x = (3, 4, 0)^T$ the target is $-5e_1$. The normal $v = (8, 4, 0)^T$ can be scaled to $(2, 1, 0)^T$ without changing the reflector. Substitution gives

$$H_1 = \begin{bmatrix} -3/5 & -4/5 & 0 \\ -4/5 & 3/5 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad H_1 A = \begin{bmatrix} -5 & -2 \\ 0 & 4 \\ 0 & 3 \end{bmatrix}.$$

The active tail of column two is now $(4, 3)^T$. To send it to $(-5, 0)^T$, use normal $(9, 3)^T$, proportional to $(3, 1)^T$. Embed its reflector in the bottom-right corner:

$$H_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -4/5 & -3/5 \\ 0 & -3/5 & 4/5 \end{bmatrix}, \qquad H_2 H_1 A = \begin{bmatrix} -5 & -2 \\ 0 & -5 \\ 0 & 0 \end{bmatrix}.$$

Which order gives Q? Undo the left multiplications: $U_f = (H_2 H_1)^* = H_1 H_2$. Its first two columns are $-q_1, -q_2$ from the Gram–Schmidt example. The top two rows of the transformed matrix are $-R$, so the minus signs cancel in the product. The sign convention is the entire difference between these two exact reduced factorizations.

In general, step j uses $H_j = \operatorname{diag}(I_{j-1}, \widetilde{H}_j)$ on rows j through m. Earlier columns already vanish in those rows. If $H_t \cdots H_1 A = [R^T \ 0]^T$, the full unitary factor is $H_1 \cdots H_t$. In software, store the normals and apply each reflector to a trailing block B by the rank-one update $B \leftarrow B - v(2v^* B/(v^* v))$. This also lets us compute $U_f^* b$ without forming a dense U_f.

5.4 What should a numerical QR test measure?

For computed factors, examine two quantities:

$$d_{\text{fact}} = \|A - \widehat{Q}\widehat{R}\|_2, \qquad d_{\text{orth}} = \|\widehat{Q}^* \widehat{Q} - I_n\|_2.$$

The first tests reconstruction. Divide it by $\|A\|_2$ when comparing matrices on different scales. The second tests whether coefficient lengths and projection formulas remain trustworthy. For example, doubling one column of an exact Q and halving the corresponding row of R leaves the product unchanged, yet the associated diagonal entry of $\widehat{Q}^* \widehat{Q}$ is now 4.

Method	Work for dense real $m \times n$ input (leading terms)	Numerical behavior and use
CGS, one pass	About $2mn^2$	Block inner products are convenient. Nearly dependent columns can produce severe loss of orthogonality.
MGS, one pass	About $2mn^2$	Sequential updates often maintain better orthogonality. Useful for appending one Krylov basis vector.
Householder	About $2mn^2 - 2n^3/3$	Standard dense QR. A product of nearly isometric steps gives backward stability and typically near-working-precision orthogonality.

The Gram–Schmidt counts include explicit thin Q. The Householder count stores Q implicitly; explicitly generating a thin Q adds work of the same order. Backward stability here means the computed triangular factor corresponds to a matrix close to A under an exactly unitary transformation, up to dimension-dependent rounding bounds. It gives no guarantee that an intrinsically sensitive least-squares solution has small forward error.

5.4.1 Experiment — making columns approach a common direction

Use the $(n+1) \times n$ real matrix with first row all ones and remaining block δI_n. Two columns differ by a vector of length $\sqrt{2}\,\delta$. Since $A^*A = \mathbf{1}\mathbf{1}^T + \delta^2 I_n$, its extreme eigenvalues are $n + \delta^2$ and δ^2 for $n > 1$. Hence $\kappa_2(A) = \sqrt{n + \delta^2}/\delta$. The scale $\|A\|_2$ stays almost constant. This lets us vary column dependence without confounding it with the scale of the reconstruction error.

```
n = 8
deltas = np.logspace(-1, -10, 10)
methods = {
    "CGS": lambda A: gram_schmidt(A, modified=False),
    "MGS": lambda A: gram_schmidt(A, modified=True),
    "Householder": lambda A: np.linalg.qr(A, mode="reduced"),
}
errors = {name: [] for name in methods}
```

```
conditions = np.sqrt(n + deltas**2) / deltas
for delta in deltas:
    A = np.vstack((np.ones((1, n)), delta * np.eye(n)))
    for name, factor in methods.items():
        Q, R = factor(A)
        residual = np.linalg.norm(A - Q @ R, ord=2)
        defect = np.linalg.norm(Q.T @ Q - np.eye(n), ord=2)
        errors[name].append((residual, defect))

fig, axes = plt.subplots(1, 2, figsize=(7, 3.4), layout="constrained")
floor = 1e-19
for col, ax in enumerate(axes):
    for (name, rows), marker in zip(errors.items(), ("o", "s", "^")):
        values = np.array(rows)[:, col]
        ax.loglog(conditions, np.maximum(values, floor),
                  marker=marker, ms=4, label=name)
    ax.set_xlabel(r"Condition number $\kappa_2(A)$")
    ax.grid(True, which="major", alpha=0.25)
axes[0].set_title("Factorization residual")
axes[1].set_title("Orthogonality defect")
axes[0].legend(fontsize=8)
plt.show()
```

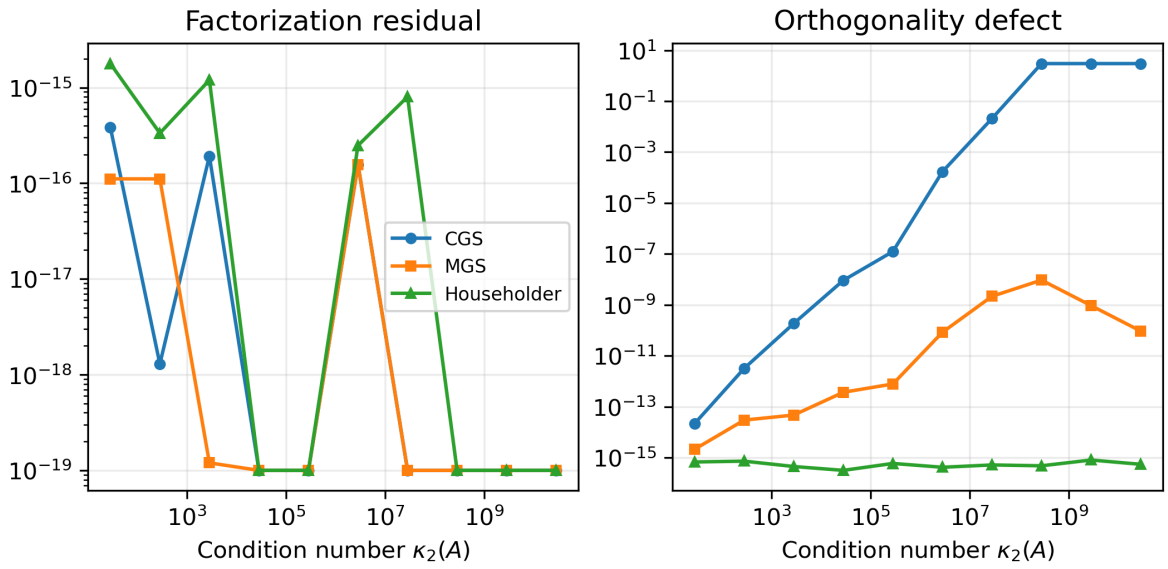


Figure 5.1: Single-pass Gram–Schmidt and Householder QR in binary64. Each point corresponds to the same matrix in both panels. Errors below 1e-19, including zero, are plotted at 1e-19.

Read the two panels at the same horizontal position. At small δ, all three methods can still reconstruct the input to high absolute accuracy. CGS can meanwhile produce an orthogonality defect of order one. The first column is almost a coordinate vector, and the small corrections to inner products can fall below binary64 resolution. MGS removes successive components from the residual and does substantially better on this family. Householder QR maintains orthogonality close to rounding precision. For example, at $\delta = 10^{-10}$, one binary64 run gave orthogonality defects of about 3, 9.4×10^{-11}, and 7.2×10^{-16} for CGS, MGS, and Householder respectively. All three reconstruction residuals were below 10^{-24}. The exact residuals, including zeros for particularly simple arithmetic, can vary with the numerical libraries. These curves illustrate mechanisms on one family; they are not universal error laws.

5.5 Least squares reduces to a triangular system

Given $m > n$ and $b \in \mathbb{F}^m$, seek x minimizing $\|b - Ax\|_2$. Write $b = QQ^*b + (I - QQ^*)b$. The first summand and Ax lie in U; the second summand lies in $U^\perp$. Pythagoras and the isometry of Q give

$$\begin{aligned} \|b - Ax\|_2^2 &= \|Q(Q^*b - Rx)\|_2^2 + \|(I - QQ^*)b\|_2^2 \\ &= \|Q^*b - Rx\|_2^2 + \|(I - QQ^*)b\|_2^2. \end{aligned}$$

Only the first term depends on x. Invertibility of R makes that term zero for exactly one x. Thus **solve** $Rx = Q^*b$. The vector Q^*b contains the coordinates of the closest point in the orthonormal basis; back substitution expresses that same point in the original columns. The residual $r = b - Ax = (I - QQ^*)b$ satisfies $Q^*r = 0$ and hence $A^*r = 0$. With implicit full QR, apply the stored reflectors to b: the first n transformed entries form the right-hand side and the norm of the remaining entries is the minimal residual norm.

5.5.1 Worked example — fitting a point to the plane

Use the same A and take $b = (0, 0, 5)^T$. Dotting with q_1, q_2 gives $Q^*b = (0, 3)^T$. Back substitution gives

$$5x_2 = 3, \qquad 5x_1 + 2x_2 = 0, \qquad x = (-6/25, 3/5)^T.$$

The closest point and the discrepancy are

$$Ax = 3q_2 = \tfrac{1}{25}(-48, 36, 45)^T, \qquad r = \tfrac{1}{25}(48, -36, 80)^T.$$

Check $a_1^T r = (144 - 144)/25 = 0$ and $a_2^T r = (-96 - 144 + 240)/25 = 0$. The residual length is 4; the fitted vector has length 3. Their squared lengths sum to $\|b\|_2^2 = 25$. Adding any nonzero displacement within the fitted plane increases the squared distance by the squared length of that displacement.

5.5.2 Why avoid explicitly forming the normal equations?

Residual orthogonality also gives $A^*Ax = A^*b$. For full column rank, the extreme eigenvalues of A^*A are the squares of the extreme singular values of A. Thus $\kappa_2(A^*A) = \kappa_2(A)^2$. By contrast $\|Ax\|_2 = \|Rx\|_2$ for every x, so R has the same extreme stretching factors and the same κ_2 as A. Forming the Gram matrix can round away small independent directions: in the experiment its diagonal entries $1 + \delta^2$ can round to 1 even while the entries δ remain readily representable. QR avoids this extra loss. The sensitivity of a least-squares solution to changes in the original data also depends on the right-hand side and residual.

If A is rank deficient, the fitted point remains unique, but one may add any vector of $\ker A$ to a minimizer. The invertible triangular solve no longer applies. Column-pivoted QR helps identify independent columns and estimate numerical rank. An SVD permits a rank decision from singular values and produces a minimum-norm solution.

5.6 From QR to the seminar

- **Power and subspace iteration.** Normalizing a single nonzero iterate preserves its line and keeps its length manageable. For a block, normalizing each column separately still permits nearly parallel columns. Thin QR controls their mutual angles as well.
- **Chebyshev filtering.** In $Y_{k+1} = p_m(A)Q_k$, the filter changes the relative sizes of spectral components. Strong amplification can make several columns point almost along the same preferred direction. Factor $Y_{k+1} = Q_{k+1}R_{k+1}$. For independent columns, R_{k+1} is invertible, so the next basis spans precisely the filtered space. If filtering annihilates a direction, or rounding has already made it indistinguishable, QR cannot reconstruct it. Completing a deficient block would introduce a freely chosen direction.
- **Rayleigh–Ritz.** For an approximate eigenvector Qz, require $AQz - \theta Qz$ to be orthogonal to $\operatorname{im} Q$. Multiplication by Q^* gives the small problem $(Q^*AQ)z = \theta z$ because $Q^*Q = I$. Loss of orthogonality undermines this reduction.
- **Arnoldi and Lanczos.** Arnoldi extends a Krylov basis by orthogonalizing Aq_j against earlier vectors. For Hermitian A, Lanczos admits a short recurrence in exact arithmetic. In finite precision, reorthogonalization can be important for maintaining independent approximate eigenvector information.
- **QR iteration.** For square $A_k = Q_kR_k$, reversing the factors gives $A_{k+1} = R_kQ_k = Q_k^*A_kQ_k$. Each step preserves eigenvalues by unitary similarity. Here QR changes coordinates for a whole operator; in filtered subspace iteration it provides a basis for a trial space. Session 5 develops the eigenvalues and spectral theorem needed to analyze these operations.

Course reading. The revision plan in §1.4.4 and the seminar link in §1.3.3 of the book motivate this sequence. For the lecture algorithms see Kanschat and Scheichl, *Numerical Linear Algebra*, §§2.1–2.2 and Appendix A. For least squares and implicit reflector application see Scheichl and Zech, §§3.8–3.9, and Bastian, §§11–12.

5.7 Independent exercises

1. **Dimensions and coordinates.** A 5×3 matrix B has independent columns. Give the dimensions of both QR factorizations, the ranks of QQ^* and $I - QQ^*$ for thin Q, and the orthonormal coordinates of Bx.
2. **Hand factorization.** Compute positive-diagonal thin QR for the columns $a_1 = (1, 0, 1)^T$, $a_2 = (1, 1, 1)^T$. Verify both identities and interpret r_{22} geometrically.
3. **Reflecting onto an axis.** For $x = (3, -4)^T$, construct the stable-sign reflector, verify Hx and H^TH, and explain using the inner product why every unitary matrix preserves distances.

4. **Projection and triangular solve.** For the matrix in Exercise 2 and $b = (2, 3, 0)^T$, find the closest point to b in the column plane, its original coordinates, and the distance. Check optimality using both columns.
5. **A numerical diagnosis.** A binary64 run on a matrix of norm near one gives $d_{\text{fact}} = 10^{-16}$ and $d_{\text{orth}} = 10^{-2}$. Which identity is reliable at working precision? Would you use the columns directly in the projector formula? Explain how a second pass or Householder QR changes the computation. Use the experiment to determine why testing reconstruction alone misses the problem.
6. **Filtering and rank.** Take $A = \operatorname{diag}(2, 1, 0)$, $Q_0 = [(e_1 + e_2)/\sqrt{2},\ e_3]$, and $p(t) = t$. Find the filtered image and its dimension. Repeat with $p(t) = t + \varepsilon$ for small real $\varepsilon > 0$. What does thin QR preserve, and what does the small diagonal of R tell you in this example?

5.8 Solutions — consult after attempting the exercises

1. Dimensions and coordinates. Thin factors have sizes 5×3 and 3×3. Full factors have sizes 5×5 and 5×3, with a 2×3 zero block below the triangular part. The projector onto the column space has rank three; its complement has rank two. Since $Bx = Q(Rx)$, Rx gives the requested coordinates.

2. Hand factorization. Normalize a_1 to get $q_1 = (1, 0, 1)^T/\sqrt{2}$. Then $r_{12} = q_1^T a_2 = \sqrt{2}$, and $a_2 - \sqrt{2}q_1 = (0, 1, 0)^T = q_2$. Thus

$$Q = \begin{bmatrix} 1/\sqrt{2} & 0 \\ 0 & 1 \\ 1/\sqrt{2} & 0 \end{bmatrix}, \qquad R = \begin{bmatrix} \sqrt{2} & \sqrt{2} \\ 0 & 1 \end{bmatrix}.$$

Both columns of Q have norm one and their dot product vanishes. Reconstruction gives $\sqrt{2}q_1 = a_1$ and $\sqrt{2}q_1 + q_2 = a_2$. Here $r_{22} = 1$ measures the perpendicular distance of a_2 from the line through a_1.

3. Reflecting onto an axis. Choose $\alpha = -5$, so $v = (8, -4)^T$ is proportional to $(2, -1)^T$. Then

$$H = \tfrac{1}{5}\begin{bmatrix} -3 & 4 \\ 4 & 3 \end{bmatrix}, \qquad Hx = \tfrac{1}{5}(-25, 0)^T.$$

The columns of the integer matrix have squared length 25 and dot product $-12 + 12 = 0$, so $H^T H = I$. For any unitary U and $d = x - y$, $\|Ux - Uy\|_2^2 = d^* U^* U d = d^* d$. This proves distance preservation for complex vectors as well.

4. Projection and triangular solve. The orthonormal coordinates are $Q^T b = (\sqrt{2}, 3)^T$, so the projection is $\sqrt{2}q_1 + 3q_2 = (1, 3, 1)^T$. Back substitution gives $x_2 = 3$ and $\sqrt{2}(x_1 + 3) = \sqrt{2}$, hence $x_1 = -2$. Indeed $-2a_1 + 3a_2 = (1, 3, 1)^T$. The residual $(1, 0, -1)^T$ has dot product zero with both a_1 and a_2 and length $\sqrt{2}$. This certifies the closest point geometrically.

5. A numerical diagnosis. Reconstruction is accurate on the input scale, but the Gram matrix deviates from identity by 10^{-2} in operator norm. The usual projector derivation uses $Q^*Q = I$; a defect this large makes the columns unsuitable for a working-precision projection. In the experiment the CGS reconstruction curve stays small even where its orthogonality curve grows: the coefficients still express the input accurately in the computed, imperfect basis. MGS uses the evolving residual in each dot product. A second pass explicitly removes components left by the first pass and updates R. Refactoring the original input with Householder transformations constructs the basis through unitary operations instead. Check both diagnostics after either remedy.

6. Filtering and rank. The first filtered block has columns $(2, 1, 0)^T/\sqrt{2}$ and zero. Its image is a line. The first normalized direction is $(2, 1, 0)^T/\sqrt{5}$; no second filtered direction exists. For the shifted polynomial the columns are $u = (2+\varepsilon, 1+\varepsilon, 0)^T/\sqrt{2}$ and εe_3. They are orthogonal, so positive-diagonal QR gives $Q = [u/\|u\|_2,\ e_3]$ and $R = \mathrm{diag}(\|u\|_2, \varepsilon)$. The image is now a plane. Here the small r_{22} records attenuation of a genuinely independent direction. Its angle with the first column is still 90°. Against perturbations comparable in norm to ε, the second direction can be unreliable even though normalization in exact arithmetic is straightforward.

5.9 End-of-session mastery checklist

- ☐ Explain $Ax = Q(Rx)$ and use it to identify the coordinate change and the preserved space.
- ☐ Give thin and full dimensions, state the uniqueness convention, and identify the orthogonal projector.
- ☐ Compute thin QR by Gram–Schmidt and verify reconstruction and orthonormality separately.
- ☐ Locate the differing dot product in CGS and MGS and explain when reorthogonalization helps.
- ☐ Derive a Householder reflector, choose a stable sign or phase, and prove unitarity.
- ☐ Eliminate a subdiagonal tail without disturbing earlier zeros and accumulate the factors in the correct order.
- ☐ Project a right-hand side, solve the triangular least-squares system, and check residual orthogonality.
- ☐ Explain the condition-number penalty incurred by explicitly forming the normal equations.
- ☐ Interpret both numerical diagnostics even when they give very different impressions.
- ☐ Explain which step in filtered subspace iteration changes the span, which step represents it in an orthonormal basis, and what happens when rank is lost.

6 Session 5 — Eigenvalues and the Spectral Theorem

Session 4 gave us orthonormal coordinates for a subspace. Can we choose coordinates that also simplify the *action of an operator*? Consider the map $(x, y) \mapsto (2x + y, x + 2y)$. Each output mixes both inputs. Yet a vector with $x = y$ is simply tripled, and one with $x = -y$ stays fixed. If we resolve the input along those two lines, even a hundred applications become a pair of scalar calculations.

Study route (90 minutes). Spend 15 minutes on invariant lines and multiplicity, 10 on change of basis, 15 on the spectral theorem, and 20 on powers, energy, and the numerical experiment. Reserve 25 minutes for the exercises and 5 for the final check. Keep a blank page for reconstructing the argument from $Av = \lambda v$ to a formula for $A^k v$.

6.1 Invariant lines and the shifted kernel

Let $T : V \to V$ be a linear operator on a finite-dimensional space over $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. A nonzero vector v is an **eigenvector** with **eigenvalue** $\lambda \in \mathbb{F}$ if $Tv = \lambda v$. The line spanned by v is invariant since $T(cv) = c\lambda v$. Conversely, an invariant line gives an eigenvector by choosing any nonzero vector on it. Over $\mathbb{R}$, $|\lambda|$ is the length multiplier. A negative sign reverses direction, and $\lambda = 0$ collapses the whole line to the origin. The zero vector itself would satisfy the equation for every scalar, hence the requirement $v \neq 0$.

An operator acts on vectors. Its matrix A acts on their coordinate columns in a chosen basis. We use v also for the coordinate column when the basis is understood. The equation $Av = \lambda v$ compares two vectors in the *same* space. For a map between different spaces the comparison would require an additional identification. Invertibility of A itself is unnecessary. Indeed, zero is an eigenvalue precisely when A is singular.

Finding the special directions. Move the right-hand side to the left. For square A, Session 1's kernel criterion gives

$$\begin{aligned}\lambda \text{ is an eigenvalue} &\iff (A-\lambda I)v = 0 \text{ for some } v \neq 0\\ &\iff \ker(A-\lambda I) \neq \{0\}\\ &\iff A-\lambda I \text{ is singular}\\ &\iff \det(A-\lambda I) = 0.\end{aligned}$$

Thus a determinant supplies candidate scalars, and solving a homogeneous system supplies their directions. We use the characteristic polynomial $\chi_A(t) = \det(tI - A)$, which is monic and has the same roots as $\det(A - tI)$. The set of eigenvalues is the **spectrum**. For each eigenvalue the **eigenspace** $E_\lambda = \ker(A - \lambda I)$ is a subspace because it is a kernel. It includes zero, and every other member is an eigenvector.

A change of coordinates $v = Sw$ gives $Av = \lambda v$ exactly when $(S^{-1}AS)w = \lambda w$. The eigenvalue and the underlying subspace belong to T. The coordinate description of that subspace changes to $S^{-1}E_\lambda$.

Example (the field matters). A real quarter-turn has matrix $R = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$. It takes every real nonzero vector to a perpendicular vector, so it preserves no real line. Its characteristic polynomial $t^2 + 1$ has roots i and $-i$ over $\mathbb{C}$. For example, $R(1,-i)^T = (i,1)^T = i(1,-i)^T$. A complex invariant line can therefore exist even when no real line has that property.

6.2 Counting roots and counting independent directions

The **algebraic multiplicity** a_λ counts occurrences of the factor $t-\lambda$ in $\chi_A(t)$. The **geometric multiplicity** $g_\lambda = \dim E_\lambda$ counts independent eigenvectors available at that eigenvalue. To see why $1 \le g_\lambda \le a_\lambda$, extend a basis of E_λ to a basis of the whole space. In those coordinates the first g_λ columns are the corresponding columns of λI, so the matrix is block upper triangular. Its characteristic polynomial contains the factor $(t-\lambda)^{g_\lambda}$.

Eigenvectors from distinct eigenvalues are linearly independent. More precisely, if $\sum_{j=1}^{s} u_j = 0$ with $u_j \in E_{\lambda_j}$ for distinct λ_j, apply $\prod_{j\neq r}(A-\lambda_j I)$. Every term disappears except $(\prod_{j\neq r}(\lambda_r - \lambda_j))u_r$. Its scalar multiplier is nonzero, so $u_r = 0$ for every r. Consequently the sum of the eigenspaces is direct. Its dimension is $\sum_\lambda g_\lambda$.

Example (the same polynomial, two outcomes). Compare

$$D = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}, \qquad J = \begin{pmatrix} 2 & 1 \\ 0 & 2 \end{pmatrix}.$$

Both have $\chi(t) = (t-2)^2$. For D, the equation $(D - 2I)v = 0$ imposes no restriction, so $E_2 = \mathbb{F}^2$. For J, $(J - 2I)(x,y)^T = (y,0)^T$, so $E_2 = \operatorname{span}\{(1,0)^T\}$. The vector $(0,1)^T$ acquires

a first component under J. There is only one independent eigendirection. We call J **defective**. Over $\mathbb{C}$ the characteristic polynomial always has n roots counted with multiplicity, but the calculation for J shows exactly where a basis vector can be missing.

6.3 Diagonalization as a change of coordinates

Suppose the columns of $V = [v_1, \dots, v_n]$ form an eigenbasis and set $\Lambda = \operatorname{diag}(\lambda_1, \dots, \lambda_n)$ in the same order. The jth column of AV is $\lambda_j v_j$, the jth column of $V\Lambda$. Hence

$$AV = V\Lambda, \qquad V^{-1}AV = \Lambda, \qquad A = V\Lambda V^{-1}.$$

Independence makes V invertible. Session 1's change-of-basis formula now has a concrete interpretation. For an input x, $V^{-1}x$ gives eigenbasis coordinates, Λ scales each coordinate independently, and V returns the result to the original coordinates. The two matrices A and Λ represent the same operator and are **similar**.

Conversely, an invertible V satisfying $AV = V\Lambda$ supplies an eigenbasis column by column. Thus A is **diagonalizable over** $\mathbb{F}$ precisely when $\mathbb{F}^n = \bigoplus_\lambda E_\lambda$, or equivalently when $\sum_\lambda g_\lambda = n$. Having n distinct eigenvalues in the field suffices. Repeated eigenvalues also work if the characteristic polynomial splits in the field and every $g_\lambda = a_\lambda$, as D illustrates.

Worked example (an oblique eigenbasis). Let $B = \begin{pmatrix} 2 & 1 \\ 0 & 1 \end{pmatrix}$. Triangularity gives roots $2, 1$. The equation for 2 gives $y = 0$ and that for 1 gives $x + y = 0$. Choose $(1, 0)^T$ and $(-1, 1)^T$, respectively. Then

$$V = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}, \quad V^{-1} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \quad V \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} V^{-1} = \begin{pmatrix} 2 & 1 \\ 0 & 1 \end{pmatrix}.$$

For input $(0, 1)^T$ the eigenbasis coordinates are $(1, 1)^T$. Scaling them gives $(2, 1)^T$ in that basis and $(1, 1)^T$ in the original basis. The eigenvectors have dot product -1. Normalizing each column keeps the oblique angle, while Gram–Schmidt across the two lines would generally spoil the eigenvector property.

6.4 Why symmetry gives orthonormal eigencoordinates

Write $A^* = \overline{A}^T$. A real matrix is **symmetric** when $A^T = A$. A complex matrix is **Hermitian** when $A^* = A$. With Session 2's convention $\langle x, y \rangle = y^* x$, linear in the first argument, Hermitian symmetry says $\langle Ax, y \rangle = \langle x, Ay \rangle$. Two consequences drive the theorem.

1. **Reality.** If $Av = \lambda v$, then $v^*Av = \lambda v^*v$. The left side equals its own conjugate since $A^* = A$, and $v^*v > 0$. Dividing shows $\lambda \in \mathbb{R}$.
2. **Orthogonality.** If $Au = \lambda u$ and $Av = \mu v$, then $\mu u^*v = u^*Av = (Au)^*v = \overline{\lambda} u^*v$. Both eigenvalues are real. If they differ, $u^*v = 0$.

Finite-dimensional spectral theorem.

Every Hermitian $A \in \mathbb{C}^{n\times n}$ has an orthonormal eigenbasis. Equivalently $A = Q\Lambda Q^*$ with $Q^*Q = QQ^* = I$ and Λ real diagonal. Every real symmetric A admits the real form $A = Q\Lambda Q^T$ with Q orthogonal.

Proof. Over $\mathbb{C}$, the characteristic polynomial has a root, so choose a unit eigenvector q. For a real symmetric matrix we can also start over $\mathbb{C}$. Its eigenvalue is real by the argument above, and the real and imaginary parts of an eigenvector separately satisfy the real eigenvector equation. At least one part is nonzero and can be normalized. Now take $x \in q^\perp$. Then $q^*Ax = (Aq)^*x = \lambda q^*x = 0$. Thus $q^\perp$ is invariant, and the restriction of A to it is again self-adjoint in its inherited inner product. Apply the same argument in this space of dimension $n-1$. Induction supplies the remaining orthonormal eigenvectors. The factorization follows from $AQ = Q\Lambda$ and $Q^{-1} = Q^*$.

Distinct eigenspaces are mutually orthogonal. Inside a repeated eigenspace, any linear combination stays in that eigenspace, so Session 2's Gram–Schmidt procedure can be used freely there. The full decomposition is an orthogonal direct sum. The inverse coordinate map is just Q^*, and a coordinate error δc satisfies $\|Q\delta c\|_2^2 = \delta c^*Q^*Q\delta c = \|\delta c\|_2^2$. This length preservation explains a numerical advantage over an oblique eigenbasis. More generally a complex **normal** matrix ($A^*A = AA^*$) admits unitary diagonalization. Its diagonal entries can be complex, as the quarter-turn illustrates.

6.5 Spectral projectors and matrix powers

For an orthonormal eigenbasis write $c_i = q_i^*v$. Session 3 identifies $P_i = q_iq_i^*$ as the orthogonal projector onto $\operatorname{span}\{q_i\}$, since $P_iv = (q_i^*v)q_i$. Orthonormality gives $P_iP_j = 0$ for $i \neq j$, $P_i^2 = P_i$, and $\sum_i P_i = I$. Expanding the matrix product gives the **spectral decomposition**

$$A = Q\Lambda Q^* = \sum_i \lambda_i q_i q_i^*.$$

For a repeated eigenvalue use $P_\lambda = \sum_{i:\lambda_i=\lambda} q_iq_i^*$. This is the unique orthogonal projector onto the entire eigenspace, independent of the chosen basis there. Then $A = \sum_\lambda \lambda P_\lambda$, summing once per distinct eigenvalue.

Derive the power formula. Multiply two copies of the factorization. The middle Q^*Q cancels, giving $A^2 = Q\Lambda^2 Q^*$. Each additional multiplication adds one diagonal factor. For $k = 0$ use $A^0 = I$ and $\Lambda^0 = I$, including on a zero eigenspace. Hence for any integer $k \geq 0$,

$$A^k v = Q\Lambda^k Q^* v = \sum_i \lambda_i^k (q_i^* v) q_i = \sum_\lambda \lambda^k P_\lambda v.$$

Read this from right to left in the matrix product. Compute the coordinates once, raise scalar multipliers to the desired power, and reassemble. No component can feed into another. A component with $|\lambda| < 1$ decays, one with $|\lambda| > 1$ grows in magnitude if present, and a negative eigenvalue alternates its sign. An initially absent component remains absent however large its multiplier.

Worked example (from a kernel to a power). Return to the opening map with

$$A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}, \qquad v = (1, 0)^T.$$

The equation $(2 - t)^2 - 1 = 0$ gives 3 and 1. The first row of $A - 3I$ gives $y = x$, and that of $A - I$ gives $y = -x$. Normalize to obtain $q_+ = (1, 1)^T/\sqrt{2}$, $q_- = (1, -1)^T/\sqrt{2}$. In that order $Q = [q_+, q_-]$ has $Q^T Q = I$ and $\Lambda = \mathrm{diag}(3, 1)$. Explicitly,

$$P_+ = \tfrac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \qquad P_- = \tfrac{1}{2}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}, \qquad A = 3P_+ + P_-.$$

The first columns of the projectors give $P_+ v = (1, 1)^T/2$ and $P_- v = (1, -1)^T/2$. Thus

$$A^k v = \tfrac{1}{2}(3^k + 1,\, 3^k - 1)^T, \qquad A^3 v = (14, 13)^T.$$

Check directly. Starting from $(1, 0)^T$, successive applications give $(2, 1)^T$, $(5, 4)^T$, then $(14, 13)^T$. The two entries approach equality *relative to their size*, because the ratio of the two eigencoordinate magnitudes is 3^{-k}. Starting instead on $y = -x$ would give a stationary vector for every k.

6.6 Positive definiteness and the geometry of energy

For Hermitian A, the scalar x^*Ax is real. We call A **positive definite** if $x^*Ax > 0$ for every nonzero x, and **positive semidefinite** if $x^*Ax \geq 0$ for every x. The same definitions with transpose apply to real symmetric matrices. Write $c = Q^*x$. Then

$$x^* A x = c^* \Lambda c = \sum_i \lambda_i |c_i|^2.$$

If every eigenvalue is positive, every nonzero x contributes at least one positive term. If all eigenvalues are nonnegative, the sum is nonnegative. Conversely, substituting each unit eigenvector yields $q_i^* A q_i = \lambda_i$. This proves both equivalences, including strict positivity in the definite case. For a semidefinite matrix, zero energy occurs exactly in its kernel.

For the running matrix, $x^T A x = 2x_1^2 + 2x_1 x_2 + 2x_2^2 = 3c_+^2 + c_-^2$. The level set $x^T A x = 1$ is an ellipse with semiaxes $1/\sqrt{3}$ along q_+ and 1 along q_-. A larger eigenvalue means that the energy reaches one after a shorter displacement. Compare the two panels below. The same direction that A stretches most is the shortest semiaxis of its unit-energy ellipse.

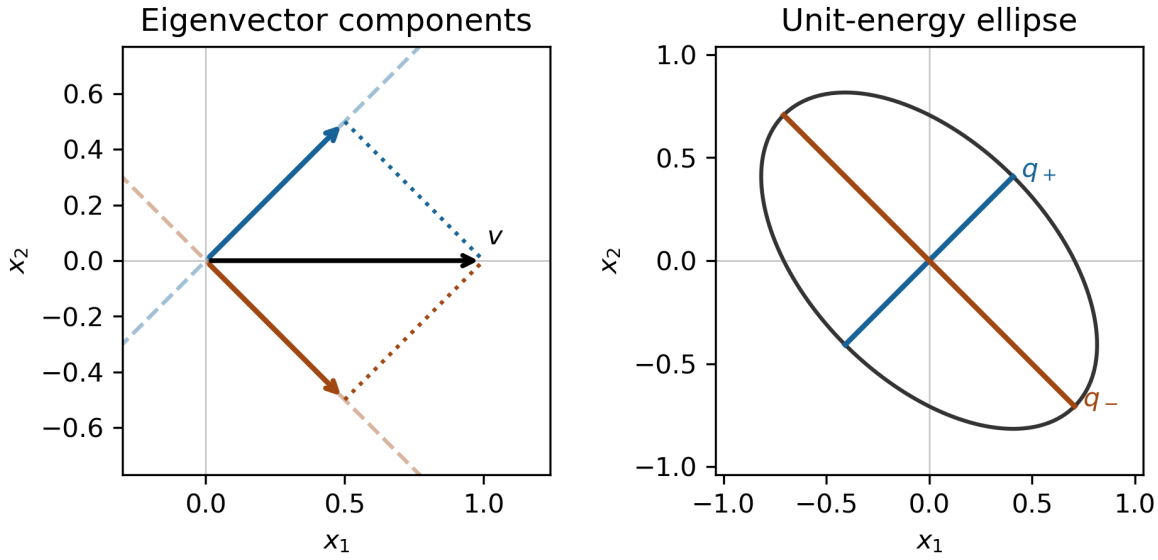


Figure 6.1: Left: the two arrows along the dashed eigenlines are the projections of v. Dotted segments complete their vector sum. Right: the semiaxes of the unit-energy ellipse lie on the same eigenlines.

In real quadratic optimization the energy $F(x) = \frac{1}{2} x^T A x - b^T x$ has gradient $Ax - b$ for symmetric A. Positive definiteness gives strictly positive curvature in every direction and a unique minimizer $A^{-1}b$. In a stiffness matrix this form can measure deformation energy. The same structure permits Cholesky factorization $A = LL^*$ and underlies the conjugate gradient method for minimizing the quadratic energy of a real SPD system.

6.7 Numerical experiment

Use a Hermitian eigensolver for the running example. NumPy returns eigenvalues in ascending order and the corresponding eigenvectors in columns. A column may have the opposite sign from a hand calculation. Its projector and its contribution to $A^k v$ remain identical.

```
import numpy as np

A = np.array([[2., 1.], [1., 2.]])
v = np.array([1., 0.])
k = 3
lam, Q = np.linalg.eigh(A)
reconstructed = (Q * lam) @ Q.T  # scale column j by lam[j]
spectral = Q @ (lam**k * (Q.T @ v))
direct = np.linalg.matrix_power(A, k) @ v

print("Eigenvalues:", lam)
print("Orthogonality error:", np.linalg.norm(Q.T @ Q - np.eye(2)))
print("Reconstruction error:", np.linalg.norm(A - reconstructed))
print("Spectral and direct results:", spectral, direct)
for computed, expected in [(Q.T @ Q, np.eye(2)),
                           (reconstructed, A), (spectral, direct)]:
    np.testing.assert_allclose(computed, expected, rtol=0, atol=1e-12)
```

```
Eigenvalues: [1. 3.]
Orthogonality error: 3.1560822113208575e-16
Reconstruction error: 6.473657049138938e-16
Spectral and direct results: [14. 13.] [14. 13.]
```

In exact arithmetic both matrix errors vanish and the two vectors equal $(14, 13)^T$. Floating-point normalization introduces rounding, so inspect the errors and compare against a tolerance. The absolute tolerance here is chosen for this small example and its scale.

6.8 What the seminar algorithms exploit

- **Power iteration.** Suppose $|\lambda_1| > |\lambda_2| \geq \cdots$ and $c_1 \neq 0$. For $j > 1$ the magnitude of component j relative to component 1 is $|c_j/c_1|\,|\lambda_j/\lambda_1|^k$. The ratio of eigenvalue magnitudes determines how rapidly directions separate. Normalization preserves those ratios. A tie in magnitude removes this separation, and a negative dominant eigenvalue can make the normalized vector alternate sign.

- **Subspace iteration.** A block X carries several input vectors. In eigencoordinates $Q^*A^kX = \Lambda^kQ^*X$, so each row of coefficients is scaled by the corresponding power. Several wanted directions can survive if the initial block represents them independently. Session 4's thin QR supplies a fresh orthonormal basis of the current block's range when it has full column rank. The target is an invariant subspace U satisfying $AU \subseteq U$.
- **Polynomial filtering.** For $p(t) = \sum_{j=0}^{d} a_j t^j$, the identity $A^jq_i = \lambda_i^j q_i$ gives $p(A)q_i = \sum_j a_j\lambda_i^j q_i = p(\lambda_i)q_i$. A Chebyshev filter is designed to make $|p(\lambda)|$ large on wanted eigenvalues relative to its size on the unwanted interval. Filtering changes the weights while preserving the eigenvectors.
- **Rayleigh–Ritz.** Let W have orthonormal columns spanning a trial subspace. The projection of AWz back into the subspace is WW^*AWz. Its small coordinate vector is Hz with $H = W^*AW$, Hermitian when A is Hermitian. If $Hz = \theta z$, the lifted residual $AWz - \theta Wz$ is orthogonal to the trial space. This gives a way to extract approximate eigenpairs from the block.

Course reading. The revision route follows §§1.3, 1.3.3 and 1.4.4 of the book. For the lecture's spectral terminology and similarity conventions see Kanschat and Scheichl, *Numerical Linear Algebra*, §3.1 and Appendix A.1.2. Section 3.2 begins the later conditioning discussion. For the real symmetric case see Scheichl and Zech, *Numerik 0*, §3.2, and Bastian, *Numerik 0*, §§5.3 and 5.6.

6.9 Independent exercises

1. **Shift and solve (3 min).** For $C = \begin{pmatrix} 0 & 2 \\ 0 & 2 \end{pmatrix}$, find both eigenvalues and eigenspaces. Give their dimensions and an eigenbasis. What happens geometrically on the zero eigenspace?
2. **A repeated root (4 min).** Let $B_s = \begin{pmatrix} 4 & s & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1 \end{pmatrix}$. Give both multiplicities at every eigenvalue and decide for which real s the matrix is diagonalizable. Identify the equation responsible for the change.
3. **Reconstruct the proof (4 min).** Starting from $A^* = A$, prove that all eigenvalues are real and that the orthogonal complement of a unit eigenvector is invariant. Explain how this yields an orthonormal eigenbasis even if the characteristic polynomial has a repeated root.
4. **Mastery problem (7 min).** For $S = \begin{pmatrix} 3 & -1 \\ -1 & 3 \end{pmatrix}$ and $w = (2, 0)^T$, find the eigenvalues and an orthonormal eigenbasis. Write the two spectral projectors and the decomposition of S. Compute S^4w, decide definiteness, and identify the shorter semiaxis of $x^TSx = 1$.
5. **A complex eigenspace (3 min).** For $H = \begin{pmatrix} 1 & i \\ -i & 1 \end{pmatrix}$, find unit eigenvectors. Classify its quadratic form and give a nonzero vector of zero energy. Check orthogonality using

the complex inner product.

6. **Design a filter (4 min).** A Hermitian operator has eigenvalues $-2, 1, 4$, with orthogonal spectral projectors P_{-2}, P_1, P_4. Construct a polynomial of degree at most two whose matrix action is exactly P_4. Prove your claim for arbitrary input. What happens if the initial vector has no component in E_4?

6.10 Worked solutions

1. Shift and solve. Triangularity gives $\chi_C(t) = t(t-2)$. At 0 the kernel equation is $2y = 0$, so $E_0 = \operatorname{span}\{(1,0)^T\}$. At 2 the first row gives $-2x + 2y = 0$, so $E_2 = \operatorname{span}\{(1,1)^T\}$. Both have dimension one. The displayed spanning vectors are independent and form an eigenbasis. The first line maps entirely to the origin.

2. A repeated root. The diagonal gives algebraic multiplicities two at 4 and one at 1. At 4 the equations reduce to $sy = 0$ and $z = 0$. When $s = 0$ the eigenspace is the entire xy-plane and has dimension two. For $s \neq 0$ it is the x-axis and has dimension one. At 1, $3y = 0$ and $3x + sy = 0$ give $x = y = 0$ for every s, leaving the z-axis. Thus the dimensions add to three exactly when $s = 0$.

3. Reconstruct the proof. For $Av = \lambda v$ with $v \neq 0$, conjugating v^*Av leaves it unchanged. Its ratio to v^*v is therefore real and equals λ. Choose a unit eigenvector q using a root of the characteristic polynomial over $\mathbb{C}$ (for real symmetric A take a nonzero real or imaginary part and normalize). For $q^*x = 0$, $q^*Ax = (Aq)^*x = \lambda q^*x = 0$. Thus the restriction to $q^\perp$ maps that space into itself and retains the self-adjoint inner product identity. Induct on its dimension, with the one-dimensional case immediate. This removes one orthogonal direction per step, regardless of root multiplicities.

4. Mastery problem. The characteristic polynomial $(t-3)^2 - 1$ gives $2, 4$. At 2 the kernel condition is $x = y$, and at 4 it is $x = -y$. Unit eigenvectors are $u = (1,1)^T/\sqrt{2}$ and $z = (1,-1)^T/\sqrt{2}$, respectively. Thus $S = 2P_2 + 4P_4$ with

$$P_2 = uu^T = \tfrac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \qquad P_4 = zz^T = \tfrac{1}{2}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}.$$

The projected inputs are $(1,1)^T$ and $(1,-1)^T$. Raising only the multipliers gives $S^4 w = 16(1,1)^T + 256(1,-1)^T = (272,-240)^T$. Both eigenvalues are positive, so S is positive definite. It doubles lengths along u and quadruples them along z. Unit energy is reached at distances $1/\sqrt{2}$ and $1/2$, respectively, giving the shorter semiaxis along z.

5. A complex eigenspace. The characteristic polynomial $(t-1)^2 - 1$ gives $0, 2$. At 0 solve $x + iy = 0$, and at 2 solve $-x + iy = 0$. Take $u = (-i,1)^T/\sqrt{2}$ and $z = (i,1)^T/\sqrt{2}$, respectively. Then $u^*z = (i \cdot i + 1)/2 = 0$. Since $H^* = H$ and its eigenvalues are nonnegative with one zero, it is positive semidefinite and singular. In particular $u^*Hu = 0$.

6. Design a filter. Both unwanted eigenvalues must be roots. Choose $p(t) = (t+2)(t-1)/18$ so $p(4) = 1$. On E_λ each power A^j acts by λ^j, hence $p(A)$ acts by $p(\lambda)$. Decompose any input into its three orthogonal projections. The first two vanish after filtering and the third is unchanged, so $p(A)v = P_4 v$. If the wanted component was absent the output is zero. This exact filter uses known eigenvalues. Interval filtering in the seminar uses bounds on a spectral interval instead.

6.11 End-of-session mastery checklist

On a blank page, check whether you can do each of the following.

- ☐ Define an eigenpair over a specified field, identify its invariant line, and derive the characteristic equation from a shifted kernel.
- ☐ Read algebraic multiplicity from a polynomial and geometric multiplicity from a kernel. Use their dimensions to decide diagonalizability.
- ☐ Derive $AV = V\Lambda$ column by column and explain the coordinate maps in $V\Lambda V^{-1}$.
- ☐ State both forms of the spectral theorem. Prove reality, orthogonality across distinct eigenvalues, and invariance of the orthogonal complement.
- ☐ Construct an orthonormal eigenbasis from a small symmetric matrix, form its projectors, and recover the matrix from their weighted sum.
- ☐ Derive $A^k = Q\Lambda^k Q^*$ and compute $A^k v$ from projected components. Predict growth, decay, sign alternation, and the effect of an absent component.
- ☐ Derive the positive-definite and semidefinite criteria from x^*Ax and identify the principal axes of a real unit-energy ellipse.
- ☐ Explain why powers favor components with larger eigenvalue magnitude and why polynomial filters can favor a different part of the spectrum.

Ready for Session 6. Redo the mastery problem without the solution. Check your projectors by verifying $P_2 + P_4 = I$, then use them to explain why $\|Sx\|_2$ for a unit vector lies between 2 and 4. Identifying the directions that attain those endpoints is the bridge to matrix norms and conditioning.

7 Session 6 — Matrix Norms and Conditioning

A solver reports a relative residual of 10^{-8}. Can you promise even one accurate digit in the solution? To decide, we need to know what happens to an error vector when the matrix acts on it. If the matrix nearly erases one direction, checking the equations can hide precisely the error we care about.

Revision route (75 minutes). Spend 12 minutes on amplification and spectral radius, 12 on perturbations, 14 on residuals and the experiment, and 10 on backward error, stability, and synthesis. Reserve 27 minutes for the exercises. The mastery task is to reconstruct the relative error bound and exhibit a system where equality holds even though the residual tends to zero.

7.1 Measuring amplification

A residual is a vector with one entry per equation. To compare it with the data, we need a common scale. A **vector norm** assigns a nonnegative real size to each vector, with $\|x\| = 0$ exactly when $x = 0$, $\|\alpha x\| = |\alpha|\,\|x\|$, and $\|x + y\| \le \|x\| + \|y\|$. For real or complex coordinates, the usual choices are

$$\|x\|_1 = \sum_i |x_i|, \qquad \|x\|_2 = \left(\sum_i |x_i|^2 \right)^{1/2}, \qquad \|x\|_\infty = \max_i |x_i|.$$

They measure total absolute size, Euclidean length, and the largest coordinate magnitude. Their unit balls in the real plane are a diamond, a disk, and an axis-aligned square. Choose a norm according to the error you want to control, and keep that choice throughout an estimate.

7.1.1 From vector size to operator size

For a matrix A mapping one coordinate space into another, its **induced norm** is the greatest amplification of input size,

$$\|A\| = \sup_{v \neq 0} \frac{\|Av\|}{\|v\|} = \max_{\|v\|=1} \|Av\|.$$

Homogeneity allows normalization. In finite dimensions the unit sphere is compact, so the continuous function $v \mapsto \|Av\|$ attains its maximum. Every individual ratio lies below that maximum. Thus $\|Av\| \leq \|A\|\,\|v\|$, also for $v = 0$. This step converts an exact vector identity into a scalar error bound. Applying it twice gives $\|ABv\| \leq \|A\|\,\|B\|\,\|v\|$. Maximizing over unit vectors proves **submultiplicativity**, $\|AB\| \leq \|A\|\,\|B\|$, for compatible dimensions and norms.

The subscript p denotes the same p-norm on the input and output. Writing a_{ij} for the matrix entries gives

$$\|A\|_1 = \max_j \sum_i |a_{ij}|, \qquad \|A\|_\infty = \max_i \sum_j |a_{ij}|.$$

Why columns and rows? In the 1-norm, Av combines columns with weights v_j. The triangle inequality bounds its size by the largest column sum times $\sum_j |v_j|$. A coordinate vector selecting a largest column attains the bound. In the infinity norm, each output coordinate is a row acting on v. Its magnitude is at most the absolute row sum times $\|v\|_\infty$. For a maximizing row, choose unit-modulus input entries whose phases make all its nonzero terms positive real. This attains the row bound.

For the Euclidean norm, let A^* denote the conjugate transpose (ordinary transpose over the reals). Since A^*A is Hermitian positive semidefinite, Session 5 gives an orthonormal eigenbasis. In that basis, v^*A^*Av for a unit vector is a weighted average of eigenvalues with weights equal to squared coordinate magnitudes. The largest average is the largest eigenvalue, attained on its eigenvector. Hence

$$\|A\|_2^2 = \max_{\|v\|_2=1} \|Av\|_2^2 = \max_{\|v\|_2=1} v^*A^*Av = \lambda_{\max}(A^*A).$$

The square roots of these eigenvalues are the **singular values** (for square matrices). In particular $\|A\|_2 = \sigma_{\max}(A)$, the longest semiaxis of the image of the unit sphere. Unitary changes of coordinates preserve Euclidean length and hence preserve these stretching factors.

Worked example. Take $B = \begin{pmatrix} 0 & 3 \\ 0 & 4 \end{pmatrix}$. Its largest column sum is 7, its largest row sum is 4, and $B^*B = \operatorname{diag}(0, 25)$ gives $\|B\|_2 = 5$. Indeed $B(s,t)^T = (3t, 4t)^T$ has length $5|t|$, maximized on the unit circle at $(0, \pm 1)^T$.

The **Frobenius norm** $\|A\|_F = (\sum_{ij} |a_{ij}|^2)^{1/2}$ measures the Euclidean size of the whole array. It is submultiplicative (apply Cauchy–Schwarz to each entry of a product and sum). For the

n-dimensional identity I, every norm induced by the same vector norm on both sides satisfies $\|I\| = 1$, whereas $\|I\|_F = \sqrt{n}$. This rules out such an induced-norm interpretation for $n > 1$.

7.2 One application versus repeated applications

The **spectral radius** of a square matrix is $\rho(A) = \max_{\lambda \in \sigma(A)} |\lambda|$, where $\sigma(A)$ is the set of complex eigenvalues. For an eigenpair $Av = \lambda v$ with $v \neq 0$, we have $|\lambda| \, \|v\| = \|Av\| \leq \|A\| \, \|v\|$. Taking the maximum gives $\rho(A) \leq \|A\|$ for any induced norm on the complex space. For real matrices the usual $1, 2, \infty$ formulas give the same operator norms over either field.

For Hermitian A, an orthonormal eigenbasis diagonalizes A and A^*A with respective entries λ_i and $|\lambda_i|^2$. Thus $\|A\|_2 = \rho(A)$. The same proof applies to every **normal** matrix ($A^*A = AA^*$), by unitary diagonalization.

Revisit B. Its triangular form gives eigenvalues $0, 4$, so $\rho(B) = 4 < 5 = \|B\|_2$. Its most stretched input $(0, 1)^T$ changes direction. Indeed $B^2 = 4B$, so $\|B^k\|_2 = 5 \cdot 4^{k-1}$ for $k \geq 1$. The one-step maximum is 5, yet the long-run per-step growth factor is 4. More generally, for a stationary iteration with error recurrence $e_{k+1} = Te_k$, the error tends to zero for every initial error exactly when $\rho(T) < 1$. A norm bound $\|T\| < 1$ guarantees contraction at every step. Session 8 studies this distinction.

7.3 Conditioning a linear solve

Assume from here that A is square and invertible, $Ax = b$, and all matrix norms are induced by the chosen vector norm unless marked otherwise. Relative measures below assume $b \neq 0$, which also gives $x \neq 0$.

7.3.1 What changes when the data change?

Hold A fixed and replace b by $b+\delta b$. Subtracting the original equations from $A(x+\delta x) = b+\delta b$ gives $A\delta x = \delta b$. The absolute sensitivity is governed by $\delta x = A^{-1}\delta b$. To compare *relative* changes, insert the scale of the original data,

$$\frac{\|\delta x\|}{\|x\|} \leq \|A^{-1}\| \frac{\|b\|}{\|x\|} \frac{\|\delta b\|}{\|b\|} \leq \underbrace{\|A^{-1}\| \, \|A\|}_{\kappa(A)} \frac{\|\delta b\|}{\|b\|}.$$

The last step uses $\|b\| = \|Ax\| \leq \|A\| \, \|x\|$. The **condition number** $\kappa(A)$ is dimensionless because the units of forward and inverse amplification cancel. Scaling A by any nonzero scalar leaves it unchanged. Also $1 = \|I\| \leq \|A\| \, \|A^{-1}\|$.

For this particular b, the sharp worst-direction factor is $\|A^{-1}\|\,\|b\|/\|x\|$. Maximizing over x as well yields $\kappa(A)$. A large value describes possible amplification. The actual outcome depends on both the original solution direction and the direction of the perturbation. No solver entered the derivation.

7.3.2 Stretching factors and inversion

As a unit vector ranges over all directions, its squared image length ranges from $\lambda_{\min}(A^*A)$ to $\lambda_{\max}(A^*A)$. Write $\sigma_{\min}$ and $\sigma_{\max}$ for the corresponding positive square roots. Substituting $w = Av$ in the definition of the inverse norm gives

$$\|A^{-1}\|_2 = \max_{v\neq 0} \frac{\|v\|_2}{\|Av\|_2} = \frac{1}{\sigma_{\min}}, \qquad \kappa_2(A) = \frac{\sigma_{\max}}{\sigma_{\min}}.$$

Thus conditioning measures the aspect ratio of the image ellipsoid. For Hermitian positive-definite A, all eigenvalues are positive and $A^*A = A^2$, so $\kappa_2(A) = \lambda_{\max}(A)/\lambda_{\min}(A)$. Inversion multiplies each eigenvector by the reciprocal eigenvalue. For a general invertible matrix the singular-value formula applies, even when eigenvectors fail to span the space.

Worked example. Set $A_\varepsilon = \operatorname{diag}(1, \varepsilon)$ with $0 < \varepsilon \leq 1$. The unit circle maps to an ellipse with horizontal semiaxis 1 and vertical semiaxis ε. Inversion stretches a vertical data perturbation by $1/\varepsilon$ and preserves a horizontal one. For $\varepsilon = 10^{-4}$ the corresponding amplification factors are 10^4 and 1, and $\kappa_2 = 10^4$. By comparison, $10^{-4}I$ amplifies *every* direction by 10^4 under inversion but has condition number 1. Absolute amplification and relative sensitivity answer different questions.

7.3.3 If the matrix changes too

Suppose $(A+\delta A)(x+\delta x) = b+\delta b$. Expansion gives the exact identity $A\delta x = \delta b - \delta A\, x - \delta A\, \delta x$. The extra terms express a change in the operator itself. Put $d_A = \|\delta A\|/\|A\|$ and $d_b = \|\delta b\|/\|b\|$. Apply A^{-1}, take norms, and move the term containing $\|\delta x\|$ to the left. When $\kappa(A)d_A < 1$, the result is

$$\frac{\|\delta x\|}{\|x\|} \leq \frac{\kappa(A)(d_A + d_b)}{1 - \kappa(A)d_A}.$$

The condition ensures invertibility of $A + \delta A$ because $\|A^{-1}\delta A\| < 1$ and $I + A^{-1}\delta A$ has a convergent geometric-series inverse. To first order in small perturbations the numerator gives the bound. For fixed A the denominator is exactly one.

7.4 The error equation

Given a candidate $\hat{x}$, form the **residual** $r = b - A\hat{x}$ from known quantities. The **forward error** $e = x - \hat{x}$ involves the unknown exact solution. Linearity connects them:

$$Ae = Ax - A\hat{x} = b - A\hat{x} = r, \qquad \boxed{e = A^{-1}r}.$$

This identity says that recovering the missing correction is itself a solve with the original matrix. It does not require forming an inverse. Taking norms gives $\|e\| \le \|A^{-1}\|\,\|r\|$. For $b \neq 0$, insert $\|b\|/\|b\|$ and use $\|b\|/\|x\| \le \|A\|$:

$$\boxed{\frac{\|x - \hat{x}\|}{\|x\|} \le \kappa(A)\frac{\|b - A\hat{x}\|}{\|b\|}.}$$

The second factor measures the defect on the scale of the equations. The first translates that defect into a worst-case relative uncertainty in solution space. For example, a verified $\kappa(A) \le 20$ and a relative residual at most 10^{-8} certify relative error at most $2 \cdot 10^{-7}$. An estimated condition number gives a diagnostic estimate unless it is known to bound the true value from above.

7.4.1 One residual, two directions

Take A_ε as above, $x = (1, 0)^T$, $b = (1, 0)^T$, and $\hat{x} = (1, 1)^T$. Then

$$A_\varepsilon\hat{x} = (1, \varepsilon)^T, \qquad r = (0, -\varepsilon)^T, \qquad e = (0, -1)^T.$$

Both reference norms are 1. The relative residual is ε, the relative error is 1, and the upper bound is $(1/\varepsilon)\varepsilon = 1$. **Equality holds at every scale.** An entire spurious vertical component survives in the computed solution while contributing only ε to the equations. Taking $\varepsilon \downarrow 0$ makes the residual arbitrarily small while keeping every matrix invertible.

Keep the matrix and b fixed but choose $\hat{x} = (1 - \varepsilon, 0)^T$. The equally small horizontal residual $(\varepsilon, 0)^T$ now corresponds to relative error ε. Conversely, for the well-conditioned system $A = I$, $b = (1, 0)^T$, every residual *equals* its error vector. The first approximation $(1, 1)^T$ then has relative residual 1, immediately exposing the error.

7.4.2 Experiment in two dimensions

Predict the slope of each curve as ε decreases. The experiment prescribes a candidate so that it isolates what a residual can certify independently of how the candidate was obtained.

```
import numpy as np
import matplotlib.pyplot as plt

eps_values = 10.0 ** (-np.arange(0, 13, dtype=float))
x = np.array([1.0, 0.0])
x_hat = np.array([1.0, 1.0])
measurements = []

for eps in eps_values:
    A = np.diag([1.0, eps])
    b = A @ x
    relative_residual = np.linalg.norm(b - A @ x_hat) / np.linalg.norm(b)
    relative_error = np.linalg.norm(x - x_hat) / np.linalg.norm(x)
    condition = np.linalg.cond(A, 2)
    measurements.append((condition, relative_residual, relative_error,
                         condition * relative_residual))

condition, residual, error, bound = np.array(measurements).T
for observed, predicted in zip((condition, residual, error, bound),
                               (1 / eps_values, eps_values, 1.0, 1.0)):
    np.testing.assert_allclose(observed, predicted, rtol=1e-13, atol=0)

fig, ax = plt.subplots(figsize=(6.2, 3.4), constrained_layout=True)
ax.loglog(eps_values, condition, "^-", label="Condition number")
ax.loglog(eps_values, residual, "v--", label="Relative residual")
ax.loglog(eps_values, error, "-", color="black", label="Relative error")
ax.loglog(eps_values, bound, "o", mfc="none", color="black", label="Bound")
ax.invert_xaxis()
ax.set(xlabel=r"Vertical stretching factor $\varepsilon$", ylabel="Magnitude")
ax.grid(True, which="major", alpha=0.25)
ax.legend(fontsize=8, ncol=2)
plt.show()
```

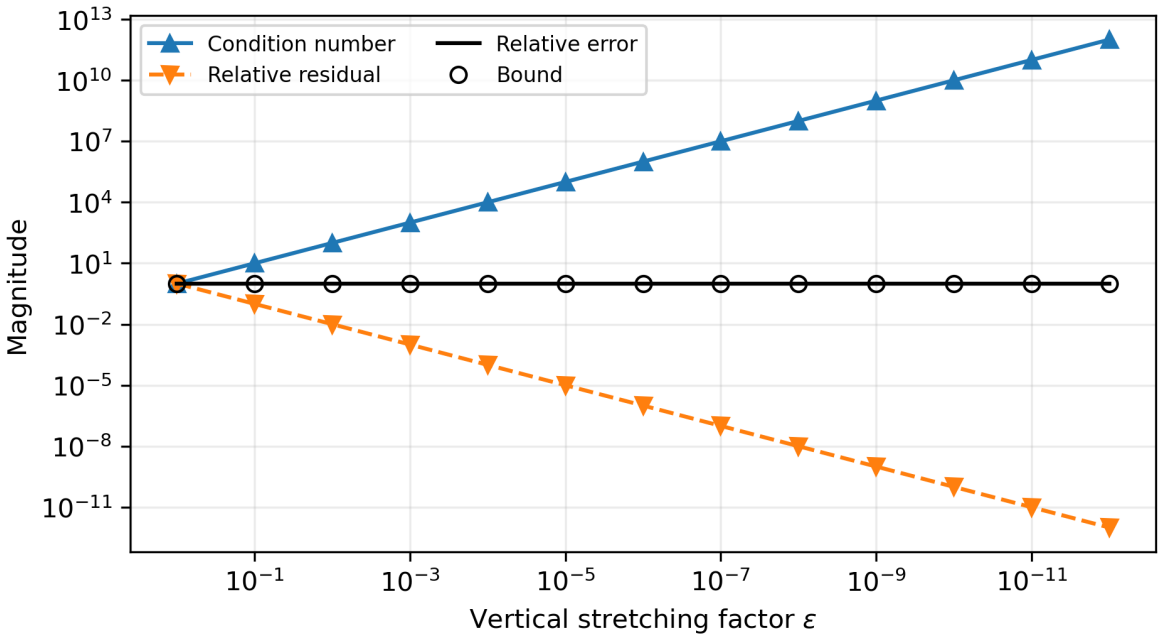


Figure 7.1: As the vertical stretching factor decreases, the relative residual shrinks while the relative error remains one. Open circles mark the upper bound, which is attained at every point.

In exact arithmetic the four quantities are $1/\varepsilon, \varepsilon, 1, 1$. At 10^{-8} this gives $10^{8}, 10^{-8}, 1, 1$. The code computes all four from the arrays and checks agreement to relative tolerance 10^{-13}. The range down to 10^{-12} stays far above underflow. The vertical residual subtracts a nonzero entry from zero, so it avoids the cancellation that can complicate real residual evaluations.

7.5 Backward error and stability

Ask how the *data* would have to change to make the computed answer exact. With A fixed, $A\hat{x} = b - r$, so the unique admissible change is $\delta b = -r$. The relative **right-hand-side backward error** is therefore $\eta_b = \|r\|/\|b\|$. For the vertical error above, $\hat{x}$ is an exact answer for data only ε away in relative terms.

A different model allows simultaneous, unstructured changes to both inputs. In the Euclidean and induced spectral norms let $\eta_{A,b}$ be the smallest t for which $(A + \delta A)\hat{x} = b + \delta b$, with $\|\delta A\|_2 \le t\|A\|_2$ and $\|\delta b\|_2 \le t\|b\|_2$. Then

$$\eta_{A,b} = \frac{\|r\|_2}{\|A\|_2\|\hat{x}\|_2 + \|b\|_2}.$$

Indeed $r = \delta A\hat{x} - \delta b$ gives the lower bound by the triangle inequality. For nonzero $r, \hat{x}$, set $q = r/\|r\|_2$, $v = \hat{x}/\|\hat{x}\|_2$ and take $\delta A = t\|A\|_2 q v^*$, $\delta b = -t\|b\|_2 q$. The two contributions point along q and attain the bound at the stated t. Zero residual gives zero backward error, and $\hat{x} = 0$ requires $\delta b = -b$. Restrictions such as symmetry, sparsity, or componentwise tolerances define other models. To bound forward error from $\eta_{A,b}$, use the simultaneous perturbation estimate with $d_A, d_b \le \eta_{A,b}$ and check its denominator.

Backward stability means that an algorithm, over its stated class of inputs, returns exact solutions for admissibly perturbed data with backward error bounded on the scale of Cu, where u is unit roundoff and C is a suitably modest factor under the stated assumptions. A single observed backward error assesses one result. An algorithmic stability claim requires a bound across inputs. On an ill-conditioned system, even an error on the scale of roundoff can be amplified by the inverse.

Worked example — same problem, different algorithms. Solve the scalar system $1x = 1$ in binary64 arithmetic with rounding to nearest, ties to even. Direct division gives 1. Another procedure computes $((b + M) - M)/A$ with $M = 2^{53}$. This is the same formula in exact arithmetic. In binary64, the consecutive representable numbers at and just above M are M and $M + 2$. The midpoint $M + 1$ rounds to M, so the procedure returns 0. Here $\kappa = 1$, and both the relative residual and relative forward error of the second answer are 1. The damaging operation was introduced by the algorithm. Session 4 used the same diagnostic discipline for QR by checking *both* the reconstruction residual $\|A - \widehat{Q}\widehat{R}\|$ and the orthogonality defect $\|I - \widehat{Q}^*\widehat{Q}\|$.

7.6 Central Synthesis — Conditioning, Backward Error, and Forward Error

Start with the candidate produced by the algorithm. Substitution into the equations gives a defect. Specify which inputs may change, then translate the defect into a backward error. A sensitivity estimate for **that same data model** converts the backward error into a statement about forward accuracy. In the fixed-A model the entire argument fits in one line:

$$A\hat{x} = b - r \quad \Longrightarrow \quad x - \hat{x} = A^{-1}r \quad \Longrightarrow \quad \frac{\|x - \hat{x}\|}{\|x\|} \le \kappa(A)\eta_b.$$

Reading a solver report. The following quantities assume consistent norms and nonzero reference data where needed.

Diagnostic	Quantity	Available without x?
Absolute residual	$\|b - A\hat{x}\|$	Yes, in the units of b
Relative residual	$\|r\|/\|b\|$	Yes, relative to data scale

Diagnostic	Quantity	Available without x?
Absolute forward error	$\|x - \hat{x}\|$	Usually requires a reference solution
Relative forward error	$\|x - \hat{x}\|/\|x\|$	Usually requires a reference solution
Condition number	$\|A\| \, \|A^{-1}\|$	Yes, often estimated using solves
Backward error	η_b or $\eta_{A,b}$	Yes, after specifying the model

A small upper bound certifies accuracy. A large upper bound leaves accuracy undecided, as the two residual directions for A_ε illustrate. Good conditioning helps only when the algorithm delivers small backward error. In practice the computed residual also contains rounding from the matrix-vector product and subtraction. If $A\hat{x}$ is very close to b, that rounding can dominate the defect. Recomputing the residual in higher precision, as in mixed-precision iterative refinement, makes the diagnostic more informative.

7.6.1 Connections to the lecture and seminar

- **Direct methods (Session 7).** For a backward-stable QR solve, or an LU solve with suitably controlled growth, a small backward error and large condition number can coexist. Pivoting and the choice of factorization concern the algorithm. Sensitivity of the original solve is common to all of them.
- **Stationary iterations (Session 8).** The iteration matrix governs how errors propagate between steps. The system matrix governs how the current residual translates into solution error. A residual stopping tolerance needs the latter relationship.
- **Eigensolvers (Sessions 9 and 11).** For a unit candidate $\hat{v}$ and approximate eigenvalue $\hat{\lambda}$, the eigenpair residual $s = A\hat{v} - \hat{\lambda}\hat{v}$ is computable. For Hermitian A, expanding in an orthonormal eigenbasis gives $\|s\|_2^2 = \sum_i |\lambda_i - \hat{\lambda}|^2 |c_i|^2$, where c_i are the coefficients of $\hat{v}$. If several eigenvalues cluster near $\hat{\lambda}$, many mixtures give small residuals. Accuracy of an individual eigenvector requires spectral separation.
- **Chebyshev filtering (Sessions 9–11).** In exact arithmetic a polynomial $p(A)$ weights each eigenvector component by $p(\lambda_i)$. Separation between wanted and unwanted filter values drives subspace improvement. In finite precision, polynomial evaluation introduces errors, and unequal amplification can make block columns nearly dependent. QR restores a usable basis to the extent that its directions are still resolved. Check basis orthogonality and Ritz residuals alongside the predicted filter response.

Course reading. The Review I route follows §§1.3, 1.3.3, and 1.4.4 of the book. Kanschat and Scheichl, *Numerical Linear Algebra*, Appendix B.1 gives the operator-norm and spectral-radius framework, with Chapters 3–4 supplying eigenvalue and iteration contexts. Scheichl

and Zech, *Numerik 0*, §§2.1–2.2 and 3.1–3.3 develop conditioning, stability, matrix norms, and perturbation bounds. Bastian, *Numerik 0*, §§3, 5, and 6 connects finite arithmetic to linear-system sensitivity.

7.7 Independent exercises

1. **Three views of one matrix (3 min).** For $C = \begin{pmatrix} 1 & -2 \\ 0 & 0 \end{pmatrix}$, compute the induced $1, \infty, 2$ norms and $\rho(C)$. Exhibit a Euclidean unit input attaining the spectral norm.
2. **A first step that grows (3 min).** For $T = \begin{pmatrix} 0 & 6 \\ 0 & 0 \end{pmatrix}$, find $\rho(T)$ and $\|T\|_2$. Starting from $(0,1)^T$, calculate the first two errors in $e_{k+1} = Te_k$. Explain their behavior.
3. **Two data errors (4 min).** Take $A = \operatorname{diag}(4, 1/2)$, $x = (1,0)^T$, and $b = Ax$. Compute $\kappa_2(A)$. Perturb b first by $(0.04, 0)^T$, then by $(0, 0.04)^T$. Find the relative data and solution changes. Which direction attains the global bound?
4. **Reconstruct the certificate (5 min).** Given $Ax = b$ and a candidate y, derive the absolute and relative forward-error bounds from the definition of the induced norm. State every nonzero or invertibility assumption. Recover the relative right-hand-side perturbation estimate by choosing y appropriately.
5. **Audit an accuracy claim (6 min).** A report gives a relative residual of 10^{-9} and an estimated κ_2 of 10^9. It claims nine accurate digits. Assess the claim, then construct two invertible diagonal-system examples with those exact diagnostics, one with relative error 1 and one with relative error 10^{-9}. You may use the same matrix for both.
6. **Two solvers and two data models (6 min).** A matrix has verified $\kappa_2(A) = 3$. On the same data, solvers P and Q produce relative residuals 10^{-14} and 0.2, respectively. Give upper *and lower* bounds on their relative forward errors. Which stability claims about the algorithms follow? For Q, suppose additionally that $\|A\|_2\|\hat{x}\|_2 = 2\|b\|_2$. Calculate the simultaneous backward error and explain why it differs from 0.2.

7.8 Worked solutions

1. Three views. The absolute column sums are $1, 2$ and row sums are $3, 0$, giving $\|C\|_1 = 2$ and $\|C\|_\infty = 3$. A unit input $(s,t)^T$ produces length $|s-2t| \le \sqrt{5}$ by Cauchy–Schwarz. Equality holds at $(1,-2)^T/\sqrt{5}$, so $\|C\|_2 = \sqrt{5}$. Triangularity gives eigenvalues $1, 0$ and spectral radius 1.

2. A first step that grows. Both eigenvalues are zero. Acting on $(s,t)^T$ gives $(6t, 0)^T$, so $\|T\|_2 = 6$. Starting on the vertical axis gives $e_1 = (6,0)^T$ and $e_2 = 0$. The maximally stretched direction maps into the kernel. Indeed $T^2 = 0$ for all initial vectors, consistent with $\rho(T) = 0$.

3. Two data errors. The singular values are $4, 1/2$, so $\kappa_2 = 8$. Since $b = (4, 0)^T$, both relative perturbations are $0.04/4 = 0.01$. Multiplication by $A^{-1} = \mathrm{diag}(1/4, 2)$ gives corrections $(0.01, 0)^T$ and $(0, 0.08)^T$. The original solution has unit norm, so the relative changes are 0.01 and 0.08. The vertical perturbation attains $8 \cdot 0.01$.

4. Reconstruct the certificate. Let A be square and invertible and take a vector norm with its induced matrix norm. Set $e = x - y$ and $r = b - Ay$. Then $Ae = r$, hence $e = A^{-1}r$. The defining supremum bounds $\|e\|$ by $\|A^{-1}\| \, \|r\|$ (including $r = 0$). For $b \neq 0$ we have $x \neq 0$ and $\|b\|/\|x\| \leq \|A\|$. Dividing the absolute bound by $\|x\|$ and inserting this ratio gives $\|e\|/\|x\| \leq \kappa(A)\|r\|/\|b\|$. If $Ay = b + \delta b$, then $r = -\delta b$ and $e = -(y - x)$. Taking norms gives the desired relative perturbation estimate.

5. Audit an accuracy claim. Even if the condition estimate is exact, the upper bound is 1. An estimated condition number introduces additional uncertainty. Choose $A = \mathrm{diag}(1, 10^{-9})$ and $x = b = (1, 0)^T$. With $y = (1, 1)^T$, $r = (0, -10^{-9})^T$ and $e = (0, -1)^T$. With $y = (1 - 10^{-9}, 0)^T$, $r = e = (10^{-9}, 0)^T$. Both candidates have the reported diagnostics in exact arithmetic. Their relative errors are 1 and 10^{-9} respectively. The report omits the direction information that distinguishes them.

6. Two solvers. Write $h = \|r\|/\|b\|$ and $f = \|e\|/\|x\|$. From $r = Ae$ and $x = A^{-1}b$ we get $h \leq \|A\| \, \|A^{-1}\| f$. Combine this with $f \leq \kappa(A)h$ to obtain $h/3 \leq f \leq 3h$. P has $10^{-14}/3 \leq f \leq 3 \cdot 10^{-14}$. Q has $1/15 \leq f \leq 0.6$. The lower bound establishes an actual loss of accuracy for Q. Conditioning is identical in both runs. P delivered small backward error on this instance, which alone cannot establish backward stability over a class of inputs. Q's result contradicts any claimed fixed-A stability guarantee requiring a relative backward error smaller than 0.2 on these data. For the simultaneous model the denominator is $3\|b\|_2$, giving $\eta_{A,b} = 0.2/3 = 1/15$. Allowing perturbations in the operator as well as the data makes a smaller common budget sufficient.

7.9 End-of-session mastery checklist

Before Session 7, close the chapter and check that you can independently:

- Define an induced norm, prove $\|Av\| \leq \|A\| \, \|v\|$, and use the same argument for a product.
- Calculate induced $1, \infty, 2$ norms for a small matrix and identify a maximally stretched input.
- Define $\rho(A)$, prove the induced-norm bound, and explain the Euclidean equality for normal matrices.
- Interpret $\kappa(A)$ through inversion, singular values, and directional sensitivity.
- Derive the relative right-hand-side perturbation estimate and identify what changes when A varies.
- Derive $e = A^{-1}r$ and the relative forward-error bound, stating its assumptions.
- Construct a family with vanishing relative residual and order-one relative forward error.

- Distinguish forward error, right-hand-side backward error, and simultaneous backward error.
- Explain backward stability and why ill-conditioning can limit the forward accuracy of a stable solve.
- Explain why Ritz residuals, spectral separation, and basis orthogonality matter in a filtered eigensolver.

Ready for direct methods. Given a computed LU or QR solution, you can now say which defect you can measure, which perturbation it represents, and what further information would justify a claim about solution accuracy.

8 Session 7 — Direct Numerical Methods

Given $Ax = b$, how can we turn the equations into a form we can solve one unknown at a time? Session 1 supplies the algebraic idea of composing linear maps. Session 6 supplies a separate question about the answer we compute: how much uncertainty does a small residual leave in the solution? A direct solver brings these questions together. Its transformations determine the work, the intermediate numbers and, ultimately, how much accuracy survives.

One-hour route. Spend 12 minutes on elimination, substitution and LU, 9 on pivoting and the experiment results, 8 on Cholesky, and 8 on comparing and assessing solvers. Reserve 23 minutes for the exercises. The experiment runs during rendering. Studying its implementation is optional.

Throughout, $A \in \mathbb{F}^{n\times n}$ with $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$, and A^* denotes the conjugate transpose. Derivations use exact arithmetic unless rounding is mentioned.

8.1 Elimination changes the equations

In $Ax = b$, applying an invertible map E to the equation space gives $EAx = Eb$. Every solution satisfies the transformed equations, and multiplication by E^{-1} recovers the originals. In particular, the unknown x keeps its original coordinates. Each elementary row operation has an explicit inverse: swap the same rows again, scale by the reciprocal of a nonzero scale factor, or undo a row addition by subtracting the same multiple. Carry the right-hand side through every operation.

Starting with $A^{(1)} = A$, suppose the first $k-1$ columns already have zeros below their diagonals. The **pivot** $a_{kk}^{(k)}$ is the coefficient used to eliminate the remaining entries below it in column k. For a nonzero pivot and each $i > k$, set

$$m_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}, \qquad R_i \leftarrow R_i - m_{ik} R_k.$$

The new column entry is zero by construction. The pivot row has zeros in the preceding columns, so those columns stay triangular. **Gaussian elimination** stops at upper-triangular form and uses substitution. **Gauss–Jordan elimination** continues by normalizing pivots and clearing entries above them to obtain reduced row echelon form.

Running example. Start with the augmented matrix below. Subtract twice row 1 from row 2 and add row 1 to row 3:

$$\left[\begin{array}{ccc|c} 2 & 1 & 1 & 3 \\ 4 & 5 & 0 & 14 \\ -2 & 2 & 5 & -3 \end{array}\right] \longrightarrow \left[\begin{array}{ccc|c} 2 & 1 & 1 & 3 \\ 0 & 3 & -2 & 8 \\ 0 & 3 & 6 & 0 \end{array}\right].$$

The first two multipliers were 2 and -1. Now subtract row 2 from row 3. The multiplier is 1, using the *current* entries 3/3. The final system is

$$Ux = y, \qquad U = \begin{pmatrix} 2 & 1 & 1 \\ 0 & 3 & -2 \\ 0 & 0 & 8 \end{pmatrix}, \qquad y = \begin{pmatrix} 3 \\ 8 \\ -8 \end{pmatrix}.$$

The three diagonal entries $2, 3, 8$ are the pivots, including the last coefficient used in the solve.

8.2 Triangular systems have a natural order

The last equation of $Ux = y$ determines x_n. At row i, all terms to the right of $u_{ii}x_i$ are already known. Isolating that term gives **backward substitution**. A lower-triangular system $Ly = b$ has the reverse dependency and gives **forward substitution**:

$$x_i = \frac{y_i - \sum_{j=i+1}^{n} u_{ij}x_j}{u_{ii}}, \quad i = n, n-1, \dots, 1,$$

$$y_i = \frac{b_i - \sum_{j=1}^{i-1} \ell_{ij}y_j}{\ell_{ii}}, \qquad i = 1, 2, \dots, n.$$

An empty sum is zero. For a **unit lower-triangular** matrix every $\ell_{ii} = 1$, so the divisions disappear. A square triangular matrix has a unique solution for every right-hand side exactly when every diagonal entry is nonzero: its determinant is their product. With a zero on the diagonal, consistency determines whether there are infinitely many solutions or none.

In the example the last equation gives $x_3 = -1$, the second gives $x_2 = (8 + 2x_3)/3 = 2$, and the first gives $x_1 = (3 - x_2 - x_3)/2 = 1$. Across all rows there are $n(n-1)/2$ off-diagonal products to subtract, hence $O(n^2)$ work. Substitution computes the action of an inverse on the one vector we need. Forming the whole inverse would solve for all its columns and introduce extra work and rounding.

8.3 LU remembers the elimination

Let e_k be the kth coordinate vector and m_k the column containing m_{ik} for $i > k$ and zeros elsewhere. Since $e_k^T A^{(k)}$ extracts row k, one stage is

$$A^{(k+1)} = E_k A^{(k)}, \qquad E_k = I - m_k e_k^T, \qquad E_k^{-1} = I + m_k e_k^T.$$

The inverse follows from $e_k^T m_k = 0$. After the last stage,

$$U = E_{n-1} \cdots E_1 A, \qquad A = LU, \qquad L = E_1^{-1} \cdots E_{n-1}^{-1}.$$

The inverse factors occur in reverse order from the elimination factors. In this product every cross term with indices $j < k$ contains $e_j^T m_k = 0$. Thus $L = I + \sum_k m_k e_k^T$: write each multiplier, with its original sign, below the diagonal in the column where it was used.

Verification and reuse. In the running example,

$$\underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 1 & 1 \end{pmatrix}}_{L} \underbrace{\begin{pmatrix} 2 & 1 & 1 \\ 0 & 3 & -2 \\ 0 & 0 & 8 \end{pmatrix}}_{U} = \begin{pmatrix} 2 & 1 & 1 \\ 4 & 5 & 0 \\ -2 & 2 & 5 \end{pmatrix} = A.$$

For instance, row 3 of the product is minus row 1 of U plus rows 2 and 3. Solve $Ly = b$ from the top: $y_1 = 3$, $y_2 = 14 - 2(3) = 8$, and $y_3 = -3 + 3 - 8 = -8$. This recovers exactly the transformed right-hand side. Backward substitution then gives the previously obtained $x = (1, 2, -1)^T$. For a new load vector b, only these two substitutions change. The same factors solve a whole block $AX = B$ by solving $LY = B$ and $UX = Y$ column by column or in a block.

When does this work without exchanges? For nonsingular A, unpivoted elimination succeeds exactly when every leading principal minor $d_k = \det A_{1:k,1:k}$ is nonzero. To see why, set $d_0 = 1$. Once the first $k-1$ stages succeed, row additions restricted to the leading k rows triangularize the leading k-by-k block. Thus $d_k = d_{k-1} u_{kk}$. Induction gives the criterion and the pivot formula $u_{kk} = d_k / d_{k-1}$. Nonsingularity alone guarantees only $d_n \neq 0$.

For a real dense matrix, updating a trailing j-by-j block takes about $2j^2$ flops when a multiplication and a subtraction count separately. Summing over $j = 1, \ldots, n-1$ gives $2n^3/3 + O(n^2)$; multiplier divisions affect lower-order terms. Each further right-hand side costs about $2n^2$ flops for both substitutions together.

8.4 Pivot choice and element growth

A zero pivot on a perfectly conditioned system. Take

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad b = (2, 3)^T.$$

Unpivoted elimination asks for division by zero. Yet $A^* A = I$, so both singular values are one and $\kappa_2(A) = 1$. With the row swap $P = A$, we obtain $PA = I = LU$ with $L = U = I$. The transformed right-hand side $Pb = (3, 2)^T$ is already the solution. It was the ordering of the equations that failed.

Partial pivoting. At stage k, select a row $p \geq k$ maximizing $|a_{pk}^{(k)}|$ and exchange it with row k. If the maximum is zero, the active column is zero and the remaining square system is singular in exact arithmetic. Otherwise the multipliers have magnitude at most one. Each exchange is multiplication on the left by a permutation matrix. Their chronological action on the original equations gives a net permutation P. Think of replaying elimination with those rows placed in their final order from the start: the record of multipliers for each equation must move with that equation. In particular, a late exchange swaps the two rows of the **already filled columns** of L. The result is

$$PA = LU, \qquad Ly = Pb, \qquad Ux = y.$$

We use this convention throughout. A source writing $A = \widetilde{P}LU$ uses $\widetilde{P} = P^{-1} = P^T$. One swap is its own inverse, but a sequence of swaps need not be. **Complete pivoting** searches the whole active submatrix and may swap columns as well. A column permutation reorders unknowns and must be undone when reporting the solution.

A small nonzero pivot. Consider

$$A_\varepsilon = \begin{pmatrix} \varepsilon & 1 \\ 1 & 1 \end{pmatrix}, \qquad 0 < \varepsilon \ll 1.$$

Without a swap, $m_{21} = 1/\varepsilon$ and $u_{22} = 1 - 1/\varepsilon$. The updated right-hand side $b_2 - b_1/\varepsilon$ may also be large. Back substitution reconstructs $x_1 = (b_1 - \hat{x}_2)/\varepsilon$: even a tiny absolute error in $\hat{x}_2$ is divided by ε. When b_1 and $\hat{x}_2$ are nearly equal, cancellation exposes the error already in these operands. With a swap, $m_{21} = \varepsilon$, $u_{22} = 1 - \varepsilon$, and the final division for x_1 uses the pivot 1.

The **growth factor** compares the largest absolute entry seen in the coefficient matrices during elimination with the largest in the original matrix. Exclude stored multipliers and the augmented right-hand side. For this family unpivoted growth is of order $1/\varepsilon$, while partial pivoting gives growth factor one. Large entries increase the scale of local rounding errors, which may survive even when the entries later cancel. Partial pivoting controls each multiplier, but repeated updates can still cause large growth on exceptional matrices. Its generally good practical accuracy depends in part on the growth actually encountered.

8.4.1 Experiment: hold conditioning nearly fixed and change the pivot

Take $x = (1, 7/4)^T$ and $\varepsilon = 3 \cdot 2^{-k}$. For the exponents below, $b = (\varepsilon + 7/4, 11/4)^T$ is exactly representable in binary64, as are A_ε and x. Thus we know the exact solution of the *stored* data. As $\varepsilon \to 0$, the limiting symmetric matrix has eigenvalues $(1 \pm \sqrt{5})/2$. Its singular values are their absolute values, so κ_2 tends to $(3 + \sqrt{5})/2 \approx 2.618$. Predict which solver's error can increase while this condition number stays nearly fixed.

```
import numpy as np

def solve_two_by_two(A, b, pivot):
    """One elimination step and back substitution in binary64."""
    U = np.array(A, dtype=np.float64, copy=True)
    y = np.array(b, dtype=np.float64, copy=True)
    if U.shape != (2, 2) or y.shape != (2,):
        raise ValueError("Expected a 2-by-2 matrix and a length-2 vector")
    if pivot and abs(U[1, 0]) > abs(U[0, 0]):
        U = U[[1, 0], :]
        y = y[[1, 0]]
    if U[0, 0] == 0:
        raise ValueError("Zero first pivot")
    multiplier = U[1, 0] / U[0, 0]
    U[1, 1] -= multiplier * U[0, 1]
    y[1] -= multiplier * y[0]
    if U[1, 1] == 0:
        raise ValueError("Zero second pivot")
    x2 = y[1] / U[1, 1]
    x1 = (y[0] - U[0, 1] * x2) / U[0, 0]
    return np.array([x1, x2])

exact = np.array([1.0, 1.75])
print(" k  cond2  method       rel.res.  rel.error")
for k in (10, 25, 40, 50):
    eps = 3.0 * 2.0**(-k)
    A = np.array([[eps, 1.0], [1.0, 1.0]])
    b = A @ exact
    answers = [
        ("no swap", solve_two_by_two(A, b, False)),
        ("partial", solve_two_by_two(A, b, True)),
        ("library", np.linalg.solve(A, b)),
    ]
    for method, computed in answers:
        residual = np.linalg.norm(b - A @ computed) / np.linalg.norm(b)
        error = np.linalg.norm(computed - exact) / np.linalg.norm(exact)
        print(f"{k:2}  {np.linalg.cond(A):5.3f}  {method:8}"
              f"  {residual:9.2e}  {error:9.2e}")
```

```
 k  cond2  method       rel.res.  rel.error
10  2.628  no swap    2.31e-14   3.76e-14
10  2.628  partial    0.00e+00   0.00e+00
```

```
10  2.628  library    0.00e+00   0.00e+00
25  2.618  no swap    7.62e-10   1.23e-09
25  2.618  partial    0.00e+00   0.00e+00
25  2.618  library    0.00e+00   0.00e+00
40  2.618  no swap    2.50e-05   4.04e-05
40  2.618  partial    0.00e+00   0.00e+00
40  2.618  library    0.00e+00   0.00e+00
50  2.618  no swap    2.56e-02   4.13e-02
50  2.618  partial    0.00e+00   0.00e+00
50  2.618  library    0.00e+00   0.00e+00
```

Read across methods at fixed k first. They receive identical data; the library solve provides a practical reference, while the known x provides the forward-error measurement. At $k = 50$ the unpivoted first component loses about 0.083 in absolute value. The condition number is still about 2.62. A displayed zero residual is a zero in the arithmetic used for the check. Re-evaluating a residual at higher precision can distinguish small errors hidden by that rounding.

8.5 Cholesky uses positive definiteness

Session 5 characterized a **Hermitian positive-definite** (HPD) matrix by $A = A^*$ and $z^*Az > 0$ for all $z \neq 0$, equivalently by Hermitian symmetry and strictly positive eigenvalues. The real case is symmetric positive definite (SPD).

Cholesky theorem. Every HPD matrix admits a unique factorization $A = LL^*$ with L lower triangular and strictly positive real diagonal. For real SPD matrices, $A = LL^T$. Conversely, any such product is Hermitian and $z^*LL^*z = \|L^*z\|_2^2 > 0$ for $z \neq 0$, since L^* is invertible.

Why no pivot search is necessary. Partition the first column from the rest:

$$A = \begin{pmatrix} a & w^* \\ w & B \end{pmatrix}, \qquad S = B - ww^*/a.$$

Testing on e_1 gives $a > 0$. For any $z \neq 0$, test the quadratic form on $v = \binom{-w^*z/a}{z}$, interpreted as a stacked scalar and column. Direct multiplication gives $v^*Av = z^*Sz > 0$. Thus the trailing matrix S after eliminating column one is again HPD. Cholesky takes $\sqrt{a}$ as its first diagonal entry, $w/\sqrt{a}$ below it, and factors S recursively. This is symmetric Gaussian elimination: the trailing update preserves Hermitian symmetry, so only one triangle needs to be updated. General LU would maintain two factors without using this relationship.

How to compute the entries. Expand $a_{ij} = \sum_{s \le j} \ell_{is}\overline{\ell_{js}}$ for $i \ge j$, and process columns in increasing order:

$$\ell_{jj} = \sqrt{a_{jj} - \sum_{s<j} |\ell_{js}|^2},$$
$$\ell_{ij} = \frac{a_{ij} - \sum_{s<j} \ell_{is}\overline{\ell_{js}}}{\ell_{jj}}, \qquad i > j.$$

Each radicand is the leading entry of an HPD trailing matrix, hence real and positive. Taking its positive square root fixes the only sign or phase ambiguity. The column formulas then determine every subsequent entry uniquely.

Worked example. Solve with

$$A = \begin{pmatrix} 4 & 2 & 0 \\ 2 & 5 & -2 \\ 0 & -2 & 2 \end{pmatrix}, \qquad b = (4, 4, -2)^T.$$

The first column of L is $(2, 1, 0)^T$. In column two, $\ell_{22} = \sqrt{5-1} = 2$ and $\ell_{32} = (-2-0)/2 = -1$. Finally $\ell_{33} = \sqrt{2 - 0^2 - (-1)^2} = 1$. Multiplication verifies both the factorization and positive definiteness:

$$\begin{pmatrix} 2 & 0 & 0 \\ 1 & 2 & 0 \\ 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 0 \\ 0 & 2 & -1 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 4 & 2 & 0 \\ 2 & 5 & -2 \\ 0 & -2 & 2 \end{pmatrix}.$$

Forward substitution in $Ly = b$ gives $y_1 = 2$, $y_2 = (4-2)/2 = 1$, $y_3 = -2 + 1 = -1$. In $L^*x = y$, the last row gives $x_3 = -1$, then $2x_2 - x_3 = 1$ gives $x_2 = 0$, and $2x_1 + x_2 = 2$ gives $x_1 = 1$.

Where the hypotheses matter. A positive semidefinite Hermitian matrix has $z^*Az \ge 0$; if singular, it cannot have a Cholesky factor with strictly positive diagonal (try $\mathrm{diag}(1, 0)$). An indefinite Hermitian matrix has directions of both signs. For $\left(\begin{smallmatrix} 1 & 2 \\ 2 & 1 \end{smallmatrix}\right)$ both diagonal entries are positive, yet the second radicand is $1 - 2^2 = -3$. A non-Hermitian matrix cannot equal LL^* because every such product is Hermitian. In floating point a very small positive radicand can be lost in subtraction. An implementation checks for breakdown; a failure near the boundary of positive definiteness needs interpretation in light of input accuracy and rounding.

8.6 Choosing and assessing a solver

Classical leading work below assumes a dense, real, nonsingular square matrix. QR uses Householder reflectors stored implicitly. Each method takes $O(n^2)$ additional work per right-hand side.

Method	When to use it	Factorization work	Reliability
LU, no swaps	Leading minors nonzero and pivot behavior understood	$2n^3/3$	Small relative pivots can give large growth.
LU, partial pivoting	General square solve	$2n^3/3$	Bounded multipliers; accuracy still depends on growth.
Cholesky	Hermitian positive definite	$n^3/3$	Positive exact pivots; backward stable on successful completion under standard rounding assumptions.
Householder QR	General square solve; also useful for least squares	$4n^3/3$	Unitary transformations preserve the 2-norm; normwise backward stable.

Cholesky roughly halves the leading LU factorization cost by updating one triangle. Session 4's QR solve applies the reflectors to form Q^*b, then solves $Rx = Q^*b$. In all four methods the factors act directly on any new right-hand side. Explicitly constructing A^{-1} adds work even if many right-hand sides are expected.

Residual, backward error and forward error. Given a computed $\hat{x}$, set $r = b - A\hat{x}$. The residual answers how far the computed vector misses the equations. Since $A\hat{x} = b - r$, it is also an explicit backward perturbation to the right-hand side. With perturbations allowed in both A and b, a common normwise backward-error measure is

$$\eta = \frac{\|r\|_2}{\|A\|_2\|\hat{x}\|_2 + \|b\|_2}.$$

It is the smallest common relative bound on $\|\Delta A\|_2/\|A\|_2$ and $\|\Delta b\|_2/\|b\|_2$ needed for $(A + \Delta A)\hat{x} = b + \Delta b$, assuming $b \neq 0$. A backward-stable solve delivers a solution to data perturbed on the scale of rounding, up to dimension-dependent factors. A measured small backward error describes this particular result; a stability analysis concerns the algorithm over its class of inputs.

For nonsingular A and $b \neq 0$, Session 6 gives

$$x - \hat{x} = A^{-1}r, \qquad \frac{\|x - \hat{x}\|_2}{\|x\|_2} \leq \kappa_2(A)\frac{\|r\|_2}{\|b\|_2}.$$

A small relative residual and moderate condition number certify small forward error. A large condition number weakens that certificate: an error pointing along a small-singular-value direction can have a very small image under A. Even a backward-stable solver can then lose forward accuracy.

8.7 Where the factors reappear

Sparse systems and Session 8. Discretizing a differential operator often couples each unknown only to a few neighbors. During elimination two neighbors of the eliminated unknown can become coupled to one another. A newly nonzero factor entry in a previously zero position is *fill-in*. Its extent depends on ordering and can dominate storage and work. Dense costs can therefore be prohibitive, and the cost of a sparse factorization depends on more than the initial number of nonzeros. A direct method follows a finite elimination and substitution schedule in exact arithmetic. An iterative method produces successive approximations and stops using an accuracy criterion. A factorization of a simpler approximation to A can also supply an inexpensive solve inside an iterative method as a preconditioner.

Eigenvalue methods and the seminar. At a fixed shift σ outside the spectrum, inverse iteration repeatedly solves $(A - \sigma I)y = x$ with changing x. One pivoted LU of the shifted matrix can serve every step. An eigencomponent with eigenvalue λ is scaled by $1/(\lambda - \sigma)$. In Chebyshev-filtered subspace iteration the corresponding weight is $p(\lambda)$: a polynomial recurrence applies A repeatedly to vectors or blocks, avoiding a shifted factorization and its possible fill-in. QR orthonormalization still has a role after filtering. Sessions 9–11 return to these choices.

8.8 Central Synthesis — From Elimination to Reliable Numerical Solvers

Question	What the factorization tells us	What to check
Mathematical equivalence	Each invertible row operation acts on both sides. Elimination ends with triangular equations having the same solution.	Track the right-hand side and the row order.

Question	What the factorization tells us	What to check
Computational efficiency	LU records all elimination steps; substitutions reuse them. HPD structure makes the second Cholesky factor the adjoint of the first.	Match the structure and number of right-hand sides to the factorization.
Numerical stability	Pivot choices determine multipliers and affect element growth. An algebraically valid elimination can produce poor factors after rounding.	Inspect a scaled residual or backward error, then use conditioning to assess forward accuracy.

The two pivot examples separate two diagnoses. The exchange matrix forces a row swap even in exact arithmetic. The small-pivot family permits both algorithms in exact arithmetic, yet choosing the smaller pivot can lose accuracy. In either case $\kappa_2(A)$ is a property of the input shared by both algorithms. Conversely, selecting a stable solver leaves the sensitivity of an ill-conditioned problem in place.

Course reading. The revision plan and seminar connection are in §§1.3–1.4 of the course notes book. For elimination, pivoting and Cholesky, see Scheichl and Zech, *Num0 notes*, §§3.4–3.7, and Bastian, *num0*, §§7–10. For the orthogonal alternative, see Kanschat and Scheichl, *Numerical Linear Algebra*, Chapter 2.

8.9 Exercises

1. **Recover a triangular system (3 min).** Eliminate below the first pivot of $A = \left(\begin{smallmatrix} 2 & -1 \\ 6 & 1 \end{smallmatrix}\right)$ with $b = (1, 11)^T$. Record the multiplier, L and U. Solve via $Ly = b$ and $Ux = y$.
2. **Undo two stages (3 min).** Suppose $E_1 = I - 2e_2e_1^T + e_3e_1^T$, $E_2 = I - 3e_3e_2^T$, and $E_2E_1A = U$. Derive L in $A = LU$. Show by multiplication what goes wrong if you reverse the two inverse factors.
3. **Choose a Cholesky solve (4 min).** For $C = \left(\begin{smallmatrix} 9 & 3 \\ 3 & 5 \end{smallmatrix}\right)$ construct a Cholesky factor, certify positive definiteness and solve $Cx = (6, -2)^T$. Classify $\left(\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}\right)$, $\mathrm{diag}(2, -1)$ and $\left(\begin{smallmatrix} 2 & 1 \\ 0 & 2 \end{smallmatrix}\right)$ with respect to the same algorithm.
4. **Diagnose a pivot (3 min).** Let $J = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$. Determine its first unpivoted pivot, determinant and κ_2. Repeat for tJ, $t > 0$. For the experiment's A_ε, identify which intermediate quantity grows and why swapping helps. Which of these observations concerns the problem's sensitivity?

5. **A late row exchange (6 min).** Take

$$A = \begin{pmatrix} 4 & 4 & 0 \\ 2 & 2 & 1 \\ 1 & 3 & 2 \end{pmatrix}, \qquad b = (4, 4, 3)^T.$$

   Decide whether unpivoted elimination succeeds. Construct P, L, U using partial pivoting, verify $PA = LU$ by multiplication, and solve the two triangular systems. Check the residual. Distinguish an exact zero residual from a small residual evaluated in floating point.
6. **A good residual for a poor answer (4 min).** Take $D = \mathrm{diag}(1, 10^{-8})$, $b = (1, 0)^T$ and $\hat{x} = (1, 1)^T$. Compute the relative residual, relative forward error and $\kappa_2(D)$. Give a right-hand-side perturbation making $\hat{x}$ exact. Can these data alone establish that a particular solver is unstable?

8.10 Worked solutions

1. The multiplier is $6/2 = 3$. Subtracting three times the first equation from the second gives $4x_2 = 8$. Thus

$$L = \begin{pmatrix} 1 & 0 \\ 3 & 1 \end{pmatrix}, \qquad U = \begin{pmatrix} 2 & -1 \\ 0 & 4 \end{pmatrix}.$$

Forward substitution gives $y = (1, 11 - 3)^T = (1, 8)^T$. Back substitution gives $x_2 = 2$ and $x_1 = (1 + 2)/2 = 3/2$.

2. Multiplying $E_2 E_1 A = U$ first by E_2^{-1} and then by E_1^{-1} gives

$$L = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 3 & 1 \end{pmatrix}.$$

In reversed order row 3 of the product becomes $3(2, 1, 0) + (-1, 0, 1) = (5, 3, 1)$. The spurious 6 in column one records an operation on a row already modified at the wrong stage.

3. The first column is $(3, 1)^T$ and the remaining diagonal is $\sqrt{5 - 1} = 2$. With $L = \left(\begin{smallmatrix} 3 & 0 \\ 1 & 2 \end{smallmatrix}\right)$, multiplication gives $LL^T = C$. Invertibility of L implies $z^T C z = \|L^T z\|_2^2 > 0$ for every nonzero real z. Forward substitution gives $y = (2, -2)^T$ and backward substitution gives $x = (1, -1)^T$. The all-ones matrix has quadratic form $(z_1 + z_2)^2$ on real vectors and is singular positive semidefinite. The diagonal matrix is indefinite. The final matrix is nonsymmetric. None admits a real Cholesky factor with strictly positive diagonal.

4. The first pivot is zero in both cases. The determinants are -1 and $-t^2$, respectively. Since $(tJ)^T(tJ) = t^2 I$, both singular values are t, giving $\kappa_2 = 1$ for every $t > 0$, including $t = 1$. In A_ε the multiplier $1/\varepsilon$ and updated entry $1 - 1/\varepsilon$ grow in magnitude. Exchanging the rows gives

a multiplier ε and updated entry $1-\varepsilon$. The singular-value ratio measures relative worst-case sensitivity. The intermediate quantities describe a particular elimination order.

5. The first pivot is 4, with multipliers $1/2, 1/4$. The remaining rows become $(0,0,1)$ and $(0,2,2)$, so the second unpivoted pivot vanishes. Swap rows two and three, including their stored first-column multipliers. The final multiplier is zero. Then

$$P = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \quad L = \begin{pmatrix} 1 & 0 & 0 \\ 1/4 & 1 & 0 \\ 1/2 & 0 & 1 \end{pmatrix}, \quad U = \begin{pmatrix} 4 & 4 & 0 \\ 0 & 2 & 2 \\ 0 & 0 & 1 \end{pmatrix}.$$

The rows of LU are $(4,4,0)$, $(1,1,0)+(0,2,2)=(1,3,2)$ and $(2,2,0)+(0,0,1)=(2,2,1)$, exactly those of PA. Here $Pb=(4,3,4)^T$. Forward substitution gives $y=(4,3-1,4-2)^T=(4,2,2)^T$. From the bottom, $x_3=2$, $x_2=(2-4)/2=-1$, $x_1=(4+4)/4=2$. Direct multiplication gives $Ax=(4,4,3)^T$ and $r=0$. With nonsingular A, an exact zero proves $\hat{x}=x$. For a numerical residual one must allow for rounding in its evaluation and use conditioning to turn a relative residual into a forward-error bound.

6. The exact solution is $(1,0)^T$ and $r=(0,-10^{-8})^T$. The relative residual is 10^{-8}, the relative forward error is 1, and $\kappa_2(D)=10^8$. The residual bound is attained. Perturbing b by $-r=(0,10^{-8})^T$ makes $\hat{x}$ exact. This example shows how the poorly scaled direction weakens the residual certificate. Judging the stability of an algorithm would also require its precision, the backward-error scale expected at that precision, and information about the procedure that produced the result.

8.10.1 End-of-session mastery checklist

Before Session 8, **Basic Iterative Methods**, check that you can independently:

- Prove that simultaneous elementary operations on A and b preserve the solution set.
- Eliminate by hand, identify pivots and multipliers, and derive $A=LU$ by undoing the stages.
- Explain what each factor stores, perform both substitutions and reuse factors for new loads.
- Construct $PA=LU$, including a late exchange, and apply the same permutation to b.
- Distinguish pivot breakdown, singularity, poor conditioning and numerical instability with examples.
- State and justify the HPD hypothesis, compute a Cholesky factor and explain the saving in work.
- Choose among pivoted LU, Cholesky and Householder QR using structure, cost and accuracy.
- Interpret a computed residual using backward error and conditioning.
- Explain how triangular solves enter iterative and shifted eigenvalue methods, and identify the main operation in a polynomial filter.

9 Session 8 — Basic Iterative Methods

A factorization gives us a way to solve $Ax = b$. An iteration poses a different design question. Can we compute an inexpensive correction from the data we have, and prove that applying it repeatedly removes an error we cannot observe?

Study route (75 minutes). Allow 12 minutes for splittings and fixed points, 10 for the two hand calculations, 15 for the convergence theorem and its guarantees, and 10 for stopping, the figure, and synthesis. Use the remaining 28 minutes for the exercises. Reading the Python implementation line by line is optional.

9.1 From a linear system to a fixed point

In Session 7, elimination reduced a system to triangular solves. Given suitable pivots, LU solves a nonsingular system in finitely many operations in exact arithmetic. Dense factorization costs order n^3 operations and order n^2 storage. Sparse elimination can introduce entries where the original matrix had zeros (*fill-in*). On a large grid, storing the factors can be a greater obstacle than storing the original equations. A sparse matrix-vector product needs work proportional to the number of stored nonzeros. A sparse triangular solve can also visit each stored coefficient once, although its rows depend on one another. The question is how many such inexpensive sweeps we need for a given accuracy.

A **fixed point** of a map F satisfies $x^* = F(x^*)$. Starting with $x^{(0)}$, evaluate $x^{(k+1)} = F(x^{(k)})$. For continuous F, any limit of the whole sequence is a fixed point, since passing to the limit gives $x^* = F(x^*)$. Existence alone gives no assurance that the sequence approaches it.

We study **stationary linear iterations**. The update is affine, $F(x) = Bx + c$, and *stationary* means that the same B and c are used at every sweep. The fixed-point equation is $(I - B)x = c$. It has a unique solution exactly when $I - B$ is invertible.

9.1.1 Build the update by choosing an easy solve

Let $A \in \mathbb{F}^{n\times n}$ be nonsingular, $b \in \mathbb{F}^n$, and $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. Split $A = M - N$ with M invertible. Since $Ax = b$ is equivalent to $Mx = Nx + b$, evaluate the right-hand side at the current approximation and solve for the next one:

$$Mx^{(k+1)} = Nx^{(k)} + b, \qquad x^{(k+1)} = Bx^{(k)} + c, \quad B = M^{-1}N, \quad c = M^{-1}b.$$

The identity $I - B = M^{-1}A$ verifies consistency and uniqueness at once. Multiplying the fixed-point equation by M recovers precisely the original system. Subtracting $Mx^{(k)}$ from both sides of the sweep gives another useful reading:

$$r^{(k)} = b - Ax^{(k)}, \qquad Md^{(k)} = r^{(k)}, \qquad x^{(k+1)} = x^{(k)} + d^{(k)}.$$

Thus $M^{-1}r^{(k)}$ approximates the exact correction $A^{-1}r^{(k)}$. Taking $M = A$ would give that correction in one step, at the price of solving the original problem. A more economical M changes the propagation matrix $B = I - M^{-1}A$. In code, solve with M using its structure. Forming its inverse can destroy sparsity and adds unnecessary work.

9.2 Two choices of splitting, one system

Write $A = D + L + U$, retaining the actual signs of entries in the diagonal, strictly lower, and strictly upper parts. Assume $a_{ii} \neq 0$ for each i. This makes both D and $D + L$ invertible. A nonsingular A can have a zero diagonal entry, so this assumption needs its own check.

9.2.1 Jacobi uses a snapshot

Take $M = D$, $N = -(L + U)$. Reading $Dx^{(k+1)} = b - (L + U)x^{(k)}$ by rows gives

$$B_J = -D^{-1}(L + U), \qquad c_J = D^{-1}b, \qquad x_i^{(k+1)} = \frac{b_i - \sum_{j \neq i} a_{ij}x_j^{(k)}}{a_{ii}}.$$

Every row reads the same old vector. Keep separate input and output arrays during the sweep. Rows can then be distributed across processors and the outputs become the common input for the next sweep.

9.2.2 Gauss–Seidel reuses a corrected component

Visit rows in order $1, \ldots, n$. By row i, the first $i - 1$ components have already been updated. Moving their terms to the left gives $(D + L)x^{(k+1)} = b - Ux^{(k)}$, hence

$$B_{GS} = -(D+L)^{-1}U, \qquad c_{GS} = (D+L)^{-1}b,$$
$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j<i} a_{ij}x_j^{(k+1)} - \sum_{j>i} a_{ij}x_j^{(k)}\right).$$

This is forward substitution with $M = D+L$. One array suffices. An entry below the diagonal creates a dependency on the newly computed component. A different ordering can change the sweep and its convergence behavior.

9.2.3 Worked example — when does the second unknown first change?

Use the same start $x^{(0)} = (0,0)^T$ for both methods, with

$$A = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}, \qquad b = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad x^* = \begin{pmatrix} 2/3 \\ 1/3 \end{pmatrix}.$$

Write the old coordinates as (u, v) and the new ones as (u', v'). Both methods compute $u' = (1+v)/2$. Jacobi takes $v' = u/2$ and Gauss–Seidel takes $v' = u'/2 = (1+v)/4$. Reading off the coefficients and constant terms gives

$$B_J = \begin{pmatrix} 0 & 1/2 \\ 1/2 & 0 \end{pmatrix}, \quad c_J = \begin{pmatrix} 1/2 \\ 0 \end{pmatrix}, \qquad B_{GS} = \begin{pmatrix} 0 & 1/2 \\ 0 & 1/4 \end{pmatrix}, \quad c_{GS} = \begin{pmatrix} 1/2 \\ 1/4 \end{pmatrix}.$$

Sweep k	Jacobi (u, v)	Gauss–Seidel (u, v)
0	$(0, 0)$	$(0, 0)$
1	$(1/2, 0)$	$(1/2, 1/4)$
2	$(1/2, 1/4)$	$(5/8, 5/16)$
3	$(5/8, 1/4)$	$(21/32, 21/64)$

In the first Jacobi sweep the second row still sees $u = 0$. In the first Gauss–Seidel sweep it sees $u' = 1/2$, and its equation is already satisfied at the end of the sweep. The next first-row update can disturb that equation again. To decide whether these corrections solve *both* equations in the limit, track an entire error vector across a sweep.

9.3 The error obeys a homogeneous equation

Use the signed error $e^{(k)} = x^{(k)} - x^*$ throughout this session. This reverses the exact-minus-approximate convention of Session 6 and preserves all norm estimates. The solver stores $x^{(k)}$. The vector x^* is fixed and generally unknown. We can reason about their difference because both obey the same affine rule:

$$\begin{aligned} x^{(k+1)} &= Bx^{(k)} + c, \\ x^* &= Bx^* + c, \\ e^{(k+1)} &= B(x^{(k)} - x^*) = Be^{(k)}. \end{aligned}$$

The translation c cancels. Centered at the solution, every sweep is the same linear transformation. Applying it k times yields

$$\boxed{e^{(k)} = B^k e^{(0)}}.$$

Since the starting guess is arbitrary, $e^{(0)}$ ranges over the entire space. Convergence for every start therefore requires all directions to decay. For the running example, B_J swaps the coordinates and halves them, so every Euclidean error norm halves at every sweep. Meanwhile $B_{GS}(u, v)^T = v(1/2, 1/4)^T$. One sweep sends every error onto the line spanned by $(2, 1)^T$, where every subsequent sweep multiplies it by $1/4$. This advantage is specific to the system and splitting.

9.4 When do all matrix powers decay?

Convergence theorem. For any fixed finite square matrix B over $\mathbb{R}$ or $\mathbb{C}$ and fixed vector c, the following are equivalent in exact arithmetic:

1. The map $x \mapsto Bx + c$ has a unique fixed point, and the iterates approach it for every initial vector.
2. $B^k \to 0$ as $k \to \infty$.
3. $\rho(B) = \max_{\lambda \in \sigma(B)} |\lambda| < 1$, taking eigenvalues over $\mathbb{C}$.

Proof. Under the first condition, choose initial errors equal to the coordinate vectors in turn. The error formula shows that every column of B^k tends to zero. In finite dimensions this implies convergence to zero in any matrix norm. Conversely, if $B^k \to 0$, the eigenvector identity $B^k v = \lambda^k v$ forces $|\lambda| < 1$ for every eigenvalue. For real matrices the column argument also gives decay on complex vectors by linearity.

To finish, assume $\rho(B) < 1$. For diagonalizable $B = V\Lambda V^{-1}$, cancellation of adjacent factors yields $B^k = V\Lambda^k V^{-1}$. Each diagonal entry λ_i^k vanishes. For a defective matrix, take its

complex Jordan form. A block $J = \lambda I + H$ of size s has $H^s = 0$. For $k \geq s$ its binomial expansion is

$$J^k = \sum_{j=0}^{s-1} \binom{k}{j} \lambda^{k-j} H^j.$$

For $0 < |\lambda| < 1$, exponential decay dominates every fixed polynomial factor in k. A block with $\lambda = 0$ vanishes after s powers. Thus all blocks tend to zero, and conjugating back gives $B^k \to 0$. Finally 1 is absent from the spectrum, so $x^* = (I - B)^{-1}c$ exists uniquely and $x^{(k)} - x^* = B^k e^{(0)} \to 0$. This completes the proof.

How fast? For diagonalizable B, $\|B^k\|_2 \leq \kappa_2(V)\rho(B)^k$. For normal B, a unitary eigenbasis gives $\|B^k\|_2 = \rho(B)^k$. In general the asymptotic worst-case exponential factor is $\lim_{k\to\infty} \|B^k\|^{1/k} = \rho(B)$, by the same block argument and the eigenvector lower bound. An initial error can miss a slowly decaying eigendirection. Consecutive error ratios can also vary.

A sufficient norm test. In any chosen induced norm, $q = \|B\| < 1$ gives $\|e^{(k)}\| \leq q^k \|e^{(0)}\|$ by submultiplicativity. This test can be inconclusive. For example,

$$B = \begin{pmatrix} 0 & 4 \\ 0 & 0 \end{pmatrix}$$

has $\|B\|_2 = 4$ and sends $(0, 1)^T$ to $(4, 0)^T$ in the first step. Yet $B^2 = 0$ and $\rho(B) = 0$. This nonnormal example exhibits transient amplification before complete decay. The decisive condition concerns the powers of the **iteration matrix**. The eigenvalues of the original A do not in general give those of B.

9.5 Two guarantees that avoid computing eigenvalues

Strict row diagonal dominance. If $|a_{ii}| > \sum_{j\neq i} |a_{ij}|$ for every row, Jacobi and forward Gauss–Seidel both converge from every start. For Jacobi the maximum absolute row sum of B_J is directly less than one. For Gauss–Seidel let $\ell_i = \sum_{j<i} |a_{ij}|$ and $u_i = \sum_{j>i} |a_{ij}|$. At a coordinate i attaining the maximum new error, the homogeneous row update gives

$$(|a_{ii}| - \ell_i)\|e^{(k+1)}\|_\infty \leq u_i \|e^{(k)}\|_\infty.$$

The largest of the ratios $u_i/(|a_{ii}| - \ell_i)$ is strictly below one. Thus the entire sweep contracts. These tests are sufficient. The spectral-radius criterion is exact.

Hermitian positive definiteness. If $A = A^*$ and $z^*Az > 0$ for all $z \neq 0$, Gauss–Seidel converges. To see why, measure error in the energy norm $\|e\|_A^2 = e^*Ae$. A row update changes one coordinate by t. Along coordinate vector s_i, the energy becomes

$$\|e + ts_i\|_A^2 = \|e\|_A^2 + 2\,\mathrm{Re}(\bar{t}(Ae)_i) + a_{ii}|t|^2.$$

Its unique minimizer is $t = -(Ae)_i/a_{ii}$, exactly the current row correction. Every nonzero correction strictly decreases energy. If a full sweep makes no correction, every row of Ae was zero and $e = 0$. By continuity on the compact energy unit sphere, the maximum energy-norm ratio for a full sweep is strictly less than one. Jacobi evaluates all corrections at the old point, so this successive minimization argument does not apply.

Divergence in a uniquely solvable SPD system. Let $w = (1, 1, 1)^T$, $A = (I + 3ww^T)/4$, and $b = Aw$. The diagonal is one and every off-diagonal entry is $3/4$. Along w, A has eigenvalue $5/2$; on $w^\perp$ it has eigenvalue $1/4$. Hence A is SPD and $x^* = w$ is unique. Here $B_J = I - A$ has eigenvalue $-3/2$ along w and $3/4$ on $w^\perp$. From zero the initial error is $-w$, so it alternates sign and grows by $3/2$ per sweep. An initial error orthogonal to w decays by $3/4$ instead. This system shows both the limit of the SPD guarantee for Jacobi and the importance of the words *every initial guess*.

9.6 What can we measure without the solution?

Since $b = Ax^*$, our sign convention gives

$$r^{(k)} = b - Ax^{(k)} = -Ae^{(k)}, \qquad e^{(k)} = -A^{-1}r^{(k)}.$$

The residual measures how much the right-hand side must change to make the current approximation exact (the required change is $-r^{(k)}$). The error measures distance to the solution. In any compatible vector norm and its induced matrix norm, for $b \neq 0$,

$$\frac{\|e^{(k)}\|}{\|x^*\|} \leq \underbrace{\|A^{-1}\|\|A\|}_{\kappa(A)} \frac{\|r^{(k)}\|}{\|b\|}.$$

Here we used $\|b\| \leq \|A\|\|x^*\|$. An ill-conditioned matrix can map a large error to a small residual. Assess the requested accuracy in light of this factor.

Diagnostic	How to use it
Absolute and relative residual	Recompute $b - Ax^{(k)}$. Stop when its norm is at most $\tau_{\text{abs}} + \tau_{\text{rel}}\|b\|$. The absolute term also handles $b = 0$.
Size of the last change	A small correction can signal slow progress or rounding. A verified contraction factor can turn it into an error bound.
Iteration budget and finite values	Report exhaustion, stagnation, or nonfinite values separately from meeting the residual tolerance.

For a contraction with known $q = \|B\| < 1$, successive changes shrink at least geometrically since $x^{(k+1)} - x^{(k)} = B(x^{(k)} - x^{(k-1)})$. Sum the tail from the current iterate to the limit to obtain, for $k \geq 1$,

$$\|x^* - x^{(k)}\| \leq \frac{q}{1-q}\|x^{(k)} - x^{(k-1)}\|.$$

A factor q close to one makes this certificate weak. In floating point, a computed sweep can be written as $\hat{e}^{(k+1)} = B\hat{e}^{(k)} + \delta^{(k)}$, with $\delta^{(k)}$ the local arithmetic error. Earlier perturbations propagate through powers of B and each sweep introduces a new one. Even for $q < 1$, uniformly bounded perturbations of size ϵ contribute up to $\epsilon/(1-q)$ to the error. Recomputing the residual also incurs rounding. Exact-arithmetic convergence establishes neither attainable accuracy nor backward stability of an implementation.

9.7 Computational experiment — predict the slopes

For the hand example, the characteristic polynomials are $t^2 - 1/4$ for Jacobi and $t(t - 1/4)$ for Gauss–Seidel. Predict the two error slopes before reading the figure. The code computes the spectral radii independently and implements each sweep directly. Dense arrays keep the small experiment inspectable. The exact solution is used only for measurement.

```
import numpy as np
import matplotlib.pyplot as plt

def jacobi(A, b, x):
    return x + (b - A @ x) / np.diag(A)

def gauss_seidel(A, b, x):
    y = x.copy()
```

```
    for i in range(len(b)):
        lower = A[i, :i] @ y[:i]
        upper = A[i, i+1:] @ x[i+1:]
        y[i] = (b[i] - lower - upper) / A[i, i]
    return y

def history(A, b, exact, x0, sweep, max_steps, rtol=1e-11):
    if np.any(np.diag(A) == 0):
        raise ValueError("A diagonal entry is zero")
    x = np.array(x0, dtype=float)
    rows = []
    threshold = 1e-14 + rtol * np.linalg.norm(b)
    for k in range(max_steps + 1):
        residual = np.linalg.norm(b - A @ x)
        error = np.linalg.norm(x - exact)
        if not np.isfinite([residual, error]).all():
            raise FloatingPointError("Nonfinite history entry")
        rows.append((error, residual))
        if residual <= threshold:
            return np.array(rows), "residual tolerance"
        if k < max_steps:
            y = sweep(A, b, x)
            if np.array_equal(y, x):
                return np.array(rows), "stagnation"
            x = y
    return np.array(rows), "iteration limit"
```

```
A = np.array([[2., -1.], [-1., 2.]])
b = np.array([1., 0.])
exact = np.array([2/3, 1/3])
start = np.zeros(2)
fig, axes = plt.subplots(1, 2, sharey=True, layout="constrained")
methods = [
    ("Jacobi", jacobi, np.diag(np.diag(A)), "-", "C0"),
    ("Gauss-Seidel", gauss_seidel, np.tril(A), "--", "C1"),
]
for name, sweep, M, style, color in methods:
    B = np.linalg.solve(M, M - A)
    rho = max(abs(np.linalg.eigvals(B)))
    values, status = history(A, b, exact, start, sweep, 80)
    assert status == "residual tolerance"
    # Check the sweep on a vector with both coordinates nonzero.
```

```
    z = np.array([0.75, -0.25])
    assert np.allclose(sweep(A, b, z), B @ z + np.linalg.solve(M, b))
    label = f"{name} ({rho:.2f})"
    for ax, column in zip(axes, values.T):
        ax.semilogy(column, style, color=color, label=label)

for ax, title in zip(axes, ["Error norm", "Residual norm"]):
    ax.set(title=title, xlabel="Sweep k")
    ax.grid(True, which="major", alpha=0.25)
axes[0].set_ylabel("Euclidean norm")
axes[1].legend(title="Method (spectral radius)", fontsize=8)
plt.show()
```

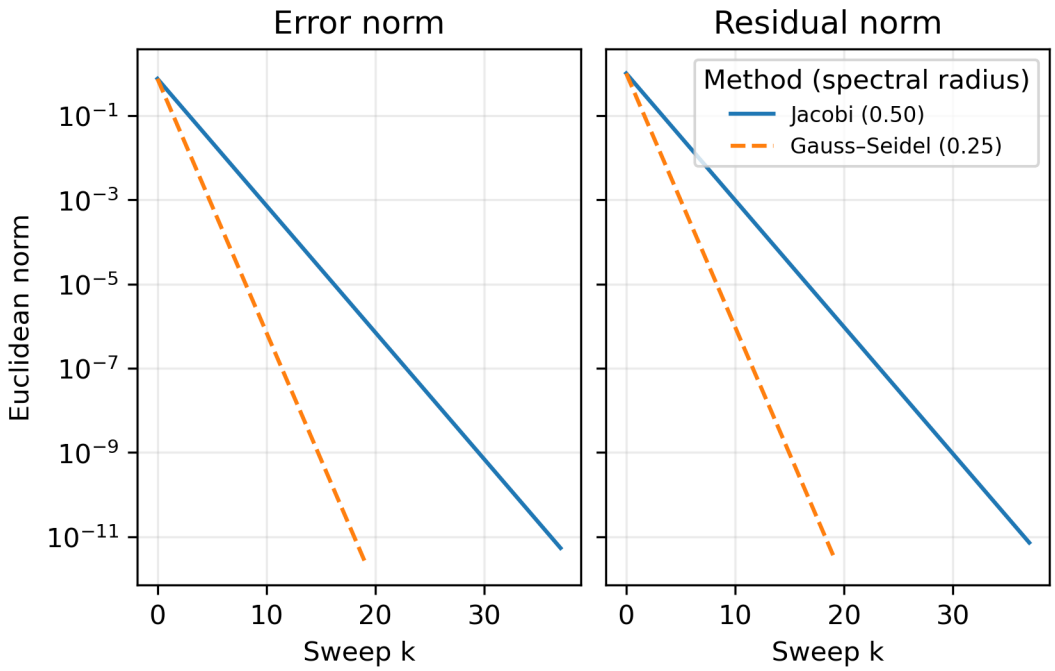


Figure 9.1: Euclidean error and residual for the two sweeps from zero. Each curve ends at the first accepted residual.

The Jacobi curve loses a factor of two per sweep. The Gauss–Seidel curve loses a factor of four per sweep from the first updated vector onward. In this particular run even the first Gauss–Seidel error lies on the same line as the initial error. For Jacobi, the residual alternates between coordinate axes; for Gauss–Seidel its second coordinate is zero after each sweep. Both residual histories therefore have a direct interpretation in terms of the equations still violated.

The straight lines here arise from the special matrices and starting vector. The nilpotent example above explains why a single sweep of another convergent method can increase the error.

Optional divergence check. Run eight sweeps on the SPD counterexample. The reported ratio is a measurement of the growing mode selected by the initial error.

```
w = np.ones(3)
C = (np.eye(3) + 3 * np.outer(w, w)) / 4
values, status = history(C, C @ w, w, np.zeros(3), jacobi, 8)
ratios = values[1:, 0] / values[:-1, 0]
assert np.allclose(ratios, 1.5)
print(f"{status} after 8 sweeps; error factor {ratios[-1]:.2f}")
```

```
iteration limit after 8 sweeps; error factor 1.50
```

9.8 From sparse solves to spectral filtering

For large systems, the splitting gives two reusable operations: apply A and solve approximately for a correction. A few sweeps can also serve as part of a preconditioner. In multigrid, suitable relaxation methods such as weighted Jacobi can damp oscillatory error on a grid, leaving smoother components for a coarser grid. Total cost depends on both the work per sweep and the number of sweeps.

Krylov methods build approximations from a growing subspace. Starting with $r_0 = b - Ax^{(0)}$, a typical unpreconditioned approximation lies in $x^{(0)} + \mathcal{K}_m(A, r_0)$, where

$$\mathcal{K}_m(A, r_0) = \operatorname{span}\{r_0, Ar_0, \dots, A^{m-1}r_0\}.$$

Here the choice of approximation can adapt as the space grows. In our stationary methods the error polynomial in B is already prescribed: $p_k(t) = t^k$.

Session 9 reuses the calculation $B^k v = \lambda^k v$ in a different role. Power iteration repeatedly applies an operator and normalizes, so directions with larger eigenvalue magnitudes can grow *relative to* other directions. Subspace iteration carries several directions together. In Session 10 a filter $p_m(A)$ multiplies an eigencomponent by $p_m(\lambda)$. Its purpose is to separate wanted and unwanted spectral components. A convergent stationary solver must drive **every** initial error component to zero. This difference in objectives determines which spectral comparison matters.

9.9 Central Synthesis — Iterative Solvers as Spectral Error Dynamics

Question	Calculation	What it establishes
Where is the fixed point?	$A = M - N$ gives $Mx = Nx + b$.	The fixed points solve the original system. Invertibility of A gives uniqueness.
What does one sweep do?	Solve $Md = r^{(k)}$, then add d.	The choice of easy solve determines $B = I - M^{-1}A$.
What happens after many sweeps?	Subtract the fixed-point identity. Then $e^{(k+1)} = Be^{(k)}$ and $e^{(k)} = B^k e^{(0)}$.	Every start succeeds exactly when $B^k \to 0$, equivalently $\rho(B) < 1$.
How accurate is the result in hand?	Recompute $r = b - A\hat{x}$ and relate it to $e = -A^{-1}r$.	Conditioning governs the forward-error certificate. Rounding and the stopping rule affect the delivered answer.

For the SPD counterexample, both splittings target exactly the same unique point, yet Jacobi can move away from it at every step. For the running example both converge, yet the two methods require different numbers of sweeps to satisfy the same tolerance. Solvability, convergence of a chosen algorithm, and accuracy of a computed answer each require their own argument.

Course reading. Kanschat and Scheichl, *Numerical Linear Algebra*, §4.2 and Appendix B.1, connect stationary iterations with the spectral radius. Bastian, *num0*, §§13–14, develops splittings, convergence conditions, and stopping. Scheichl and Zech, *Num0 notes*, §6.3, gives the relaxation viewpoint. The notation here reserves M for the easy solve and B for error propagation.

9.10 Exercises

1. **Reconstruct a splitting (3 min).** For $A = \mathrm{diag}(2, 3)$, $b = (2, 3)^T$, take $M = 4I$. Find N, B, c and the fixed point. Which coordinate of a generic initial error decays more slowly?
2. **Recover the theorem (5 min).** Starting with an affine iteration and a fixed point, prove the power formula by induction. Explain why testing every initial error tests the columns of B^k, why a unit-modulus eigenvalue prevents convergence from every start, and why defective blocks with eigenvalues inside the unit circle still decay.

3. **Read a stopping report (4 min).** Let $A = \mathrm{diag}(1, 10^{-6})$, $b = (1, 0)^T$ and $\widehat{x} = (1, 1)^T$. Compute relative residual, relative error and the conditioning bound in the Euclidean norm. Separately, for $F(x) = 0.999x + 0.001$ starting at zero, what does a first change of 0.001 certify about the error at the new iterate?
4. **Which initial errors succeed? (5 min)** Take $B = \begin{pmatrix} -2 & 1 \\ 0 & 1/2 \end{pmatrix}$ and $c = 0$. Find the unique fixed point and all starts that converge to it. Contrast applying B^k to a generic start here with applying powers to select a dominant eigendirection.
5. **Mastery problem — predict and audit two sweeps (11 min).** Take $A = \begin{pmatrix} 4 & 2 \\ -1 & 2 \end{pmatrix}$, $b = (6, 1)^T$, and start at zero. Derive the two component rules, both iteration matrices and their constant vectors. Compute their spectra, predict convergence, and perform two sweeps of each. Compute the second residual for each. Use the powers of the iteration matrices to explain which features of the observed behavior should persist.

9.11 Worked solutions

1. Reconstruct a splitting. From $N = M - A = \mathrm{diag}(2, 1)$ we obtain $B = \mathrm{diag}(1/2, 1/4)$ and $c = (1/2, 3/4)^T$. Solving $(I - B)x = c$ gives $(1, 1)^T$, which also satisfies $Ax = b$. After k sweeps the first signed error is $2^{-k}e_1^{(0)}$ and the second is $4^{-k}e_2^{(0)}$. The first coordinate dominates eventually if its initial error is nonzero.

2. Recover the theorem. Subtraction gives $e^{(k+1)} = Be^{(k)}$. At $k = 0$, $B^0e^{(0)} = e^{(0)}$. If the formula holds at k, multiplication by B establishes it at $k + 1$. Taking the initial errors to be coordinate vectors tests every column; decay of all columns implies $B^k \to 0$ in finite dimensions. For a unit-modulus eigenvalue, $\|B^k v\| = |\lambda|^k \|v\| = \|v\|$ on a complex eigenvector. Decay on every real start would imply decay on its real and imaginary parts and hence on v as well. Inside the unit disk a Jordan block contributes terms of the form $\binom{k}{j}\lambda^{k-j}$ for finitely many j. These vanish by exponential decay for $\lambda \neq 0$ and nilpotence for $\lambda = 0$. Thus every block decays. Also 1 is excluded, so $(I - B)x = c$ has one solution.

3. Read a stopping report. Here $x^* = (1, 0)^T$, $e = (0, 1)^T$, and $r = (0, -10^{-6})^T$. The relative residual is 10^{-6} and the relative error is 1. The condition number is 10^6, so its upper bound on the relative error is 1, attained in this direction. In the scalar problem the fixed point is 1 and $x^{(1)} = 0.001$. The a posteriori estimate gives $0.999/(1 - 0.999)$ times 0.001, hence 0.999. This equals the actual error magnitude. The small change certifies little progress toward a substantially accurate solution.

4. Which initial errors succeed? The diagonal entries are the eigenvalues -2 and $1/2$. Since 1 is absent, the unique fixed point is zero. Eigenvectors are $(1, 0)^T$ for -2 and $(2, 5)^T$ for $1/2$. Writing the start as $\alpha(1, 0)^T + \beta(2, 5)^T$, its two coefficients become $(-2)^k\alpha$ and $2^{-k}\beta$. Precisely the line $\alpha = 0$ converges to zero. For generic starts the growth spoils the solve, while

normalization would make the *direction*, up to sign, approach the dominant eigenline. The same powers explain both phenomena.

5. Mastery problem. Write (u, v) for the old coordinates. In both methods $u' = (6-2v)/4$. Jacobi uses $v' = (1+u)/2$; Gauss–Seidel uses $v' = (1+u')/2 = 5/4 - v/4$. Thus

$$B_J = \begin{pmatrix} 0 & -1/2 \\ 1/2 & 0 \end{pmatrix}, \qquad c_J = (3/2, 1/2)^T,$$
$$B_{GS} = \begin{pmatrix} 0 & -1/2 \\ 0 & -1/4 \end{pmatrix}, \qquad c_{GS} = (3/2, 5/4)^T.$$

Jacobi has characteristic polynomial $t^2 + 1/4$ and eigenvalues $\pm i/2$. The triangular Gauss–Seidel matrix has eigenvalues $0, -1/4$. Both radii are below one. Substituting successively and multiplying by A for the last column gives

Method	$x^{(1)}$	$x^{(2)}$	$b - Ax^{(2)}$
Jacobi	$(3/2, 1/2)^T$	$(5/4, 5/4)^T$	$(-3/2, -1/4)^T$
Gauss–Seidel	$(3/2, 5/4)^T$	$(7/8, 15/16)^T$	$(5/8, 0)^T$

The exact solution is $(1, 1)^T$. Jacobi rotates the error through a quarter turn and halves its Euclidean length; $B_J^2 = -I/4$. After two sweeps its signed error is $(1/4, 1/4)^T$. For Gauss–Seidel the range of B_{GS} is the line spanned by $(2, 1)^T$, on which B_{GS} multiplies by $-1/4$. Its first error $(1/2, 1/4)^T$ becomes $(-1/8, -1/16)^T$. Its second-row residual vanishes at the end of every sweep by construction.

9.12 End-of-session mastery checklist

Before Session 9, close the chapter and check that you can independently:

- Explain the sparse-system motivation and trade the cost of a sweep against the number of sweeps.
- Start from $A = M - N$, identify B, c, verify the fixed point, and recover the residual correction.
- Derive Jacobi and Gauss–Seidel by choosing the part of A to solve with. Perform two sweeps of each and identify exactly which entries use new values.
- Derive $e^{(k+1)} = Be^{(k)}$ and prove $e^{(k)} = B^k e^{(0)}$.
- Prove the equivalence of convergence from every start, decay of matrix powers, and $\rho(B) < 1$, including the defective case.
- Distinguish a norm-based sufficient test from the spectral criterion. Give an example of transient growth.

- State the diagonal-dominance and positive-definiteness guarantees with the correct methods. Exhibit an invertible system on which Jacobi diverges.
- Relate the observable residual to the signed error, interpret a stopping tolerance, and explain what rounding changes.
- Explain how the same spectral-component calculation describes error elimination, selection of a dominant direction, and polynomial filtering.

10 Session 9 — Power and Subspace Iteration

10.1 From a linear solve to a spectral question

Given $A \in \mathbb{F}^{n\times n}$, with $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$, seek $v \neq 0$ and λ such that $Av = \lambda v$. If A has a million rows and we want just one eigenpair, what operation should we ask the matrix to perform? Forming the coefficients of $\det(tI - A)$ and finding all its roots introduces a potentially ill-conditioned polynomial problem, does work for the entire spectrum, and still leaves us to find v. In contrast, a single sparse product Av costs work proportional to the number of stored entries plus the vector dimensions. Sometimes a routine applying A is all we have.

Session 7 used factorizations to solve $Ax = b$. In Session 8, a stationary iteration propagated the solution error as $e^{(k)} = B^k e^{(0)}$. Convergence for every start required $\rho(B) < 1$. Here we deliberately retain a component. Even when **every** component of $A^k v$ shrinks, its *direction* can reveal an eigenvector. The decisive quantity compares two rates of shrinkage or growth.

Study route (90 minutes). Spend 20 minutes on the single-vector derivation and diagnostics, 25 on invariant subspaces and the block proof, and 15 on the experiment and seminar connection. Reserve 30 minutes for the exercises, including checking your solutions.

10.2 What survives repeated multiplication?

Use the spectral theorem from Session 5. For Hermitian $A = A^*$, write $A = Q\Lambda Q^*$, where $Q = [q_1, \dots, q_n]$ is unitary and Λ is real diagonal. The star denotes conjugate transpose. Normalize the start to unit length. Orthogonal expansion (Session 2) gives $v^{(0)} = \sum_i c_i q_i$ with $c_i = q_i^* v^{(0)}$. Since each application multiplies q_i by λ_i, induction gives

$$A^k v^{(0)} = \sum_{i=1}^{n} c_i \lambda_i^k q_i.$$

Order by magnitude, $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$, and assume $c_1 \neq 0$. Factoring out the largest weight gives

$$A^k v^{(0)} = \lambda_1^k \left(c_1 q_1 + \sum_{i=2}^{n} c_i (\lambda_i/\lambda_1)^k q_i \right).$$

Each unwanted coefficient divided by the wanted coefficient tends to zero. The prefactor can be huge, tiny, or alternate in sign. We can discard its magnitude after *each* product because multiplying an input by a nonzero scalar preserves the output line whenever the output is nonzero.

10.2.1 Normalized power iteration

Starting with a unit vector, repeat

$$w^{(k+1)} = Av^{(k)}, \qquad v^{(k+1)} = \frac{w^{(k+1)}}{\|w^{(k+1)}\|_2}.$$

Check for a zero product before dividing. It means the current vector lies in the kernel. Under the stated gap and overlap assumptions the wanted coefficient survives every finite step in exact arithmetic. By induction $v^{(k)} = A^k v^{(0)}/\|A^k v^{(0)}\|_2$. Normalization limits accumulation of extreme scales across steps. The implementation needs one product per update and never forms A^k.

10.2.2 The direction error and its rate

The acute angle θ_k between two **lines** satisfies $\cos\theta_k = |q_1^* v^{(k)}|$. The absolute value identifies unit representatives that differ by a sign or a complex phase. With $P_1 = q_1 q_1^*$, Session 3 gives the tangent as the length perpendicular to the target divided by the length parallel to it. Taking these lengths before normalization cancels a common factor.

$$\begin{aligned}
\tan\theta_k &= \frac{\|(I - P_1)A^k v^{(0)}\|_2}{\|P_1 A^k v^{(0)}\|_2} \\
&= \frac{\left(\sum_{i=2}^{n} |c_i|^2 |\lambda_i/\lambda_1|^{2k}\right)^{1/2}}{|c_1|} \\
&\le \left|\frac{\lambda_2}{\lambda_1}\right|^k \tan\theta_0.
\end{aligned}$$

Power convergence theorem. For an orthonormal eigenbasis, a strict magnitude gap at the first eigenvalue, and nonzero initial overlap, this bound implies convergence of the line. A smaller ratio means fewer steps for the same guaranteed angular reduction. The absolute magnitude gap $g = |\lambda_1| - |\lambda_2|$ has the units of A. The factor $1 - g/|\lambda_1|$ is dimensionless and unchanged by scaling A by a nonzero scalar. A nearly orthogonal start can have a large $\tan\theta_0$. Missing slow modes can make convergence faster than the bound.

The normalized vector approaches the moving representative $(\lambda_1/|\lambda_1|)^k(c_1/|c_1|)q_1$ in the sense that their difference tends to zero. A negative dominant eigenvalue alternates the sign. For a complex normal matrix the same proof uses a unitary eigenbasis, and a nonreal dominant eigenvalue rotates the phase. In all these cases the orthogonal projector onto the line has a single limit.

10.2.3 Worked example 1 — rotate into eigencoordinates

Take $A = \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix}$ and $v^{(0)} = e_1$. Its orthonormal eigenvectors are $q_1 = (1,1)^T/\sqrt{2}$ and $q_2 = (1,-1)^T/\sqrt{2}$, with eigenvalues 4 and 2. Thus $e_1 = (q_1+q_2)/\sqrt{2}$ and $A^k e_1 = (4^k q_1 + 2^k q_2)/\sqrt{2}$. Three applications in the original coordinates give

$$\begin{aligned} w^{(1)} &= (3,1)^T, & v^{(1)} &= (3,1)^T/\sqrt{10}, \\ w^{(2)} &= (10,6)^T/\sqrt{10}, & v^{(2)} &= (5,3)^T/\sqrt{34}, \\ w^{(3)} &= (18,14)^T/\sqrt{34}, & v^{(3)} &= (9,7)^T/\sqrt{130}. \end{aligned}$$

For $(x,y)^T$ the ratio of the two eigenbasis coefficients in magnitude is $|x-y|/|x+y|$. Across the four vectors it is $1, 1/2, 1/4, 1/8$. Here the tangent bound is exact. In ordinary coordinates the entries are becoming equal because the *difference* direction is being suppressed.

10.2.4 Worked example 2 — where selection fails

Let $A = \mathrm{diag}(2,-2,1)$ and $v^{(0)} = (1,1,0)^T/\sqrt{2}$. The first two weights have equal magnitude. The lines alternate between the spans of $(1,1,0)^T$ and $(1,-1,0)^T$, which are perpendicular. Both lie in the same invariant plane. Equal magnitudes of *different* eigenvalues can therefore prevent even line convergence. For a repeated copy of the *same* dominant eigenvalue, the whole dominant eigenspace receives one common weight. With a gap below it and nonzero projection onto it, the limiting line is the line of that projection and depends on the start.

If the dominant component is absent, its coefficient stays zero. Starting Example 1 at q_2 gives the eigenpair for 2 forever. For a general diagonalizable matrix the expansion uses coefficients in a possibly ill-conditioned eigenvector basis, and the orthogonal sum-of-squares proof no longer applies. With defective matrices even the pure-power expansion can fail. For example a Jordan block $\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ has upper-right entry k in its kth power.

10.3 Estimating and checking an eigenpair

For $v \neq 0$ define the Rayleigh quotient $\rho_A(v) = v^*Av/(v^*v)$. It is the coefficient of the orthogonal projection of Av onto v. Hence $r = Av - \rho_A(v)v$ satisfies $v^*r = 0$, and $r = 0$ precisely for an eigenvector. If $Av = \lambda v$, substitution gives $\rho_A(v) = \lambda$, including its sign. In a loop, reuse the product Av to evaluate the quotient and residual for the **current** iterate.

In Example 1, $A(5,3)^T = (18,14)^T$, so $\rho_A(v^{(2)}) = 132/34 = 66/17$. The exact value is 4. More generally, for a unit vector $v = \sum_i a_i q_i$ the Hermitian quotient is $\sum_i |a_i|^2 \lambda_i$. Therefore $|\rho_A(v) - \lambda_1| \leq \max_i |\lambda_i - \lambda_1| \sin^2 \theta$. A good direction can give an even more accurate eigenvalue.

As in Session 6, a small residual needs a scale and an interpretation. For unit v, $\|r\|_2^2 = \sum_i |a_i|^2 |\lambda_i - \rho_A(v)|^2$. Thus some eigenvalue lies within $\|r\|_2$ of the quotient. In a tight cluster, a mixture of well-separated *directions* can have a small residual because their eigenvalues are close. Even a zero residual can identify a different eigenpair from the one sought. For scale-independent monitoring one can compare $\|r\|_2$ with a tolerance times $\|A\|_2 \|v\|_2$ or an available operator scale.

10.4 Which object should a block approximate?

A subspace U is **invariant** under A if $A(U) \subseteq U$. Once a vector is in U, repeated products stay there. An eigenline is the one-dimensional case. Any span of eigenvectors is invariant by linearity. By contrast the line through e_1 in Example 1 leaves itself on the first multiplication. An invariant subspace can contain very few eigenvectors: the entire plane of the Jordan block above is invariant, but its only eigenline is the first coordinate axis.

For Hermitian A, an invariant U also has invariant orthogonal complement: if $x \perp U$, then $u^*Ax = (Au)^*x = 0$ for every $u \in U$. The restrictions are Hermitian and admit orthonormal eigenbases. In particular a group of eigenvalues identifies a spectral subspace. When eigenvalues inside the group coincide, individual basis vectors are nonunique. A gap to the **outside** can still distinguish the group cleanly. This is a reason to ask an eigensolver for a subspace.

Start with p independent columns, $X_0 \in \mathbb{F}^{n \times p}$, $1 \leq p < n$. The raw block iteration is $X_{k+1} = AX_k$, hence $X_k = A^k X_0$. Its geometric iterate is $\mathcal{S}_k = \text{range}(X_k)$. Right multiplication by any invertible $p \times p$ matrix changes the coordinates of a basis, preserving its span. This is the block analogue of choosing a different length for a vector. It gives us freedom to choose a numerically favorable basis at every step.

10.4.1 Why separate column normalizations are insufficient

Let $A = \text{diag}(8,2,1)$ and start with columns $e_1 + e_2$ and $e_1 - e_2$. After k steps the columns are $8^k e_1 \pm 2^k e_2$. For every finite k their sum and difference recover e_1 and e_2 up to nonzero factors.

The subspace is already exactly the desired plane. Nevertheless the cosine between the columns is $(8^{2k} - 2^{2k})/(8^{2k} + 2^{2k})$, tending to one. Normalizing each column leaves this angle unchanged.

The two nonzero singular values are $\sqrt{2}\,8^k$ and $\sqrt{2}\,2^k$ (compute $X_k X_k^*$). Thus $\kappa_2(X_k) = 4^k$, also after independent column normalization in this example since the two norms agree. In exact arithmetic there are still two independent directions. In floating point, a perturbation of order machine precision times the large singular value can swamp the small one. Extreme overall scaling introduces a further risk of overflow or underflow. The task for orthogonalization is to represent the second direction **while it is still resolved**.

10.4.2 Orthogonal subspace iteration

Take a reduced QR factorization of the starting block and use its orthonormal factor Q_0. Repeat

$$Y_{k+1} = AQ_k, \qquad Y_{k+1} = Q_{k+1}R_{k+1}.$$

Session 4 supplies the reduced QR with $Q_{k+1}^* Q_{k+1} = I_p$ and upper triangular R_{k+1}. At full column rank R_{k+1} is invertible, so $Q_{k+1} = Y_{k+1}R_{k+1}^{-1}$ proves equality of the ranges. This identity is a proof, not an instruction to form an inverse. Induction now gives $\mathrm{range}(Q_k) = \mathrm{range}(A^k X_0)$. Multiplication advances the plane. QR assigns it coordinates in which all singular values of the basis are one. Householder QR provides this change of basis reliably for a resolved block. If the product loses rank, an orthonormal completion returned by a QR routine can contain directions outside the product range. It cannot recover missing information.

10.4.3 Worked example 3 — a plane moves, QR represents it

Use

$$A = \begin{pmatrix} 3 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad Q_0 = [e_1,\, (e_2 + e_3)/\sqrt{2}].$$

Example 1 identifies the wanted eigenvalues $4, 2$ in the first coordinate plane. The initial columns project to independent vectors there. The product columns are $y_1 = (3,1,0)^T$ and $y_2 = (1,3,1)^T/\sqrt{2}$. Subtract the component of y_2 along y_1 to get $(-4,12,5)^T/(5\sqrt{2})$. Normalizing gives

$$Q_1 = \left[\frac{(3,1,0)^T}{\sqrt{10}},\, \frac{(-4,12,5)^T}{\sqrt{185}}\right], \qquad R_1 = \begin{pmatrix} \sqrt{10} & 6/\sqrt{20} \\ 0 & \sqrt{185}/(5\sqrt{2}) \end{pmatrix}.$$

The dot product numerator is $-12 + 12 = 0$ and the squared lengths before normalization are 10 and 185. Thus $Q_1^T Q_1 = I_2$. To see what happened to the plane, write it as a height z over

(x, y). Initially $z = y$. After one product both new columns satisfy $z = (-x + 3y)/8$. The maximum absolute height above a unit horizontal vector has fallen from 1 to $\sqrt{10}/8$. This slope is the tangent of the largest angle to the target plane. The next proof generalizes exactly this calculation.

10.5 The block convergence estimate

Partition the orthonormal eigenbasis as $[U_1, U_2]$, where U_1 spans the wanted p directions. Write Λ_1, Λ_2 for the corresponding diagonal blocks. Assume $|\lambda_p| > |\lambda_{p+1}|$ in decreasing magnitude order. Decompose the starting block as $X_0 = U_1 C + U_2 D$, with $C = U_1^* X_0$ and $D = U_2^* X_0$. The exact overlap condition is that the square matrix C is invertible: the projected starting vectors must span the *entire* target. It is possible for every starting vector to have a wanted component and for all those components to lie on the same line.

Claim. Under these assumptions, in exact arithmetic every product has rank p and the largest principal angle Θ_k between $\mathcal{S}_k$ and range(U_1) satisfies

$$\tan \Theta_k \leq \left| \frac{\lambda_{p+1}}{\lambda_p} \right|^k \tan \Theta_0.$$

Proof. In eigencoordinates the top and bottom blocks of $A^k X_0$ are $\Lambda_1^k C$ and $\Lambda_2^k D$. The top block is invertible since the gap implies $|\lambda_p| > 0$. This proves the rank assertion. Right-multiply by the inverse of that top block. The resulting basis is $U_1 + U_2 T_k$, where

$$T_k = \Lambda_2^k D C^{-1} \Lambda_1^{-k}.$$

Every vector in the trial subspace now has a unique representation $U_1 s + U_2 T_k s$. Its tangent of the angle to the target is $\|T_k s\|_2 / \|s\|_2$ for $s \neq 0$, by orthogonality. Maximizing gives $\tan \Theta_k = \|T_k\|_2$. Submultiplicativity (Session 6) and the norms of the diagonal factors give the claim, since $T_0 = DC^{-1}$. This proof also covers complex normal A. For $p = 1$, T_k is a column of unwanted-to-wanted coefficient ratios, precisely the single-vector argument.

The numerator uses the **strongest unwanted** mode and the denominator the **weakest wanted** mode. Internal ties are harmless if the boundary gap holds. A nearly singular overlap can make the initial angle close to a right angle. Full rank of X_0 alone says nothing about that overlap: for $A = \mathrm{diag}(4, 2, 1)$ the block $[e_1, e_3]$ has full rank and stays in the wrong invariant plane forever.

10.6 How can we tell whether a computed plane is good?

For an orthonormal basis Q_k, the projector is $P_k = Q_k Q_k^*$. Replacing Q_k by $Q_k W$ for unitary W leaves P_k unchanged. Even $-Q_k$ represents exactly the same plane while $\|Q_k - (-Q_k)\|_2 = 2$.

For equal-dimensional spaces the spectral norm of the difference of their orthogonal projectors equals the sine of their largest principal angle. Thus $\|P_k - U_1 U_1^*\|_2 = \sin\Theta_k$ when the target is known. Without it, $\|P_{k+1} - P_k\|_2$ measures movement. A wrong invariant plane can be motionless, and a small boundary gap can make progress slow.

A complementary test uses only a product and a small matrix:

$$H_k = Q_k^* A Q_k, \qquad F_k = A Q_k - Q_k H_k = (I - P_k) A Q_k.$$

Here F_k denotes the **invariance residual**, reserving R_k for the triangular QR factor. The columns of H_k give the coordinates of the projected images of the basis vectors. F_k contains exactly the parts outside the trial space. Hence $F_k = 0$ if and only if the space is invariant. Under $Q_k \mapsto Q_k W$, F_k becomes $F_k W$, preserving its norm. In practice one can monitor $\|F_k\|_2$ relative to an operator scale alongside the spectral region represented by H_k. Small leakage measures near-invariance and cannot by itself identify which cluster has been found.

10.7 Experiment — one matrix, three iterations

Use a real symmetric matrix with known eigenvalues $8, 2, 1, 1/2$. Rotate the eigenbasis into ordinary coordinates so that a small spectral component can be hidden by cancellation. In eigencoordinates take the first starting vector proportional to $(1, 1, 1, 0)^T$ and the second block column proportional to $(1, -1, -1, 0)^T$. Adding and subtracting the columns isolates the first eigenvector and a mixture of the second and third. Predict the two curves before running the code. The line has tangent $\sqrt{(2/8)^{2k} + (1/8)^{2k}}$, while the plane has tangent $(1/2)^k$. Its projector distance is therefore $t/\sqrt{1+t^2}$ for $t = (1/2)^k$.

```
import numpy as np
import matplotlib.pyplot as plt

U = 0.5 * np.array([[1, 1, 1, 1], [1, -1, 1, -1],
                    [1, 1, -1, -1], [1, -1, -1, 1]])
A = (U * [8., 2., 1., 0.5]) @ U.T
X = U @ np.array([[1., 1.], [1., -1.], [1., -1.], [0., 0.]])
Q = np.linalg.qr(X, mode="reduced")[0]
v = X[:, 0] / np.linalg.norm(X[:, 0])
target = U[:, :2] @ U[:, :2].T
eps = np.finfo(float).eps

def resolved(s, shape):
    return s[-1] > max(shape) * eps * s[0]

def plane_error(basis):
```

```
    return np.linalg.norm(basis @ basis.T - target, ord=2)

steps = np.arange(37)
line, raw, orth, cond, defect = [], [], [], [], []
for k in steps:
    a = U.T @ v
    line.append(np.linalg.norm(a[1:]) / abs(a[0]))
    left, s, _ = np.linalg.svd(X, full_matrices=False)
    ok = resolved(s, X.shape)
    raw.append(plane_error(left) if ok else np.nan)
    cond.append(s[0] / s[-1] if s[-1] > 0 else np.inf)
    orth.append(plane_error(Q))
    defect.append(np.linalg.norm(Q.T @ Q - np.eye(2), ord=2))
    if k == steps[-1]:
        break

    w = A @ v
    length = np.linalg.norm(w)
    if length == 0 or not np.isfinite(length):
        raise ArithmeticError("Power product cannot be normalized")
    v = w / length
    X = A @ X
    Y = A @ Q
    if not resolved(np.linalg.svd(Y, compute_uv=False), Y.shape):
        raise ArithmeticError("QR input has lost numerical rank")
    Q = np.linalg.qr(Y, mode="reduced")[0]

fig, (ax, bx) = plt.subplots(1, 2, figsize=(7, 3.6),
                             constrained_layout=True)
# Exact zeros cannot be plotted on logarithmic axes.
def positive(values):
    values = np.asarray(values, dtype=float)
    return np.where((values > 0) & np.isfinite(values), values, np.nan)

t = 0.5 ** steps
ax.semilogy(steps, positive(line), "C0.-", label="Power tangent")
ax.semilogy(steps, positive(raw), "C3x", label="Raw plane")
ax.semilogy(steps, orth, "C2-", label="QR plane")
ax.semilogy(steps, t / np.hypot(1, t), "k:", label="Exact plane")
bx.semilogy(steps, positive(cond), "C3.-", label="Raw condition number")
bx.semilogy(steps, positive(defect), "C2.", label="QR orthogonality error")
bx.axhline(eps, color="0.4", ls=":", label="Machine epsilon")
```

```
for axis, ylabel in [(ax, "Angle or projector error"), (bx, "Basis diagnostic")]:
    axis.set(xlabel="Iteration", ylabel=ylabel)
    axis.grid(True, which="major", alpha=0.25)
    axis.legend(fontsize=7)
plt.show()
```

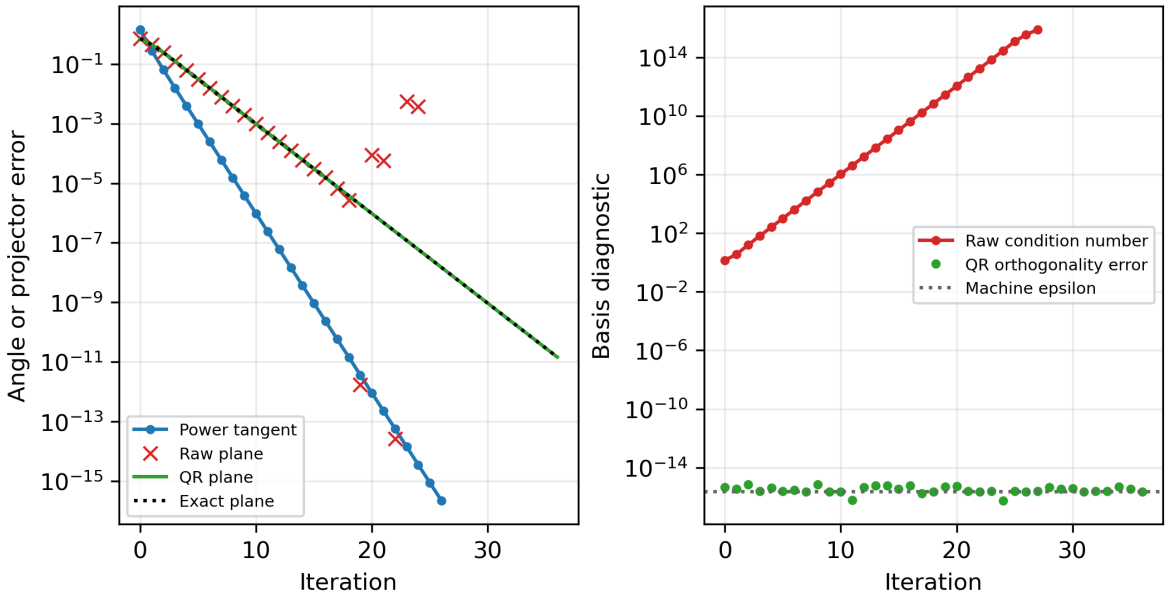


Figure 10.1: Direction and plane errors (left), and basis quality (right). The raw plane is plotted only while its second singular direction passes the numerical rank test.

Reading the curves. The line initially contracts faster than the plane because its slowest relative weight is $1/4$ versus $1/2$ for the plane. Once components reach rounding scale the exact-arithmetic rates cease to predict further measured improvement. The QR plane follows the predicted curve even as the raw columns become difficult to distinguish. Here the raw dominant singular value grows relative to the second roughly as 4^k. Before the threshold is reached, the raw-plane error turns upward: the weaker direction is already sensitive to rounding. In a binary64 run the raw block last passed the test at step 24, while at step 36 the QR projector error was about 1.46×10^{-11}, in agreement with 2^{-36}. The precise rank-loss step can vary with the numerical libraries. Computing a second orthogonal vector from a numerical completion at that point would give a misleading plane comparison. The right panel measures two different properties: the sensitivity of the raw basis and the deviation of the QR Gram matrix from identity. Exact zeros are omitted on logarithmic axes. Explicit projectors and diagnostic SVDs are affordable for this four-row experiment.

10.8 Central Synthesis — From Spectral Amplification to Invariant Subspace Computation

For one vector, spectral decomposition turns a product into independent scalar multiplications. Repetition raises each weight to the kth power, so ratios select a line. For a block, every column undergoes the same weighting. Provided the wanted projections span their target, the graph T_k measures the largest surviving unwanted component per unit wanted component. The boundary ratio controls the decay of this graph. It says nothing about how nearly parallel the raw columns may have become along the way.

Task	Operation	What it accomplishes
Select components	Apply A, or $p_m(A)$	Weight an eigenvector by λ_i, or by $p_m(\lambda_i)$.
Approximate a space	Iterate the image of a block	Reduce the tilt of the trial space toward unwanted directions.
Represent the space	Compute reduced QR after each product	Preserve the full-rank image with orthonormal coordinates.
Extract eigenpairs	Compress to Q^*AQ and solve the small eigenproblem	Choose directions *within* the computed space and lift them with Q.

In the seminar algorithm the update changes from $Y = AQ$ to $Y = p_m(A)Q$. From $A^j q_i = \lambda_i^j q_i$, linearity gives $p_m(A)q_i = p_m(\lambda_i)q_i$. For a Hermitian matrix, the same block proof applies with the per-sweep factor

$$\frac{\max_{i \text{ unwanted}} |p_m(\lambda_i)|}{\min_{i \text{ wanted}} |p_m(\lambda_i)|}.$$

The denominator must be positive, the ratio strictly below one, and the initial wanted overlap invertible. This can select a spectral region whose original eigenvalues are small in magnitude. QR after filtering preserves the computed full-rank image and keeps its directions resolved for the next sweep. Session 10 develops interval scaling, Chebyshev polynomials and their recurrence for applying the filter. Session 11 develops Rayleigh–Ritz extraction, residual orthogonality and their place in the complete iteration.

Reading connections. The course book's §§1.3, 1.3.3 and 1.4.4 locate this session at the start of Week 3. Kanschat and Scheichl, *Numerical Linear Algebra*, §§3.3.1–3.3.2 give the vector and orthogonal iterations (Algorithms 3.3.3 and 3.3.20). Their §5.1, especially Algorithm 5.1.10, leads to the projected problem. For supporting definitions see Scheichl and Zech, §3.2 on symmetric matrices and the spectral norm, and Bastian, §5.3 on eigenvalues and eigenvectors.

10.9 Exercises

1. **Compute a line (4 min).** For $A = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$ start at e_1. Decompose it in an orthonormal eigenbasis. Compute the first two normalized iterates, their angular tangents to the dominant line, and the Rayleigh quotient after the second step.
2. **Prove a rate (5 min).** For Hermitian A with a strict first magnitude gap and $c_1 \neq 0$, derive the tangent bound from $A^k v^{(0)}$. When is it an equality for every k? Compare spectra $(5, 4)$ and $(101, 100)$ for equal starting angles.
3. **Diagnose selection (4 min).** For $A = \operatorname{diag}(-5, 5, 2)$ describe the normalized vectors and lines from e_1, $(e_1 + e_2)/\sqrt{2}$, and e_3. Identify the dominant plane and its block convergence factor. Which starts give an exact eigenpair residual of zero immediately?
4. **A misleading basis (4 min).** For the block $[8^k e_1 + 2^k e_2, 8^k e_1 - 2^k e_2]$ find an orthonormal basis for its range and its orthogonal projector. Explain the consequence of perturbations of relative size 4^{-k}. If a QR routine returns the two orthonormal columns in reverse order at the next step, what do the projector change and invariance residual report for $A = \operatorname{diag}(8, 2, 1)$?
5. **Track a plane (8 min).** Let $A = \operatorname{diag}(6, 3, 1)$. Identify the dominant plane and choose an orthonormal starting block containing e_1 whose other column is equally inclined to e_2 and e_3 in their plane. Take one product, compute a reduced QR, verify the initial overlap condition, and find the initial and updated projector distances to the target. Predict the distance after k steps. Would $[e_1, e_3]$ work?
6. **Redirect the iteration (4 min).** For the same spectrum $(6, 3, 1)$ seek the plane for eigenvalues $3, 1$. Compare ordinary iteration with the filter $p(t) = 7 - t$. Determine the filtered convergence factor under suitable initial overlap. Explain what QR and the later projected eigenproblem each contribute.

10.10 Worked solutions

1. The sum and difference directions $q_+ = (1, 1)^T/\sqrt{2}$ and $q_- = (1, -1)^T/\sqrt{2}$ have eigenvalues $3, -1$. The start is $(q_+ + q_-)/\sqrt{2}$. The first product is $(1, 2)^T$ and its normalization divides by $\sqrt{5}$. The next product is $(5, 4)^T/\sqrt{5}$, giving $(5, 4)^T/\sqrt{41}$. The ratios $|x-y|/|x+y|$ are $1/3$ and $1/9$. The unwanted coefficient alternates in sign. The second quotient is $(5, 4)(13, 14)^T/41 = 121/41$.

2. Project $\sum_i c_i \lambda_i^k q_i$ onto q_1 and its complement. Orthogonality gives squared lengths $|c_1 \lambda_1^k|^2$ and $\sum_{i>1} |c_i \lambda_i^k|^2$. Divide the latter by the former and take the square root. Bounding each unwanted weight by $|\lambda_2/\lambda_1|^k$ gives the claim. Equality for every k holds precisely when each nonzero unwanted coefficient has $|\lambda_i| = |\lambda_2|$ (including the case of no unwanted coefficient). The absolute gaps both equal one. The tangent contracts by $4/5$ in the first two-mode problem and $100/101$ in the second, so the first is faster for a nonzero finite starting tangent.

3. From e_1 the vectors are $(-1)^k e_1$ and the line is fixed. From the mixed vector they are $((-1)^k e_1 + e_2)/\sqrt{2}$ and the two lines alternate. From e_3 both vector and line are fixed. The first and third starts have zero residual, although the third selects the smallest magnitude. The dominant plane is $\operatorname{span}(e_1, e_2)$, with a boundary ratio $2/5$ for a two-column start with invertible wanted projection.

4. The sum and difference of the columns give multiples of e_1, e_2. These unit vectors form an orthonormal basis and the projector is $\operatorname{diag}(1, 1, 0)$ at every step. Because the smaller singular value is 4^{-k} times the larger, a relative perturbation of that size in spectral norm can erase the smaller singular direction entirely. Column normalization cannot change the angle. Swapping the basis order preserves the projector, so its change is zero. The residual is also zero since both images remain in the plane. Individual columns can change by a large amount even with these two quantities exactly zero.

5. The wanted plane is $\operatorname{span}(e_1, e_2)$. Use $Q_0 = [e_1, (e_2 + e_3)/\sqrt{2}]$. The wanted projection in target coordinates is $\operatorname{diag}(1, 1/\sqrt{2})$, which is invertible. The product columns $6e_1$ and $(3e_2 + e_3)/\sqrt{2}$ are orthogonal. Dividing by their lengths gives

$$Q_1 = [e_1, (3e_2 + e_3)/\sqrt{10}], \qquad R_1 = \operatorname{diag}(6, \sqrt{5}).$$

The Gram matrix of Q_1 is I_2. One principal angle stays zero because the planes share e_1. The other's sine falls from $1/\sqrt{2}$ to $1/\sqrt{10}$, the out-of-plane component of the unit second column. These are the projector distances. After k steps the second direction is proportional to $3^k e_2 + e_3$, so the distance is $1/\sqrt{3^{2k} + 1}$. It is the tangent that contracts exactly by $1/3$ each step. With $[e_1, e_3]$ the projected block has rank one and the iteration remains in that starting plane.

6. Ordinary iteration generally favors 6 and 3 when their projected block has full rank. The filter assigns values $1, 4, 6$ to $6, 3, 1$, respectively. The weakest wanted weight is 4, the unwanted weight 1, and the block factor is $1/4$. The required overlap is full rank on the plane for $3, 1$. QR gives an orthonormal basis for the filtered image. The later eigensolve on $Q^* A Q$ resolves approximate eigenpairs of the original A within this space.

10.11 End-of-session mastery checklist

- ☐ Expand a start in eigencoordinates, derive the normalized power iteration and its tangent bound, and state the gap, overlap and nonbreakdown assumptions.
- ☐ Explain the difference between a magnitude gap and a convergence factor, and predict the effects of a negative, phase-rotating or tied dominant eigenvalue on a vector and its line.
- ☐ Compute a Rayleigh quotient and residual, and explain what each can establish about an approximate eigenpair.

- ☐ Define invariance, give an example, and derive the block update from repeated image spaces.
- ☐ Identify the boundary gap and full-rank overlap; derive the graph bound using the strongest unwanted and weakest wanted weights.
- ☐ Derive the QR update, prove full-rank span preservation, and explain numerical rank loss using two nearly parallel unit columns.
- ☐ Compare subspaces with projectors and assess invariance with a residual even when basis vectors rotate.
- ☐ Predict how polynomial weights redirect or accelerate selection, and distinguish the roles of filtering, orthonormalization and eigenpair extraction.

11 Session 10 — Matrix Polynomials and Chebyshev Filtering

Session 9 gave us a way to compute a dominant invariant subspace using only products with A and QR factorizations. Suppose the wanted eigenvalues of a Hermitian matrix lie above an unwanted cluster. Can we use the same operations, but make a single sweep distinguish the two groups more sharply? The answer starts with a scalar problem on the real line. Once we prescribe how much weight each eigenvalue should receive, a polynomial recurrence transfers that design to the matrix.

Study route (90 minutes). Spend 10 minutes on spectral mapping and selection, 15 on Chebyshev polynomials and minimax, 10 on interval design, 10 on implementation, and 15 on the block estimate, experiment and synthesis. Reserve 30 minutes for practice and checking. By the end, you should be able to design a filter from interval bounds, predict its effect on a trial space, and apply it using only a block-product routine.

11.1 From powers to designed spectral weights

Write $A = Q\Lambda Q^*$, where $A = A^*$, $Q = [q_1, \dots, q_n]$ is unitary, and $\Lambda = \operatorname{diag}(\lambda_1, \dots, \lambda_n)$ is real. For $v = \sum_i c_i q_i$, orthogonality gives $c_i = q_i^* v$. Session 9 used

$$A^k v = \sum_i c_i \lambda_i^k q_i.$$

Relative to a nonzero component at $\lambda_1 \neq 0$, the magnitude of component i changes by $|\lambda_i/\lambda_1|^k$. Thus power iteration favors the largest magnitude present initially. Normalizing the vector leaves these ratios unchanged. Even a large negative eigenvalue competes on equal terms with a positive one of the same magnitude.

For a scalar polynomial $p_m(t) = \sum_{j=0}^{m} a_j t^j$, define

$$p_m(A) = a_0 I + a_1 A + \cdots + a_m A^m.$$

This is defined for every square A because $A^0 = I$ and every power has the same dimensions. Since $AA^j = A^j A$, the sum commutes with A. The choice $p_m(t) = t^m$ recovers m successive products. Other choices combine the images $v, Av, \dots, A^m v$ with coefficients that can produce cancellation in selected directions.

Spectral mapping. If $Aq = \lambda q$, induction gives $A^j q = \lambda^j q$ for all $j \geq 0$. Linearity then gives $p_m(A)q = p_m(\lambda)q$. In the Hermitian case, collecting these identities columnwise and multiplying by Q^* yields

$$p_m(A) = Q p_m(\Lambda) Q^*, \qquad p_m(A)v = \sum_i c_i p_m(\lambda_i) q_i.$$

Here $p_m(\Lambda)$ has diagonal entries $p_m(\lambda_i)$. This is the simplest example of applying a scalar function to an operator. Every original eigenvector remains an eigenvector, including when its mapped eigenvalue is zero. Distinct eigenvalues can map to the same value, so the transformed operator can have larger eigenspaces.

More generally, for any finite square matrix, with spectrum taken over $\mathbb{C}$,

$$\sigma(p_m(A)) = p_m(\sigma(A)).$$

Indeed, a complex Schur form is upper triangular in unitary coordinates. Its polynomial is triangular with diagonal entries $p_m(\lambda_i)$, which are exactly its eigenvalues. This argument includes defective matrices. The **orthogonal component estimates** below use an orthonormal eigenbasis, available for normal matrices. The real interval design also uses the reality of the Hermitian spectrum.

11.2 What makes a filter effective?

Partition the eigenvalues into nonempty wanted and unwanted sets σ_{w} and σ_{u}, taking whole eigenspaces. For a polynomial $p = p_m$ define

$$\alpha = \min_{\lambda \in \sigma_{\mathrm{w}}} |p(\lambda)| > 0, \qquad \beta = \max_{\lambda \in \sigma_{\mathrm{u}}} |p(\lambda)|, \qquad \eta_m = \frac{\beta}{\alpha}.$$

Decompose $v = v_{\mathrm{w}} + v_{\mathrm{u}}$ orthogonally. Squaring the coefficients in the spectral expansion and summing shows that $\|p(A)v_{\mathrm{u}}\|_2 \leq \beta \|v_{\mathrm{u}}\|_2$ and $\|p(A)v_{\mathrm{w}}\|_2 \geq \alpha \|v_{\mathrm{w}}\|_2$. For $v_{\mathrm{w}} \neq 0$,

$$\frac{\|p(A)v_{\mathrm{u}}\|_2}{\|p(A)v_{\mathrm{w}}\|_2} \leq \eta_m \frac{\|v_{\mathrm{u}}\|_2}{\|v_{\mathrm{w}}\|_2}.$$

A value $\eta_m < 1$ guarantees relative suppression, and smaller is better per sweep. Both bounds scale by $|c|$ when p is replaced by cp for $c \neq 0$. Consequently we can choose a convenient normalization. A filter cannot create a wanted component absent from the start.

Exact selection at known points. Take $A = \mathrm{diag}(1, 2, 3)$ and $p(t) = (t-1)(t-3)$. Evaluating at the diagonal entries gives $p(A) = \mathrm{diag}(0, -1, 0)$, so $(x, y, z)^T$ maps to $(0, -y, 0)^T$. One application selects the middle eigenvalue whenever $y \neq 0$, although ordinary power iteration would favor 3 if $z \neq 0$. The minus sign preserves the selected line. This realizes the selection principle in §1.3.3 with $\eta = 0$. For a large unwanted cluster its individual eigenvalues are usually unknown. What can we guarantee if we only know an interval containing them?

11.3 A polynomial bounded on an entire interval

Define the Chebyshev polynomial of the first kind by $T_m(\cos\theta) = \cos(m\theta)$. As θ runs from 0 to π, $t = \cos\theta$ runs through $[-1, 1]$. The cosine addition identity

$$\cos((m+1)\theta) + \cos((m-1)\theta) = 2\cos\theta\cos(m\theta)$$

gives $T_0(t) = 1$, $T_1(t) = t$, and

$$\boxed{T_{m+1}(t) = 2tT_m(t) - T_{m-1}(t).}$$

This constructs a degree-m polynomial for each m and extends its definition to all real arguments. Substitution gives $T_2 = 2t^2 - 1$ and $T_3 = 2t(2t^2 - 1) - t = 4t^3 - 3t$. Only the two preceding values are needed to advance one degree.

Inside the interval. The cosine definition yields $|T_m(t)| \le 1$. For $m \ge 1$, at $t_j = \cos(j\pi/m)$, $j = 0, \ldots, m$, one has $T_m(t_j) = (-1)^j$. These $m + 1$ successive peaks attain the same absolute height with alternating signs. This equioscillation certifies optimality below. Increasing the degree adds oscillations while preserving the height bound.

Outside the interval. For $t > 1$, set $u = \operatorname{arcosh} t$. The hyperbolic cosine addition identity shows that $\cosh(mu)$ satisfies the same recurrence and initial values, giving

$$T_m(t) = \cosh(mu) = \tfrac{1}{2}(e^{mu} + e^{-mu}).$$

The recurrence also gives $T_m(-t) = (-1)^m T_m(t)$. Thus the magnitude grows exponentially in degree at either fixed exterior point, although growth is slow near an endpoint where u is small. Both exterior tails matter when deciding which eigenvalues the filter might admit.

11.3.1 Why this polynomial? A normalized minimax problem

Fix a real $t_* > 1$ and an integer $m \ge 1$. Among real polynomials r of degree at most m satisfying $r(t_*) = 1$, the polynomial

$$h(t) = \frac{T_m(t)}{T_m(t_*)}$$

minimizes $\max_{-1\le t\le 1} |r(t)|$. Its maximum there is $M = 1/T_m(t_*)$. The normalization expresses a design requirement: retain unit weight at one specified wanted point. Without it, the zero polynomial would trivially minimize the maximum.

Proof sketch. Suppose some admissible r had maximum strictly below M. At each alternating extremum t_j, $h(t_j) - r(t_j)$ has the sign of $h(t_j) = (-1)^j M$. The difference therefore has at least one zero in each of the m disjoint open gaps between adjacent extrema. It also vanishes at t_*, giving at least $m + 1$ distinct zeros. A nonzero polynomial of degree at most m

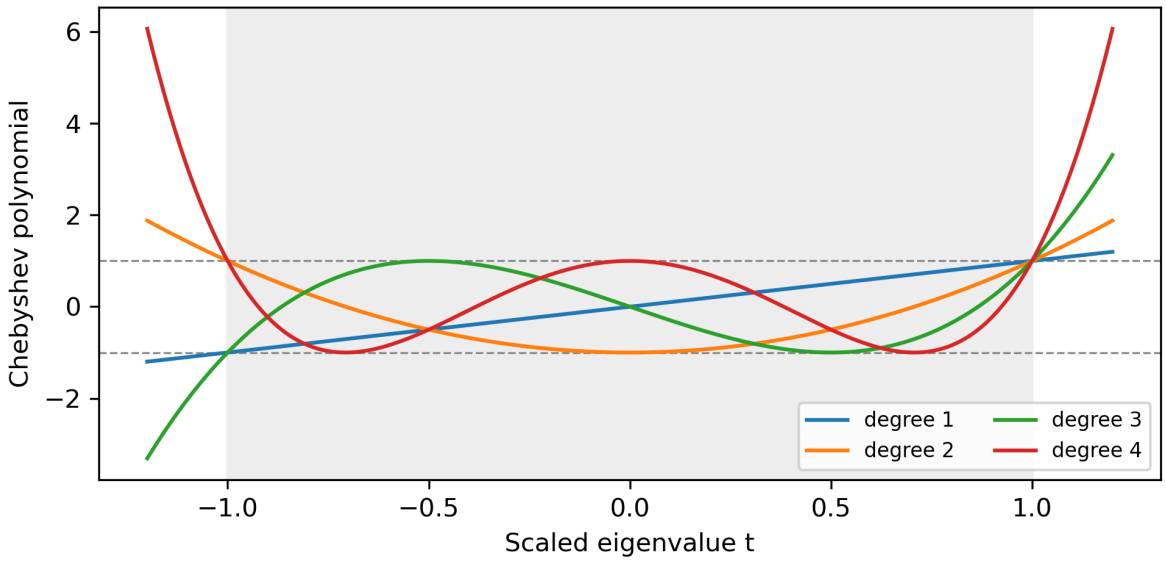


Figure 11.1: The shaded interval contains every oscillation between the two dashed levels. Beyond either endpoint the higher degrees separate rapidly.

cannot have this many, and $h - r$ is nonzero at the extrema. Hence no smaller maximum is possible.

The objective is a **uniform bound on one interval with one exterior value fixed**. It does not assert optimality for a known discrete spectrum, two disjoint unwanted intervals, or a prescribed passband shape. The exact three-point example illustrates how additional information can change the design.

11.4 From spectral bounds to a filter

Let $[a, b]$, $a < b$, enclose the unwanted spectrum. Subtract its midpoint $c = (a + b)/2$ and divide by its half-width $d = (b - a)/2$. Then

$$s(t) = \frac{t - c}{d} = \frac{2t - (a + b)}{b - a}, \qquad \widetilde{A} = \frac{A - cI}{d}.$$

Since $s(a) = -1$ and $s(b) = 1$, s maps the whole interval onto $[-1, 1]$. The scaled matrix has the same eigenvectors with eigenvalues $s(\lambda_i)$. Choose $p_m = T_m \circ s$, optionally dividing by $T_m(s(t_*))$ to give unit response at a wanted point t_*. For a wanted region $[t_*, \infty)$ with $t_* > b$, the weakest guaranteed wanted magnitude occurs at its near edge. Normalizing there gives a wanted magnitude of at least one throughout the region. The same magnitude argument works on the left by parity.

Bounds alone do not make an interior target exterior. If a wanted point lies inside the interval on which the filter has uniform bound M, its magnitude there is at most M too. Individual unwanted points could happen to be near roots, but the interval-wide design gives no such guarantee. Selecting an interior band with unwanted spectrum on both sides requires additional structure, such as approximating a passband on separated sets.

Worked design. Return to $A = \operatorname{diag}(1, 2, 3)$, now wanting 3 and enclosing the other two eigenvalues in $[1, 2]$. Here $s(t) = 2t - 3$, $s(3) = 3$ and $T_2(3) = 17$. Thus

$$p_2(t) = \frac{T_2(2t-3)}{17} = \frac{8t^2 - 24t + 17}{17}.$$

Eigenvalue	Scaled argument	$T_2(s(\lambda))$	Normalized weight
1	-1	1	1/17
2	1	1	1/17
3	3	17	1

For $v = (1, 2, 1)^T$, the result is $(1/17, 2/17, 1)^T$. The unwanted-to-wanted norm ratio falls from $\sqrt{5}$ to $\sqrt{5}/17$, exactly attaining the bound. Two ordinary products give $(1, 8, 9)^T$ with ratio $\sqrt{65}/9$. Both procedures use two products when evaluated on vectors. Their different polynomials explain the different outcomes.

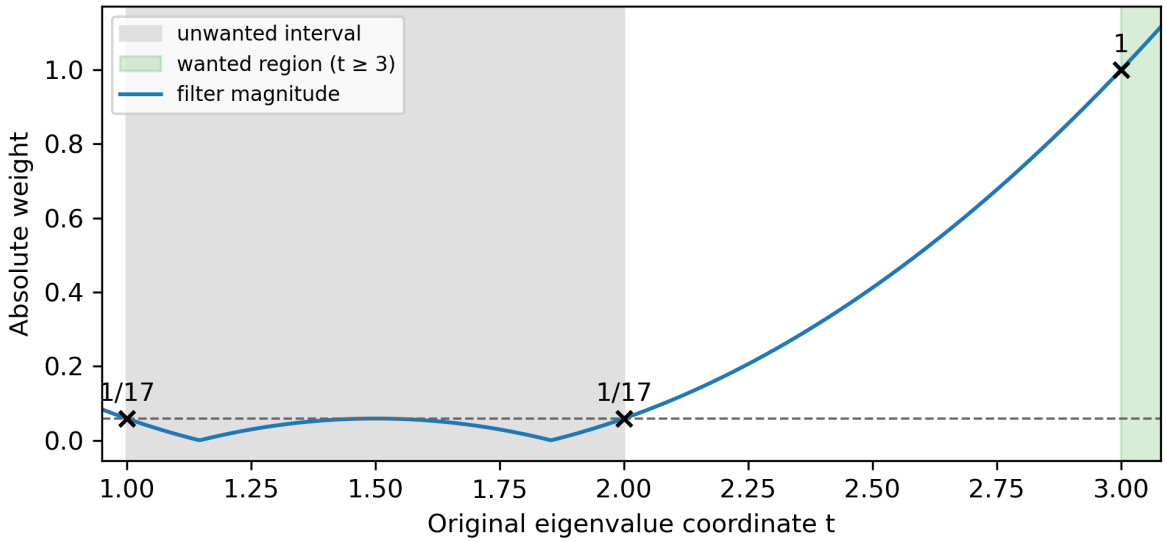


Figure 11.2: Normalized quadratic response. The crosses mark the three eigenvalues. The entire unwanted interval stays below the dashed level, even between the known eigenvalues.

Choosing the degree. If $s(t_*) = \gamma > 1$ is a lower bound for all wanted scaled eigenvalues, then $\eta_m \le 1/\cosh(m \operatorname{arcosh} \gamma)$. A sufficient degree for $\eta_m \le \varepsilon$, $0 < \varepsilon < 1$, is

$$m = \left\lceil \frac{\operatorname{arcosh}(1/\varepsilon)}{\operatorname{arcosh} \gamma} \right\rceil.$$

With $\gamma = 3$, degree two gives $1/17$ and degree three gives $1/99$. For a factor at most 0.01, three falls just short and four suffices ($T_4(3) = 577$). A narrower separation from the unwanted interval requires more work for the same guarantee.

11.5 Applying the filter with matrix products

Even for sparse A, its powers may fill in as longer paths connect more indices. Forming and storing them can cost far more than a sparse product. Expanding a high-degree filter in monomials can also introduce large coefficients and severe cancellation. Instead, implement $\widetilde{A}X = (AX - cX)/d$ for a vector or a thin block X. Set

$$W_0 = X, \qquad W_1 = \widetilde{A}X, \qquad W_{j+1} = 2\widetilde{A}W_j - W_{j-1}.$$

Induction. The initial blocks are $T_0(\widetilde{A})X$ and $T_1(\widetilde{A})X$. Substituting the corresponding formulas for W_j, W_{j-1} into the update yields $(2\widetilde{A}T_j(\widetilde{A}) - T_{j-1}(\widetilde{A}))X = T_{j+1}(\widetilde{A})X$. This works for every column simultaneously. Divide the final block by $T_m(s(t_*))$ for the normalized result.

In the worked example, $W_0 = (1, 2, 1)^T$, $W_1 = (-1, 2, 3)^T$ and $W_2 = 2\widetilde{A}W_1 - W_0 = (1, 2, 17)^T$. Dividing by 17 recovers the spectral calculation. Degree m requires m applications of A and a constant number of working blocks. For r columns and a sparse matrix the filtering work is $O(mr(\operatorname{nnz}(A) + n))$, with $O(nr)$ additional storage. Thin QR afterward costs $O(nr^2)$ for $n \ge r$.

11.5.1 What changes in floating point?

The identity $W_j = T_j(\widetilde{A})X$ is exact arithmetic. In computation each product and block update injects a perturbation that subsequent recurrence steps propagate. Exterior modes can grow until a raw block overflows, even if its final normalized value is near one. Very weak components can underflow or fall below roundoff relative to stronger wanted components. Dividing at the end cannot recover lost directions. Also, an unwanted eigenvalue just beyond an underestimated endpoint is in a growth region. An excessively wide interval can instead bring the nearest wanted point too close to the scaled endpoint.

One remedy for raw growth is to normalize **at each degree**. For $\gamma = s(t_*) > 1$, write $d_j = T_j(\gamma)$, $Z_j = W_j/d_j$, and $r_j = d_{j-1}/d_j$. The scalar recurrence gives

$$r_1 = 1/\gamma, \qquad r_{j+1} = 1/(2\gamma - r_j).$$

Dividing the block recurrence by d_{j+1} gives $Z_0 = X$, $Z_1 = \widetilde{A}X/\gamma$, and

$$Z_{j+1} = r_{j+1}(2\widetilde{A}Z_j - r_j Z_{j-1}).$$

The reference component keeps unit weight at every degree. Components farther outside can still grow. Check interval estimates, the numerical rank of the block before QR, and the precision of the product and recurrence if observed convergence departs from prediction. A different normalization or higher precision may be appropriate. No uniform floating-point stability guarantee follows from the scalar minimax result.

11.6 How much does a filtered sweep improve a subspace?

Let $U = [U_{\mathrm{w}}, U_{\mathrm{u}}]$ be a unitary eigenbasis of A and let Q_k be an orthonormal trial block with exactly as many columns r as the wanted dimension. Session 9 used $Y = AQ_k$. Replace this with $Y = p_m(A)Q_k$ and compute the thin QR factorization $Y = Q_{k+1}R_{k+1}$. Filtering changes the space. QR gives the same full-rank image an orthonormal basis, so weaker independent directions remain usable on the next sweep. QR cannot repair an already lost direction.

One-sweep angle bound. Assume $C = U_{\mathrm{w}}^* Q_k$ is invertible and the filter is nonzero on every wanted eigenvalue. Put $S = U_{\mathrm{u}}^* Q_k$. In eigenbasis coordinates the trial block has rows grouped as $[C^T, S^T]^T$. Right multiplication by C^{-1} gives the graph basis $\binom{I}{G}$, $G = SC^{-1}$. For a vector in that graph with coordinates (x, Gx), the tangent of its angle to the wanted space is $\|Gx\|_2/\|x\|_2$. Maximizing over $x \neq 0$ yields $\tan\theta_k = \|G\|_2$, where θ_k is the largest principal angle.

Write D_{w} and D_{u} for the diagonal filter weights in the two groups. After filtering, the coordinate blocks are $D_{\mathrm{w}}C$ and $D_{\mathrm{u}}S$. The upper block is still invertible. The new graph therefore satisfies

$$G_{\mathrm{new}} = D_{\mathrm{u}} G D_{\mathrm{w}}^{-1}, \qquad \boxed{\tan\theta_{k+1} \le \|D_{\mathrm{u}}\|_2 \, \|D_{\mathrm{w}}^{-1}\|_2 \tan\theta_k = \eta_m \tan\theta_k.}$$

QR changes only the coordinates within this graph. For a fixed filter with $\eta_m < 1$, the tangent decreases at least geometrically in exact arithmetic. Taking $p(t) = t$ and ordering by decreasing magnitude recovers $|\lambda_{r+1}|/|\lambda_r|$. With a designed filter the boundary ratio is measured in the *transformed* spectrum. A cluster inside the wanted space need not be resolved to approximate its span.

11.6.1 Experiment: a prediction at equal work

Use four eigenvalues $(4, 3, 2, 1)$ in a rotated orthonormal basis u_i. Target the first two. Start with the plane spanned by $u_1 + u_4$ and $u_2 + u_3$. Its wanted overlap is invertible and its largest initial tangent is one. Under ordinary iteration the two graph entries have magnitudes $(1/4)^k$

and $(2/3)^k$. Under the quadratic above they have magnitudes $(1/49)^k$ and $(1/17)^k$, since $T_2(s(4)) = T_2(5) = 49$. Thus the worse pair attains the bound for both methods.

For equal-dimensional spaces the projector distance is $\|Q_k Q_k^* - U_{\mathrm{w}} U_{\mathrm{w}}^*\|_2 = \sin\theta_k$. Convert the predicted tangent z to $z/\sqrt{1+z^2}$ **before** comparing with that distance. Explicit projectors are inexpensive in this small diagnostic.

```
def filtered_block(apply, X, a, b, target, degree):
    """Normalized Chebyshev action for a right exterior reference point."""
    center, halfwidth = (a + b) / 2, (b - a) / 2
    gamma = (target - center) / halfwidth

    def scaled(Y):
        return (apply(Y) - center * Y) / halfwidth

    if degree == 0:
        return X.copy()
    ratio = 1 / gamma
    previous, current = X, ratio * scaled(X)
    for j in range(1, degree):
        next_ratio = 1 / (2 * gamma - ratio)
        following = next_ratio * (2 * scaled(current) - ratio * previous)
        previous, current, ratio = current, following, next_ratio
    return current

def orth(X):
    return np.linalg.qr(X, mode="reduced")[0]

rng = np.random.default_rng(26)
U = orth(rng.normal(size=(4, 4)))
A = (U * np.array([4., 3., 2., 1.])) @ U.T
apply = lambda X: A @ X
Q0 = orth(U[:, :2] + U[:, [3, 2]])
Pstar = U[:, :2] @ U[:, :2].T

fig, ax = plt.subplots(figsize=(6.6, 3.5))
for name, cost, factor, step, color in [
    ("ordinary", 1, 2/3, apply, "tab:blue"),
    ("quadratic", 2, 1/17,
     lambda X: filtered_block(apply, X, 1, 2, 3, 2), "tab:orange"),
]:
    Q = Q0.copy()
    sweeps = np.arange(16 // cost + 1)
```

```
    errors = []
    for k in sweeps:
        if k:
            Q = orth(step(Q))
        errors.append(np.linalg.norm(Q @ Q.T - Pstar, ord=2))
    z = factor ** sweeps
    ax.semilogy(cost * sweeps, z / np.hypot(1, z), color=color,
                label=f"{name} prediction")
    ax.semilogy(cost * sweeps, errors, "o", ms=4, mfc="none",
                color=color, label=f"{name} measured")
    print(f"{name}, 4 block products: {errors[4 // cost]:.6g}")
ax.set(xlabel="Cumulative products with A (two-column block)",
       ylabel="Projector distance")
ax.grid(alpha=.25)
ax.legend(fontsize=8)
fig.tight_layout()
plt.show()
```

```
ordinary, 4 block products: 0.193786
quadratic, 4 block products: 0.00346019
```

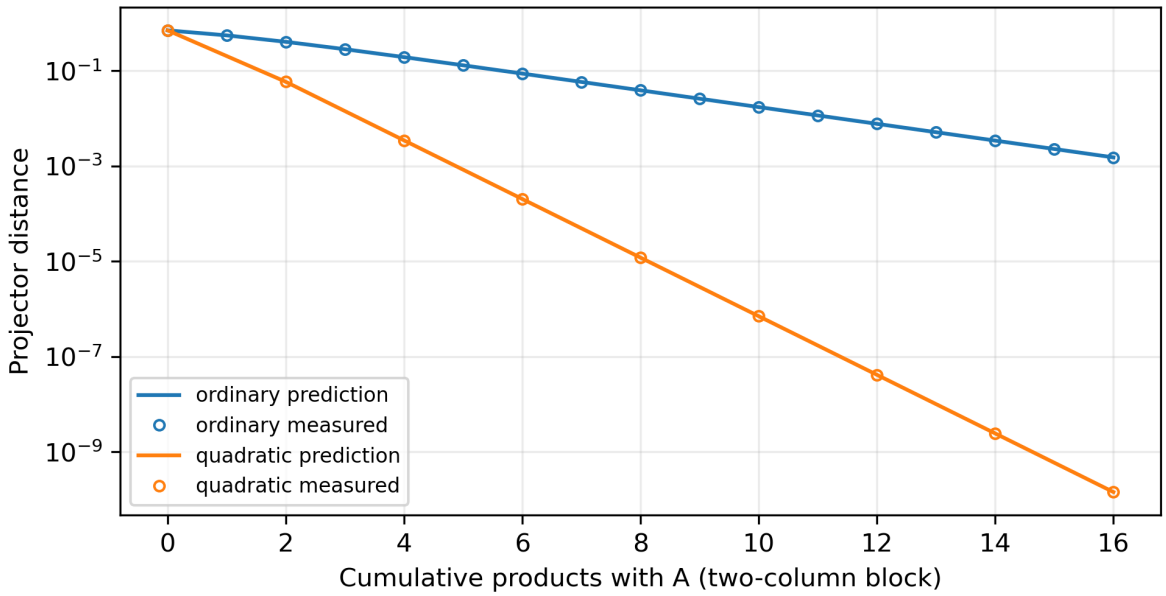


Figure 11.3: Projector distance measured after QR. Markers show the iterates and lines give the exact predictions. Each quadratic sweep uses two products with A.

At four products, ordinary iteration has completed four sweeps and has tangent $(2/3)^4 = 16/81$. The filter has completed two sweeps and has tangent $(1/17)^2 = 1/289$. The corresponding measured distances are about 0.194 and 0.00346. Per sweep, compare 1/17 with 2/3. Per two products, compare 1/17 with $(2/3)^2$. Both comparisons favor the quadratic **for this spectrum and start**. Equal product counts still omit QR costs, block updates and memory traffic. With a different start the bound can be strict, and at rounding scale the computed projector distance need no longer track the predicted curve.

11.7 How the seminar algorithm fits together

Stage	Mathematical decision	Computational consequence
1. Spectral information	Enclose the unwanted eigenvalues and identify an exterior wanted edge.	A missed unwanted outlier may receive a large weight. A conservative bound can require a higher degree.
2. Polynomial design	Scale the unwanted interval, fix a reference weight, and choose a degree from the required separation.	The minimax bound converts interval information into a worst-case factor per sweep.
3. Numerical application	Evaluate the scalar design on a block through the matrix recurrence.	Each degree adds one block product. Scaling and precision determine the range of intermediate values that can be represented reliably.
4. Subspace update	Compute a thin QR of the filtered block.	The orthonormal basis can be filtered again or used to extract approximate eigenpairs of the original matrix.

Session 11 develops Rayleigh–Ritz extraction using Q^*AQ. Keeping A here matters: the wanted eigenvalues are those of the original operator, and the polynomial may map distinct eigenvalues to the same value.

11.8 Central Synthesis — From Polynomial Approximation to Spectral Subspace Selection

The identity $p(A)q_i = p(\lambda_i)q_i$ turns a question about an n-dimensional operator into a scalar design decision. To keep a trial space close to the wanted eigenspace, control the **weakest wanted** weight as well as the **strongest unwanted** weight. Their ratio bounds precisely the part of the trial space that tilts away from the target.

Interval information turns the design decision into an approximation problem. After an affine change of variable, equioscillation certifies the smallest possible worst-case weight on the unwanted interval for a fixed exterior response. The same Chebyshev polynomial has a three-term recurrence, so a matrix-product routine can realize this design one degree at a time. Finally, QR preserves the image for the next sweep. On the worked interval $[1, 2]$, these steps turn the scalar number $T_2(3) = 17$ into a factor $1/17$ in a subspace-angle estimate, at the price of two block products.

11.9 Independent practice (30 minutes, including checks)

1. **Redirect dominance (4 minutes).** For $A = \mathrm{diag}(-5, 1, 2)$, design a degree-two filter with unit weight at 1 and zero weights at the other two eigenvalues. Apply it to $(1, 2, 3)^T$. Which line would ordinary power iteration select?
2. **Design from bounds (6 minutes).** The unwanted spectrum lies in $[2, 6]$ and every wanted eigenvalue is at least 8. Derive the normalized quadratic, its guaranteed ratio, and the smallest degree certifying a factor at most 0.02.
3. **Run the recurrence by hand (5 minutes).** With the bounds in exercise 2, use $A = \mathrm{diag}(2, 4, 8)$ and $v = (1, 1, 1)^T$. Compute the first two raw recurrence steps and normalize. Why does one coefficient change sign?
4. **Diagnose a failed design (4 minutes).** Use the chapter's $[1, 2]$ filter with wanted point 3. An unwanted eigenvalue at 0 was missed. Predict its degree-two weight. Would increasing the degree restore separation?
5. **Predict a block sweep (6 minutes).** In the experiment replace the second starting vector by $u_2 + 2u_3$. Predict the filtered projector distance after two block products. Then replace that vector by u_3 alone. Which hypothesis fails? Can QR fix it?
6. **Mastery check (5 minutes).** Without formulas in front of you, explain how to turn an interval and a wanted edge into a factor, a block recurrence and a reliable trial basis. Identify one limitation of each of the spectral design and its numerical evaluation.

11.9.1 Solution checks

1. $p(t) = -(t + 5)(t - 2)/6$ has the prescribed values. The image is $(0, 2, 0)^T$. Ordinary iteration selects the line of the first coordinate since $|-5|$ is largest and its initial

coefficient is nonzero.

2. $s(t) = (t-4)/2$, $s(8) = 2$ and $p_2(t) = (2s(t)^2 - 1)/7$. The ratio is at most $1/7$. The threshold requires $T_m(2) \geq 50$. Successive values at degrees $2, 3, 4$ are $7, 26, 97$, so degree four suffices and degree three falls short of the interval guarantee.
3. The scaled eigenvalues are $-1, 0, 2$. Thus $W_1 = (-1, 0, 2)^T$, $W_2 = (1, -1, 7)^T$ and the answer is $(1/7, -1/7, 1)^T$. At the midpoint the response is $T_2(0) = -1$ before normalization. A negative weight reverses the coefficient's sign while its magnitude still obeys the interval bound.
4. $s(0) = -3$ and $s(3) = 3$. By parity their magnitudes agree at **every** degree. The normalized quadratic gives unit weight to both. Redesign the unwanted set description before choosing a degree.
5. After one quadratic sweep the two graph entries have magnitudes $1/49$ and $2/17$. Thus $\tan\theta = 2/17$ and the distance is $2/\sqrt{293}$. With u_3 alone, the wanted overlap has rank one. Neither polynomial weighting nor a change of basis by QR introduces the missing u_2 component.
6. Map the unwanted interval onto $[-1, 1]$ and normalize T_m at the scaled wanted edge. Bound the unwanted-to-wanted ratio and choose m to meet the tolerance. Apply the scaled operator to successive blocks using the recurrence and take thin QR of the result. A missed exterior outlier can defeat an exact filter, and lost numerical rank can defeat an accurately chosen interval. These require different remedies.

12 Session 11 — Rayleigh–Ritz and Integration

Session 9 produced a plane whose images under repeated matrix multiplication approached a wanted invariant plane. Session 10 replaced multiplication by A with a polynomial $p_m(A)$ that selects spectral components deliberately. Suppose the resulting plane is already very accurate. Which way should we point **within** that plane to obtain an eigenvector? Even an exact invariant plane admits infinitely many orthonormal bases, most of which mix its eigenvectors.

Study route (75 minutes). Allow 15 minutes for compression and the Rayleigh quotient, 17 for residuals and the three worked examples, and 15 for the algorithm and synthesis. Reserve 28 minutes for the five exercises. The computational laboratory and generalized problem are supplementary. Throughout, $A = A^* \in \mathbb{F}^{n\times n}$, where $\mathbb{F}$ is $\mathbb{R}$ or $\mathbb{C}$, and the star denotes conjugate transpose. Vector norms are Euclidean and matrix norms are induced 2-norms unless indicated otherwise.

12.1 What equation can a trial space satisfy?

Let $\mathcal{U}$ have dimension p and let $Q \in \mathbb{F}^{n\times p}$ contain an orthonormal basis. Every candidate in $\mathcal{U}$ has the unique form $u = Qz$, where $z = Q^*u$. Session 1 reduces the unknowns from n coordinates to p. Session 2 supplies the isometry $\|Qz\|_2 = \|z\|_2$, so coordinate errors and physical vector errors have the same length.

Requiring $AQz = \theta Qz$ would require $\mathcal{U}$ to contain an exact eigenvector. A generic line in the plane already shows the difficulty: its image need not lie along it. Ask instead whether the equation holds when tested against every direction we can represent. For $z \neq 0$ define $r = AQz - \theta Qz$. Then

$$\begin{aligned} r \perp \mathcal{U} &\iff (Qw)^* r = 0 \quad \text{for every } w \in \mathbb{F}^p \\ &\iff Q^* r = 0 \\ &\iff Q^*AQz = \theta Q^*Qz = \theta z. \end{aligned}$$

> **Note**
>
> **Rayleigh–Ritz extraction.** Form $H = Q^*AQ$, solve $Hz_i = \theta_i z_i$ with $\|z_i\|_2 = 1$, and lift $u_i = Qz_i$. The **Ritz value** is θ_i and the **Ritz vector** is u_i. The pair is defined relative to $\mathcal{U}$. Its residual in the original problem is $r_i = Au_i - \theta_i u_i$.

The dimensions follow the calculation: AQ is $n \times p$ and $Q^*(AQ)$ is $p \times p$. Since $H^* = Q^*A^*Q = H$, the spectral theorem gives $HZ = Z\Theta$ with a unitary Z and real diagonal Θ. The lifted matrix $U = QZ$ satisfies $U^*U = I_p$. Thus extraction rotates the coordinates within the same space. If a wanted direction is orthogonal to $\mathcal{U}$, none of these combinations can reach it.

12.1.1 What has been projected?

With $P = QQ^*$, Session 3 gives the orthogonal decomposition

$$A(Qz) = \underbrace{QHz}_{PA(Qz)} + \underbrace{(I-P)A(Qz)}_{\perp \mathcal{U}}.$$

Think of Au as a vector and PAu as its shadow in the trial plane. A Ritz direction u is one whose shadow after applying A is exactly θu. The remaining perpendicular component is then precisely its residual. The map $PA|_{\mathcal{U}} \colon \mathcal{U} \to \mathcal{U}$ is the **compression**. Its coordinate matrix is H. On the whole ambient space $PAP = QHQ^*$ also annihilates $\mathcal{U}^\perp$; this accounts for the additional zero directions in that representation.

A change of orthonormal basis $Q' = QS$, with S unitary, gives $H' = S^*HS$. The coordinates $z' = S^*z$ lift to $Q'z' = Qz$. Ritz values and lifted Ritz eigenspaces depend only on the plane. Individual unit vectors retain a sign or complex phase choice, and a repeated Ritz value permits any orthonormal basis of its lifted eigenspace.

12.1.2 A variational interpretation

For $v \neq 0$ write $\rho_A(v) = v^*Av/(v^*v)$. Since $\rho_A(Qz) = \rho_H(z)$, each unit Ritz vector satisfies $\theta_i = u_i^*Au_i$. Write $z = Zc$ with $\|c\|_2 = 1$. Then

$$\rho_A(Qz) = \sum_{i=1}^{p} |c_i|^2 \theta_i.$$

This weighted average attains its smallest and largest values at eigenvectors of H. Consequently the extremal Ritz values are the minimum and maximum of ρ_A over the trial space. Restriction gives $\lambda_{\min}(A) \leq \theta_{\min} \leq \theta_{\max} \leq \lambda_{\max}(A)$. Enlarging the space can only decrease its minimum and increase its maximum. Interior Ritz directions are stationary directions: for a unit u and

a variation $w \in \mathcal{U}$, the derivative at $t = 0$ of $\rho_A(u + tw)$ for real t is $2\,\mathrm{Re}(w^*(Au - \rho_A(u)u))$. Vanishing for every w is exactly the Galerkin condition (in complex space one may also test iw). This specifies the sense of optimality. Minimizing the ambient residual over all unit vectors in the space would be a different optimization problem.

12.2 How much of the action escapes the space?

Definition. The block invariance residual is $R = AQ - QH$. Substitution of H gives

$$R = (I - P)AQ, \qquad Q^*R = Q^*AQ - H = 0.$$

Its norm measures the largest perpendicular component produced by applying A to any unit vector in $\mathcal{U}$. Hence $R = 0$ exactly when $A\mathcal{U} \subseteq \mathcal{U}$. With small $\|R\|_2$ relative to the scale of A, we call the space approximately invariant in this residual sense. Identifying it with a particular *wanted* space requires further evidence.

For an eigenvector z_i of H,

$$r_i = AQz_i - QHz_i = Rz_i, \qquad Q^*r_i = 0, \qquad \|r_i\|_2 \le \|R\|_2.$$

Thus exact invariance gives exact eigenpairs throughout the lifted basis. One exact pair only requires $Rz_i = 0$. For the full Ritz basis $AU - U\Theta = RZ$, so its matrix 2-norm is $\|R\|_2$ and the sum of squared column residual norms is $\|R\|_{\mathrm{F}}^2$. Orthogonality holds even when the residual is large. It verifies extraction algebraically; its **magnitude** measures the remaining defect.

12.2.1 Worked example 1 — Extracting two directions from the Session 9 plane

Take $A = \mathrm{diag}(6, 3, 1)$. One ordinary subspace sweep starting at $\mathrm{span}\{e_1, e_2 + e_3\}$ gives the plane spanned by $v = e_1$ and $w = (3e_2 + e_3)/\sqrt{10}$. To see extraction do real work, represent the same plane by $Q = [(v + w)/\sqrt{2}, (v - w)/\sqrt{2}]$. Its columns are orthonormal because v, w are orthonormal. The plane is proper and $Aw = (9e_2 + e_3)/\sqrt{10}$ leaves it.

Since $v^*Av = 6$, $w^*Aw = 28/10$ and $v^*Aw = 0$, direct multiplication gives

$$H = \frac{1}{5}\begin{pmatrix} 22 & 8 \\ 8 & 22 \end{pmatrix}.$$

Adding and subtracting columns identifies unit eigenvectors $z_+ = (1, 1)^T/\sqrt{2}$ and $z_- = (1, -1)^T/\sqrt{2}$. Their eigenvalues are 6 and 14/5. Lifting gives $Qz_+ = v$ and $Qz_- = w$. For the first pair $r_+ = 0$. For the second,

$$r_- = \frac{3e_2 - 9e_3}{5\sqrt{10}}, \qquad v^* r_- = 0, \qquad w^* r_- = \frac{9-9}{50} = 0, \qquad \|r_-\|_2 = \frac{3}{5}.$$

Both columns of Q are therefore orthogonal to r_-. The wanted eigenvalues are 6 and 3. Extraction recovers 6 exactly because e_1 belongs to the plane. The second value is low by $1/5$ because w still has weight $1/10$ on the eigenvalue 1.

12.2.2 Worked example 2 — Tilting an exact invariant plane

Keep e_1 and replace w by $w_\phi = \cos\phi\, e_2 + \sin\phi\, e_3$ for $0 \le \phi \le \pi/2$. The reduced matrix in this basis is diagonal. Its second Ritz value and residual are

$$\theta = 3 - 2\sin^2\phi, \qquad r = -2\sin\phi\cos\phi\,(-\sin\phi\, e_2 + \cos\phi\, e_3).$$

The projector distance to the wanted plane is $\sin\phi$: on the last two coordinates the difference of the projectors has trace zero and determinant $-\sin^2\phi$, hence eigenvalues $\pm\sin\phi$. At $\phi = 0$ both residuals vanish. For a small tilt the second residual is first order in the angle, while the Ritz value error is second order. At $\phi = \pi/2$ both residuals again vanish, but now the second value is 1. Exact invariance can describe the wrong plane.

12.3 What does a small residual certify?

Eigenvalue inclusion. For Hermitian A, any real θ and any unit u,

$$\boxed{\operatorname{dist}(\theta, \sigma(A)) \le \|Au - \theta u\|_2.}$$

Proof. Expand $u = \sum_j c_j v_j$ in an orthonormal eigenbasis of A. Then

$$\|r\|_2^2 = \sum_j |c_j|^2 |\lambda_j - \theta|^2 \ge \min_j |\lambda_j - \theta|^2 \sum_j |c_j|^2.$$

The final sum is one. Thus the real interval $[\theta - \|r\|_2, \theta + \|r\|_2]$ contains at least one eigenvalue. In Example 1 the interval $[2.2, 3.4]$ isolates 3 if the other spectral locations are known. Overlapping inclusion intervals alone cannot certify distinct eigenvalues or their multiplicities.

Direction estimate. Let $\mathcal{S}$ be a chosen spectral subspace spanned by a set of eigenvectors indexed by J, and let $P_{\mathcal{S}}$ be its orthogonal projector. Suppose the complementary spectrum is separated from θ by $\delta = \min_{j \notin J} |\lambda_j - \theta| > 0$. Retaining the complementary terms in the same sum gives

$$\sin \angle(u, \mathcal{S}) = \|(I - P_{\mathcal{S}})u\|_2 = \left(\sum_{j \notin J} |c_j|^2 \right)^{1/2} \leq \frac{\|r\|_2}{\delta}.$$

For a simple eigenvalue this bounds the angle to its eigenline. For a cluster it controls leakage outside the entire cluster. Computable certification needs an independently justified lower bound on δ. A gap between two computed Ritz values can overlook a missing competitor.

12.3.1 Worked example 3 — A very small residual at a fixed angle

Let $A = \mathrm{diag}(1, 1+\varepsilon, 4)$ with $0 < \varepsilon < 1$. The line spanned by $u = (e_1 + e_2)/\sqrt{2}$ gives $\theta = 1 + \varepsilon/2$ and $r = \varepsilon(-e_1 + e_2)/(2\sqrt{2})$. As ε tends to zero the residual norm $\varepsilon/2$ tends to zero while the angle to each of the first two eigenlines stays 45°. For either individual eigenline the competitor is only $\varepsilon/2$ away from θ. For the two-dimensional cluster the outside separation is $3 - \varepsilon/2$, and u already belongs to that cluster exactly. At $\varepsilon = 0$ every direction in the plane is an eigenvector. This is the Session 6 residual-versus-forward-error distinction in spectral coordinates.

12.4 A complete filtered eigensolver

Consider an upper-edge target of d eigenpairs. Enclose the unwanted spectrum in $[a, b]$, $a < b$, with the wanted eigenvalues above b. Set $s(t) = (2t - a - b)/(b - a)$ and $p_m(t) = T_m(s(t))$. On the unwanted interval $|p_m| \leq 1$; above it $T_m(s)$ grows with $s > 1$ for $m \geq 1$. A common nonzero normalization leaves the filtered range unchanged. More general target regions require a filter with the corresponding passband.

For block width $p = d$, let V_1, V_2 contain orthonormal wanted and unwanted eigenvectors. If $V_1^* Q$ is invertible, the columns of Q describe a graph over the wanted space with matrix $G = (V_2^* Q)(V_1^* Q)^{-1}$. In spectral coordinates one filter step changes G to $D_2 G D_1^{-1}$, where D_j contains the filter values on group j. Thus the graph norm contracts by at most $\max_{\text{unwanted}} |p_m| / \min_{\text{wanted}} |p_m|$ if this ratio is below one. QR preserves the image of a full-rank filtered block. Extraction then sees a better plane. Neither QR nor the reduced solve supplies missing initial overlap.

```
Choose target, target count d, block width p >= d, interval, degree m >= 1.
Choose a full-rank starting block with sufficient wanted overlap.
Q = thin orthonormal basis of the starting block
For each sweep up to the iteration limit:
    W0 = Q
    W1 = s(A) Q                        # affine action on a block
```

```
    For j = 1, ..., m-1:
        W0, W1 = W1, 2 s(A) W1 - W0
    Check numerical rank of W1 and compute its thin QR basis Q.
    C = A Q
    H = Q* C
    Solve H Z = Z Theta with a Hermitian eigensolver.
    U = Q Z
    Select d columns using the original Ritz values and target region.
    For each selected i compute r_i = C z_i - theta_i u_i.
    If count, region and residual tests pass, return selected pairs.
    Continue with Q or with all columns of U.
Report unconverged pairs if the sweep limit is reached.
```

A full $U = QZ$ yields the same next filtered range as Q, because Z is unitary. Keeping only a subset changes the next search space. With $p > d$, retaining extra columns can help resolve a boundary cluster. A target that cuts a repeated eigenvalue has no uniquely distinguished partial eigenspace within that multiplicity.

12.4.1 Decisions that affect the result

- **Block quality.** Check $\|Q^*Q - I\|_2$ after thin Householder QR. Examine the singular values of the filtered block or use rank-revealing QR when needed. If a direction has fallen below numerical resolution, reduce the degree or replenish the missing columns and orthogonalize again. A full-sized QR output alone does not establish numerical rank.
- **Bounds and degree.** A missed unwanted eigenvalue outside $[a, b]$ can be amplified. An overly wide interval weakens separation or includes a wanted value. A higher degree usually improves the edge response at greater cost and can cause overflow or loss of weaker directions. Session 10's scaled recurrence controls intermediate growth. A fixed design uses the same interval and degree each sweep. An adaptive design updates them using spectral estimates and must recheck its coverage.
- **Reduced solve.** Compute $C = AQ$ once. Form $H = Q^*C$ and check its Hermitian defect relative to scale. Roundoff-sized skew can be removed with $(H + H^*)/2$. A large defect calls for checking the matrix action or conjugation. Use a Hermitian eigensolver and test the lifted pairs against the original A.
- **Stopping and cost.** For unit u_i, one possible test is $\|r_i\|_2 \le \tau_{\text{abs}} + \tau_{\text{rel}}(M + |\theta_i|)$ with M a documented scale for $\|A\|_2$ and tolerances chosen for the application. Require all requested pairs in the target region and check orthogonality. Small changes in θ_i can precede accurate directions. At finite precision residual evaluation itself incurs errors of order machine precision times a problem-dependent scale. Each degree-m sweep needs m matrix-block products for filtering and one for C. Forming $CZ - U\Theta$ reuses C; an

independent fresh check with AU costs an additional product. For $p \ll n$ the dense eigensolve costs $O(p^3)$, while QR and compression each cost $O(np^2)$.

12.5 Connections to the lecture and seminar

In Galerkin projection for $Ax = b$, seeking $x = x_0 + Qz$ and testing $b - Ax$ against Q gives $Q^*AQz = Q^*(b - Ax_0)$. For the eigenproblem the same test gives $Hz = \theta z$: the unknown right-hand side depends on the candidate itself. Both use all representable test directions to replace an ambient equation by a coordinate equation.

Arnoldi builds an orthonormal basis of $\operatorname{span}\{v, Av, \dots, A^{k-1}v\}$ up to breakdown. Its compression is upper Hessenberg; for Hermitian A, Lanczos uses the resulting tridiagonal structure in exact arithmetic. These methods retain intermediate Krylov directions. A filtered sweep combines powers via a polynomial and retains a block of fixed width. Both constructions can pass their trial basis to the same Rayleigh–Ritz extraction.

Reading connection. In Kanschat and Scheichl's lecture notes, Chapter 2 supplies orthogonalization, Chapter 3 supplies vector and subspace iteration, and §5.1 (especially Algorithms 5.1.2 and 5.1.10) connects extraction to projected iteration for large sparse problems. In the seminar, sparse matrix-block multiplication implements the polynomial action while the dense spectral calculation takes place on the small compression.

12.5.1 Optional extension — The mass matrix changes the geometry

Finite-element discretization of a vibration problem yields stiffness and mass matrices $A = A^*$ and $B = B^* > 0$, with $Av = \lambda Bv$. The discrete inner product is $\langle x, y \rangle_B = x^*By$. Testing the equation against a full-rank trial basis gives

$$Q^*AQz = \theta\, Q^*BQz.$$

A B-orthonormal basis has $Q^*BQ = I$ and reduces this to an ordinary Hermitian problem. Otherwise the reduced mass matrix is positive definite and one solves a small generalized Hermitian problem. Specifically, $B^{-1}Au - \theta u$ is B-orthogonal to the trial space, equivalently $Q^*(Au - \theta Bu) = 0$. This distinguishes the operator residual from the residual written in stiffness–mass form.

12.6 Central Synthesis — From Linear Algebra Foundations to a Complete Filtered Eigensolver

Represent → Transform → Filter → Orthonormalize → Project → Extract → Verify. The first decision is what information a thin block can carry. A p-column basis rep-

resents infinitely many vectors but only p independent directions. The spectral theorem lets us predict how an operator changes every one of those directions using scalar weights. Polynomial approximation then designs those weights using interval information, without needing the eigenvectors in advance.

Operation	Foundation in Review I	What passes to the next stage
Represent	Bases and inner products (1–2)	A block and coordinates with measurable lengths and angles
Transform	Spectral theorem and repeated iteration (5, 8–9)	A prediction for relative growth of spectral components
Filter	Matrix polynomials and recurrence (10)	A new range with increased relative wanted weight
Orthonormalize	QR and direct methods (4, 7)	A numerically usable basis of that range
Project	Orthogonal projection (3)	The small Hermitian coordinate matrix Q^*AQ
Extract	Hermitian spectral theory and Rayleigh–Ritz (5, 11)	Directions in the plane with $PAu_i = \theta_i u_i$
Verify	Norms, conditioning and spectral gaps (6, 9, 11)	Defects for the original problem and gap-dependent direction estimates

Two gaps answer different questions. The **ratio of filter responses** governs how much unwanted weight survives the next sweep. The **distance from a Ritz value to competing original eigenvalues** governs how much directional accuracy follows from a measured residual. A filter can accelerate subspace convergence while leaving a pair of close original eigenvalues just as difficult to distinguish. When reading the seminar paper, identify the operator in each estimate: $p_m(A)$ controls the update, Q^*AQ determines the coordinates, and A determines the final residual.

12.7 Optional computational laboratory — Follow a plane through extraction

Construct $A = V\Lambda V^T$ using the discrete sine basis $V_{ij} = \sqrt{2/(n+1)}\sin(ij\pi/(n+1))$, $1 \le i, j \le n$. The wanted eigenvalues are $4, 5$; all others lie in $[0, 3]$. In this example $s(t) = 2t/3-1$. With degree three the weaker wanted response is $T_3(5/3) = 365/27$ and the unwanted magnitudes

are at most one. Thus the graph norm contracts by at most 27/365 per sweep. The initial block has the identity as its wanted coordinate block before QR, ensuring full overlap.

The exact eigenvectors serve only to construct the test and evaluate diagnostics. The iteration uses products with A, a supplied interval and a residual test. The initial extraction is recorded at sweep zero. Each column of the residual block measures one pair, while the matrix norm of $AQ - QH$ measures the whole plane.

```
import numpy as np
import matplotlib.pyplot as plt

n, width, degree = 12, 2, 3
indices = np.arange(1, n + 1)
V = np.sqrt(2 / (n + 1)) * np.sin(
    np.pi * np.outer(indices, indices) / (n + 1)
)
spectrum = np.r_[np.linspace(0, 3, n - width), 4., 5.]
A = (V * spectrum) @ V.T
target = V[:, -width:]
P_target = target @ target.T
rng = np.random.default_rng(11)
start = target + V[:, :-width] @ rng.normal(size=(n - width, width))
eye = np.eye(width)
assert A.shape == (n, n) and start.shape == (n, width)
assert np.allclose(V.T @ V, np.eye(n))
assert np.allclose(A @ V, V * spectrum)

def basis(block):
    assert block.shape == (n, width)
    assert np.linalg.matrix_rank(block) == width
    Q, _ = np.linalg.qr(block, mode="reduced")
    assert np.linalg.norm(Q.T @ Q - eye, 2) < 1e-12
    return Q

def filtered(block, m):
    def scaled(X):
        return (2 / 3) * (A @ X) - X

    assert m >= 1
    previous, current = block, scaled(block)
    for j in range(1, m):
        previous, current = current, 2 * scaled(current) - previous
    return current
```

```
def extract(Q):
    C = A @ Q
    H = Q.T @ C
    assert C.shape == Q.shape and H.shape == (width, width)
    assert np.allclose(H, H.T, atol=1e-13, rtol=0)
    theta, Z = np.linalg.eigh((H + H.T) / 2)
    U = Q @ Z
    residuals = C @ Z - U * theta
    block_residual = C - Q @ H
    assert Z.shape == H.shape and U.shape == Q.shape
    assert theta.shape == (width,) and residuals.shape == Q.shape
    assert np.allclose(U.T @ U, eye)
    assert np.linalg.norm(Q.T @ residuals, 2) < 1e-13
    assert np.allclose(residuals, A @ U - U * theta, atol=1e-13)
    assert np.allclose(residuals, block_residual @ Z, atol=1e-13)
    return theta, U, np.linalg.norm(residuals, axis=0), block_residual

Q = basis(start)
values, defects, distances, block_norms = [], [], [], []
absolute_tol, relative_tol, scale = 1e-13, 1e-10, 5.
for sweep in range(31):
    theta, U, norms, R = extract(Q)
    values.append(theta)
    defects.append(norms)
    distances.append(np.linalg.norm(Q @ Q.T - P_target, 2))
    block_norms.append(np.linalg.norm(R, 2))
    nearest = np.min(abs(theta[:, None] - spectrum), axis=1)
    assert np.all(nearest <= norms + 1e-13)
    thresholds = absolute_tol + relative_tol * (scale + abs(theta))
    if np.all(theta > 3) and np.all(norms <= thresholds):
        break
    if sweep < 30:
        Q = basis(filtered(U, degree))
else:
    raise RuntimeError("Residual criterion missed within 30 sweeps")

assert np.allclose(theta, [4., 5.], atol=1e-10, rtol=0)
assert distances[-1] < 1e-8
print(f"Stopped after {sweep} sweeps")
for value, defect in zip(theta, norms):
    print(f"Ritz value {value:.10f}, residual {defect:.2e}")
print(f"Projector distance {distances[-1]:.2e}")
```

```
Stopped after 9 sweeps
Ritz value 4.0000000000, residual 7.52e-11
Ritz value 5.0000000000, residual 1.24e-14
Projector distance 3.20e-11
```

Read the panels together. When a Ritz value already looks flat at the plotted scale, its residual can still fall by orders of magnitude. The upper bound from the graph analysis concerns tilt; the residual of a particular pair can decay faster when its wanted component has a larger filter response. Only in a benchmark with a known target can we plot the actual projector distance directly. The independent AU product in the code is a validation cost beyond the products needed for extraction.

12.8 Independent exercises (28 minutes)

1. **Test space and coordinates (3 min).** Let X have independent, possibly nonorthogonal columns. Starting from residual orthogonality, derive the reduced problem. Show how a thin QR turns it into an ordinary Hermitian problem.
2. **A full extraction (6 min).** For $A = \begin{pmatrix} 3 & 1 & 1 \\ 1 & 3 & 0 \\ 1 & 0 & 2 \end{pmatrix}$ use $Q = [e_1, e_2]$. Find both Ritz values and lifted vectors, compute both residuals and verify orthogonality. Compute $\|R\|_2$ and explain its relation to the two residual norms.
3. **Invariance and a converse (4 min).** Prove that vanishing residuals for a full orthonormal Ritz basis imply invariance. Give an example with just one vanishing Ritz residual and nonzero block residual.
4. **Interpreting a certificate (5 min).** A Hermitian matrix has exactly one eigenvalue in $[2, 2.2]$, it is simple, and all remaining eigenvalues lie in $[4, 8]$. A unit vector has $\theta = 2.1$ and $\|r\|_2 = 0.03$. Identify the certified interval and bound the angle to the corresponding eigenline. What would survive if there were two eigenvalues in the first interval?
5. **Capstone — A filtered sweep by hand (10 min).** Let $A = \mathrm{diag}(1, 2, 3)$, target the two eigenvalues above 1.5, and start with the columns e_3 and $e_1 + e_2$. Use the degree-one Chebyshev filter $T_1(2t - 1)$ for the unwanted interval $[0, 1]$. Predict its spectral effect, filter the columns, compute thin QR, form H, extract both pairs, and check the original residuals and their orthogonality. With an absolute residual tolerance of 0.05, decide which pairs pass. Predict what one further sweep changes. Explain how to turn this calculation into a stopping loop.

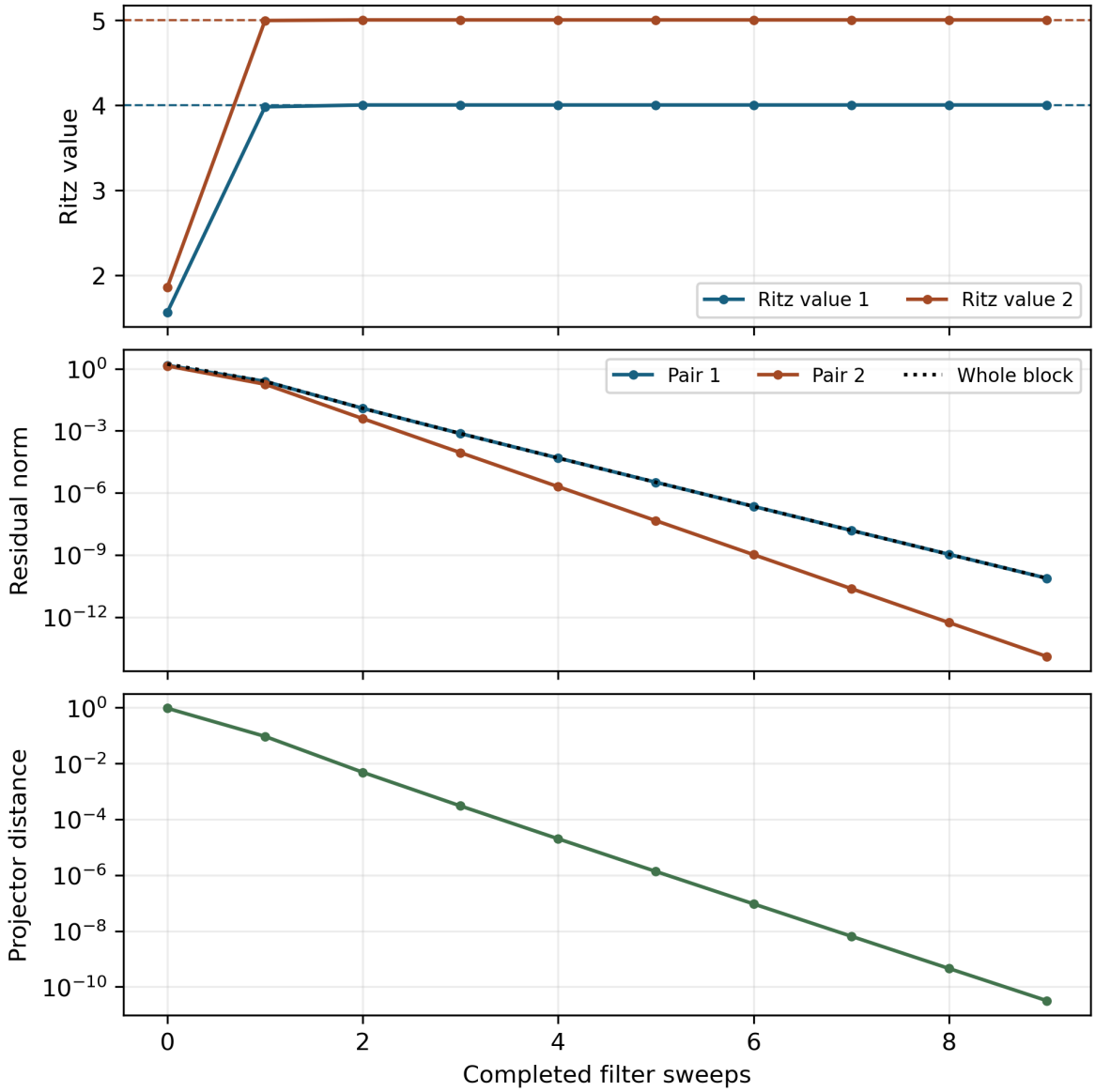


Figure 12.1: Upper-edge extraction by sweep. Dashed lines in the top panel are the exact wanted eigenvalues. The lower panels separate defects in the original equation from distance to the known target plane.

12.9 Worked solutions

1. Test space and coordinates. For $u = Xy$, orthogonality gives $0 = X^*(AXy - \theta Xy)$, hence $(X^*AX)y = \theta(X^*X)y$. The Gram matrix is positive definite since $y^*X^*Xy = \|Xy\|^2 > 0$ for $y \neq 0$. Write $X = QS$ with invertible triangular S. Substitution gives $S^*(Q^*AQ)Sy = \theta S^*Sy$. Cancel S^* and put $z = Sy$ to obtain $Hz = \theta z$ and $u = Qz$. Implementing extraction with Q avoids explicitly forming or inverting the Gram matrix.

2. A full extraction. The leading 2×2 block is $H = \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix}$. Its eigenvalues are $4, 2$ with coordinates $(1, \pm 1)^T/\sqrt{2}$, respectively. Lifting appends a zero. For each lifted vector, $Au_\pm = \theta_\pm u_\pm + e_3/\sqrt{2}$, so both residuals equal $e_3/\sqrt{2}$, perpendicular to both trial columns. Here $R = [e_3, 0]$ has norm 1. Its action on either Ritz coordinate vector has norm $1/\sqrt{2}$; its action on $(1, 0)^T$ has norm 1. The largest leakage over all directions can exceed each individual Ritz residual.

3. Invariance and a converse. With $U = QZ$, the matrix of residuals is RZ. If every residual vanishes, $RZ = 0$ and multiplication by Z^* gives $R = 0$. Hence $AQ = QH$ and A maps the span of Q into itself. For a counterexample to the single-vector implication take $A = \mathrm{diag}(3, 2, 0)$ and $Q = [e_1, (e_2 + e_3)/\sqrt{2}]$. Then $H = \mathrm{diag}(3, 1)$ and the residual columns are 0 and $(e_2 - e_3)/\sqrt{2}$.

4. Interpreting a certificate. Inclusion gives $[2.07, 2.13]$, wholly inside the interval containing the unique eigenvalue. The nearest possible competing eigenvalue is at least $4 - 2.1 = 1.9$ from θ. Thus $\sin\angle(u, v) \leq 0.03/1.9 < 0.0158$, or an acute angle below 0.91°. If the first interval contained two eigenvalues, inclusion would still certify one eigenvalue in the smaller interval. The same angle estimate would apply to the two-dimensional spectral subspace; an individual direction would need an internal separation estimate.

5. Capstone. The weights are $1, 3, 5$, so the weakest wanted component gains a factor three against the unwanted component. In the given column order the filtered block is $[5e_3, e_1 + 3e_2]$. Its columns are orthogonal, so $Q = [e_3, w]$, $w = (e_1 + 3e_2)/\sqrt{10}$, and the triangular factor is $\mathrm{diag}(5, \sqrt{10})$. Normalizing the starting columns first changes only the triangular factor. Compression gives $H = \mathrm{diag}(3, 19/10)$. The coordinate eigenvectors are the two coordinate axes, lifting to e_3, w. The first residual is zero. The second is

$$r = Aw - \tfrac{19}{10}w = \frac{-9e_1 + 3e_2}{10\sqrt{10}}.$$

Here $e_3^*r = 0$ and $w^*r = (-9 + 9)/100 = 0$. Its norm is 0.3. The exact pair for 3 passes; the pair for 2 misses the residual tolerance and its value is low by 0.1. The next filtered plane contains e_3 and $(e_1 + 9e_2)/\sqrt{82}$. Its unwanted-to-wanted component ratio drops from $1/3$ to $1/9$; the new second Ritz value is $163/82$ and the residual norm is $9/82$. Continue filtering, orthonormalizing, extracting with A, and checking both the target region and the two residuals until both pass or the sweep budget is exhausted.

12.10 Final mastery checklist

- ☐ Derive $H = Q^*AQ$, prove it is Hermitian, and interpret it as the coordinate matrix of $PA|_{\mathcal{U}}$.
- ☐ Define Ritz values and vectors, compute them by hand, and explain both the Rayleigh quotient and the extremal characterization.
- ☐ Prove residual orthogonality, compute R, derive $r_i = Rz_i$, and characterize exact invariance.
- ☐ Derive the eigenvalue inclusion bound and explain how spectral separation changes a value certificate into a direction estimate.
- ☐ Explain why a cluster can be accurately represented even when its individual eigenvectors are poorly determined.
- ☐ Reconstruct the entire filtered iteration. State the overlap and filter assumptions, explain why QR matters, and separate the change of plane from extraction within it.
- ☐ Choose a scale and stopping test, distinguish the residuals of A from filter diagnostics, and count matrix-block products.

12.11 Final revision-program assessment

Try answering each question aloud without referring to the chapter. For each answer supply one equation and one concrete consequence for an implementation.

1. Two programs generate the same sequence of exact subspaces. Why might one lose a wanted direction in floating point, and which two diagnostics would distinguish rank loss from loss of orthogonality?
2. How can a filter give rapid convergence to a cluster whose individual eigenvectors remain difficult to certify? Which two kinds of separation enter your explanation?
3. A run returns a tiny block residual for the wrong invariant plane. Which assumptions or checks from initialization, filter design and target selection could expose the problem?
4. Starting from ordinary subspace iteration, what changes when A is replaced in the update by $T_m(s(A))$? Why does the extraction stage still require the original A?
5. Given only a matrix-block product routine and a target interval, how would you combine direct and iterative methods to construct, extract and test candidate eigenpairs? Identify which conclusions require additional spectral information.

Part II

Review II

13 Module A — Structural Linear Algebra

Linear maps, kernels, images, direct sums, quotients, isomorphism theorems, and their applications

Advanced undergraduate · 8 study sessions · Approximately 14–18 hours

13.1 Introduction: The central idea

The purpose of this module is to develop a structural understanding of linear algebra.

In an introductory course, we learn how to solve linear systems, calculate ranks, compute null spaces, and manipulate matrices.

Here, we will ask a deeper question:

What do these computations reveal about the mathematical structure of a linear transformation?

Given a linear map

$$T : V \longrightarrow W,$$

we want to understand what information it preserves, what information it loses, how it organizes the domain and codomain into subspaces, and why these constructions generalize far beyond matrices.

One of the most important results we will establish is

$$\boxed{V/\ker T \cong \operatorname{im} T.}$$

This equation connects several concepts:

- The kernel describes which distinctions between inputs disappear under T.
- The quotient identifies inputs that T cannot distinguish.
- The image describes all outputs that T can produce.
- The isomorphism tells us that, after identifying indistinguishable inputs, the transformation becomes bijective onto its image.

We will derive this result rigorously, interpret it geometrically, compute it in concrete examples, and investigate its manifestations in numerical linear algebra, differential equations, and scientific computing.

13.2 The six perspectives

Every major concept will be developed through the following perspectives.

Perspective	What we want to understand
Rigorous	Definitions, assumptions, statements, and proofs
Intuitive and geometric	What the construction represents and why it is natural
Computational	How to construct, calculate, or implement it
Structural	How it relates to other concepts and theorems
Universal	How it generalizes to abstract algebra and category theory
Applied	How it appears in numerical methods, functional analysis, PDEs, and scientific computing

These perspectives will not always receive equal emphasis. We will introduce abstraction when it illuminates the concrete mathematics, rather than making every calculation unnecessarily abstract.

13.3 Roadmap

I. Understand linear maps

Kernels · Images · Rank–nullity · RREF

II. Decompose and identify

Subspace sums · Direct sums · Complements · Quotients

III. Discover the common structure

First isomorphism theorem · Universal properties · Exact sequences · Projections

IV. Transfer to applications

Singular systems · Graph Laplacians · Differential operators · Invariant subspaces

13.3.1 Sessions

Session	Main topic
1	Linear maps, kernels, images, and rank–nullity
2	RREF, matrix representations, and parameterization of kernels
3	Subspace sums, direct sums, and complements
4	Quotient spaces and equivalence classes
5	The first isomorphism theorem
6	Universal properties, exact sequences, and projections
7	Applications to numerical systems, graphs, and differential equations
8	Invariant subspaces, spectral connections, and synthesis

Each session includes a checkpoint. A set of exercises, solutions, and a computational laboratory follows the theoretical material.

13.4 Session 1 — The structure of a linear transformation

13.4.1 From matrices to linear maps

Let V, W be vector spaces over a field $\mathbb{F}$, usually $\mathbb{R}$ or $\mathbb{C}$.

A transformation

$$T : V \to W$$

is linear if

$$T(\alpha v + \beta w) = \alpha T(v) + \beta T(w)$$

for every $v, w \in V$ and $\alpha, \beta \in \mathbb{F}$.

The property implies

$$T\left(\sum_{i=1}^{k} a_i v_i\right) = \sum_{i=1}^{k} a_i T(v_i).$$

In particular, if $(v_1, \dots, v_n)$ is a basis of V, knowing the images $T(v_1), \dots, T(v_n)$ determines the transformation completely.

Indeed, every $v \in V$ has a unique expansion

$$v = \sum_{i=1}^{n} a_i v_i,$$

and therefore

$$T(v) = \sum_{i=1}^{n} a_i T(v_i).$$

This is the fundamental reason we can represent linear maps using matrices.

A matrix is a coordinate representation of a linear map, not the linear map itself. Once bases are chosen in the domain and codomain, the columns of the matrix record the coordinates of the images of the domain basis vectors.

This distinction becomes important when studying changes of basis, eigenvalues, and numerical algorithms.

13.4.2 The kernel: indistinguishable inputs

The **kernel** of a linear map is

$$\boxed{\ker T = \{v \in V : T(v) = 0\}.}$$

It is a subspace of V.

The most illuminating property of the kernel is

$$\boxed{T(v) = T(w) \iff v - w \in \ker T.}$$

Proof

By linearity,

$$\begin{aligned} T(v) = T(w) &\iff T(v) - T(w) = 0 \\ &\iff T(v - w) = 0 \\ &\iff v - w \in \ker T. \end{aligned}$$

This tells us exactly what information a linear map loses.

Two inputs produce the same output if and only if they differ by an element of the kernel.

Example

Consider

$$T : \mathbb{R}^3 \to \mathbb{R}^2, \qquad T(x, y, z) = (x + y, z).$$

To find the kernel, solve

$$(x + y, z) = (0, 0).$$

Thus

$$x = -y, \qquad z = 0.$$

Therefore,

$$\boxed{\ker T = \text{span}\{(1, -1, 0)\}.}$$

The transformation observes the sum $x + y$, but cannot distinguish different pairs with the same sum.

For example,

$$T(2, 3, 4) = T(5, 0, 4) = (5, 4).$$

Their difference is

$$(-3, 3, 0) \in \ker T.$$

The kernel contains precisely the directions invisible to the transformation.

Injectivity

A linear map is injective if and only if its kernel is trivial:

$$\boxed{T \text{ injective} \iff \ker T = \{0\}.}$$

To prove the forward direction, observe that $T(v) = 0 = T(0)$ implies $v = 0$ when T is injective.

For the converse, if $T(v) = T(w)$, then $v - w \in \ker T$. A trivial kernel therefore implies $v = w$.

13.4.3 The image: attainable outputs

The **image** of T is

$$\boxed{\operatorname{im} T = \{T(v) : v \in V\}.}$$

It is a subspace of W.

The kernel tells us what the transformation loses; the image tells us what it can produce.

For our example,

$$T(x, y, z) = (x + y, z),$$

we have

$$\operatorname{im} T = \mathbb{R}^2,$$

because every $(a, b) \in \mathbb{R}^2$ is obtained as

$$T(a, 0, b) = (a, b).$$

Thus T is surjective, although not injective.

Notice the distinction between the **codomain** W and the **image** $\operatorname{im} T$. The codomain is part of the specification of T, whereas the image consists of its actually attainable outputs.

13.4.4 Rank–nullity

Assume V is finite-dimensional.

Define

$$\text{nullity}(T) = \dim \ker T,$$

$$\text{rank}(T) = \dim \text{im}\, T.$$

The **rank–nullity theorem** states that

$$\boxed{\dim V = \text{nullity}(T) + \text{rank}(T).}$$

Proof by extending a kernel basis

Let

$$(k_1, \dots, k_s)$$

be a basis of $\ker T$.

Extend it to a basis of V:

$$(k_1, \dots, k_s, u_1, \dots, u_r).$$

We claim that

$$(Tu_1, \dots, Tu_r)$$

is a basis of $\text{im}\, T$.

Spanning. Every $v \in V$ can be written as

$$v = \sum_{i=1}^{s} a_i k_i + \sum_{j=1}^{r} b_j u_j.$$

Applying T,

$$T(v) = \sum_{j=1}^{r} b_j T(u_j),$$

because $T(k_i) = 0$.

Hence these images span $\operatorname{im} T$.

Independence. Suppose

$$\sum_{j=1}^{r} b_j T(u_j) = 0.$$

Then

$$T\left(\sum_{j=1}^{r} b_j u_j\right) = 0,$$

which means

$$\sum_{j=1}^{r} b_j u_j \in \ker T.$$

It can therefore be expressed as a linear combination of the k_i. But the extended basis is linearly independent, so all $b_j = 0$.

Thus the $T(u_j)$ form a basis of the image.

Consequently,

$$\dim \operatorname{im} T = r, \qquad \dim \ker T = s,$$

and since the basis of V contains $s + r$ vectors,

$$\dim V = s + r.$$

What the proof reveals

The proof does more than count dimensions.

It demonstrates that we can choose a basis of V consisting of two groups:

- Vectors annihilated by T.
- Additional vectors whose images form a basis of the image.

This is the beginning of a direct-sum decomposition.

Checkpoint 1

Prove that if $\dim V > \dim W$, then every linear map $T : V \to W$ has a nontrivial kernel.

Then explain why no linear map $\mathbb{R}^4 \to \mathbb{R}^3$ can be injective.

13.5 Session 2 — RREF as a method for discovering structure

Gaussian elimination is often introduced as a procedure for solving linear equations.

We will interpret it instead as a method for exposing the structure of a linear transformation.

13.5.1 Running example

Consider

$$\boxed{A = \begin{pmatrix} 1 & 2 & -1 & 3 \\ 2 & 4 & 1 & 9 \\ 3 & 6 & 0 & 12 \end{pmatrix} : \mathbb{R}^4 \to \mathbb{R}^3.}$$

The third row is the sum of the first two, so we already expect some redundancy in the equations.

Gaussian elimination gives

$$A \sim \begin{pmatrix} 1 & 2 & -1 & 3 \\ 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 0 \end{pmatrix} \sim \underbrace{\begin{pmatrix} 1 & 2 & 0 & 4 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}}_{R=\mathrm{rref}(A)}.$$

The pivot columns are columns 1 and 3, so

$$\operatorname{rank}(A) = 2.$$

13.5.2 Why row reduction preserves the null space

Every elementary row operation is equivalent to multiplying the matrix on the left by an invertible matrix.

Thus

$$R = EA$$

for some invertible $E \in \mathbb{R}^{3\times 3}$.

Therefore,

$$\begin{aligned} Rx = 0 &\iff EAx = 0 \\ &\iff Ax = 0. \end{aligned}$$

So

$$\boxed{\ker R = \ker A.}$$

The transformation EA is obtained by applying an invertible transformation to the outputs of A.

Since an invertible transformation sends only zero to zero, the inputs annihilated by A are unchanged.

This explains conceptually why row reduction preserves the homogeneous solution space.

13.5.3 Constructing a null-space basis

The equations $Rx = 0$ are

$$\begin{aligned} x_1 + 2x_2 + 4x_4 &= 0, \\ x_3 + x_4 &= 0. \end{aligned}$$

The pivot variables are x_1, x_3, while x_2, x_4 are free.

Let

$$x_2 = s, \qquad x_4 = t.$$

Then

$$x_1 = -2s - 4t, \qquad x_3 = -t.$$

Every solution has the form

$$x = \begin{pmatrix} -2s - 4t \\ s \\ -t \\ t \end{pmatrix} = s \begin{pmatrix} -2 \\ 1 \\ 0 \\ 0 \end{pmatrix} + t \begin{pmatrix} -4 \\ 0 \\ -1 \\ 1 \end{pmatrix}.$$

Thus

$$\boxed{\ker A = \operatorname{span} \left\{ \begin{pmatrix} -2 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -4 \\ 0 \\ -1 \\ 1 \end{pmatrix} \right\}.}$$

Why are these vectors not merely generators, but a basis?

13.5.4 The free-variable parameterization is an isomorphism

Define

$$\Phi : \mathbb{R}^2 \to \ker A$$

by

$$\boxed{\Phi(s,t) = \begin{pmatrix} -2s - 4t \\ s \\ -t \\ t \end{pmatrix}.}$$

This map is linear.

It is injective because if

$$\Phi(s_1, t_1) = \Phi(s_2, t_2),$$

then comparing the second and fourth coordinates gives

$$s_1 = s_2, \qquad t_1 = t_2.$$

It is surjective because we have shown that every element of $\ker A$ can be expressed in this form.

Therefore,

$$\boxed{\mathbb{R}^2 \cong \ker A.}$$

Now consider the standard basis of $\mathbb{R}^2$:

$$e_1 = (1, 0), \qquad e_2 = (0, 1).$$

Since an isomorphism maps a basis to a basis,

$$\Phi(e_1), \quad \Phi(e_2)$$

form a basis of $\ker A$.

That is precisely the familiar procedure of setting one free variable equal to 1 and the other free variables equal to 0.

The computational rule is an application of a general theorem: a linear isomorphism transports bases between vector spaces.

13.5.5 The general RREF structure

Suppose $A \in \mathbb{F}^{m \times n}$ has rank r.

After identifying and conceptually reordering pivot and free coordinates, its RREF has the block form

$$R = \begin{pmatrix} I_r & C \\ 0 & 0 \end{pmatrix}.$$

Write

$$x = \begin{pmatrix} x_P \\ x_F \end{pmatrix},$$

where $x_P \in \mathbb{F}^r$ and $x_F \in \mathbb{F}^{n-r}$.

Then

$$Rx = 0 \iff x_P + Cx_F = 0.$$

Thus

$$\boxed{x = \begin{pmatrix} -Cx_F \\ x_F \end{pmatrix}.}$$

The map

$$\Phi(x_F) = \begin{pmatrix} -Cx_F \\ x_F \end{pmatrix}$$

is an isomorphism from $\mathbb{F}^{n-r}$ onto $\ker A$.

Its matrix is

$$N = \begin{pmatrix} -C \\ I_{n-r} \end{pmatrix}.$$

The columns of N, restored to the original coordinate ordering, form a null-space basis.

This gives

$$\dim \ker A = n - r,$$

which is exactly rank–nullity.

13.5.6 Column dependencies and the image

Write

$$A = (a_1\ a_2\ a_3\ a_4).$$

Then

$$Ax = x_1a_1 + x_2a_2 + x_3a_3 + x_4a_4.$$

Therefore,

$$Ax = 0$$

means that the coefficients $x_1, \ldots, x_4$ describe a linear-dependence relation among the columns.

For our matrix,

$$a_2 = 2a_1, \qquad a_4 = 4a_1 + a_3.$$

The corresponding dependence relations are

$$-2a_1 + a_2 = 0$$

and

$$-4a_1 - a_3 + a_4 = 0.$$

These are exactly the two kernel basis vectors obtained above.

The pivot columns of the original matrix are

$$a_1 = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}, \qquad a_3 = \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix}.$$

Thus

$$\boxed{\operatorname{im} A = \operatorname{span}\left\{ \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix} \right\}.}$$

Remember that row reduction preserves column-dependence relations, but generally changes the column space as a subspace of the codomain. That is why the pivot indices are found using RREF, while the actual column-space basis vectors are taken from the original matrix.

Checkpoint 2

Explain why the columns of

$$N = \begin{pmatrix} -2 & -4 \\ 1 & 0 \\ 0 & -1 \\ 0 & 1 \end{pmatrix}$$

are linearly independent without computing an RREF of N.

Then explain why the identity $AN = 0$ alone would not be sufficient to guarantee that they form a basis of $\ker A$.

13.6 Session 3 — Subspace sums, direct sums, and complements

One of the most important themes in linear algebra is the decomposition of a vector space into simpler components.

A good decomposition allows us to analyze a transformation by studying how it acts on the individual components.

13.6.1 The sum of two subspaces

Let $U, W \subseteq V$ be subspaces.

Their sum is defined by

$$\boxed{U + W = \{u + w : u \in U,\ w \in W\}.}$$

The sum is itself a subspace.

It is also the **smallest subspace** containing both U and W.

Indeed, every subspace containing U and W must contain all sums $u + w$, so it must contain $U + W$.

This is why the sum is called the *join* of U and W in the lattice of subspaces of V.

Example

Let

$$U = \text{span}\{(1, 0)\}, \qquad W = \text{span}\{(0, 1)\}.$$

Then

$$U + W = \mathbb{R}^2.$$

But

$$U \cup W$$

is just the union of the two coordinate axes.

The union is not closed under addition, whereas the sum is.

13.6.2 When is a sum direct?

The sum $U + W$ is called a **direct sum** when

$$\boxed{U \cap W = \{0\}.}$$

We write $U \oplus W$.

The key result is the equivalence

$$\boxed{U \cap W = \{0\} \iff \text{every vector in } U + W \text{ has a unique decomposition } u + w.}$$

Proof

Suppose

$$u_1 + w_1 = u_2 + w_2.$$

Then

$$u_1 - u_2 = w_2 - w_1.$$

The left-hand side belongs to U, while the right-hand side belongs to W.

If their intersection is trivial, both sides must be zero.

Therefore,

$$u_1 = u_2, \qquad w_1 = w_2.$$

Conversely, suppose $0 \neq z \in U \cap W$.

Then

$$0 = 0 + 0 = z + (-z)$$

gives two different decompositions, so uniqueness fails.

Thus directness means **independent contributions without ambiguity**.

13.6.3 Internal and external direct sums

There is an important distinction.

An internal direct sum describes subspaces already sitting inside a common vector space:

$$V = U \oplus W.$$

An external direct sum constructs a new vector space from two possibly unrelated spaces:

$$U \oplus W = \{(u, w) : u \in U,\ w \in W\},$$

with componentwise operations.

The external sum always exists.

When $U, W \subseteq V$, the addition map

$$F : U \oplus W \to U + W, \qquad F(u, w) = u + w$$

connects the two constructions.

It is always surjective. It is injective precisely when $U \cap W = \{0\}$.

Therefore, when the internal sum is direct, this addition map is an isomorphism.

This also suggests a deeper reason for the dimension formula.

13.6.4 The dimension formula for subspace sums

For finite-dimensional subspaces U, W,

$$\boxed{\dim(U + W) = \dim U + \dim W - \dim(U \cap W).}$$

Let's derive this using the addition map.

We have

$$F : U \oplus W \to U + W, \qquad F(u, w) = u + w.$$

Its kernel is

$$\begin{aligned} \ker F &= \{(u, w) : u + w = 0\} \\ &= \{(z, -z) : z \in U \cap W\}. \end{aligned}$$

The map

$$z \longmapsto (z, -z)$$

is an isomorphism from $U \cap W$ onto $\ker F$.

Therefore,

$$\dim \ker F = \dim(U \cap W).$$

Applying rank–nullity to F,

$$\dim(U \oplus W) = \dim \ker F + \dim(U + W).$$

Since the external direct sum has dimension $\dim U + \dim W$, rearranging gives the formula.

A structural insight

The intersection measures the redundancy in the addition map.

If $U \cap W \neq \{0\}$, different pairs (u, w) can produce the same vector $u + w$.

If $U \cap W = \{0\}$, this redundancy disappears.

This is the same kernel-versus-identification phenomenon we encountered in Session 1.

13.6.5 Complements

A subspace $U \subseteq V$ is a **complement** of $K \subseteq V$ if

$$\boxed{V = K \oplus U.}$$

Every subspace of a finite-dimensional vector space has at least one complement.

To prove this, take a basis of K, extend it to a basis of V, and let U be the span of the additional basis vectors.

Complements are generally not unique.

For example, if

$$K = \text{span}\{(1, 0)\} \subseteq \mathbb{R}^2,$$

both

$$U_1 = \text{span}\{(0, 1)\}$$

and

$$U_2 = \text{span}\{(1, 1)\}$$

are complements of K.

13.6.6 A complement of the kernel

Let $T : V \to W$ be linear, with finite-dimensional V.

Choose any complement U of $\ker T$:

$$\boxed{V = \ker T \oplus U.}$$

The restriction

$$T|_U : U \to \operatorname{im} T$$

is an isomorphism.

Proof

If $u \in U$ and $Tu = 0$, then

$$u \in U \cap \ker T = \{0\}.$$

So $T|_U$ is injective.

For surjectivity, take any $y \in \operatorname{im} T$, so $y = Tv$ for some $v \in V$.

Write

$$v = k + u, \qquad k \in \ker T, \quad u \in U.$$

Then

$$y = Tv = T(k + u) = Tu.$$

Thus $T|_U$ is surjective onto the image.

We obtain

$$\boxed{U \cong \operatorname{im} T.}$$

This is a stronger structural version of rank–nullity: not only do the dimensions match, but we can identify a subspace of the domain on which T becomes an isomorphism onto its image.

13.6.7 Our running matrix: an explicit decomposition

For

$$A : \mathbb{R}^4 \to \mathbb{R}^3,$$

we found

$$K = \ker A = \operatorname{span}\{(-2, 1, 0, 0), (-4, 0, -1, 1)\}.$$

Because the pivot coordinates are 1 and 3, choose

$$U = \operatorname{span}\{e_1, e_3\}.$$

Then

$$\boxed{\mathbb{R}^4 = K \oplus U.}$$

For any $x = (x_1, x_2, x_3, x_4)^T$, the decomposition is

$$\boxed{x = \underbrace{\begin{pmatrix} -2x_2 - 4x_4 \\ x_2 \\ -x_4 \\ x_4 \end{pmatrix}}_{\in K} + \underbrace{\begin{pmatrix} x_1 + 2x_2 + 4x_4 \\ 0 \\ x_3 + x_4 \\ 0 \end{pmatrix}}_{\in U}.}$$

The first component uses the same free coordinates as x, but adjusts the pivot coordinates to satisfy $Ax = 0$.

The second component corrects the remaining difference and has zero free coordinates.

Geometric interpretation

The same decomposition can be seen in two dimensions. Take $K = \operatorname{span}\{(1, 0)\}$ and $U = \operatorname{span}\{(1, 1)\}$. The lines are oblique, and they intersect only at the origin. In Figure 13.1, the two sides of the parallelogram determine the unique components of x.

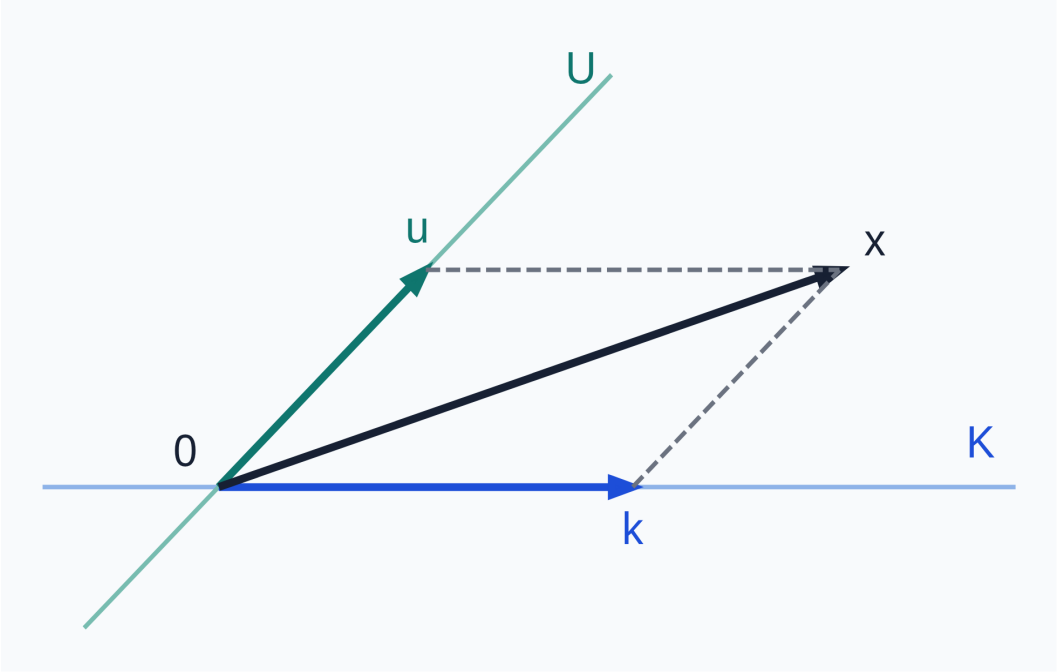


Figure 13.1: The vector x = (3, 1) decomposes uniquely into k = (2, 0) on K and u = (1, 1) on U.

Checkpoint 3

Prove that if $V = \ker T \oplus U$, then

$$\dim U = \operatorname{rank} T.$$

Then explain why U generally cannot be identified with $\operatorname{im} T$ as literally the same subset, even though they are isomorphic.

13.7 Session 4 — Quotient spaces: identifying vectors modulo a subspace

In Session 1, we learned that a linear map cannot distinguish inputs differing by a kernel vector.

In Session 3, we learned to separate a vector space into complementary components.

Now we introduce a third way to understand the same structure: instead of choosing a complement, we can identify all vectors differing by elements of a chosen subspace.

This is the idea behind quotient spaces.

13.7.1 Defining the equivalence relation

Let $S \subseteq V$ be a subspace.

Define

$$\boxed{v \sim w \iff v - w \in S.}$$

This is an equivalence relation.

Reflexivity follows because $v - v = 0 \in S$.

Symmetry follows because if $v - w \in S$, then $w - v = -(v - w) \in S$.

Transitivity follows because if $v - w \in S$ and $w - u \in S$, then

$$v - u = (v - w) + (w - u) \in S.$$

The equivalence class of v is

$$[v] = v + S = \{v + s : s \in S\}.$$

It is called a **coset** of S.

The quotient set is

$$\boxed{V/S = \{v + S : v \in V\}.}$$

The elements of V/S are cosets, not individual vectors of V.

13.7.2 Geometric example

Let

$$V = \mathbb{R}^2, \qquad S = \operatorname{span}\{(1,0)\}.$$

Then

$$(x_1, y_1) \sim (x_2, y_2)$$

precisely when $y_1 = y_2$.

Therefore each equivalence class is a horizontal line.

Every horizontal line becomes one element of the quotient. All points with the same vertical coordinate are identified.

The quotient is isomorphic to $\mathbb{R}$, since one scalar (the vertical coordinate) completely describes each equivalence class.

But V/S is not literally the vertical axis. Its elements are horizontal lines.

The vertical axis provides one representative for each equivalence class, as shown in Figure 13.2.

13.7.3 Making the quotient into a vector space

Define addition and scalar multiplication by

$$(v + S) + (w + S) = (v + w) + S$$

and

$$\alpha(v + S) = \alpha v + S.$$

Before accepting these definitions, we must show they are **well-defined**.

Suppose $v' = v + s_1$ and $w' = w + s_2$, with $s_1, s_2 \in S$.

Then

$$v' + w' = (v + w) + (s_1 + s_2).$$

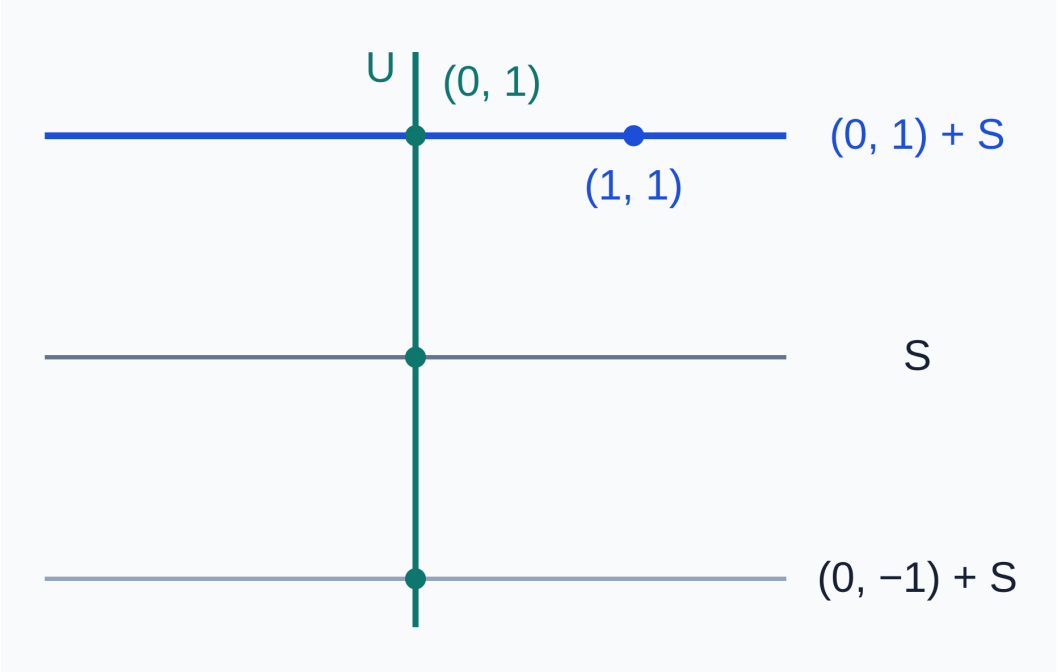


Figure 13.2: Three horizontal cosets of S. The vertical complement U meets each coset at its unique representative.

Because S is a subspace,

$$s_1 + s_2 \in S.$$

Thus

$$(v' + w') + S = (v + w) + S.$$

Similarly,

$$\alpha v' - \alpha v = \alpha s_1 \in S,$$

so scalar multiplication is independent of the representative.

The vector-space axioms then follow from those of V.

The zero vector of V/S is the coset $S = 0 + S$.

13.7.4 The canonical quotient map

Define

$$\boxed{q : V \to V/S, \qquad q(v) = v + S.}$$

This map is linear and surjective.

Its kernel is

$$\begin{aligned} \ker q &= \{v : q(v) = S\} \\ &= \{v : v + S = S\} \\ &= S. \end{aligned}$$

Therefore,

$$\boxed{\ker q = S.}$$

This is a revealing fact: the quotient map is a linear transformation that collapses precisely the subspace S.

It is universal among linear maps that do so, as we will prove in Session 6.

13.7.5 Dimension of a quotient space

Suppose

$$\dim V = n, \qquad \dim S = k.$$

Then

$$\boxed{\dim(V/S) = n - k.}$$

Proof

Choose a basis $(s_1, \dots, s_k)$ of S, and extend it to a basis of V:

$$(s_1, \dots, s_k, u_1, \dots, u_{n-k}).$$

We claim that the cosets

$$u_1 + S, \dots, u_{n-k} + S$$

form a basis of V/S.

Spanning: Every $v \in V$ can be expressed as

$$v = \sum_i a_i s_i + \sum_j b_j u_j.$$

Since the s_i vanish in the quotient,

$$v + S = \sum_j b_j (u_j + S).$$

Independence: Suppose

$$\sum_j b_j (u_j + S) = S.$$

Then

$$\sum_j b_j u_j \in S.$$

Therefore it can also be written as a linear combination of the s_i.

Independence of the extended basis forces all $b_j = 0$.

Thus the quotient has a basis containing $n - k$ cosets.

13.7.6 Quotients and complements

Suppose

$$V = S \oplus U.$$

Every $v \in V$ can be written uniquely as

$$v = s + u.$$

Because $s \in S$,

$$v + S = u + S.$$

Thus every coset contains exactly one vector from U.

Define

$$\Psi : U \to V/S, \qquad \Psi(u) = u + S.$$

This is a linear isomorphism.

Hence

$$\boxed{V/S \cong U.}$$

There is an important distinction between the two objects.

The quotient V/S is determined by V and S.

A complementary subspace U is generally not unique.

Therefore, the quotient exists independently of any choice of complement, even though a complement provides a convenient concrete model for it.

Checkpoint 4

Explain why the quotient V/S should be thought of as a space of equivalence classes rather than as a particular complementary subspace.

Then prove that if $V = S \oplus U$, the restriction of the quotient map q to U is an isomorphism $U \to V/S$.

13.8 Session 5 — The first isomorphism theorem

We have now developed enough theory to prove the central theorem of the module.

13.8.1 Motivation: turning a noninjective map into an isomorphism

Let

$$T : V \to W$$

be linear.

Suppose T is not injective.

Then there are distinct vectors $v, w \in V$ such that

$$T(v) = T(w).$$

We know that this happens precisely when

$$v - w \in \ker T.$$

Therefore, if we identify vectors differing by elements of $\ker T$, the ambiguity should disappear.

That is exactly what the quotient

$$V / \ker T$$

does.

The first isomorphism theorem makes this intuition rigorous.

13.8.2 Statement

For every linear map $T : V \to W$,

$$\boxed{V/\ker T \cong \operatorname{im} T.}$$

More precisely, define

$$\widetilde{T} : V/\ker T \to \operatorname{im} T$$

by

$$\boxed{\widetilde{T}(v + \ker T) = T(v).}$$

The theorem states that $\widetilde{T}$ is a linear isomorphism.

13.8.3 Proof

We must establish four properties.

Well-definedness

Suppose two representatives describe the same coset:

$$v + \ker T = w + \ker T.$$

Then

$$v - w \in \ker T.$$

Consequently,

$$T(v - w) = 0,$$

which implies

$$T(v) = T(w).$$

Thus $\widetilde{T}$ does not depend on the representative used.

Linearity

For $v, w \in V$,

$$\begin{aligned}\widetilde{T}(\alpha(v + \ker T) + \beta(w + \ker T)) &= \widetilde{T}((\alpha v + \beta w) + \ker T) \\ &= T(\alpha v + \beta w) \\ &= \alpha T(v) + \beta T(w).\end{aligned}$$

Therefore $\widetilde{T}$ is linear.

Injectivity

Suppose

$$\widetilde{T}(v + \ker T) = 0.$$

Then $T(v) = 0$, so $v \in \ker T$.

Hence

$$v + \ker T = \ker T,$$

the zero element of the quotient.

Therefore,

$$\ker \widetilde{T} = \{0\},$$

and $\widetilde{T}$ is injective.

Surjectivity

Take any $y \in \operatorname{im} T$.

By definition, $y = T(v)$ for some $v \in V$.

Thus

$$y = \widetilde{T}(v + \ker T).$$

So $\widetilde{T}$ is surjective.

We have proved that $\widetilde{T}$ is an isomorphism.

13.8.4 The canonical factorization of a linear map

The theorem gives the factorization

$$\boxed{V \xrightarrow{q} V/\ker T \underset{\cong}{\xrightarrow{\widetilde{T}}} \operatorname{im} T \overset{\iota}{\hookrightarrow} W.}$$

Here q is the quotient map and ι is the inclusion.

The original transformation is their composition:

$$\boxed{T = \iota \circ \widetilde{T} \circ q.}$$

This separates the behavior of any linear map into three parts.

The quotient map removes the ambiguity caused by the kernel.

The induced map $\widetilde{T}$ establishes a one-to-one correspondence with the image.

The inclusion regards that image as a subspace of the entire codomain.

A noninvertible linear map contains an invertible transformation once the kernel and image have been properly accounted for.

This is the essential structural content of the theorem.

13.8.5 Rank–nullity follows from the isomorphism theorem

We already proved that

$$\dim(V/S) = \dim V - \dim S.$$

Setting $S = \ker T$ gives

$$\dim(V/\ker T) = \dim V - \dim \ker T.$$

But isomorphic finite-dimensional spaces have equal dimensions, so

$$\dim(V/\ker T) = \dim \operatorname{im} T.$$

Therefore,

$$\dim V - \dim \ker T = \dim \operatorname{im} T.$$

Rearranging,

$$\boxed{\dim V = \dim \ker T + \dim \operatorname{im} T.}$$

We have derived rank–nullity a second way.

The two proofs emphasize different ideas:

The basis-extension proof counts independent directions.

The quotient-space proof identifies indistinguishable inputs and shows that the remaining information is in bijection with the image.

13.8.6 The theorem applied to our running matrix

For our matrix $A : \mathbb{R}^4 \to \mathbb{R}^3$,

$$K = \ker A = \operatorname{span}\{(-2, 1, 0, 0), (-4, 0, -1, 1)\}.$$

We constructed the complement

$$U = \operatorname{span}\{e_1, e_3\}.$$

Thus

$$\mathbb{R}^4 = K \oplus U.$$

Every coset in $\mathbb{R}^4/K$ has exactly one representative in U.

A basis of the quotient is therefore

$$(e_1 + K,\ e_3 + K).$$

The induced isomorphism maps these basis elements to

$$\widetilde{A}(e_1 + K) = Ae_1 = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

and

$$\widetilde{A}(e_3 + K) = Ae_3 = \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix}.$$

These are exactly the basis vectors of $\operatorname{im} A$.

Thus we have the explicit identifications

$$\boxed{\mathbb{R}^4/K \cong U \cong \operatorname{im} A.}$$

The map from the quotient to the image is canonical: it is determined by A.

The identification of the quotient with U depends on the complement we selected.

13.8.7 Connection to nonuniqueness of linear systems

Consider the equation

$$Ax = b.$$

Suppose it is consistent, and x_0 is one solution.

Then every solution has the form

$$\boxed{x = x_0 + k, \qquad k \in \ker A.}$$

Proof

If $Ax = b = Ax_0$, then

$$A(x - x_0) = 0,$$

so $x - x_0 \in \ker A$.

Conversely, if $k \in \ker A$, then

$$A(x_0 + k) = Ax_0 + Ak = b.$$

Hence the full solution set is the affine subspace

$$x_0 + \ker A.$$

This is not usually a vector subspace because it need not contain zero.

However, it is precisely one coset of $\ker A$.

Thus the quotient $V/\ker A$ can be interpreted as the space of equivalence classes of solutions that share the same output.

The first isomorphism theorem then says that each such equivalence class corresponds to exactly one attainable right-hand side b.

Checkpoint 5

Prove that if $Ax = b$ is consistent, all its solutions form a single coset of $\ker A$.

Then explain why the first isomorphism theorem is stronger than simply stating that the number of free variables equals the nullity.

13.9 Session 6 — Universal properties, exact sequences, and projections

We now examine the constructions from the previous sessions at a higher level of abstraction.

The objective is not to begin a full category theory course, but to recognize how the same ideas can be characterized by their relationships with maps.

13.9.1 What is a universal property?

A universal property describes a mathematical object by how maps into or out of that object behave.

Rather than specifying only its elements, we specify a mapping problem for which the object provides a unique solution.

This often reveals why a construction generalizes to other algebraic structures.

13.9.2 The universal property of the quotient

Let $S \subseteq V$, with quotient map

$$q : V \to V/S.$$

Suppose

$$L : V \to Z$$

is linear and $S \subseteq \ker L$.

Then there exists a unique linear map

$$\overline{L} : V/S \to Z$$

such that

$$\boxed{L = \overline{L} \circ q.}$$

The map is necessarily defined by

$$\overline{L}(v + S) = L(v).$$

Why is this well-defined?

If $v + S = w + S$, then $v - w \in S \subseteq \ker L$, so

$$L(v) = L(w).$$

Linearity follows immediately.

Uniqueness follows because every coset equals $q(v)$ for some v, so the identity

$$\overline{L}(q(v)) = L(v)$$

determines $\overline{L}$ everywhere.

The maps form the commuting triangle in Figure 13.3.

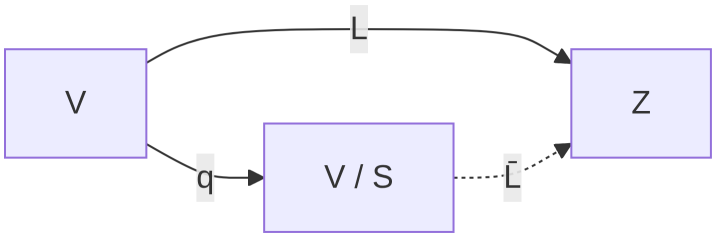


Figure 13.3: The dashed arrow is the unique induced map from V / S to Z.

Both paths from V to Z give the same result.

Interpretation

The quotient is the universal construction that forces all vectors in S to become zero.

Any linear map already annihilating S automatically factors through the quotient.

This gives a conceptual explanation of why quotient spaces arise throughout algebra.

13.9.3 Direct sums as universal constructions

The external direct sum $V \oplus W$ has canonical inclusions

$$i_V(v) = (v, 0), \qquad i_W(w) = (0, w).$$

Suppose we have linear maps

$$f : V \to Z, \qquad g : W \to Z.$$

There exists a unique linear map

$$h : V \oplus W \to Z$$

such that

$$h \circ i_V = f, \qquad h \circ i_W = g.$$

The map must be

$$\boxed{h(v, w) = f(v) + g(w).}$$

This is the **coproduct universal property**.

There is also a product universal property.

Let π_V, π_W be the coordinate projections from $V \oplus W$.

Given linear maps

$$f : Z \to V, \qquad g : Z \to W,$$

there is a unique map

$$h : Z \to V \oplus W$$

defined by

$$h(z) = (f(z), g(z))$$

such that

$$\pi_V \circ h = f, \qquad \pi_W \circ h = g.$$

Thus finite direct sums of vector spaces serve simultaneously as products and coproducts.

This is an example of a categorical structure called a *biproduct*.

It illustrates how ordinary linear algebraic operations can be understood through universal mapping properties.

13.9.4 Exact sequences

Consider a sequence of linear maps

$$\cdots \longrightarrow V_{i-1} \xrightarrow{T_{i-1}} V_i \xrightarrow{T_i} V_{i+1} \longrightarrow \cdots.$$

The sequence is **exact at** V_i if

$$\boxed{\operatorname{im} T_{i-1} = \ker T_i.}$$

In particular, exactness implies

$$T_i \circ T_{i-1} = 0.$$

Every vector coming from the preceding map is annihilated by the next, and every vector annihilated by the next comes from the preceding map.

Short exact sequences

A particularly important form is

$$\boxed{0 \longrightarrow K \xrightarrow{i} V \xrightarrow{p} I \longrightarrow 0.}$$

Exactness means that

$$i \text{ is injective}, \qquad p \text{ is surjective}, \qquad \operatorname{im} i = \ker p.$$

For any linear map $T : V \to W$, we obtain the canonical short exact sequence

$$\boxed{0 \longrightarrow \ker T \longrightarrow V \xrightarrow{T} \operatorname{im} T \longrightarrow 0.}$$

Here, the arrow labeled T is regarded as a map onto its image.

This sequence packages the essential information about kernels and images into one expression.

Rank–nullity from exactness

For finite-dimensional spaces, a short exact sequence

$$0 \to K \to V \to I \to 0$$

implies

$$\boxed{\dim V = \dim K + \dim I.}$$

Thus rank–nullity can be seen as a dimension consequence of exactness.

13.9.5 Why exact sequences matter beyond vector spaces

For vector spaces over a field, every short exact sequence splits.

Informally, this means that a sequence

$$0 \to K \to V \to I \to 0$$

allows us to find an isomorphism

$$\boxed{V \cong K \oplus I.}$$

The reason is that every subspace has a complement.

However, the splitting generally requires a choice and is not canonical.

In other algebraic structures, such as modules over arbitrary rings, short exact sequences need not split.

This is why exact sequences carry information beyond dimension counting in abstract algebra.

They describe how objects fit together, including whether one can be decomposed into independent components.

13.9.6 Projections correspond to direct-sum decompositions

Suppose

$$V = K \oplus U.$$

Every $v \in V$ has a unique decomposition

$$v = k + u.$$

Define

$$P : V \to V, \qquad P(k+u) = k.$$

This map is a projection onto K along U.

By construction,

$$\boxed{\operatorname{im} P = K, \qquad \ker P = U.}$$

Moreover,

$$P^2 = P.$$

Indeed,

$$P(P(v)) = P(k) = k = P(v).$$

A linear operator satisfying

$$\boxed{P^2 = P}$$

is called **idempotent**.

Converse: every idempotent operator gives a direct sum

Suppose $P : V \to V$ is linear and $P^2 = P$.

For every $v \in V$,

$$v = P(v) + (v - P(v)).$$

The first term belongs to $\operatorname{im} P$.

The second belongs to $\ker P$, because

$$P(v - P(v)) = P(v) - P^2(v) = 0.$$

Thus

$$V = \operatorname{im} P + \ker P.$$

The intersection is trivial.

If $w \in \operatorname{im} P \cap \ker P$, then $w = P(v)$ for some v and $P(w) = 0$. Hence

$$w = P(v) = P^2(v) = P(w) = 0.$$

Therefore,

$$\boxed{V = \operatorname{im} P \oplus \ker P.}$$

This is a general algebraic characterization of projections.

13.9.7 Our running example: constructing a projection

Recall the decomposition

$$\mathbb{R}^4 = \ker A \oplus U, \qquad U = \operatorname{span}\{e_1, e_3\}.$$

The projection onto $\ker A$ along U is

$$P(x_1, x_2, x_3, x_4) = (-2x_2 - 4x_4,\ x_2,\ -x_4,\ x_4).$$

In matrix form,

$$\boxed{P = \begin{pmatrix} 0 & -2 & 0 & -4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix}.}$$

You can verify that

$$P^2 = P, \qquad AP = 0.$$

The second identity follows because every output of P lies in $\ker A$.

But P is not symmetric.

So this projection is generally oblique, not orthogonal.

The distinction will become central in Module C, when we introduce inner products and adjoints.

Checkpoint 6

Prove that an idempotent operator is the identity on its image.

Then explain why the identity $P^2 = P$ does not imply $P = P^T$.

Finally, explain why the existence of algebraic projections requires no inner product.

13.10 Session 7 — Applications: when the abstract structure becomes useful

We now apply the ideas developed in Sessions 1–6.

The goal is not merely to identify a kernel or quotient in a familiar setting. We want to see how the structure answers a concrete question that would otherwise be difficult to formulate or solve.

We will examine three examples: rank-deficient linear systems, graph Laplacians, and differential operators.

13.10.1 Application I: rank-deficient linear systems

Consider again

$$A = \begin{pmatrix} 1 & 2 & -1 & 3 \\ 2 & 4 & 1 & 9 \\ 3 & 6 & 0 & 12 \end{pmatrix}.$$

We know that

$$\operatorname{rank} A = 2, \qquad \dim \ker A = 2.$$

Now consider the linear system

$$Ax = b.$$

There are two different questions to investigate:

1. Does a solution exist?
2. If it exists, is it unique?

The image and kernel answer these questions separately.

Existence

A solution exists precisely when

$$\boxed{b \in \operatorname{im} A.}$$

Every column of our matrix has its third entry equal to the sum of its first two entries.

Therefore every vector in the image satisfies

$$b_3 = b_1 + b_2.$$

Conversely, the image has dimension 2, and the set of vectors satisfying this equation is a two-dimensional subspace of $\mathbb{R}^3$.

Hence

$$\boxed{\operatorname{im} A = \{b \in \mathbb{R}^3 : b_3 = b_1 + b_2\}.}$$

Consequently, $Ax = b$ is consistent if and only if

$$\boxed{b_3 = b_1 + b_2.}$$

For example,

$$b = (1, 2, 3)^T$$

is attainable, whereas

$$b = (1, 0, 0)^T$$

is not.

Uniqueness

Suppose x_0 is one solution.

Then all solutions are

$$\boxed{x = x_0 + k, \qquad k \in \ker A.}$$

For $b = (1, 2, 3)^T$, we can take

$$x_0 = e_1 = (1, 0, 0, 0)^T.$$

The full solution family is

$$x = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + s \begin{pmatrix} -2 \\ 1 \\ 0 \\ 0 \end{pmatrix} + t \begin{pmatrix} -4 \\ 0 \\ -1 \\ 1 \end{pmatrix}.$$

Thus there are infinitely many solutions, indexed by two parameters.

Why this matters numerically

For a rank-deficient matrix, the nonuniqueness is not a numerical accident. It is part of the mathematical structure of the problem.

A numerical algorithm must determine which solution to return.

For example, one might choose the unique solution belonging to a selected complement of $\ker A$.

With Euclidean geometry, an especially natural choice is the solution orthogonal to the kernel. This is the minimum-Euclidean-norm solution, which is computed using the Moore–Penrose pseudoinverse.

Those ideas will be developed further in Module C.

The numerical complication: near rank deficiency

Suppose a matrix is not exactly rank deficient, but has a very small singular value.

Then small changes in the right-hand side or rounding errors may produce large changes in the solution.

This is a conditioning issue.

Thus the mathematical distinction between a trivial and a nontrivial kernel leads naturally to numerical questions involving singular values, condition numbers, regularization, and stable algorithms.

A further question is whether the rank is exact or only *numerically* small relative to a chosen tolerance.

That is why robust numerical linear algebra generally uses singular values or rank-revealing factorizations rather than treating an RREF computed in floating-point arithmetic as an exact algebraic answer.

13.10.2 Application II: graphs, flows, and discrete differential operators

Consider the path graph in Figure 13.4.

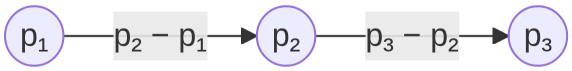


Figure 13.4: The two arrows fix the sign convention for the incidence matrix of the path graph.

Assign a scalar potential p_i to each vertex.

For example, p_i might represent electrical potential, temperature, or a scalar state variable.

Collect the potentials in

$$p = \begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix}.$$

We want to measure differences across the edges:

$$p_2 - p_1, \qquad p_3 - p_2.$$

Define

$$\boxed{D = \begin{pmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix}.}$$

Then

$$Dp = \begin{pmatrix} p_2 - p_1 \\ p_3 - p_2 \end{pmatrix}.$$

The matrix D is a discrete gradient operator, represented by an oriented graph incidence matrix.

What is its kernel?

Solve

$$Dp = 0.$$

This gives

$$p_2 = p_1, \qquad p_3 = p_2.$$

Thus every potential must have the same value:

$$p = c \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}.$$

Therefore,

$$\boxed{\ker D = \operatorname{span}\{\mathbf{1}\}, \qquad \mathbf{1} = (1,1,1)^T.}$$

What does this mean?

Adding the same constant to every potential does not change any potential difference:

$$D(p + c\mathbf{1}) = Dp.$$

The discrete gradient cannot detect absolute potential levels. It detects only relative differences.

This is an example of a **gauge freedom**.

The natural quotient

Because constant shifts are invisible, the essential input space is

$$\mathbb{R}^3/\operatorname{span}\{\mathbf{1}\}.$$

Our graph is connected, and D has rank 2. Since its codomain is $\mathbb{R}^2$,

$$\operatorname{im} D = \mathbb{R}^2.$$

The first isomorphism theorem gives

$$\boxed{\mathbb{R}^3/\operatorname{span}\{\mathbf{1}\} \cong \mathbb{R}^2.}$$

This has a concrete interpretation:

Two independent edge differences contain exactly the same information as three vertex potentials, provided we ignore a common additive constant.

Choosing a representative

We can remove the ambiguity by fixing one potential, for example

$$p_1 = 0.$$

Then, given edge differences d_1, d_2,

$$p_2 = d_1, \qquad p_3 = d_1 + d_2.$$

Thus we obtain the unique representative

$$p = \begin{pmatrix} 0 \\ d_1 \\ d_1 + d_2 \end{pmatrix}.$$

This is another example of choosing a complement of the kernel.

The quotient is intrinsic; setting $p_1 = 0$ is a convenient choice of coordinates.

Constructing the graph Laplacian

Now define

$$L = D^T D.$$

Calculating,

$$\boxed{L = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{pmatrix}.}$$

This is the graph Laplacian for our path graph.

Notice that

$$L\mathbf{1} = 0.$$

In fact,

$$\ker L = \ker D.$$

To see this, suppose $Lp = 0$. Then

$$0 = p^T Lp = p^T D^T Dp = \|Dp\|_2^2.$$

Therefore $Dp = 0$.

Conversely, $Dp = 0$ clearly implies $Lp = 0$.

Hence

$$\boxed{\ker L = \operatorname{span}\{\mathbf{1}\}.}$$

The Laplacian is singular because absolute potential levels are undetermined.

Compatibility conditions

Consider the system

$$Lp = b.$$

Because $L\mathbf{1} = 0$ and L is symmetric, every vector in its image is orthogonal to $\mathbf{1}$:

$$\mathbf{1}^T(Lp) = (L\mathbf{1})^T p = 0.$$

Thus a necessary condition for solvability is

$$\boxed{b_1 + b_2 + b_3 = 0.}$$

It is also sufficient because $\operatorname{rank} L = 2$, and the vectors satisfying this condition form a two-dimensional subspace.

Therefore,

$$Lp = b$$

is solvable precisely when the entries of b sum to zero.

This is exactly the same kernel-image structure we studied abstractly.

Why this is important

Graph Laplacians appear in electrical networks, diffusion, numerical PDEs, graph algorithms, machine learning, and scientific computing.

Their constant-vector null space is not an incidental numerical defect. It reflects a fundamental invariance: only differences of potential matter.

For more general connected graphs, the same principle applies. The discrete gradient has constant vectors in its kernel, while the transpose of the incidence matrix describes a discrete divergence of edge flows.

This creates a natural bridge toward discrete differential operators and the algebraic structure of conservation laws.

13.10.3 Application III: differential operators and function spaces

The preceding examples used finite-dimensional vectors.

But the definitions of kernel, image, quotient, and linear map do not require finite-dimensional spaces.

Let's investigate an infinite-dimensional example.

Consider the vector spaces

$$V = C^1([0,1]), \qquad W = C([0,1]).$$

Here $C^1([0,1])$ consists of continuously differentiable real-valued functions, while $C([0,1])$ consists of continuous functions.

Define the differentiation operator

$$\boxed{D : V \to W, \qquad D(f) = f'.}$$

This is linear because

$$(\alpha f + \beta g)' = \alpha f' + \beta g'.$$

What is its kernel?

We have

$$Df = 0 \iff f' = 0.$$

On a connected interval, a continuously differentiable function with zero derivative is constant.

Therefore,

$$\boxed{\ker D = \{f : f(x) = c,\ c \in \mathbb{R}\}.}$$

The kernel is a one-dimensional vector space of constant functions.

What is its image?

Let $g \in C([0,1])$.

Define

$$f(x) = \int_0^x g(t)\,dt.$$

By the fundamental theorem of calculus,

$$f'(x) = g(x).$$

Thus $Df = g$.

Since g was arbitrary,

$$\boxed{\operatorname{im} D = C([0,1]).}$$

So differentiation is surjective from $C^1([0,1])$ onto $C([0,1])$, but not injective.

The quotient interpretation

The first isomorphism theorem gives

$$\boxed{C^1([0,1])/\{\text{constant functions}\} \cong C([0,1]).}$$

This is a genuinely infinite-dimensional example of the same theorem.

It says that differentiation determines a continuously differentiable function uniquely up to an additive constant.

The integral

$$f(x) = \int_0^x g(t)\,dt$$

chooses the unique representative satisfying

$$f(0) = 0.$$

Other representatives differ by a constant:

$$f_c(x) = \int_0^x g(t)\,dt + c.$$

Relation to initial conditions

This example gives a structural interpretation of why one initial condition determines a unique solution to

$$f'(x) = g(x).$$

The differential equation determines the solution only up to an element of $\ker D$, which has dimension one.

One scalar condition, such as

$$f(0) = a,$$

fixes the remaining freedom.

The same idea extends to higher-order differential operators.

For instance,

$$L : C^2([0,1]) \to C([0,1]), \qquad L(f) = f''$$

has kernel

$$\ker L = \operatorname{span}\{1, x\}.$$

Thus solutions to $f'' = g$ differ by an affine function $a + bx$, and two independent initial conditions select a unique solution.

Why this matters for functional analysis and PDEs

This is the beginning of operator theory.

Instead of studying matrices acting on coordinate vectors, we study linear operators acting on functions.

The same algebraic structures reappear:

$$\ker L, \qquad \operatorname{im} L, \qquad V/\ker L.$$

But in functional analysis, additional questions become important.

For example, are the relevant spaces complete? Is the operator continuous or bounded? Is its image closed? Does a closed complementary subspace exist?

In infinite-dimensional normed spaces, these questions cannot be answered using algebra alone.

In particular, a subspace may have an algebraic complement without having a closed topological complement.

This is one of the major reasons functional analysis is more subtle than finite-dimensional linear algebra.

The algebraic foundations, however, are exactly the structures developed in this module.

Checkpoint 7

For the operator

$$L(f) = f''$$

on $C^2([0,1])$, prove that

$$\ker L = \operatorname{span}\{1, x\}.$$

Then explain why prescribing $f(0)$ and $f'(0)$ selects a unique solution of $f'' = g$ for every continuous g.

Finally, interpret this as choosing a unique representative from a coset of $\ker L$.

13.11 Session 8 — Invariant subspaces and the bridge to numerical eigenvalue methods

The applications in Session 7 showed how kernels, images, and quotient spaces arise in different mathematical settings.

We now connect Module A to the spectral theory that underlies your Numerical Linear Algebra course and HASC seminar.

The main new idea is the **invariant subspace**.

13.11.1 Eigenspaces and invariant subspaces

Let

$$T : V \to V$$

be a linear operator.

An eigenvector is a nonzero vector v satisfying

$$Tv = \lambda v.$$

The corresponding eigenspace is

$$\boxed{E_\lambda = \ker(T - \lambda I).}$$

This is already a direct connection to Module A.

An eigenspace is not a completely new type of object. It is the kernel of a shifted linear operator.

Now generalize the idea.

A subspace $U \subseteq V$ is **invariant under** T if

$$\boxed{T(U) \subseteq U.}$$

Thus, whenever a vector belongs to U, applying T keeps it inside U.

Every eigenspace is invariant because

$$T(v) = \lambda v \in E_\lambda$$

whenever $v \in E_\lambda$.

But the converse is not generally true: an invariant subspace need not have a basis of eigenvectors.

For example, the real plane $\mathbb{R}^2$ is invariant under a rotation through 90°, although that rotation has no real eigenvectors.

13.11.2 Invariant subspaces and block matrices

Suppose

$$V = U \oplus W$$

and both U, W are invariant under T.

Choose a basis of U and a basis of W, then combine them to obtain a basis of V.

In this adapted basis, T has a block-diagonal matrix:

$$\boxed{[T] = \begin{pmatrix} T_U & 0 \\ 0 & T_W \end{pmatrix}.}$$

This follows because $T(U) \subseteq U$ and $T(W) \subseteq W$, so the transformation does not mix the two components.

This is a fundamental connection between direct-sum decompositions and spectral methods.

A suitable invariant-subspace decomposition can turn one large problem into smaller independent problems.

If only U is invariant and its chosen complement W is not, the matrix is generally block upper triangular rather than block diagonal:

$$[T] = \begin{pmatrix} T_U & B \\ 0 & C \end{pmatrix}.$$

This distinction is important.

A direct sum alone does not imply block diagonality; invariance of both summands does.

13.11.3 A symmetric matrix example

Consider

$$\boxed{B = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 9 \end{pmatrix}.}$$

The eigenvalues are

$$\lambda_1 = 1, \qquad \lambda_2 = 3, \qquad \lambda_3 = 9.$$

Corresponding eigenvectors are

$$v_1 = \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}, \qquad v_2 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \qquad v_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

Observe that

$$Bv_1 = v_1, \qquad Bv_2 = 3v_2, \qquad Bv_3 = 9v_3.$$

Define

$$U = \operatorname{span}\{v_1, v_2\}, \qquad W = \operatorname{span}\{v_3\}.$$

Then

$$\mathbb{R}^3 = U \oplus W.$$

Both subspaces are invariant, so B has a block-diagonal representation in a basis adapted to this decomposition.

In fact, after normalizing the eigenvectors, the matrix is diagonal in that eigenbasis.

This illustrates how spectral theory refines the general direct-sum constructions from Module A.

13.11.4 The spectral theorem: a preview

The matrix B is symmetric:

$$B = B^T.$$

The spectral theorem states that every real symmetric matrix admits an orthonormal eigenbasis.

Thus

$$\boxed{B = Q\Lambda Q^T}$$

for some orthogonal Q and real diagonal Λ.

We will study the full theorem in Module D.

For now, notice one important consequence of symmetry.

Suppose

$$Bv = \lambda v, \qquad Bw = \mu w.$$

Because $B = B^T$,

$$\begin{aligned} \lambda\langle v, w\rangle &= \langle Bv, w\rangle \\ &= \langle v, Bw\rangle \\ &= \mu\langle v, w\rangle. \end{aligned}$$

Therefore,

$$(\lambda - \mu)\langle v, w\rangle = 0.$$

If $\lambda \neq \mu$,

$$\boxed{\langle v, w\rangle = 0.}$$

Distinct eigenspaces of a real symmetric matrix are orthogonal.

This explains why symmetric eigenvalue problems have such particularly useful geometric structure.

13.11.5 Polynomial functions of matrices

Let

$$p(t) = a_0 + a_1 t + \cdots + a_d t^d.$$

Define

$$p(B) = a_0 I + a_1 B + \cdots + a_d B^d.$$

If $Bv_i = \lambda_i v_i$, then

$$B^k v_i = \lambda_i^k v_i.$$

Therefore,

$$\begin{aligned} p(B)v_i &= \sum_{k=0}^{d} a_k B^k v_i \\ &= \sum_{k=0}^{d} a_k \lambda_i^k v_i \\ &= p(\lambda_i) v_i. \end{aligned}$$

Thus

$$\boxed{p(B)v_i = p(\lambda_i)v_i.}$$

A polynomial of a matrix preserves its eigendirections, while changing the scaling factors associated with them.

An elementary spectral filter

For our example, take

$$p(t) = \frac{9-t}{8}.$$

Then

$$p(1) = 1, \qquad p(3) = \frac{3}{4}, \qquad p(9) = 0.$$

If

$$x = c_1 v_1 + c_2 v_2 + c_3 v_3,$$

then

$$p(B)x = c_1 v_1 + \frac{3}{4} c_2 v_2.$$

The component along v_3 disappears entirely.

Repeated application attenuates the component along v_2 relative to v_1.

This is a simple example of polynomial spectral filtering.

Chebyshev filtering uses carefully chosen higher-degree polynomials to create more effective separation between desired and unwanted spectral regions.

The mathematical mechanism is the same: an appropriately chosen polynomial changes the relative magnitudes of spectral components.

13.11.6 Connection to subspace iteration

Power iteration repeatedly applies a matrix to a vector, typically with normalization:

$$x_{k+1} = \frac{Bx_k}{\|Bx_k\|}.$$

Under suitable spectral assumptions, the vector becomes increasingly aligned with an eigendirection corresponding to an eigenvalue of largest magnitude.

Subspace iteration generalizes this idea by evolving an entire subspace.

If $X_k \in \mathbb{R}^{n\times s}$ has linearly independent columns spanning a trial subspace, a basic iteration computes

$$Y_{k+1} = BX_k$$

When Y_{k+1} has s independent columns, orthonormalizing them gives X_{k+1}.

A filtered variant replaces B by $p(B)$:

$$Y_{k+1} = p(B)X_k.$$

The polynomial is designed so that desired spectral components are amplified relative to unwanted ones. The filtered block must retain s independent columns for this step to preserve the trial subspace dimension.

This gives a direct connection to the structures we've studied:

- Eigenvectors belong to kernels of $B - \lambda I$.
- Eigenspaces and their sums are invariant subspaces.
- Direct sums describe decompositions into spectral components.
- Polynomial filters act on each eigenspace through scalar multiplication.
- Orthogonalization and projection exploit additional inner-product geometry.

13.11.7 The Rayleigh–Ritz connection

Suppose B is symmetric and $Q \in \mathbb{R}^{n\times s}$ has orthonormal columns spanning a trial subspace U.

The Rayleigh–Ritz procedure forms

$$H = Q^T BQ.$$

If

$$Hy = \theta y,$$

then

$$v = Qy$$

is a Ritz vector, and θ is its Ritz value.

The residual is

$$r = Bv - \theta v.$$

Now observe:

$$\begin{aligned} Q^T r &= Q^T(BQy - \theta Qy) \\ &= Q^T BQy - \theta Q^T Qy \\ &= Hy - \theta y \\ &= 0. \end{aligned}$$

Therefore,

$$\boxed{r \perp U.}$$

This is an example of how a numerical approximation is characterized by a subspace and an orthogonality condition.

We will derive Rayleigh–Ritz more systematically in Modules C–E.

For your HASC seminar, the key structural insight is that the algorithm seeks a spectral subspace and uses polynomial filtering, orthogonalization, and projected eigenvalue problems to approximate it.

Finite-precision computation introduces a further challenge: the algebraic invariance and orthogonality properties may no longer hold exactly after rounding or inexact operations.

That is where residuals, perturbation theory, conditioning, and numerical stability become essential.

Checkpoint 8

Explain why $p(B)$ preserves every eigenspace of B.

Then prove that if U is invariant under B, it is also invariant under $p(B)$.

Finally, explain why a decomposition into invariant subspaces can simplify a numerical eigenvalue problem.

13.12 The conceptual synthesis of Module A

We can now connect the main results.

Start with a linear map $T : V \to W$. The kernel $\ker T$ gives its invisible directions, and the image $\operatorname{im} T$ gives its attainable outputs.

- **Choose a complement.** Write $V = \ker T \oplus U$ to obtain an explicit decomposition.
- **Form the quotient.** Use $V/\ker T$ to identify all inputs with the same output.

Both constructions recover the effective domain of T.

$$\boxed{U \cong V/\ker T \cong \operatorname{im} T.}$$

The quotient-to-image isomorphism is canonical. The identification with a particular complement U depends on the chosen decomposition.

The ideas developed in this module therefore fit together as follows.

Construction	Main meaning	Central result
Kernel	Inputs annihilated by a map	T injective iff $\ker T = \{0\}$
Image	Outputs reachable by a map	T surjective iff $\operatorname{im} T = W$
Direct sum	Independent decomposition	$V = K \oplus U$
Quotient	Identification modulo a subspace	$\dim(V/S) = \dim V - \dim S$
First isomorphism theorem	Essential bijection underlying a map	$V/\ker T \cong \operatorname{im} T$
Exact sequence	Relationship between kernels and images	$\operatorname{im} f = \ker g$
Projection	Extracts one direct-sum component	$P^2 = P$
Invariant subspace	A subspace preserved by an operator	$T(U) \subseteq U$

The same constructions recur in the applications we explored:

- Singular linear systems have solution families that are cosets of the kernel.
- Graph Laplacians have null spaces reflecting constant-shift freedoms.
- Differential operators have kernels describing the nonuniqueness of solutions.
- Spectral algorithms exploit invariant subspaces and decompositions of the domain.

The significance of Module A is therefore not just that these concepts can be calculated. It is that they provide a common language for understanding many different mathematical problems.

13.13 Exercises — Test your understanding

The following exercises are designed to test more than mechanical computation.

Some ask for explicit calculations, while others require you to explain why a construction works or to recognize the same structure in another setting.

Try completing each exercise before consulting the solutions.

13.13.1 Part I — Linear algebraic foundations

Exercise 1 — Kernel, image, and rank

Let

$$M = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 1 \end{pmatrix}.$$

Find a basis of $\ker M$ and a basis of $\operatorname{im} M$.

Verify rank–nullity and explain the relationship between the nonpivot column and the kernel basis.

Exercise 2 — Direct-sum decomposition

For the matrix M in Exercise 1, let

$$U = \operatorname{span}\{e_1, e_3\}.$$

Prove that

$$\mathbb{R}^3 = \ker M \oplus U.$$

Find the unique decomposition

$$x = k + u, \qquad k \in \ker M, \quad u \in U,$$

for

$$x = (3, 4, 5)^T.$$

Exercise 3 — Dimensions of subspace sums

In $\mathbb{R}^3$, consider

$$U = \operatorname{span}\{e_1, e_2\}, \qquad W = \operatorname{span}\{e_1, e_3\}.$$

Find bases of $U \cap W$ and $U + W$.

Verify the subspace dimension formula.

Explain why this sum is not direct.

13.13.2 Part II — Quotients and structural theorems

Exercise 4 — Quotient-space coordinates

Let

$$S = \operatorname{span}\{(1, 1, 1)\} \subseteq \mathbb{R}^3.$$

Show that the cosets $e_1 + S$ and $e_2 + S$ form a basis of $\mathbb{R}^3/S$.

Express

$$(2, -1, 4) + S$$

in this quotient basis.

Exercise 5 — First isomorphism theorem

Consider

$$T : \mathbb{R}^3 \to \mathbb{R}^2, \qquad T(x, y, z) = (x + y, z).$$

Compute its kernel and image.

Construct the induced map

$$\widetilde{T} : \mathbb{R}^3 / \ker T \to \operatorname{im} T.$$

Prove directly that this map is an isomorphism.

Exercise 6 — Universal property of a quotient

Let

$$S = \operatorname{span}\{(1,1,0)\} \subseteq \mathbb{R}^3$$

and define

$$L : \mathbb{R}^3 \to \mathbb{R}, \qquad L(x_1, x_2, x_3) = x_1 - x_2 + 2x_3.$$

Show that L factors through $\mathbb{R}^3/S$.

Construct the induced functional on the quotient and explain why it is well-defined.

Exercise 7 — Projections

Consider

$$P = \begin{pmatrix} 1 & 2 \\ 0 & 0 \end{pmatrix}.$$

Show that $P^2 = P$.

Determine $\ker P$, $\operatorname{im} P$, and the resulting direct-sum decomposition of $\mathbb{R}^2$.

Is this an orthogonal projection with respect to the standard Euclidean inner product?

13.13.3 Part III — Applications

Exercise 8 — Graph Laplacian

Let

$$L = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{pmatrix}.$$

Find its kernel.

Show that the equation $Lp = b$ is solvable if and only if

$$b_1 + b_2 + b_3 = 0.$$

Solve

$$Lp = \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix}$$

and describe every solution.

Find the unique solution whose entries sum to zero.

Exercise 9 — A differential operator

Consider

$$D^2 : C^2([0,1]) \to C([0,1]), \qquad D^2 f = f''.$$

Find its kernel and explain why the operator is surjective.

Describe the quotient-space interpretation of the equation

$$f''(x) = 2.$$

Then impose

$$f(0) = 1, \qquad f'(0) = 3$$

and determine the unique solution.

Exercise 10 — Invariant subspaces and spectral filtering

Let

$$B = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 9 \end{pmatrix}.$$

Define

$$U = \operatorname{span}\{(1, -1, 0), (1, 1, 0)\}.$$

Prove that U is invariant under B.

Show that it remains invariant under any polynomial $p(B)$.

For

$$p(t) = \frac{9 - t}{8},$$

determine the effect of $p(B)$ on each of the three eigendirections.

Explain why this illustrates the principle behind polynomial spectral filtering.

13.14 Worked solutions

13.14.1 Solution 1

Solving $Mx = 0$ gives

$$x_3 = 0, \qquad x_1 = -2x_2.$$

Therefore,

$$\boxed{\ker M = \operatorname{span}\{(-2, 1, 0)\}.}$$

Columns 1 and 3 are independent and span $\mathbb{R}^2$, so

$$\boxed{\operatorname{im} M = \mathbb{R}^2.}$$

Thus

$$\operatorname{rank} M = 2, \qquad \operatorname{nullity} M = 1,$$

and $2 + 1 = 3$.

Column 2 equals twice column 1, giving the relation

$$-2a_1 + a_2 = 0,$$

which is encoded by the kernel vector $(-2, 1, 0)$.

13.14.2 Solution 2

We know

$$\ker M = \operatorname{span}\{(-2, 1, 0)\}.$$

Every nonzero vector in this kernel has a nonzero second coordinate, while every vector in U has second coordinate zero.

Thus

$$\ker M \cap U = \{0\}.$$

Since their dimensions add to 3,

$$\mathbb{R}^3 = \ker M \oplus U.$$

For $x = (3, 4, 5)$, the kernel component must have second coordinate 4, giving

$$k = (-8, 4, 0).$$

Then

$$u = x - k = (11, 0, 5).$$

Therefore,

$$\boxed{(3, 4, 5) = (-8, 4, 0) + (11, 0, 5).}$$

13.14.3 Solution 3

The intersection consists of vectors that belong to both coordinate planes:

$$\boxed{U \cap W = \operatorname{span}\{e_1\}.}$$

Their sum contains e_1, e_2, e_3, so

$$\boxed{U + W = \mathbb{R}^3.}$$

The dimension formula gives

$$3 = 2 + 2 - 1.$$

The sum is not direct because the intersection contains the nonzero vector e_1.

13.14.4 Solution 4

Since $\dim S = 1$,

$$\dim(\mathbb{R}^3/S) = 2.$$

The classes of e_1, e_2 are independent.

Indeed, if

$$ae_1 + be_2 \in S,$$

then

$$(a, b, 0) = c(1, 1, 1).$$

The third coordinate gives $c = 0$, hence $a = b = 0$.

Their two classes therefore form a basis of the quotient.

Now subtract $4(1, 1, 1)$ from $(2, -1, 4)$:

$$(2, -1, 4) - 4(1, 1, 1) = (-2, -5, 0).$$

Thus

$$\boxed{(2,-1,4)+S=-2(e_1+S)-5(e_2+S).}$$

13.14.5 Solution 5

We have

$$\ker T=\operatorname{span}\{(1,-1,0)\},\qquad \operatorname{im} T=\mathbb{R}^2.$$

The induced map is

$$\boxed{\widetilde{T}((x,y,z)+\ker T)=(x+y,z).}$$

It is well-defined because adding $t(1,-1,0)$ to a representative does not change $x+y$ or z.

It is linear because quotient addition and scalar multiplication are defined on representatives and T is linear.

It is injective because an equivalence class maps to $(0,0)$ precisely when its representatives belong to $\ker T$, which is the zero coset.

It is surjective because

$$\widetilde{T}((a,0,b)+\ker T)=(a,b)$$

for every $(a,b)\in\mathbb{R}^2$.

Hence it is an isomorphism.

13.14.6 Solution 6

For every $s=t(1,1,0)\in S$,

$$L(s)=t-t=0.$$

Therefore,

$$S\subseteq\ker L.$$

By the quotient universal property, L factors through $\mathbb{R}^3/S$.

The induced functional is

$$\boxed{\overline{L}((x_1, x_2, x_3) + S) = x_1 - x_2 + 2x_3.}$$

If two representatives differ by $t(1, 1, 0)$, the value of the expression is unchanged.

Thus the map is well-defined and satisfies

$$L = \overline{L} \circ q.$$

13.14.7 Solution 7

Direct multiplication gives

$$P^2 = \begin{pmatrix} 1 & 2 \\ 0 & 0 \end{pmatrix}^2 = \begin{pmatrix} 1 & 2 \\ 0 & 0 \end{pmatrix} = P.$$

The image is

$$\boxed{\operatorname{im} P = \operatorname{span}\{(1, 0)\}.}$$

To find the kernel, solve

$$x_1 + 2x_2 = 0.$$

Thus

$$\boxed{\ker P = \operatorname{span}\{(-2, 1)\}.}$$

Consequently,

$$\boxed{\mathbb{R}^2 = \operatorname{span}\{(1, 0)\} \oplus \operatorname{span}\{(-2, 1)\}.}$$

The two subspaces are not orthogonal, and

$$P \neq P^T.$$

Thus P is an oblique projection rather than an orthogonal projection.

13.14.8 Solution 8

The graph Laplacian satisfies

$$L\mathbf{1} = 0, \qquad \mathbf{1} = (1,1,1)^T.$$

Its kernel is exactly

$$\boxed{\ker L = \operatorname{span}\{\mathbf{1}\}.}$$

Since L is symmetric and has rank 2, its image is the two-dimensional subspace orthogonal to $\mathbf{1}$:

$$\operatorname{im} L = \{b : b_1 + b_2 + b_3 = 0\}.$$

Therefore, $Lp = b$ is solvable precisely when the entries of b sum to zero.

For

$$b = (1,-2,1)^T,$$

one solution is

$$p_0 = (1,0,1)^T,$$

since

$$Lp_0 = (1,-2,1)^T.$$

All solutions are

$$\boxed{p = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} + c \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}.}$$

To obtain a solution whose entries sum to zero, require

$$2 + 3c = 0.$$

Thus $c = -2/3$, giving

$$\boxed{p = \begin{pmatrix} 1/3 \\ -2/3 \\ 1/3 \end{pmatrix}.}$$

This is the unique solution orthogonal to the constant-vector kernel.

13.14.9 Solution 9

If $f'' = 0$, then f' is constant, so f is affine:

$$f(x) = a + bx.$$

Thus

$$\boxed{\ker D^2 = \operatorname{span}\{1, x\}.}$$

For any continuous g, the function

$$f(x) = \int_0^x (x - t)g(t)\,dt$$

satisfies $f'' = g$. Therefore the operator is surjective.

The quotient theorem gives

$$C^2([0,1])/\operatorname{span}\{1, x\} \cong C([0,1]).$$

For $f'' = 2$, all solutions have the form

$$f(x) = x^2 + a + bx.$$

The initial conditions give

$$a = 1, \qquad b = 3.$$

Hence the unique solution is

$$\boxed{f(x) = x^2 + 3x + 1.}$$

The initial conditions select one representative of the coset of solutions.

13.14.10 Solution 10

We have

$$B(1, -1, 0)^T = (1, -1, 0)^T$$

and

$$B(1, 1, 0)^T = 3(1, 1, 0)^T.$$

Therefore U is invariant under B.

Since $B(U) \subseteq U$, induction shows that

$$B^k(U) \subseteq U$$

for every nonnegative integer k.

Any polynomial $p(B)$ is a linear combination of these powers, so

$$\boxed{p(B)(U) \subseteq U.}$$

The third eigendirection is $\text{span}\{(0, 0, 1)\}$, with eigenvalue 9.

For $p(t) = (9 - t)/8$,

$$p(1) = 1, \qquad p(3) = \frac{3}{4}, \qquad p(9) = 0.$$

Thus the filter preserves the first eigendirection, attenuates the second, and annihilates the third.

Repeated application increasingly favors the first eigendirection relative to the second.

This illustrates the basic mechanism of polynomial spectral filtering.

13.15 Computational laboratory — MATLAB

This laboratory connects the theoretical constructions to numerical computation. It uses the same matrices from the module.

The main objective is to verify structural identities, rather than merely obtain numerical answers.

```
%% Module A: Structural Linear Algebra
clear; clc;

%% 1. Kernel, rank, and image
A = [1 2 -1 3;
     2 4  1 9;
     3 6  0 12];

R = rref(A);

% Exact kernel basis derived by hand
K = [-2 -4;
      1  0;
      0 -1;
      0  1];

% Basis of the image: ORIGINAL pivot columns
C = A(:, [1 3]);

disp('Rank of A:');
disp(rank(A));

disp('Check A*K = 0:');
disp(A*K);

disp('Rank of kernel basis:');
disp(rank(K));

disp('Rank of image basis:');
disp(rank(C));

%% 2. Direct-sum decomposition and projection
P = [0 -2 0 -4;
     0  1 0  0;
     0  0 0 -1;
```

```
     0  0 0  1];

x = [5; 2; 7; 3];

k = P*x;
u = x-k;

disp('Kernel component:');
disp(k);

disp('Complement component:');
disp(u);

fprintf('||x-k-u|| = %.3e\n', norm(x-k-u));
fprintf('||P^2-P||  = %.3e\n', norm(P*P-P));
fprintf('||A*P||    = %.3e\n', norm(A*P));

%% 3. Discrete gradient and graph Laplacian
D = [-1  1  0;
      0 -1  1];

L = D.'*D;

disp('Graph Laplacian:');
disp(L);

disp('Kernel check for constant vector:');
disp(L*ones(3,1));

%% 4. Spectral filtering
B = [2 1 0;
     1 2 0;
     0 0 9];

q1 = [1; -1; 0]/sqrt(2);
q2 = [1;  1; 0]/sqrt(2);
q3 = [0;  0; 1];

Q = [q1 q2 q3];

x = q1 + q2 + q3;
```

```
% Polynomial p(t) = (9-t)/8
pB = (9*eye(3)-B)/8;
y = pB*x;

disp('Filtered spectral coefficients:');
disp(Q.'*y);

% Expected: [1; 0.75; 0]
```

13.15.1 Questions to investigate experimentally

Try modifying the examples and checking whether the relevant identities still hold.

In particular, experiment with changing the matrix A, constructing a different complement of its kernel, and comparing the resulting projections.

For the graph example, change the number of vertices and examine how the dimension of the kernel depends on the number of connected components.

For the spectral example, try different polynomials p and observe how $p(B)$ scales different eigendirections.

When using floating-point matrices, distinguish exact algebraic statements from numerical approximations. For example, an exact zero matrix may appear numerically as a matrix with very small nonzero entries.

13.16 Final mastery assessment

The module should not be considered complete merely because you've read the material.

A stronger test is whether you can reconstruct its main ideas without referring to the notes.

- ☐ **Definitions.** Define a kernel, image, direct sum, quotient space, and invariant subspace precisely.
- ☐ **Proofs.** Prove rank–nullity, the direct-sum criterion, and the first isomorphism theorem.
- ☐ **Computation.** Compute kernels, images, complements, projections, and quotient coordinates.
- ☐ **Geometric intuition.** Explain what it means for a transformation to lose information or identify inputs.
- ☐ **Structural understanding.** Relate quotients, complements, exact sequences, and projections.
- ☐ **Universal understanding.** Explain the universal properties of quotients and direct sums.

- ☐ **Applications.** Recognize the same structure in singular systems, graph Laplacians, and differential operators.
- ☐ **Numerical connections.** Explain how invariant subspaces motivate spectral decomposition and polynomial filtering.

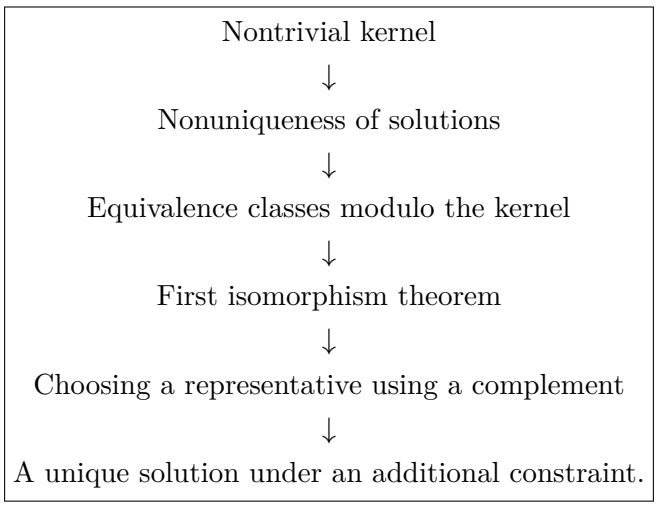
13.16.1 How I recommend studying the module

Treat each session as an independent study block, but revisit the running matrix throughout the module.

For each substantial topic, use this cycle:

1. **Read the definition and theorem.** Identify the assumptions and understand what the result claims.
2. **Reconstruct the proof.** First explain the logical argument verbally, then write it mathematically.
3. **Work through a concrete example.** Perform the computation independently rather than copying the worked solution.
4. **Identify the structural connection.** Ask which previous concepts the theorem depends on and whether it admits a more general interpretation.
5. **Connect it to an application.** Explain which feature of a numerical method, differential equation, or scientific model the construction describes.

For example, after studying quotient spaces, you should be able to connect the following without consulting the notes:



That sequence is a useful synthesis of Module A.

13.16.2 Transition to Module B

The next major topic is **duality and linear measurements**.

So far, we have studied how a linear transformation acts on vectors.

Module B will ask how it acts on linear functionals.

That leads to the dual map

$$T^\vee : W^* \to V^*, \qquad T^\vee(f) = f \circ T,$$

and to the annihilator relationships

$$\boxed{\begin{aligned} \operatorname{im} T^\vee &= (\ker T)^0, \\ \ker T^\vee &= (\operatorname{im} T)^0. \end{aligned}}$$

These are the algebraic predecessors of the orthogonality identities used in numerical linear algebra.

They also lead naturally to Riesz representation and adjoint operators in Module C.

The most important result to retain from Module A is therefore not one formula, but an overall principle:

A linear map can be understood by investigating what it annihilates, what it reaches, and how its domain can be decomposed or quotiented to reveal an underlying isomorphism.

Once that principle is familiar, many constructions in algebra, numerical analysis, functional analysis, and differential equations become instances of the same mathematical pattern.

14 Module B — Duality and Linear Measurements

Dual spaces, dual bases, dual maps, transpose, annihilators, quotient duality, and exact sequences

A self-contained study module connecting structural linear algebra to functional analysis, numerical methods, and abstract algebra.

Level	Advanced undergraduate
Prerequisites	Module A: linear maps, kernels, images, bases, quotients, direct sums, and the first isomorphism theorem
Suggested duration	8 sessions, approximately 14–18 hours
Next module	Module C: Inner-product geometry, Riesz representation, adjoints, projections, and SPD matrices

14.1 Introduction — A new perspective on linear algebra

In Module A, we studied a linear transformation

$$T : V \longrightarrow W$$

by investigating what it does to vectors.

We asked which vectors are annihilated, which outputs can be reached, and how the domain can be decomposed or quotiented to reveal an underlying isomorphism.

Module B examines the same transformation from another perspective:

Instead of studying only vectors, we study the linear measurements that can be performed on those vectors.

This leads to the dual space V^*, whose elements are linear functionals

$$f : V \longrightarrow \mathbb{F}.$$

A vector $v \in V$ is an object that can be measured. A functional $f \in V^*$ is a linear rule for measuring it.

The fundamental operation connecting them is

$$\boxed{(f, v) \longmapsto f(v).}$$

Once we introduce these measurements, several deeper connections emerge.

A linear map $T : V \to W$ induces a map in the reverse direction,

$$T^\vee : W^* \longrightarrow V^*,$$

which pulls measurements backward through T.

The transpose matrix turns out to be the coordinate representation of this dual map.

The annihilator of a subspace describes all measurements that vanish on that subspace.

And two remarkable relationships connect the dual map with the kernel and image of the original transformation:

$$\boxed{\begin{aligned} \operatorname{im} T^\vee &= (\ker T)^0, \\ \ker T^\vee &= (\operatorname{im} T)^0. \end{aligned}}$$

These are the central theorems of Module B.

Later, the Riesz representation theorem will allow us to turn algebraic duality into inner-product geometry, producing the familiar relationships between kernels, adjoints, and orthogonal complements.

14.2 How Module B fits into the curriculum

Module A — Structure of linear maps

Kernels, images, direct sums, quotients, isomorphisms

Module B — Duality

Linear measurements, dual bases, transposes, annihilators, exactness

Module C — Inner-product geometry

Riesz representation, adjoints, orthogonality, SPD matrices

Modules D–E — Spectral and numerical methods

Invariant subspaces, Rayleigh–Ritz, eigenvalue algorithms, Chebyshev filtering

A fundamental distinction will guide the module:

Algebraic duality requires only vector-space structure. Orthogonality and adjoint operators require additional inner-product structure.

We will develop algebraic duality independently before introducing the geometric identifications associated with inner products.

14.3 Six dimensions of understanding

As agreed in the overall curriculum, we will develop the material from six complementary perspectives.

Perspective	Goal in Module B
Rigorous	Define dual spaces and prove their central properties
Intuitive and geometric	Interpret covectors as linear measurements and families of level sets
Computational	Construct dual bases, compute dual maps, annihilators, and transpose matrices
Structural	Understand the relationship between duality, kernels, images, and quotients
Universal	Explain contravariance, double duality, and dual exact sequences
Applied	Connect duality to linear constraints, graph operators, variational formulations, and numerical eigenvalue methods

14.4 Study roadmap

Session	Topic	Main result
1	Linear functionals and their geometry	$V^* = \mathrm{Hom}(V, \mathbb{F})$
2	Dual bases and coordinates	$\varepsilon^i(v_j) = \delta_{ij}$
3	Dual maps and transpose	$T^\vee(f) = f \circ T$
4	Annihilators and fundamental subspaces	$\operatorname{im} T^\vee = (\ker T)^0$
5	Quotient duality and exact sequences	$(V/S)^* \cong S^0$

Session	Topic	Main result
6	Double duals and categorical structure	$V \cong V^{**}$ in finite dimensions
7	Applications	Graph operators, weak formulations, residuals, invariant subspaces
8	Synthesis, exercises, and computation	Reconstruct the theory independently

Throughout the module, V^* denotes the **algebraic dual**, and $T^\vee$ denotes the algebraic dual map. We reserve T^* for the inner-product adjoint introduced in Module C.

Unless otherwise indicated, the spaces are finite-dimensional over $\mathbb{R}$. Most algebraic results also hold over $\mathbb{C}$ or any other field.

14.5 Session 1 — Linear functionals and the geometry of dual spaces

14.5.1 What is a linear functional?

Let V be a vector space over $\mathbb{F}$.

A **linear functional** is a linear transformation

$$f : V \longrightarrow \mathbb{F}.$$

This means that

$$f(\alpha v + \beta w) = \alpha f(v) + \beta f(w)$$

for all $v, w \in V$ and scalars α, β.

A functional differs from a general linear transformation only in that its codomain is the scalar field.

For example, consider

$$f : \mathbb{R}^3 \to \mathbb{R}$$

defined by

$$f(x_1, x_2, x_3) = 2x_1 - 3x_2 + 5x_3.$$

This is a linear functional.

It takes a vector containing three coordinates and returns one scalar.

An important interpretation is that f represents a **linear measurement**.

We can think of it as a measuring instrument that extracts one scalar quantity from an input vector.

14.5.2 Why do linear functionals deserve their own space?

The collection of all linear functionals on V is denoted

$$\boxed{V^* = \mathrm{Hom}_{\mathbb{F}}(V, \mathbb{F}).}$$

It is called the **dual space** of V.

We can add functionals by defining

$$(f + g)(v) = f(v) + g(v),$$

and multiply them by scalars:

$$(\alpha f)(v) = \alpha f(v).$$

Both operations preserve linearity.

The zero element is the zero functional

$$0_{V^*}(v) = 0$$

for every $v \in V$.

Therefore V^* is itself a vector space.

There is a conceptual change here.

The elements of V are vectors.

The elements of V^* are functions.

Nevertheless, both collections are vector spaces because they satisfy the same algebraic axioms.

This is one reason abstract linear algebra is more general than the geometry of arrows in Euclidean space.

14.5.3 The natural pairing

We have a canonical operation

$$V^* \times V \to \mathbb{F}$$

defined by

$$(f, v) \longmapsto f(v).$$

This is called the **evaluation pairing** or **natural pairing**.

We will occasionally write

$$\boxed{\langle f, v \rangle_{\text{eval}} = f(v).}$$

It is bilinear:

$$\langle \alpha f + \beta g, v \rangle_{\text{eval}} = \alpha \langle f, v \rangle_{\text{eval}} + \beta \langle g, v \rangle_{\text{eval}},$$

and

$$\langle f, \alpha v + \beta w \rangle_{\text{eval}} = \alpha \langle f, v \rangle_{\text{eval}} + \beta \langle f, w \rangle_{\text{eval}}.$$

Notice that this is **not an inner product** in the usual sense.

An inner product takes two vectors from the same space and returns a scalar.

The evaluation pairing takes a functional from V^* and a vector from V.

It requires neither lengths nor angles nor any geometric metric.

This pairing is the most fundamental operation in duality.

14.5.4 Geometric meaning: a covector as a family of parallel level sets

Let

$$V = \mathbb{R}^2$$

and consider

$$f(x, y) = 2x + y.$$

Instead of representing f by a row of coefficients, let's ask what the equation

$$f(x, y) = c$$

describes geometrically.

It becomes

$$2x + y = c,$$

or

$$y = -2x + c.$$

For each scalar c, this is a straight line.

Different values of c produce parallel lines.

In Figure 14.1, moving along a level line leaves the measurement unchanged. Moving from one level line to another changes its value. The arrow shows the Euclidean representing vector, which depends on the choice of inner product.

This gives a more intrinsic geometric interpretation of a covector.

A nonzero covector on a two-dimensional real vector space can be visualized through its family of parallel level lines, with scalar labels indicating the measurement.

In three dimensions, the level sets are parallel planes.

In n dimensions, they are affine hyperplanes of dimension $n - 1$.

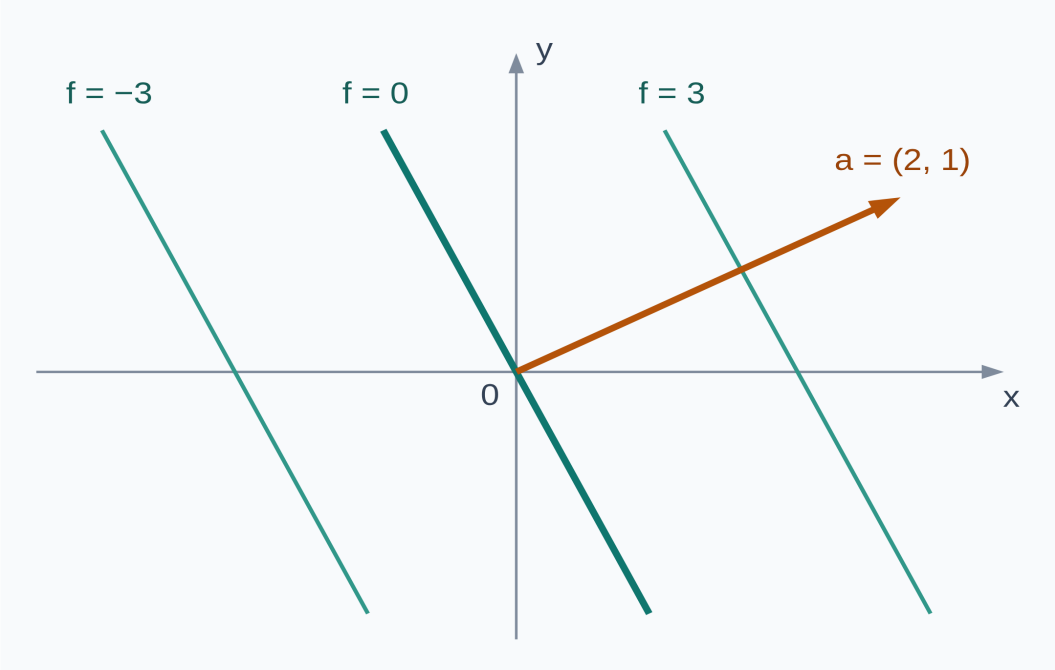


Figure 14.1: Three level sets of f(x, y) = 2x + y. The central line is the kernel. The orange arrow represents f under the standard Euclidean dot product.

The kernel consists of invisible directions

The kernel of f is

$$\ker f = \{v \in V : f(v) = 0\}.$$

For our example,

$$\ker f = \{(x, y) : 2x + y = 0\} = \text{span}\{(1, -2)\}.$$

Now let $k \in \ker f$. Then

$$f(v + k) = f(v) + f(k) = f(v).$$

Thus adding a kernel vector leaves the measurement unchanged.

This leads to a useful interpretation:

$$\boxed{\ker f = \text{the directions invisible to the measurement } f.}$$

For a nonzero functional on an n-dimensional space, rank–nullity gives

$$\dim \ker f = n - 1.$$

So a single independent linear measurement removes one degree of freedom.

The kernel contains all the remaining directions that the measurement cannot distinguish.

14.5.5 Is a covector just an ordinary vector?

In Euclidean space, we can write

$$f(x, y) = 2x + y$$

as

$$f(x, y) = \begin{pmatrix} 2 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}.$$

Using the standard dot product, this equals

$$f(v) = \langle v, a \rangle$$

for

$$a = (2, 1).$$

This makes it tempting to identify the functional f with the vector a.

But that identification uses an inner product.

Without an inner product, there is no distinguished way to convert a functional into a vector belonging to V.

The covector remains a perfectly well-defined algebraic object even when no lengths or angles have been introduced.

This is the distinction that the Riesz representation theorem will clarify in Module C.

14.5.6 Why the evaluation pairing is nondegenerate

The pairing has two important properties.

If

$$f(v) = 0$$

for every $v \in V$, then f is the zero functional.

Conversely, if

$$f(v) = 0$$

for every $f \in V^*$, then $v = 0$.

To prove the second claim, suppose $v \neq 0$.

Extend the one-element independent set $\{v\}$ to a basis of V.

Define a functional that takes the value 1 on v and 0 on the remaining basis vectors, then extend it linearly.

This functional satisfies $f(v) = 1$.

Therefore every nonzero vector can be detected by some linear functional.

This is called **nondegeneracy of the evaluation pairing**.

It tells us that the family of all linear measurements distinguishes every vector in V, even though an individual measurement generally cannot.

Checkpoint 1

Prove that every nonzero functional $f : \mathbb{R}^n \to \mathbb{R}$ is surjective.

Then explain geometrically why its kernel is an $(n-1)$-dimensional hyperplane.

Finally, distinguish carefully between the functional f, its kernel, and a vector representing f through an inner product.

14.6 Session 2 — Dual bases and coordinate representations

The dual space may initially appear quite abstract because its elements are functions rather than ordinary coordinate vectors.

Dual bases provide a systematic way to describe those functions computationally.

14.6.1 Every basis defines coordinate measurements

Let

$$\mathcal{B} = (v_1, \dots, v_n)$$

be a basis of V.

Every vector has a unique expansion

$$v = a_1 v_1 + \cdots + a_n v_n.$$

We can define a functional for each coordinate.

Let

$$\varepsilon^i(v) = a_i.$$

In other words, ε^i extracts the i-th coordinate of v relative to the basis $\mathcal{B}$.

Each ε^i is linear because coordinates depend linearly on vectors.

By construction,

$$\boxed{\varepsilon^i(v_j) = \delta_{ij},}$$

where

$$\delta_{ij} = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases}$$

The collection

$$\boxed{\mathcal{B}^* = (\varepsilon^1, \dots, \varepsilon^n)}$$

is called the **dual basis**.

Interpretation

The basis vectors are the building blocks of vectors.

The dual basis functionals are the measuring devices that recover the coefficients of those building blocks.

The relation

$$\varepsilon^i(v_j) = \delta_{ij}$$

means that the i-th coordinate measurement recognizes the i-th basis vector and ignores all others.

14.6.2 The dual basis theorem

Theorem. If $(v_1, \dots, v_n)$ is a basis of V, then its dual functionals $(\varepsilon^1, \dots, \varepsilon^n)$ form a basis of V^*.

Moreover, every $f \in V^*$ has the unique expansion

$$\boxed{f = \sum_{i=1}^{n} f(v_i)\varepsilon^i.}$$

Proof

For any $v \in V$, write

$$v = \sum_{j=1}^{n} a_j v_j.$$

Then

$$f(v) = \sum_{j=1}^{n} a_j f(v_j).$$

But

$$a_j = \varepsilon^j(v).$$

Therefore,

$$f(v) = \sum_{j=1}^{n} f(v_j)\varepsilon^j(v).$$

Since this holds for every v,

$$f = \sum_{j=1}^{n} f(v_j)\varepsilon^j.$$

Thus the ε^j span V^*.

For independence, suppose

$$\sum_{i=1}^{n} c_i\varepsilon^i = 0.$$

Evaluate at v_j:

$$0 = \sum_{i=1}^{n} c_i\varepsilon^i(v_j) = c_j.$$

As j was arbitrary, every coefficient is zero.

Hence the dual functionals are linearly independent and form a basis.

A consequence is

$$\boxed{\dim V^* = \dim V}$$

when V is finite-dimensional.

14.6.3 A nonstandard basis in $\mathbb{R}^2$

Consider

$$v_1 = (1, 0), \qquad v_2 = (1, 1).$$

These form a basis of $\mathbb{R}^2$.

We want to construct the corresponding dual basis.

Take an arbitrary vector

$$v = (x, y).$$

Write

$$(x, y) = av_1 + bv_2.$$

Then

$$(x, y) = (a + b, b).$$

Thus

$$b = y, \qquad a = x - y.$$

The coordinate-extraction functionals are

$$\boxed{\begin{aligned} \varepsilon^1(x, y) &= x - y, \\ \varepsilon^2(x, y) &= y. \end{aligned}}$$

Let's verify the defining relation:

	$v_1 = (1, 0)$	$v_2 = (1, 1)$
ε^1	1	0
ε^2	0	1

Thus $(\varepsilon^1, \varepsilon^2)$ is the dual basis.

Notice that neither functional has to be represented by the corresponding original basis vector.

The basis vectors and dual basis functionals play complementary roles.

14.6.4 How to calculate a dual basis using matrix inversion

Let

$$B = (v_1 \ \cdots \ v_n)$$

be the matrix whose columns are the basis vectors, expressed in standard coordinates.

Suppose x has coordinate column a in this basis.

Then

$$x = Ba.$$

Therefore,

$$a = B^{-1}x.$$

The i-th coordinate is the i-th row of B^{-1} applied to x.

Hence:

$$\boxed{\text{Rows of } B^{-1} = \text{coordinate rows of the dual basis functionals.}}$$

If we represent functional coefficient arrays as columns instead, the dual basis representatives are the columns of

$$B^{-T}.$$

For our example,

$$B = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \qquad B^{-1} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}.$$

Its rows represent

$$\varepsilon^1(x, y) = x - y, \qquad \varepsilon^2(x, y) = y.$$

This is a general computational method for constructing a dual basis.

14.6.5 Why vectors and covectors transform differently

Now suppose we change coordinates.

Let the original basis be $\mathcal{B}$, and let $\mathcal{C}$ be a new basis.

Let S be the change-of-coordinate matrix satisfying

$$[x]_{\mathcal{B}} = S[x]_{\mathcal{C}}.$$

Then vector coordinates transform according to

$$\boxed{[x]_{\mathcal{C}} = S^{-1}[x]_{\mathcal{B}}.}$$

What happens to a functional?

Suppose its coefficient column in the original dual basis is $c_{\mathcal{B}}$, so that

$$f(x) = c_{\mathcal{B}}^T[x]_{\mathcal{B}}.$$

Substituting the coordinate transformation gives

$$f(x) = c_{\mathcal{B}}^T S[x]_{\mathcal{C}}.$$

Therefore,

$$f(x) = (S^T c_{\mathcal{B}})^T[x]_{\mathcal{C}}.$$

So the new coefficient column is

$$\boxed{c_{\mathcal{C}} = S^T c_{\mathcal{B}}.}$$

Compare the two rules:

$$\boxed{\begin{aligned}[x]_{\mathcal{C}} &= S^{-1}[x]_{\mathcal{B}},\\ c_{\mathcal{C}} &= S^{T}c_{\mathcal{B}}.\end{aligned}}$$

Vectors and covectors have complementary transformation laws.

This is the mathematical origin of the terminology *contravariant* and *covariant* components, although conventions depend on which basis-change matrix is being specified.

Worked example

In standard coordinates, let

$$f(x,y) = 2x + y.$$

Its coefficient column is

$$c_{\text{standard}} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}.$$

Now use the basis

$$v_1 = (1,0), \qquad v_2 = (1,1),$$

with basis matrix

$$B = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

The functional coefficients in the new dual basis are

$$c_{\text{new}} = B^{T}c_{\text{standard}} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} 2 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \end{pmatrix}.$$

This matches the dual-basis theorem:

$$f(v_1) = 2, \qquad f(v_2) = 3.$$

Indeed,

$$\boxed{f = 2\varepsilon^1 + 3\varepsilon^2.}$$

The functional itself has not changed.

Only its coordinate representation has changed.

Why this distinction matters

In numerical linear algebra, vectors often represent states, while covectors represent residual functionals, constraints, or derivatives of scalar-valued quantities.

Coordinate changes must preserve the scalar evaluation $f(x)$.

The complementary coordinate transformations of vectors and covectors are what guarantee that invariance.

Later, an inner product will provide a mechanism for translating between the two kinds of coordinates.

That mechanism is precisely what underlies metric matrices, Gram matrices, and Riesz representation.

Checkpoint 2

Let

$$v_1 = (1, 1), \qquad v_2 = (1, -1).$$

Construct the dual basis explicitly.

Then calculate the coordinates of

$$f(x, y) = 3x + y$$

in that dual basis, both by evaluating $f(v_1), f(v_2)$ and by using the basis-change matrix.

14.7 Session 3 — Dual maps and the meaning of transpose

We now arrive at a central idea of the module.

A linear transformation does not merely map vectors from one space into another. It also induces a transformation between the corresponding dual spaces.

Surprisingly, that induced transformation goes in the opposite direction.

14.7.1 Pulling a measurement backward

Let

$$T : V \to W$$

be linear.

Suppose $g \in W^*$. Thus,

$$g : W \to \mathbb{F}$$

is a linear measurement on the output space.

We can turn it into a measurement on the input space by first applying T:

$$v \longmapsto T(v) \longmapsto g(T(v)).$$

The composition

$$g \circ T : V \to \mathbb{F}$$

is a linear functional on V. The two routes in Figure 14.2 define the same scalar measurement.

This motivates the definition of the **dual map**:

$$\boxed{\begin{aligned} T^\vee : W^* &\longrightarrow V^*, \\ g &\longmapsto g \circ T. \end{aligned}}$$

Notice the reversal:

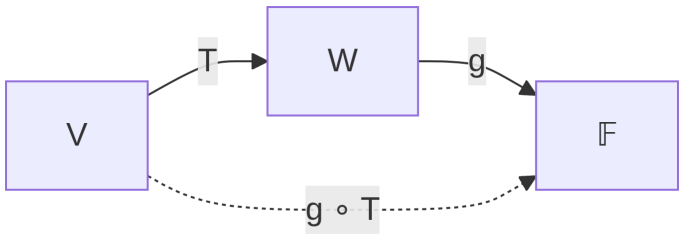


Figure 14.2: Measuring after applying T gives the same scalar as applying the pulled-back measurement directly.

$$T : V \to W, \qquad T^\vee : W^* \to V^*.$$

Vectors are pushed forward by T; linear measurements are pulled backward by $T^\vee$.

The dual map is linear. Indeed, for $g, h \in W^*$,

$$\begin{aligned} T^\vee(\alpha g + \beta h) &= (\alpha g + \beta h) \circ T \\ &= \alpha(g \circ T) + \beta(h \circ T) \\ &= \alpha T^\vee(g) + \beta T^\vee(h). \end{aligned}$$

14.7.2 Deriving the transpose matrix

Let

$$T : \mathbb{R}^n \to \mathbb{R}^m$$

be represented by $A \in \mathbb{R}^{m \times n}$.

Write a functional $g \in (\mathbb{R}^m)^*$ as

$$g(y) = c^T y, \qquad c \in \mathbb{R}^m.$$

Then

$$\begin{aligned} T^\vee(g)(x) &= g(Ax) \\ &= c^T A x \\ &= (A^T c)^T x. \end{aligned}$$

Thus the coefficient column of the pulled-back functional is $A^T c$.

We conclude:

$$\boxed{[T^\vee] = A^T}$$

when the domain and codomain dual bases correspond to the bases used to represent T.

This is the intrinsic meaning of the transpose.

The transpose is the matrix representation of the operation that pulls linear measurements backward through a linear transformation.

It is not originally a geometric operation involving perpendicular vectors.

Those geometric interpretations arise only after introducing inner products.

For complex vector spaces, the algebraic dual map is still represented by the ordinary transpose A^T, without complex conjugation.

The conjugate transpose A^H belongs to the later theory of Hilbert-space adjoints.

14.7.3 A concrete example

Consider

$$A = \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}.$$

It represents

$$T(x_1, x_2) = (x_1 + 2x_2, x_2).$$

Suppose we want to measure the first output coordinate:

$$g(y_1, y_2) = y_1.$$

Then

$$\begin{aligned} T^\vee(g)(x_1, x_2) &= g(T(x_1, x_2)) \\ &= x_1 + 2x_2. \end{aligned}$$

So the output measurement y_1 is equivalent to the input measurement $x_1 + 2x_2$.

Let's verify using the transpose.

The coefficient column of g is

$$c = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Then

$$A^T c = \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}.$$

The result represents precisely

$$x \longmapsto x_1 + 2x_2.$$

The transpose rule is therefore a consequence of composition, not an independent formula to memorize.

14.7.4 The dual map of our running matrix

Recall from Module A:

$$A = \begin{pmatrix} 1 & 2 & -1 & 3 \\ 2 & 4 & 1 & 9 \\ 3 & 6 & 0 & 12 \end{pmatrix} : \mathbb{R}^4 \to \mathbb{R}^3.$$

Its algebraic dual map is

$$A^\vee : (\mathbb{R}^3)^* \to (\mathbb{R}^4)^*.$$

In standard dual bases, it is represented by

$$A^T = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ -1 & 1 & 0 \\ 3 & 9 & 12 \end{pmatrix}.$$

Take the output functional

$$g(y_1, y_2, y_3) = y_1 - 2y_2 + y_3.$$

Its coefficient column is

$$c = (1, -2, 1)^T.$$

The pulled-back functional has coefficients

$$A^T c = \begin{pmatrix} 0 \\ 0 \\ -3 \\ -3 \end{pmatrix}.$$

Therefore,

$$\boxed{A^\vee(g)(x_1, x_2, x_3, x_4) = -3x_3 - 3x_4.}$$

We can also verify this directly.

Since

$$Ax = \begin{pmatrix} x_1 + 2x_2 - x_3 + 3x_4 \\ 2x_1 + 4x_2 + x_3 + 9x_4 \\ 3x_1 + 6x_2 + 12x_4 \end{pmatrix},$$

substituting into g gives exactly $-3x_3 - 3x_4$.

The dual map converts measurements of outputs into equivalent measurements of inputs.

14.7.5 Composition reverses order

Suppose

$$V \xrightarrow{T} W \xrightarrow{S} Z.$$

Their composition is

$$S \circ T : V \to Z.$$

What is its dual map?

For $h \in Z^*$,

$$\begin{aligned}(S \circ T)^\vee(h) &= h \circ (S \circ T) \\ &= (h \circ S) \circ T \\ &= T^\vee(S^\vee(h)).\end{aligned}$$

Therefore,

$$\boxed{(S \circ T)^\vee = T^\vee \circ S^\vee.}$$

The order of composition is reversed.

In coordinates, if T has matrix A and S has matrix B, this becomes

$$\boxed{(BA)^T = A^T B^T.}$$

So the familiar transpose identity is an expression of a more fundamental structural principle.

Categorical interpretation

The assignment

$$V \longmapsto V^*, \qquad T \longmapsto T^\vee$$

is a **contravariant functor**.

For now, that terminology means that taking duals preserves identity maps but reverses the order of composition:

$$(\mathrm{id}_V)^\vee = \mathrm{id}_{V^*},$$

$$(S \circ T)^\vee = T^\vee \circ S^\vee.$$

This is our first substantial encounter with a functorial construction.

It is particularly useful because the same pattern appears in other settings, including pullbacks of functions, differential forms, and mappings of geometric and algebraic structures.

Checkpoint 3

Derive

$$(BA)^T = A^T B^T$$

by considering the dual map of a composition, rather than by calculating individual matrix entries.

Then explain why this works over $\mathbb{C}$ without complex conjugation.

14.8 Session 4 — Annihilators and the four fundamental subspaces

Module A introduced kernels and images.

Duality allows us to study those spaces through the functionals that vanish on them.

This leads to the notion of an annihilator.

14.8.1 The annihilator of a subspace

Let $S \subseteq V$ be a subspace.

Define

$$\boxed{S^0 = \{f \in V^* : f(s) = 0 \text{ for every } s \in S\}.}$$

This is called the **annihilator** of S.

It is a subspace of V^*.

The annihilator contains all linear measurements that are insensitive to every vector in S.

Example

Let

$$S = \text{span}\{(1,1,0)\} \subseteq \mathbb{R}^3.$$

An arbitrary functional has the form

$$f(x_1, x_2, x_3) = ax_1 + bx_2 + cx_3.$$

For f to annihilate S, it must satisfy

$$f(1,1,0) = a + b = 0.$$

Therefore,

$$b = -a.$$

Every functional in the annihilator has the form

$$f(x) = a(x_1 - x_2) + cx_3.$$

Thus

$$\boxed{S^0 = \text{span}\{\varepsilon^1 - \varepsilon^2,\ \varepsilon^3\},}$$

where the ε^i are the standard coordinate functionals.

Notice that S^0 is a space of functionals, not a subspace of ordinary vectors unless we deliberately identify functionals with coordinate arrays.

14.8.2 The dimension of an annihilator

Theorem. If V is finite-dimensional and $S \subseteq V$, then

$$\boxed{\dim S^0 = \dim V - \dim S.}$$

Proof

Suppose $\dim V = n$ and $\dim S = k$.

Choose a basis

$$(s_1, \dots, s_k)$$

of S and extend it to a basis

$$(s_1, \dots, s_k, u_1, \dots, u_{n-k})$$

of V.

Let

$$(\varepsilon^1, \dots, \varepsilon^n)$$

be the corresponding dual basis.

Any functional $f \in V^*$ can be written uniquely as

$$f = \sum_{i=1}^{n} a_i \varepsilon^i.$$

It annihilates S precisely when

$$f(s_j) = 0$$

for $j = 1, \dots, k$.

But

$$f(s_j) = a_j.$$

Therefore the first k coefficients must vanish.

Hence

$$S^0 = \operatorname{span}\{\varepsilon^{k+1}, \dots, \varepsilon^n\},$$

so

$$\dim S^0 = n - k.$$

This is a dual counterpart of the quotient dimension formula.

14.8.3 Fundamental properties of annihilators

The annihilator reverses inclusion.

If

$$S \subseteq U,$$

then every functional annihilating U also annihilates S.

Thus

$$\boxed{U^0 \subseteq S^0.}$$

This is an example of **order reversal**.

We also have

$$\boxed{(S + U)^0 = S^0 \cap U^0.}$$

Indeed, a functional vanishes on every sum $s+u$ precisely when it vanishes on both subspaces.

Another useful identity in finite dimensions is

$$\boxed{(S \cap U)^0 = S^0 + U^0.}$$

One way to prove it is to apply the first identity, together with the finite-dimensional double-annihilator theorem developed in Session 6. It can also be proved by a dimension argument.

These relationships mirror the behavior of sums and intersections, but in the reverse direction.

The annihilator turns a sum into an intersection and an intersection into a sum.

14.8.4 First fundamental theorem of duality

Let

$$T : V \to W$$

be linear.

Then

$$\boxed{\operatorname{im} T^\vee = (\ker T)^0}$$

for finite-dimensional vector spaces.

In words:

The image of the dual map consists precisely of the linear functionals on V that vanish on the kernel of T.

Let's prove this carefully.

Inclusion 1

Suppose

$$f \in \operatorname{im} T^\vee.$$

Then $f = T^\vee(g)$ for some $g \in W^*$.

Thus

$$f = g \circ T.$$

If $v \in \ker T$, then

$$f(v) = g(Tv) = g(0) = 0.$$

Therefore,

$$f \in (\ker T)^0.$$

We have shown

$$\operatorname{im} T^\vee \subseteq (\ker T)^0.$$

Inclusion 2

Now let

$$f \in (\ker T)^0.$$

We want to express f as $g \circ T$ for some $g \in W^*$.

First define a functional

$$g_0 : \operatorname{im} T \to \mathbb{F}$$

by

$$\boxed{g_0(Tv) = f(v).}$$

We must show this is well-defined.

If $Tv = Tw$, then

$$v - w \in \ker T.$$

Since f vanishes on $\ker T$,

$$f(v - w) = 0,$$

so $f(v) = f(w)$.

Thus g_0 is independent of the chosen preimage.

It is also linear.

Now extend g_0 to a functional $g \in W^*$.

In finite dimensions, this is always possible: extend a basis of $\operatorname{im} T$ to a basis of W, and assign arbitrary values to the additional basis vectors.

For every $v \in V$,

$$g(Tv) = g_0(Tv) = f(v).$$

Therefore,

$$f = g \circ T = T^{\vee}(g).$$

Hence

$$(\ker T)^0 \subseteq \operatorname{im} T^{\vee}.$$

Combining both inclusions,

$$\boxed{\operatorname{im} T^{\vee} = (\ker T)^0.}$$

What does the theorem mean?

Suppose T loses some information about its input.

A functional f can be reconstructed from the output Tv only if it does not depend on the information that T loses.

That is, f must vanish on $\ker T$.

The theorem says this condition is also sufficient.

Every input measurement that ignores the kernel can be performed entirely on the output.

This is a dual version of the first isomorphism theorem.

14.8.5 Second fundamental theorem of duality

We also have

$$\boxed{\ker T^{\vee} = (\operatorname{im} T)^0.}$$

This one follows immediately from the definitions.

Let $g \in W^*$.

Then

$$\begin{aligned} g \in \ker T^\vee &\iff T^\vee(g) = 0 \\ &\iff g(Tv) = 0 \quad \forall v \in V \\ &\iff g(w) = 0 \quad \forall w \in \operatorname{im} T \\ &\iff g \in (\operatorname{im} T)^0. \end{aligned}$$

Therefore,

$$\boxed{\begin{aligned} \operatorname{im} T^\vee &= (\ker T)^0, \\ \ker T^\vee &= (\operatorname{im} T)^0. \end{aligned}}$$

These identities reveal a symmetry between kernels and images under duality.

14.8.6 The four fundamental subspaces without inner products

We can now reconstruct the four fundamental subspaces in a purely algebraic way.

For a matrix

$$A : \mathbb{R}^n \to \mathbb{R}^m,$$

we have the original kernel and image:

$$\ker A \subseteq \mathbb{R}^n, \qquad \operatorname{im} A \subseteq \mathbb{R}^m.$$

The dual transformation has

$$\ker A^\vee \subseteq (\mathbb{R}^m)^*, \qquad \operatorname{im} A^\vee \subseteq (\mathbb{R}^n)^*.$$

Under standard dual-coordinate representations, these last two spaces are represented by the null space and image of A^T.

This produces the familiar four subspaces.

Subspace	Algebraic meaning	Dimension when $\operatorname{rank} A = r$
$\ker A$	Invisible input directions	$n - r$
$\operatorname{im} A$	Attainable outputs	r
$\operatorname{im} A^\vee$	Measurements ignoring $\ker A$	r
$\ker A^\vee$	Output measurements vanishing on $\operatorname{im} A$	$m - r$

The usual *row space* is the coordinate representation of $\operatorname{im} A^\vee$, and the *left null space* is the coordinate representation of $\ker A^\vee$.

The equalities above require no inner product.

14.8.7 Worked example using our running matrix

Recall

$$A = \begin{pmatrix} 1 & 2 & -1 & 3 \\ 2 & 4 & 1 & 9 \\ 3 & 6 & 0 & 12 \end{pmatrix}.$$

Its kernel is

$$K = \operatorname{span}\{(-2, 1, 0, 0), (-4, 0, -1, 1)\}.$$

What is its annihilator K^0?

An arbitrary functional on $\mathbb{R}^4$ has the form

$$f(x) = c_1 x_1 + c_2 x_2 + c_3 x_3 + c_4 x_4.$$

To annihilate the first kernel basis vector, it must satisfy

$$-2c_1 + c_2 = 0.$$

To annihilate the second, it must satisfy

$$-4c_1 - c_3 + c_4 = 0.$$

Therefore,

$$c_2 = 2c_1, \qquad c_4 = 4c_1 + c_3.$$

Every coefficient column has the form

$$c = c_1 \begin{pmatrix} 1 \\ 2 \\ 0 \\ 4 \end{pmatrix} + c_3 \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}.$$

Thus, in coefficient coordinates,

$$\boxed{K^0 = \operatorname{span}\left\{ \begin{pmatrix} 1 \\ 2 \\ 0 \\ 4 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} \right\}.}$$

These are the nonzero rows of the RREF, expressed as coefficient columns.

They span the row space of A.

Therefore, we have verified concretely that

$$\operatorname{im} A^\vee = (\ker A)^0.$$

Now consider the image of A.

Every output satisfies

$$y_3 = y_1 + y_2.$$

Thus the functional

$$g(y_1, y_2, y_3) = y_3 - y_1 - y_2$$

vanishes on $\operatorname{im} A$.

Its coefficient column is

$$(-1, -1, 1)^T.$$

Since the image is two-dimensional, its annihilator has dimension one. Hence

$$\boxed{(\operatorname{im} A)^0 = \operatorname{span}\{(-1, -1, 1)\}.}$$

This is precisely the kernel of the transpose matrix A^T, in standard dual coordinates.

We have recovered all four fundamental subspaces from the two annihilator identities.

14.8.8 A useful rank consequence

Apply rank–nullity to the dual map

$$T^\vee : W^* \to V^*.$$

Since

$$\ker T^\vee = (\operatorname{im} T)^0,$$

we have

$$\dim \ker T^\vee = \dim W - \dim \operatorname{im} T.$$

Therefore,

$$\begin{aligned} \operatorname{rank} T^\vee &= \dim W^* - \dim \ker T^\vee \\ &= \dim W - (\dim W - \operatorname{rank} T) \\ &= \operatorname{rank} T. \end{aligned}$$

Thus

$$\boxed{\operatorname{rank} T^\vee = \operatorname{rank} T.}$$

For matrices, this becomes the familiar theorem

$$\boxed{\operatorname{rank} A^T = \operatorname{rank} A.}$$

The equality of row rank and column rank is therefore another manifestation of algebraic duality.

Checkpoint 4

Prove directly that

$$(S+U)^0 = S^0 \cap U^0.$$

Then explain why the two identities

$$\operatorname{im} T^\vee = (\ker T)^0, \qquad \ker T^\vee = (\operatorname{im} T)^0$$

are dual statements about what a transformation forgets and what it can reach.

14.9 Session 5 — Quotient duality and exact sequences

Module A established that quotient spaces identify vectors differing by elements of a given subspace.

Now we can ask a natural question:

Which linear measurements remain meaningful after making those identifications?

The answer leads to quotient duality.

14.9.1 A functional on a quotient

Let $S \subseteq V$, and let

$$q : V \to V/S$$

be the quotient map.

Suppose

$$\ell : V/S \to \mathbb{F}$$

is a linear functional on the quotient.

Composing with q, we obtain a functional on V:

$$f = \ell \circ q.$$

For every $s \in S$,

$$f(s) = \ell(q(s)) = \ell(S) = 0.$$

Thus

$$f \in S^0.$$

So every measurement on the quotient corresponds to a measurement on V that annihilates S.

Is the converse true?

Yes.

14.9.2 The quotient duality theorem

Theorem. If $S \subseteq V$, then

$$\boxed{(V/S)^* \cong S^0.}$$

The isomorphism is given by

$$\boxed{q^\vee(\ell) = \ell \circ q.}$$

Proof

First, we know that $\ell \circ q$ annihilates S, so the map takes values in S^0.

It is linear.

To prove injectivity, suppose

$$\ell \circ q = 0.$$

Because q is surjective, every element of V/S equals $q(v)$ for some v.

Therefore,

$$\ell(q(v)) = 0$$

for every coset, implying $\ell = 0$.

To prove surjectivity onto S^0, take any $f \in S^0$.

Define

$$\ell(v + S) = f(v).$$

This is well-defined: if $v - w \in S$, then $f(v - w) = 0$, so $f(v) = f(w)$.

The map ℓ is linear and satisfies

$$q^\vee(\ell) = f.$$

Hence $q^\vee$ is an isomorphism onto S^0.

14.9.3 Geometric meaning of quotient duality

Recall the example

$$V = \mathbb{R}^2, \qquad S = \operatorname{span}\{(1, 0)\}.$$

The quotient V/S consists of horizontal lines.

A functional on this quotient must assign the same scalar to all points on a given horizontal line.

Therefore, when expressed as a functional on $\mathbb{R}^2$, it cannot depend on the horizontal coordinate.

It must have the form

$$f(x, y) = cy$$

for some scalar c.

But these are precisely the functionals annihilating S.

Thus

$$(\mathbb{R}^2/S)^* \cong S^0.$$

This illustrates the general principle:

Measurements on a quotient space are exactly the measurements on the original space that ignore the subspace being collapsed.

14.9.4 Example in three dimensions

Let

$$S = \operatorname{span}\{(1,1,1)\} \subseteq \mathbb{R}^3.$$

Then

$$\dim(\mathbb{R}^3/S) = 2.$$

A functional

$$f(x,y,z) = ax + by + cz$$

annihilates S precisely when

$$a + b + c = 0.$$

Therefore S^0 has dimension two.

One basis is given by the functionals

$$f_1(x,y,z) = x - y,$$

$$f_2(x,y,z) = x - z.$$

They induce well-defined functionals on the quotient:

$$\overline{f}_1((x,y,z) + S) = x - y,$$

$$\overline{f}_2((x,y,z) + S) = x - z.$$

Adding the same constant to x, y, z changes neither difference.

Thus these are genuine measurements of equivalence classes rather than of particular representatives.

14.9.5 Duality and subspaces

Now consider the inclusion map

$$i : S \hookrightarrow V.$$

Its dual is

$$i^\vee : V^* \to S^*.$$

What does $i^\vee$ do?

It takes a functional on V and restricts it to S:

$$\boxed{i^\vee(f) = f|_S.}$$

What is its kernel?

A functional restricts to zero on S precisely when it annihilates S.

Thus

$$\boxed{\ker i^\vee = S^0.}$$

In finite dimensions, $i^\vee$ is surjective.

Indeed, every functional on S can be extended to a functional on V by extending a basis of S to one of V.

Applying the first isomorphism theorem gives

$$\boxed{V^*/S^0 \cong S^*.}$$

Compare the two identities:

$$\boxed{\begin{aligned}(V/S)^* &\cong S^0,\\ V^*/S^0 &\cong S^*.\end{aligned}}$$

These express complementary aspects of duality.

The first says that the dual of a quotient consists of functionals vanishing on the collapsed subspace.

The second says that functionals on a subspace arise by restricting functionals on the ambient space, with ambiguity measured by the annihilator.

14.9.6 Exact sequences under duality

Recall the canonical short exact sequence

$$\boxed{0 \longrightarrow S \xrightarrow{i} V \xrightarrow{q} V/S \longrightarrow 0.}$$

Taking algebraic duals reverses the arrows:

$$\boxed{0 \longrightarrow (V/S)^* \xrightarrow{q^\vee} V^* \xrightarrow{i^\vee} S^* \longrightarrow 0.}$$

Let's verify exactness.

Since q is surjective, $q^\vee$ is injective.

We know that

$$\operatorname{im} q^\vee = S^0.$$

We also know that

$$\ker i^\vee = S^0.$$

Hence

$$\operatorname{im} q^\vee = \ker i^\vee.$$

Finally, $i^\vee$ is surjective because every functional on S extends to V.

Thus the dual sequence is exact.

This gives an important theorem:

For finite-dimensional vector spaces, taking duals sends a short exact sequence to another short exact sequence, with arrows reversed.

This is one of the simplest examples of how a contravariant functor interacts with algebraic structure.

The statement also holds for algebraic duals of arbitrary vector spaces with the usual basis-extension principle. For continuous duals of infinite-dimensional topological vector spaces, additional hypotheses matter.

14.9.7 Duality and the first isomorphism theorem

Recall the factorization

$$V \longrightarrow V/\ker T \xrightarrow{\cong} \operatorname{im} T \longrightarrow W.$$

Taking duals reverses the direction.

An output functional first restricts to the image, is transferred across the isomorphism, and is then pulled back to the original domain.

The kernel of the dual map consists of functionals that vanish on the entire image.

Its image consists of functionals that vanish on the kernel.

This is exactly what we proved in Session 4:

$$\boxed{\begin{aligned}\ker T^{\vee} &= (\operatorname{im} T)^0,\\ \operatorname{im} T^{\vee} &= (\ker T)^0.\end{aligned}}$$

Thus the first isomorphism theorem, annihilators, quotient duality, and exact sequences are not unrelated results.

They express the same structure from different directions.

Checkpoint 5

Let $S \subseteq V$.

Prove that the dual of the quotient map

$$q^{\vee} : (V/S)^* \to V^*$$

is injective.

Then prove that the dual of the inclusion

$$i^{\vee} : V^* \to S^*$$

is surjective in finite dimensions.

Explain why their image and kernel fit together into an exact sequence.

14.10 Session 6 — Double duals and the universal structure of duality

We now examine a subtle question that clarifies why dual spaces are genuinely different from the original spaces.

In finite dimensions,

$$\dim V = \dim V^*.$$

So V and V^* are isomorphic.

But is there a natural, basis-independent isomorphism between them?

In general, no.

Remarkably, there *is* a canonical isomorphism between V and its **double dual** V^{**}.

Understanding this distinction is fundamental to the structural viewpoint we're developing.

14.10.1 What is the double dual?

The dual space V^* is itself a vector space.

We can therefore form its dual:

$$\boxed{V^{**} = (V^*)^*.}$$

An element of V^{**} is a linear functional acting on elements of V^*.

So there are three levels:

$$\begin{aligned} v &\in V \quad \text{a vector,} \\ f &\in V^* \quad \text{a measurement of vectors,} \\ F &\in V^{**} \quad \text{a measurement of functionals.} \end{aligned}$$

At first, this may seem like an unnecessary proliferation of objects.

But there is a very natural way for a vector v to define a functional on V^*.

14.10.2 Every vector defines an evaluation functional

Fix $v \in V$.

Define

$$J_V(v) : V^* \to \mathbb{F}$$

by

$$\boxed{J_V(v)(f) = f(v).}$$

In words, $J_V(v)$ takes a functional f and asks for its value at the vector v.

It is linear in f:

$$\begin{aligned} J_V(v)(\alpha f + \beta g) &= (\alpha f + \beta g)(v) \\ &= \alpha f(v) + \beta g(v). \end{aligned}$$

Thus $J_V(v) \in V^{**}$.

We have constructed a map

$$\boxed{J_V : V \to V^{**}, \qquad v \longmapsto J_V(v).}$$

This map is called the **canonical evaluation map**.

It requires no basis and no inner product.

14.10.3 The double-dual theorem

Theorem. If V is finite-dimensional, then

$$\boxed{J_V : V \to V^{**}}$$

is a linear isomorphism.

Proof

First, J_V is linear:

$$\begin{aligned} J_V(\alpha v + \beta w)(f) &= f(\alpha v + \beta w) \\ &= \alpha f(v) + \beta f(w) \\ &= (\alpha J_V(v) + \beta J_V(w))(f). \end{aligned}$$

Next, we prove injectivity.

Suppose

$$J_V(v) = 0.$$

Then

$$f(v) = 0$$

for every $f \in V^*$.

But we proved in Session 1 that linear functionals distinguish vectors.

Therefore $v = 0$.

Hence J_V is injective.

Finally, since V is finite-dimensional,

$$\dim V^{**} = \dim V^* = \dim V.$$

An injective linear map between spaces of the same finite dimension is surjective.

Therefore J_V is an isomorphism.

14.10.4 Why is $V \cong V^{**}$ different from $V \cong V^*$?

The distinction is **canonicity**.

A basis provides an isomorphism $V \cong V^*$ by matching the original basis with its dual basis.

But a different basis generally produces a different identification.

An inner product also gives an identification between V and V^*, but that depends on the chosen inner product.

By contrast,

$$J_V(v)(f) = f(v)$$

is determined entirely by the evaluation operation.

No choices are involved.

This is why $V \to V^{**}$ is called canonical.

It is an example of a *natural* construction.

14.10.5 Naturality of the double-dual identification

Let

$$T : V \to W$$

be linear.

Taking duals twice produces

$$T^{\vee\vee} : V^{**} \to W^{**}.$$

The commutativity of Figure 14.3 is the identity

$$\boxed{J_W \circ T = T^{\vee\vee} \circ J_V.}$$

Let's verify this.

Take $v \in V$ and $g \in W^*$.

Evaluating the left side gives

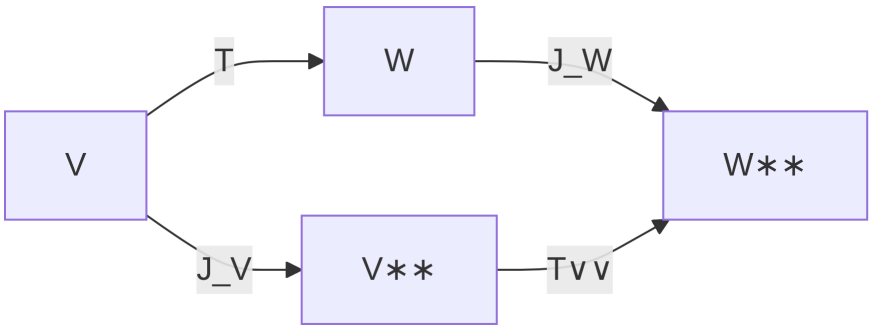


Figure 14.3: The two paths from V to W∗∗ agree. The evaluation maps J_V and J_W give the same result whether applied before or after transporting the vector.

$$J_W(Tv)(g) = g(Tv).$$

For the right side,

$$\begin{aligned}(T^{\vee\vee}(J_V(v)))(g) &= J_V(v)(T^{\vee}g)\\ &= (T^{\vee}g)(v)\\ &= g(Tv).\end{aligned}$$

Both expressions agree for every g, hence

$$J_W(Tv) = T^{\vee\vee}(J_V(v)).$$

This is a concrete example of a **natural isomorphism**.

It says that the identification of a vector with its double-dual evaluation functional is compatible with every linear transformation.

This compatibility is stronger than merely knowing that two spaces happen to have equal dimensions.

14.10.6 Double annihilators

Recall that for $S \subseteq V$,

$$S^0 \subseteq V^*.$$

We can form its annihilator again:

$$S^{00} = (S^0)^0 \subseteq V^{**}.$$

Every $s \in S$ determines an element $J_V(s) \in V^{**}$.

If $f \in S^0$, then $f(s) = 0$.

Therefore

$$J_V(s)(f) = 0$$

for every $f \in S^0$, so

$$J_V(S) \subseteq S^{00}.$$

In finite dimensions,

$$\dim S^{00} = \dim V^* - \dim S^0.$$

Since

$$\dim S^0 = \dim V - \dim S,$$

we obtain

$$\dim S^{00} = \dim S.$$

The inclusion is therefore an equality:

$$\boxed{J_V(S) = S^{00}.}$$

After identifying V canonically with V^{**}, this is commonly written

$$\boxed{S^{00} = S.}$$

This is the **double-annihilator theorem**.

It says that a subspace can be recovered from all the functionals that vanish on it.

This mirrors the nondegeneracy of the evaluation pairing.

14.10.7 Duality reverses sums and intersections

We already proved that

$$(S + U)^0 = S^0 \cap U^0.$$

We can now establish the companion identity:

$$\boxed{(S \cap U)^0 = S^0 + U^0.}$$

Here is a short structural proof using double annihilators.

Apply the first identity inside V^*:

$$(S^0 + U^0)^0 = S^{00} \cap U^{00}.$$

Under the canonical identification with V^{**}, the right-hand side corresponds to $S \cap U$.

Taking annihilators once more, and using the finite-dimensional double-annihilator theorem, gives

$$S^0 + U^0 = (S \cap U)^0.$$

Together,

$$\boxed{\begin{aligned}(S + U)^0 &= S^0 \cap U^0,\\ (S \cap U)^0 &= S^0 + U^0.\end{aligned}}$$

This is a duality between the subspace lattice operations of sum and intersection.

14.10.8 Duality and invariant subspaces

Here is a connection to spectral theory that is especially relevant to your Numerical Linear Algebra course.

Let

$$T : V \to V$$

be a linear operator and let $U \subseteq V$ be invariant:

$$T(U) \subseteq U.$$

Consider its annihilator $U^0 \subseteq V^*$.

Claim: U^0 is invariant under the dual operator $T^\vee$.

Proof

Take $f \in U^0$.

For every $u \in U$,

$$(T^\vee f)(u) = f(Tu).$$

Since U is invariant,

$$Tu \in U.$$

And since f annihilates U,

$$f(Tu) = 0.$$

Therefore $T^\vee f$ also annihilates U.

Hence

$$\boxed{T^\vee(U^0) \subseteq U^0.}$$

This is a purely algebraic relationship between invariant subspaces of a transformation and its dual.

When an inner product is introduced, it becomes a relationship involving invariant orthogonal complements and adjoints.

For self-adjoint operators, this is particularly important because invariant subspaces and their orthogonal complements can be used to decompose the operator.

14.10.9 Invariant subspaces also define quotient operators

Suppose again that $T(U) \subseteq U$.

We can define

$$\overline{T} : V/U \to V/U$$

by

$$\boxed{\overline{T}(v + U) = Tv + U.}$$

Why is this well-defined?

If $v - w \in U$, then

$$T(v - w) \in U$$

by invariance, so

$$Tv + U = Tw + U.$$

Thus T induces a linear operator on the quotient.

This is another important structural phenomenon:

An invariant subspace permits us to study both the restriction

$$T|_U : U \to U$$

and the induced transformation

$$\overline{T} : V/U \to V/U.$$

In a basis adapted to U, the matrix of T is block upper triangular:

$$[T] = \begin{pmatrix} B & C \\ 0 & D \end{pmatrix}.$$

Here B represents $T|_U$, while D represents the induced quotient operator $\overline{T}$ in corresponding quotient coordinates.

Thus invariant-subspace structure gives a systematic way to simplify linear operators.

Checkpoint 6

Explain why the canonical isomorphism $V \cong V^{**}$ does not require an inner product.

Then prove that if U is invariant under T, its annihilator U^0 is invariant under $T^\vee$.

Finally, explain why T induces a well-defined transformation on V/U.

14.11 Session 7 — Applications of duality

We now connect the abstract theory to mathematical computation.

The key idea is to recognize that many scientific problems do not merely manipulate state vectors. They also involve linear measurements, constraints, weak equations, and residual functionals.

Duality gives us a language for describing these objects before choosing an inner product or a particular coordinate representation.

14.11.1 Application I — Compatibility conditions for linear systems

Consider a linear system

$$Ax = b.$$

We know from Module A that a solution exists precisely when

$$b \in \operatorname{im} A.$$

Duality provides another way to express this condition.

Recall that

$$\ker A^\vee = (\operatorname{im} A)^0.$$

Thus a functional $g \in \ker A^\vee$ vanishes on every attainable output.

If $b = Ax$, then

$$g(b) = g(Ax) = 0.$$

Therefore, solvability requires

$$g(b) = 0$$

for every $g \in \ker A^\vee$.

In finite dimensions, this condition is also sufficient.

Indeed, if $b \notin \operatorname{im} A$, we can extend a basis of $\operatorname{im} A$ by adding b, then construct a functional that vanishes on $\operatorname{im} A$ but takes the value 1 on b.

Consequently,

$$\boxed{Ax = b \text{ is solvable } \iff g(b) = 0 \quad \forall g \in \ker A^\vee.}$$

This is an algebraic form of a solvability or compatibility condition.

Running-matrix example

Recall

$$A = \begin{pmatrix} 1 & 2 & -1 & 3 \\ 2 & 4 & 1 & 9 \\ 3 & 6 & 0 & 12 \end{pmatrix}.$$

We found

$$\ker A^\vee = \operatorname{span}\{g\},$$

where

$$g(b_1, b_2, b_3) = b_3 - b_1 - b_2.$$

Thus

$$Ax = b$$

is solvable exactly when

$$\boxed{b_3 = b_1 + b_2.}$$

This condition can be understood in two equivalent ways.

Computationally, it arises because the third row of A is the sum of the first two rows.

Structurally, it arises because the functional g belongs to the kernel of the dual map and therefore annihilates all attainable outputs.

The second interpretation generalizes beyond matrices.

Connection to the Fredholm alternative

In functional analysis, related compatibility principles are associated with the **Fredholm alternative**.

Under appropriate hypotheses on an operator T, solvability of $Tx = b$ can be characterized by orthogonality or annihilation conditions involving the kernel of an adjoint operator.

Our finite-dimensional duality identity is the algebraic foundation of that phenomenon.

In infinite-dimensional spaces, whether such conditions completely characterize solvability depends on the operator and the topology of its image. For example, bounded operators between Hilbert spaces may have nonclosed ranges.

That distinction belongs to Module F1.

14.11.2 Application II — Graph gradients, edge measurements, and discrete divergence

Let's revisit the path graph from Module A, shown in Figure 14.4.

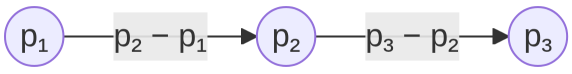


Figure 14.4: The edge orientation sets the sign of the potential differences used by D.

Suppose a scalar potential p_i is assigned to each vertex.

Define the discrete gradient operator

$$D = \begin{pmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix} : \mathbb{R}^3 \to \mathbb{R}^2.$$

Then

$$Dp = \begin{pmatrix} p_2 - p_1 \\ p_3 - p_2 \end{pmatrix}.$$

The output consists of the potential differences along the edges.

We already know that

$$\ker D = \operatorname{span}\{(1,1,1)\}.$$

Thus D cannot detect constant shifts of the potential.

What does the dual map do?

Let g be a linear functional on the edge-difference space:

$$g(d_1, d_2) = c_1 d_1 + c_2 d_2.$$

The pulled-back functional is

$$D^\vee(g)(p) = g(Dp).$$

Expanding,

$$\begin{aligned} g(Dp) &= c_1(p_2 - p_1) + c_2(p_3 - p_2) \\ &= -c_1 p_1 + (c_1 - c_2)p_2 + c_2 p_3. \end{aligned}$$

Thus the coefficient column of the pulled-back functional is

$$\boxed{D^T \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} -c_1 \\ c_1 - c_2 \\ c_2 \end{pmatrix}.}$$

Notice that the coefficients always sum to zero:

$$-c_1 + (c_1 - c_2) + c_2 = 0.$$

This is exactly what our annihilator theorem predicts.

Because

$$\ker D = \operatorname{span}\{\mathbf{1}\},$$

we have

$$\operatorname{im} D^\vee = (\ker D)^0.$$

The annihilator of the constant-vector subspace consists of functionals whose coefficients sum to zero.

Therefore every edge-based measurement, when pulled back to the vertices, is insensitive to a common constant shift.

A concrete interpretation

Take

$$g(d_1, d_2) = d_1 + d_2.$$

Then

$$g(Dp) = (p_2 - p_1) + (p_3 - p_2) = p_3 - p_1.$$

So the measurement of the sum of both edge differences becomes the measurement of the potential difference between the endpoints.

Its coefficient column is

$$D^T \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}.$$

This illustrates how transpose matrices arise naturally when measurements are transferred through discrete operators.

Relation to divergence

The operator D measures potential differences as head minus tail on each oriented edge. If c gives the flows in those orientations, $D^T c$ gives net inflow at each vertex. With the convention that divergence measures net outflow, the discrete divergence is $-D^T$.

This gradient–divergence relationship is an elementary example of a general duality between discrete differential operators.

It appears in finite-element and finite-volume methods, graph Laplacians, electrical networks, and conservation laws.

14.11.3 Application III — Variational formulations and finite elements

This is one of the most important applications for your longer-term study of functional analysis and numerical PDEs.

Suppose V is a real vector space of admissible functions.

Let

$$a : V \times V \to \mathbb{R}$$

be a bilinear form.

For example,

$$a(u, v) = \int_0^1 u'(x)v'(x)\,dx.$$

Let

$$F \in V^*$$

be a linear functional, for example

$$F(v) = \int_0^1 f(x)v(x)\,dx.$$

We want to find $u \in V$ satisfying

$$\boxed{a(u, v) = F(v) \qquad \forall v \in V.}$$

This is an abstract variational equation.

Why dual spaces arise naturally

For fixed u, the map

$$v \longmapsto a(u, v)$$

is a linear functional.

Therefore it belongs to V^*.

We can define a linear operator

$$\boxed{\mathcal{A} : V \to V^*}$$

by

$$(\mathcal{A}u)(v) = a(u, v).$$

The variational problem becomes

$$\boxed{\mathcal{A}u = F.}$$

This is a linear equation, but notice that the operator maps a vector or function u into a functional.

Its codomain is the dual space.

This is extremely common in functional analysis and numerical PDE theory.

Example: the one-dimensional Poisson equation

Consider

$$-u''(x) = f(x), \qquad 0 < x < 1,$$

with homogeneous Dirichlet boundary conditions

$$u(0) = u(1) = 0.$$

Multiply the differential equation by a smooth test function v with $v(0) = v(1) = 0$, and integrate:

$$-\int_0^1 u''(x)v(x)\,dx = \int_0^1 f(x)v(x)\,dx.$$

Integration by parts gives

$$\int_0^1 u'(x)v'(x)\,dx = \int_0^1 f(x)v(x)\,dx.$$

This is the weak or variational form.

Here,

$$a(u,v) = \int_0^1 u'v', \qquad F(v) = \int_0^1 fv.$$

The equation says that two linear functionals of v are equal for every admissible test function.

For a rigorous functional-analytic treatment, the natural setting is a Sobolev space such as $H_0^1(0,1)$, but understanding that space is not necessary for the present algebraic point.

Finite-dimensional approximation

Let

$$V_h = \text{span}\{\phi_1, \dots, \phi_n\} \subseteq V$$

be a finite-dimensional trial and test space.

We seek

$$u_h = \sum_{j=1}^{n} c_j \phi_j$$

satisfying

$$a(u_h, v_h) = F(v_h) \qquad \forall v_h \in V_h.$$

It suffices to impose this condition on the basis functions:

$$a(u_h, \phi_i) = F(\phi_i), \qquad i = 1, \dots, n.$$

Substituting the expansion,

$$\sum_{j=1}^{n} c_j a(\phi_j, \phi_i) = F(\phi_i).$$

Define the stiffness matrix

$$K_{ij} = a(\phi_j, \phi_i)$$

and the load vector

$$b_i = F(\phi_i).$$

Then

$$\boxed{Kc = b.}$$

Thus a familiar numerical linear system emerges from an operator equation involving a dual space.

Residuals as functionals

Define the residual functional

$$R_h = F - \mathcal{A}u_h.$$

It belongs to V^*.

The Galerkin condition is

$$R_h(v_h) = 0 \qquad \forall v_h \in V_h.$$

Therefore,

$$\boxed{R_h \in V_h^0.}$$

This is an annihilator statement.

It says that the residual functional vanishes on every direction belonging to the chosen finite-dimensional test space.

Once inner products are introduced, related formulations can be expressed through geometric orthogonality.

The dual-space formulation, however, already makes sense algebraically.

Why this matters

This example connects several of our concepts:

- A differential equation becomes an operator equation.
- The operator may naturally map into a dual space.
- A finite-dimensional test space determines which measurements are imposed.
- The residual is a functional.
- The Galerkin condition says that the residual belongs to an annihilator.
- Choosing bases converts the variational problem into a matrix equation.

This provides a direct bridge from Module B to functional analysis, finite-element methods, and numerical linear algebra.

14.11.4 Application IV — Dual bases and spectral decompositions

We now establish a particularly useful connection to numerical eigenvalue algorithms.

Suppose $A : V \to V$ is diagonalizable and V has dimension n.

Let

$$(v_1, \dots, v_n)$$

be a basis of eigenvectors, with

$$Av_i = \lambda_i v_i.$$

Let

$$(\ell^1, \dots, \ell^n)$$

be the corresponding dual basis.

Thus

$$\ell^i(v_j) = \delta_{ij}.$$

Every vector $x \in V$ has the expansion

$$\boxed{x = \sum_{i=1}^{n} \ell^i(x) v_i.}$$

The coefficient of each eigenvector is obtained by applying a dual-basis functional.

This is an important conceptual observation:

Spectral coordinates are themselves linear measurements.

14.11.5 Left eigenfunctionals

Let's investigate how the dual operator acts on the dual eigenbasis.

For any v_j,

$$\begin{aligned}(A^\vee \ell^i)(v_j) &= \ell^i(Av_j) \\ &= \ell^i(\lambda_j v_j) \\ &= \lambda_j \delta_{ij} \\ &= \lambda_i \delta_{ij}.\end{aligned}$$

But

$$\lambda_i \ell^i(v_j) = \lambda_i \delta_{ij}.$$

Since two functionals agreeing on a basis agree everywhere,

$$\boxed{A^\vee \ell^i = \lambda_i \ell^i.}$$

Therefore, the dual basis functionals are eigenvectors of the dual operator.

In matrix terminology, they correspond to **left eigenvectors** of A.

The original eigenvectors v_i are right eigenvectors.

This gives a very natural explanation of the distinction between left and right eigenvectors.

14.11.6 Spectral projectors from dual bases

Define

$$P_i : V \to V$$

by

$$\boxed{P_i(x) = \ell^i(x) v_i.}$$

This map extracts the component of x along v_i.

Let's verify its algebraic properties.

For i, j,

$$\begin{aligned} P_i(P_j(x)) &= P_i(\ell^j(x)v_j) \\ &= \ell^j(x)\ell^i(v_j)v_i \\ &= \delta_{ij}\ell^j(x)v_i. \end{aligned}$$

Therefore,

$$\boxed{P_iP_j = \begin{cases} P_i, & i = j, \\ 0, & i \neq j. \end{cases}}$$

In particular,

$$P_i^2 = P_i.$$

So each P_i is a projection.

Furthermore,

$$\boxed{I = \sum_{i=1}^{n} P_i,}$$

because the right-hand side reconstructs every vector from its eigenbasis coordinates.

We also have

$$\boxed{A = \sum_{i=1}^{n} \lambda_i P_i.}$$

Indeed,

$$Ax = \sum_i \lambda_i \ell^i(x)v_i.$$

This is a spectral decomposition written in terms of dual bases and projections.

14.11.7 Polynomial filtering revisited

Let p be a polynomial.

Since P_i extracts the component along v_i,

$$\boxed{p(A) = \sum_{i=1}^{n} p(\lambda_i) P_i.}$$

Therefore,

$$p(A)x = \sum_{i=1}^{n} p(\lambda_i) \ell^i(x) v_i.$$

The polynomial multiplies each spectral coefficient by $p(\lambda_i)$.

This is the algebraic mechanism underlying polynomial eigenvalue filters.

What changes for symmetric matrices?

If A is real symmetric, we can choose an orthonormal eigenbasis.

Then

$$\ell^i(x) = v_i^T x,$$

because the coordinate functional of an orthonormal basis is represented by the dot product with the corresponding basis vector.

Therefore,

$$P_i = v_i v_i^T.$$

These are orthogonal spectral projections.

For a general diagonalizable but nonsymmetric matrix, the corresponding projections are generally oblique.

They are built using left and right eigenvectors rather than one orthonormal eigenbasis.

This distinction has major numerical implications.

For nonnormal matrices, the eigenvector basis can be poorly conditioned, and the spectral coordinate functionals may be very sensitive to perturbations.

For symmetric or Hermitian matrices, the orthonormal eigenbasis avoids this particular problem.

Connection to HASC

In Chebyshev-filtered subspace iteration, we apply a polynomial $p(A)$ to vectors or subspaces in order to amplify desired spectral components relative to unwanted components.

The formula

$$p(A)x = \sum_i p(\lambda_i)\ell^i(x)v_i$$

shows exactly how filtering works when the matrix is diagonalizable.

For Hermitian matrices, this takes the especially convenient form

$$p(A)x = \sum_i p(\lambda_i)(v_i^H x)v_i$$

using an orthonormal eigenbasis.

Thus the concepts from Module B—dual bases, dual maps, and spectral coordinate functionals—connect directly to the mathematical mechanism behind the seminar's eigenvalue methods.

The more advanced questions of convergence, stability, finite precision, and inexact polynomial evaluation belong to Modules D and E.

Checkpoint 7

Prove that for a diagonalizable operator,

$$I = \sum_i P_i$$

and

$$A = \sum_i \lambda_i P_i.$$

Then derive the polynomial filtering formula.

Finally, explain why the spectral projectors for a nonsymmetric diagonalizable matrix need not be orthogonal, even though they are idempotent.

14.12 Session 8 — Consolidation, exercises, and proof reconstruction

The goal of this final session is to consolidate the theory.

At this stage, you should be able to recognize a common structural pattern:

$$\boxed{\begin{array}{c}\text{A transformation acts on vectors}\\ T : V \to W\\ \downarrow\\ \text{It induces a transformation on measurements}\\ T^\vee : W^* \to V^*\\ \downarrow\\ \text{Its kernel and image determine}\\ \text{which measurements vanish or factor through it.}\end{array}}$$

The following exercises test definitions, proofs, computations, structural understanding, and applications.

Try solving them before consulting the worked solutions.

14.12.1 Exercise 1 — Construct a dual basis

Let

$$v_1 = (1, 1), \qquad v_2 = (1, -1)$$

be a basis of $\mathbb{R}^2$.

Find the dual basis $(\varepsilon^1, \varepsilon^2)$.

Verify its defining properties.

Then express the functional

$$f(x, y) = 3x + y$$

as a linear combination of the dual basis functionals.

14.12.2 Exercise 2 — Change of coordinates for covectors

Let $B \in \mathbb{R}^{n\times n}$ be an invertible basis matrix.

Prove that the coordinate column of a vector transforms by B^{-1}, while the coefficient column of a functional transforms by B^T, when moving from standard to B-basis coordinates.

Explain why the scalar evaluation of the functional must remain unchanged.

14.12.3 Exercise 3 — Calculate a dual map

Let

$$A = \begin{pmatrix} 1 & 2 & 0 \\ 0 & 1 & 1 \end{pmatrix} : \mathbb{R}^3 \to \mathbb{R}^2.$$

Define the output functional

$$g(y_1, y_2) = 3y_1 - 2y_2.$$

Compute $A^\vee(g)$ directly by composition.

Then verify the answer using A^T.

14.12.4 Exercise 4 — Find an annihilator

Let

$$S = \operatorname{span}\{(1, 1, 0), (0, 1, 1)\} \subseteq \mathbb{R}^3.$$

Find a basis of S^0.

Verify the dimension formula

$$\dim S^0 = \dim V - \dim S.$$

14.12.5 Exercise 5 — Verify the fundamental duality identities

For the matrix A from Exercise 3, calculate

$$\ker A, \qquad \operatorname{im} A,$$

and the kernel and image of its dual map.

Verify explicitly that

$$\operatorname{im} A^{\vee} = (\ker A)^0$$

and

$$\ker A^{\vee} = (\operatorname{im} A)^0.$$

14.12.6 Exercise 6 — Dual of a quotient

Let

$$S = \operatorname{span}\{(1,1,1)\} \subseteq \mathbb{R}^3.$$

Find a basis of S^0.

Construct the corresponding functionals on the quotient $\mathbb{R}^3/S$.

Explain why they are well-defined.

14.12.7 Exercise 7 — Dual exact sequences

Consider the short exact sequence

$$0 \to S \xrightarrow{i} V \xrightarrow{q} V/S \to 0.$$

Prove that the dual sequence

$$0 \to (V/S)^* \xrightarrow{q^{\vee}} V^* \xrightarrow{i^{\vee}} S^* \to 0$$

is exact when V is finite-dimensional.

Identify the kernel and image of each nontrivial map.

14.12.8 Exercise 8 — The double-dual map

Define

$$J_V(v)(f) = f(v).$$

Prove that $J_V : V \to V^{**}$ is a linear isomorphism for finite-dimensional V.

Then prove that for every linear map $T : V \to W$,

$$J_W \circ T = T^{\vee\vee} \circ J_V.$$

Explain why this is called a natural identification.

14.12.9 Exercise 9 — Graph duality

Let

$$D = \begin{pmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix}.$$

Consider the edge functional

$$g(d_1, d_2) = 2d_1 - d_2.$$

Calculate $D^{\vee}(g)$.

Show that it annihilates the constant-vector subspace.

Then explain the relationship to the theorem

$$\operatorname{im} D^{\vee} = (\ker D)^0.$$

14.12.10 Exercise 10 — Left eigenvectors and spectral projections

Let

$$A = \begin{pmatrix} 2 & 1 \\ 0 & 3 \end{pmatrix}.$$

Its right eigenvectors can be chosen as

$$v_1 = (1, 0), \qquad v_2 = (1, 1).$$

Construct their dual basis.

Prove that the corresponding functionals are eigenfunctionals of $A^{\vee}$.

Construct the two spectral projectors P_1, P_2, verify

$$P_1 + P_2 = I,$$

and show that

$$A = 2P_1 + 3P_2.$$

Finally, determine $p(A)$ for

$$p(t) = t - 2$$

using the spectral projectors rather than ordinary matrix arithmetic.

14.13 Worked solutions

14.13.1 Solution 1 — Dual basis

The basis matrix is

$$B = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

Its inverse is

$$B^{-1} = \frac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

The rows of B^{-1} give the dual basis:

$$\boxed{\varepsilon^1(x,y) = \frac{x+y}{2}, \qquad \varepsilon^2(x,y) = \frac{x-y}{2}.}$$

Evaluating at the basis vectors gives

	v_1	v_2
ε^1	1	0
ε^2	0	1

as required.

For

$$f(x,y) = 3x + y,$$

the dual basis coefficients are

$$f(v_1) = 4, \qquad f(v_2) = 2.$$

Therefore,

$$\boxed{f = 4\varepsilon^1 + 2\varepsilon^2.}$$

Indeed,

$$4\frac{x+y}{2} + 2\frac{x-y}{2} = 3x + y.$$

14.13.2 Solution 2 — Coordinate transformations

Let x_{std} denote the standard coordinate column and x_B its coordinates in the basis matrix B.

Then

$$x_{\text{std}} = Bx_B,$$

so

$$\boxed{x_B = B^{-1}x_{\text{std}}.}$$

For a functional with standard coefficient column c_{std},

$$f(x) = c_{\text{std}}^T x_{\text{std}}.$$

Substituting,

$$f(x) = c_{\text{std}}^T B x_B = (B^T c_{\text{std}})^T x_B.$$

Thus

$$\boxed{c_B = B^T c_{\text{std}}.}$$

The two transformation rules ensure that the scalar $f(x)$ is independent of the coordinate system.

14.13.3 Solution 3 — Dual map

We have

$$Ax = \begin{pmatrix} x_1 + 2x_2 \\ x_2 + x_3 \end{pmatrix}.$$

Therefore,

$$\begin{aligned} A^\vee(g)(x) &= g(Ax) \\ &= 3(x_1 + 2x_2) - 2(x_2 + x_3) \\ &= 3x_1 + 4x_2 - 2x_3. \end{aligned}$$

Thus

$$\boxed{A^\vee(g)(x_1, x_2, x_3) = 3x_1 + 4x_2 - 2x_3.}$$

Using the coefficient column $c = (3, -2)^T$,

$$A^T c = \begin{pmatrix} 1 & 0 \\ 2 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 3 \\ -2 \end{pmatrix} = \begin{pmatrix} 3 \\ 4 \\ -2 \end{pmatrix}.$$

The two computations agree.

14.13.4 Solution 4 — Annihilator

Since the two generators of S are independent,

$$\dim S = 2.$$

Therefore,

$$\dim S^0 = 3 - 2 = 1.$$

Write

$$f(x, y, z) = ax + by + cz.$$

The conditions for $f \in S^0$ are

$$a + b = 0, \qquad b + c = 0.$$

Thus

$$a = c = -b.$$

Choosing $a = 1$, we obtain

$$\boxed{S^0 = \operatorname{span}\{f\}, \qquad f(x, y, z) = x - y + z.}$$

The annihilator has dimension one, as predicted.

14.13.5 Solution 5 — Fundamental duality identities

For

$$A = \begin{pmatrix} 1 & 2 & 0 \\ 0 & 1 & 1 \end{pmatrix},$$

the kernel equations are

$$x_1 + 2x_2 = 0, \qquad x_2 + x_3 = 0.$$

Therefore,

$$\boxed{\ker A = \operatorname{span}\{(-2, 1, -1)\}.}$$

The two rows of A are independent, so A has rank two.

Its image equals $\mathbb{R}^2$:

$$\boxed{\operatorname{im} A = \mathbb{R}^2.}$$

Since $A^\vee$ is represented by A^T, its image is represented by

$$\boxed{\operatorname{im} A^\vee = \operatorname{span}\{(1, 2, 0), (0, 1, 1)\}}$$

in standard coefficient coordinates.

Both coefficient vectors annihilate the kernel vector $(-2, 1, -1)$:

$$(1, 2, 0) \cdot (-2, 1, -1) = 0,$$

$$(0, 1, 1) \cdot (-2, 1, -1) = 0.$$

The annihilator of a one-dimensional subspace of $\mathbb{R}^3$ has dimension two.

Thus

$$\operatorname{im} A^\vee = (\ker A)^0.$$

Finally, since $A^T : \mathbb{R}^2 \to \mathbb{R}^3$ is injective,

$$\ker A^\vee = \{0\}.$$

Since $\operatorname{im} A = \mathbb{R}^2$, its annihilator is also zero:

$$(\operatorname{im} A)^0 = \{0\}.$$

Hence the second identity holds.

14.13.6 Solution 6 — Quotient duality

A functional on $\mathbb{R}^3$ has the form

$$f(x, y, z) = ax + by + cz.$$

To annihilate S, we require

$$a + b + c = 0.$$

A convenient basis is

$$f_1(x, y, z) = x - y,$$

$$f_2(x, y, z) = y - z.$$

Thus

$$\boxed{S^0 = \operatorname{span}\{f_1, f_2\}.}$$

They induce functionals on the quotient:

$$\overline{f}_1((x,y,z)+S) = x - y,$$

$$\overline{f}_2((x,y,z)+S) = y - z.$$

Adding $t(1,1,1)$ to a representative leaves both expressions unchanged.

Therefore they are well-defined.

They are independent and span the two-dimensional dual space $(\mathbb{R}^3/S)^*$.

14.13.7 Solution 7 — Dual exact sequence

The sequence is

$$0 \to (V/S)^* \xrightarrow{q^\vee} V^* \xrightarrow{i^\vee} S^* \to 0.$$

Since q is surjective, $q^\vee$ is injective:

$$\ker q^\vee = \{0\}.$$

Its image is

$$\operatorname{im} q^\vee = S^0.$$

The map $i^\vee$ restricts a functional on V to S.

Therefore,

$$\ker i^\vee = S^0.$$

Hence

$$\operatorname{im} q^\vee = \ker i^\vee.$$

Finally, $i^\vee$ is surjective because every linear functional on S extends to a linear functional on V.

Thus the sequence is exact.

14.13.8 Solution 8 — Double dual

Define

$$J_V(v)(f) = f(v).$$

The map is linear because

$$J_V(\alpha v + \beta w)(f) = \alpha f(v) + \beta f(w).$$

If $J_V(v) = 0$, then every functional vanishes on v.

By nondegeneracy of evaluation, $v = 0$. Thus J_V is injective.

Since

$$\dim V = \dim V^{**},$$

it is also surjective.

Therefore J_V is an isomorphism.

For naturality, let $T : V \to W$. For every $g \in W^*$,

$$\begin{aligned}(T^{\vee\vee}(J_V(v)))(g) &= J_V(v)(T^{\vee}g)\\ &= (T^{\vee}g)(v)\\ &= g(Tv)\\ &= J_W(Tv)(g).\end{aligned}$$

Thus

$$\boxed{T^{\vee\vee} \circ J_V = J_W \circ T.}$$

The identification is natural because it commutes with every linear map without depending on a choice of basis or inner product.

14.13.9 Solution 9 — Graph duality

The coefficient column of the edge functional is

$$c = \begin{pmatrix} 2 \\ -1 \end{pmatrix}.$$

Therefore,

$$D^T c = \begin{pmatrix} -1 & 0 \\ 1 & -1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 2 \\ -1 \end{pmatrix} = \begin{pmatrix} -2 \\ 3 \\ -1 \end{pmatrix}.$$

Thus

$$\boxed{D^\vee(g)(p_1, p_2, p_3) = -2p_1 + 3p_2 - p_3.}$$

On a constant vector $p = t(1, 1, 1)$,

$$D^\vee(g)(p) = (-2 + 3 - 1)t = 0.$$

Therefore the pulled-back functional annihilates $\ker D$.

For any edge functional g, the equality $g(Dp) = g(0) = 0$ for $p \in \ker D$ gives

$$\operatorname{im} D^\vee \subseteq (\ker D)^0.$$

The two rows of D are independent, so $\operatorname{im} D^\vee$ has dimension two. The annihilator of the one-dimensional space $\ker D$ also has dimension two. Thus the inclusion is an equality.

14.13.10 Solution 10 — Spectral projectors

The right eigenvectors are

$$v_1 = (1, 0), \qquad v_2 = (1, 1).$$

Their dual basis is

$$\ell^1(x, y) = x - y, \qquad \ell^2(x, y) = y.$$

These are left eigenfunctionals:

$$A^\vee \ell^1 = 2\ell^1, \qquad A^\vee \ell^2 = 3\ell^2.$$

The spectral projectors are

$$P_1(x,y) = \ell^1(x,y)v_1 = (x-y,0),$$

and

$$P_2(x,y) = \ell^2(x,y)v_2 = (y,y).$$

Therefore,

$$\boxed{P_1 = \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix}, \qquad P_2 = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}.}$$

Adding,

$$P_1 + P_2 = I.$$

Furthermore,

$$2P_1 + 3P_2 = \begin{pmatrix} 2 & 1 \\ 0 & 3 \end{pmatrix} = A.$$

For $p(t) = t - 2$,

$$p(2) = 0, \qquad p(3) = 1.$$

Thus

$$p(A) = p(2)P_1 + p(3)P_2 = P_2.$$

Therefore,

$$\boxed{p(A) = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}.}$$

Notice that P_1, P_2 are projections but not orthogonal projections under the standard Euclidean inner product.

This is a concrete example of spectral decomposition using a right eigenbasis and its dual left eigenbasis.

14.14 Computational laboratory — MATLAB

The following laboratory connects the abstract theory with explicit matrix computations.

Its purpose is to verify how bases, dual bases, annihilators, and spectral projectors are represented numerically.

```
%% Module B - Duality and Linear Measurements
clear; clc;

%% 1. Dual basis of a nonstandard basis
B = [1 1;
     0 1];

% Rows of B^(-1) represent the dual basis
DualRows = inv(B);

disp('Dual basis coefficient rows:');
disp(DualRows);

% Verify dual basis evaluation
disp('DualRows * B (should be identity):');
disp(DualRows * B);

% Coordinate transformation of a functional
c_old = [2; 1];
c_new = B.' * c_old;

disp('Functional coefficients in new dual basis:');
disp(c_new);

%% 2. Fundamental duality identities
A = [1 2 -1 3;
     2 4  1 9;
     3 6  0 12];
```

```
% Basis of ker(A)
K = [-2 -4;
      1  0;
      0 -1;
      0  1];

% Basis of image(A^vee), as coefficient columns
R = rref(A);
RowDual = R(1:2, :).';

% Basis of ker(A^vee)
LeftNull = [-1; -1; 1];

fprintf('||A*K|| = %.3e\n', norm(A*K));
fprintf('||K^T*RowDual|| = %.3e\n', norm(K.'*RowDual));
fprintf('||A^T*LeftNull|| = %.3e\n', norm(A.'*LeftNull));

fprintf('rank(A) = %d\n', rank(A));
fprintf('rank(A^T) = %d\n', rank(A.'));

%% 3. Dual of a graph gradient
D = [-1  1  0;
      0 -1  1];

edge_measurement = [2; -1];
vertex_measurement = D.' * edge_measurement;

disp('Pulled-back vertex measurement:');
disp(vertex_measurement);

fprintf('Sum of coefficients = %.3e\n', ...
    sum(vertex_measurement));

%% 4. Left and right eigenvectors
A2 = [2 1;
      0 3];

% Right eigenvector basis
X = [1 1;
     0 1];

% Rows of X^(-1) are the dual eigenfunctionals
```

```
LeftRows = inv(X);

P1 = X(:,1) * LeftRows(1,:);
P2 = X(:,2) * LeftRows(2,:);

disp('Spectral projector P1:');
disp(P1);

disp('Spectral projector P2:');
disp(P2);

fprintf('||P1^2-P1|| = %.3e\n', norm(P1*P1-P1));
fprintf('||P2^2-P2|| = %.3e\n', norm(P2*P2-P2));
fprintf('||P1+P2-I|| = %.3e\n', ...
    norm(P1+P2-eye(2)));

fprintf('||A2-(2P1+3P2)|| = %.3e\n', ...
    norm(A2-(2*P1+3*P2)));

% Polynomial p(t) = t-2
pA = A2-2*eye(2);

fprintf('||p(A2)-P2|| = %.3e\n', norm(pA-P2));
```

14.14.1 Interpret the numerical results

In the first experiment, $B^{-1}B = I$ verifies the defining relation between a basis and its dual basis. Each row of B^{-1} is a functional that extracts one coordinate in the chosen basis.

In the second experiment, the matrix products verify annihilation relationships. For instance,

$$K^T \,\mathrm{RowDual} = 0$$

means that every row-space functional annihilates every kernel vector.

In the graph experiment, the coefficients of the pulled-back vertex measurement sum to zero, reflecting its insensitivity to constant potential shifts.

In the spectral experiment, the left and right eigenvector bases construct projections satisfying

$$P_i^2 = P_i, \qquad P_1 + P_2 = I.$$

This is the algebraic mechanism underlying spectral decomposition.

For larger floating-point problems, the equalities will generally hold only up to numerical roundoff. Moreover, when the eigenvector basis is ill-conditioned, computing its inverse can magnify numerical errors. This will become important when studying nonnormal matrices and eigenvalue conditioning.

14.15 Final conceptual synthesis

We can now summarize Module B through one central idea:

Duality studies a mathematical object through the linear measurements that can be performed on it.

The original space V contains vectors.

The dual space V^* contains scalar-valued linear observations of those vectors.

A transformation $T : V \to W$ maps vectors forward, while its dual $T^\vee : W^* \to V^*$ pulls observations backward.

Annihilators describe measurements that vanish on prescribed subspaces, and quotient duality describes measurements that remain meaningful after certain vector differences have been identified.

Linear map

$$T : V \to W$$

Dual map

$$T^\vee : W^* \to V^*$$

Kernel–image duality

$$\boxed{\begin{aligned} \operatorname{im} T^\vee &= (\ker T)^0, \\ \ker T^\vee &= (\operatorname{im} T)^0 \end{aligned}}$$

Quotients and exact sequences

$$(V/S)^* \cong S^0$$

14.15.1 The essential relationships

Concept	Central result	Meaning
Dual space	$V^* = \mathrm{Hom}(V, \mathbb{F})$	Space of linear measurements
Dual basis	$\varepsilon^i(v_j) = \delta_{ij}$	Coordinate extraction
Dual map	$T^\vee(g) = g \circ T$	Pulling measurements backward
Transpose	$[T^\vee] = [T]^T$ in dual bases	Matrix representation of the pullback
Annihilator	$S^0 = \{f : f\|_S = 0\}$	Measurements invisible on S
Kernel–image duality	$\mathrm{im}\, T^\vee = (\ker T)^0$	Recoverable input measurements
Dual kernel	$\ker T^\vee = (\mathrm{im}\, T)^0$	Measurements vanishing on all outputs
Quotient duality	$(V/S)^* \cong S^0$	Measurements respecting identifications
Double dual	$V \cong V^{**}$	Canonical evaluation identification
Contravariance	$(S \circ T)^\vee = T^\vee \circ S^\vee$	Composition reverses
Spectral dual basis	$P_i(x) = \ell^i(x) v_i$	Extracting spectral components

14.15.2 Module B mastery checklist

Before beginning Module C, aim to complete the following without referring to the study material.

- ☐ Define dual spaces and the evaluation pairing, and explain their geometric significance.
- ☐ Prove the dual basis theorem and calculate dual bases in nonstandard coordinates.
- ☐ Derive the dual map and the transpose rule from composition of functionals.
- ☐ Prove both fundamental identities involving annihilators, kernels, and images.
- ☐ Explain quotient duality and why dualizing a finite-dimensional short exact sequence preserves exactness.
- ☐ Prove the canonical double-dual isomorphism and understand its naturality.
- ☐ Explain duality in at least two applications: graph operators, variational problems, or solvability conditions.
- ☐ Construct left eigenfunctionals and spectral projectors using a right eigenbasis and its dual.

14.16 Transition to Module C — Inner-product geometry

Module B has developed duality without requiring inner products.

Module C will add inner-product structure and investigate what new possibilities it creates.

The central question will be:

If covectors are linear measurements, how can an inner product represent each covector by a uniquely determined vector?

The Riesz representation theorem answers this question.

For a finite-dimensional real inner-product space V, every $f \in V^*$ has a unique representing vector $a_f \in V$ such that

$$\boxed{f(v) = \langle v, a_f \rangle \qquad \forall v \in V.}$$

This creates an isomorphism between the space of vectors and the space of linear measurements.

It transforms the algebraic relationships of Module B into geometric ones.

For example,

$$\operatorname{im} T^\vee = (\ker T)^0$$

becomes, under the Riesz identifications,

$$\boxed{\operatorname{im} T^* = (\ker T)^\perp,}$$

where T^* is now the inner-product adjoint.

Similarly, annihilators correspond to orthogonal complements, and the dual map corresponds to the adjoint.

That is the bridge to orthogonal projections, self-adjoint matrices, positive definiteness, least squares, and spectral theory.

One particularly important connection deserves emphasis:

In Module B, we constructed spectral coefficients using dual eigenfunctionals,

$$x = \sum_i \ell^i(x) v_i.$$

For a Hermitian matrix with an orthonormal eigenbasis, Riesz representation allows those functionals to be written as inner products:

$$\ell^i(x) = v_i^H x.$$

Therefore,

$$\boxed{x = \sum_i (v_i^H x) v_i.}$$

This is the familiar orthonormal spectral expansion.

It is also the foundation for Rayleigh–Ritz approximation, orthogonal spectral projections, and the Hermitian eigenvalue algorithms relevant to your HASC seminar.

The central achievement of Module B is understanding that algebraic duality comes first: inner-product geometry provides a particularly powerful way to represent it, but does not create it.

15 Module C — Inner-Product Geometry

Riesz Representation, Adjoint Operators, Orthogonal Projections, Self-Adjointness, Positive Definiteness, and Least Squares

Advanced undergraduate · 8 study sessions · Approximately 16–22 hours

Prerequisites	Modules A and B: linear maps, kernels, images, direct sums, quotient spaces, dual bases, dual maps, and annihilators
Main emphasis	Rigorous foundations, geometry, constructive methods, and applications
Applied connections	Least squares, SVD, Galerkin methods, numerical PDEs, generalized eigenvalue problems, and spectral algorithms
Next module	Module D: Spectral Theory

15.1 Introduction — From algebraic structure to geometry

In Module A, we studied linear transformations through their kernels, images, quotients, and direct-sum decompositions.

In Module B, we introduced the dual space V^*, whose elements are linear measurements of vectors. We learned that a map

$$T : V \longrightarrow W$$

induces a dual map

$$T^\vee : W^* \longrightarrow V^*, \qquad T^\vee(f) = f \circ T.$$

We also established the fundamental algebraic duality identities

$$\boxed{\begin{aligned} \operatorname{im} T^\vee &= (\ker T)^0, \\ \ker T^\vee &= (\operatorname{im} T)^0. \end{aligned}}$$

These results require only vector-space structure.

Module C asks what additional mathematical structure becomes available when we equip vector spaces with inner products.

The answer is substantial. Inner products allow us to introduce lengths, angles, orthogonality, distances, and geometric projections.

More surprisingly, they allow us to represent linear functionals by vectors. This is the content of the Riesz representation theorem.

That identification turns algebraic dual maps into adjoint operators and annihilators into orthogonal complements.

From there we obtain self-adjoint matrices, positive definiteness, least-squares approximation, and the geometry underlying numerical eigenvalue methods.

The central conceptual progression is:

Algebraic duality

V^*, $T^\vee$, S^0 — Module B

Introduce an inner product and Riesz representation

Inner-product geometry

$V \cong V^*$, T^*, $S^\perp$ — Module C

Develop self-adjointness and orthogonal approximation

Numerical and variational methods

Least squares · SVD · Galerkin methods · Rayleigh–Ritz · Eigenvalue algorithms

15.2 Learning outcomes

By completing the module, you should be able to understand its main concepts from six perspectives.

Perspective	Expected achievement
Rigorous	Prove finite-dimensional Riesz representation, derive adjoints, and establish orthogonal decomposition theorems
Intuitive and geometric	Interpret inner products as geometry, adjoints as pulled-back measurements, and projections as closest-point approximations
Computational	Construct Gram matrices, weighted adjoints, orthogonal projectors, least-squares solutions, and Ritz approximations

Perspective	Expected achievement
Structural	Connect annihilators with orthogonal complements, dual maps with adjoints, and direct sums with projections
Universal	Distinguish algebraic duality from inner-product-dependent identifications and understand the extension to Hilbert spaces
Applied	Recognize these concepts in numerical linear algebra, weak PDE formulations, finite elements, and spectral methods

15.3 Study roadmap

Session	Main topic	Central result
1	Inner products, norms, and Gram matrices	Geometry encoded by bilinear forms
2	Riesz representation theorem	V^* represented by vectors
3	Adjoint operators and fundamental subspaces	$\operatorname{im} T^* = (\ker T)^\perp$
4	Orthogonal projections and decompositions	Closest-point approximation
5	Self-adjoint and positive definite operators	Spectral theorem, quadratic forms, and SPD geometry
6	Least squares, pseudoinverses, and numerical stability	Normal equations and minimum-norm solutions
7	Functional analysis, variational methods, and PDEs	Galerkin orthogonality and energy minimization
8	Generalized eigenproblems and HASC connections	Rayleigh–Ritz and spectral filtering

Exercises, worked solutions, a MATLAB laboratory, and a mastery checklist follow.

Conventions

We will work primarily with finite-dimensional **real** inner-product spaces. This lets us concentrate on the mathematics without repeatedly introducing complex conjugation.

In the real case, inner products are symmetric and bilinear.

For complex spaces, we use the convention

$$\langle x, y \rangle = y^H x,$$

which is linear in the first argument and conjugate-linear in the second. Under this convention, a complex linear functional is represented as

$$f(x) = \langle x, a_f \rangle.$$

The same adjoint identity will hold:

$$\langle Tx, y \rangle = \langle x, T^* y \rangle.$$

In standard orthonormal coordinates, the adjoint matrix is the conjugate transpose A^H, rather than the ordinary transpose A^T.

We will return to the complex case when it adds something conceptually important.

15.4 Session 1 — Inner products: introducing geometry into vector spaces

15.4.1 Why introduce an inner product?

A vector space has addition and scalar multiplication.

Those operations are sufficient for discussing linear independence, bases, kernels, images, and quotient spaces.

But a vector-space structure alone does not tell us:

- How long a vector is.
- Whether two vectors are perpendicular.
- How far apart two vectors are.
- Which vector in a subspace best approximates a given vector.

An inner product provides this additional geometric structure.

15.4.2 Definition of an inner product

Let V be a real vector space.

An **inner product** on V is a function

$$\langle \cdot, \cdot \rangle : V \times V \to \mathbb{R}$$

satisfying the following properties for all $x, y, z \in V$ and $\alpha, \beta \in \mathbb{R}$:

Bilinearity

$$\langle \alpha x + \beta y, z \rangle = \alpha \langle x, z \rangle + \beta \langle y, z \rangle,$$

with linearity in the other argument as well.

Symmetry

$$\boxed{\langle x, y \rangle = \langle y, x \rangle.}$$

Positive definiteness

$$\boxed{\langle x, x \rangle \geq 0, \qquad \langle x, x \rangle = 0 \iff x = 0.}$$

The ordinary Euclidean inner product on $\mathbb{R}^n$ is

$$\boxed{\langle x, y \rangle = x^T y.}$$

But it is not the only possible inner product.

That fact will become central later.

15.4.3 From inner products to lengths and angles

An inner product defines a norm:

$$\boxed{\|x\| = \sqrt{\langle x, x \rangle}.}$$

It also defines a distance:

$$d(x, y) = \|x - y\|.$$

Two vectors are called **orthogonal** if

$$\boxed{x \perp y \iff \langle x, y \rangle = 0.}$$

For nonzero real vectors, we can define the angle $\theta \in [0, \pi]$ between them by

$$\cos\theta = \frac{\langle x, y \rangle}{\|x\|\|y\|}.$$

Why is the right-hand side always between -1 and 1?

Because of the Cauchy–Schwarz inequality.

15.4.4 The Cauchy–Schwarz inequality

Theorem. For all x, y in an inner-product space,

$$\boxed{|\langle x, y \rangle| \leq \|x\|\|y\|.}$$

Proof

If $y = 0$, the statement is immediate.

Suppose $y \neq 0$.

For any $t \in \mathbb{R}$,

$$0 \leq \|x - ty\|^2.$$

Expanding,

$$\begin{aligned} \|x - ty\|^2 &= \langle x - ty, x - ty \rangle \\ &= \|x\|^2 - 2t\langle x, y \rangle + t^2 \|y\|^2. \end{aligned}$$

Choose

$$t = \frac{\langle x, y \rangle}{\|y\|^2}.$$

Substituting gives

$$0 \leq \|x\|^2 - \frac{\langle x, y \rangle^2}{\|y\|^2}.$$

Therefore,

$$\langle x, y \rangle^2 \leq \|x\|^2 \|y\|^2.$$

Taking square roots proves the result.

Equality holds precisely when x and y are linearly dependent.

Why this theorem matters

Cauchy–Schwarz establishes much of the basic geometry of inner-product spaces.

For example, it gives the triangle inequality:

$$\begin{aligned}\|x+y\|^2 &= \|x\|^2 + 2\langle x, y\rangle + \|y\|^2 \\ &\leq (\|x\| + \|y\|)^2.\end{aligned}$$

Hence

$$\boxed{\|x+y\| \leq \|x\| + \|y\|.}$$

Thus the norm induced by an inner product really is a norm.

It also implies that, for fixed a,

$$f(x) = \langle x, a\rangle$$

is a continuous linear functional.

This observation will be essential in the Riesz representation theorem.

15.4.5 Orthogonality and the Pythagorean theorem

If $x \perp y$, then

$$\langle x, y\rangle = 0.$$

Consequently,

$$\begin{aligned}\|x+y\|^2 &= \langle x+y, x+y\rangle \\ &= \|x\|^2 + 2\langle x, y\rangle + \|y\|^2 \\ &= \|x\|^2 + \|y\|^2.\end{aligned}$$

Therefore,

$$\boxed{x \perp y \Longrightarrow \|x+y\|^2 = \|x\|^2 + \|y\|^2.}$$

Unlike many elementary geometric presentations, this result does not require a diagram of a triangle.

It follows directly from the algebraic properties of the inner product.

This is how Euclidean geometry generalizes to arbitrary inner-product spaces, including spaces of functions.

15.4.6 Positive definite matrices define inner products

Now consider a real symmetric matrix

$$G \in \mathbb{R}^{n \times n}$$

satisfying

$$x^T G x > 0 \qquad \text{for every } x \neq 0.$$

Such a matrix is called **symmetric positive definite**, or SPD.

Define

$$\boxed{\langle x, y \rangle_G = x^T G y.}$$

Let's verify that this is an inner product.

Bilinearity follows from matrix multiplication.

Symmetry follows because

$$x^T G y = y^T G^T x = y^T G x.$$

Positive definiteness follows from the assumption

$$x^T G x > 0$$

for nonzero x.

Therefore every SPD matrix defines an inner product.

Conversely, every inner product on $\mathbb{R}^n$, once a basis is fixed, can be represented using an SPD matrix.

This is an important result:

An SPD matrix can encode the geometry of a vector space.

15.4.7 Different inner products give different geometries

Consider

$$G = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix}.$$

Then

$$\langle x, y \rangle_G = 4x_1 y_1 + x_2 y_2.$$

The induced norm is

$$\|x\|_G = \sqrt{4x_1^2 + x_2^2}.$$

The vectors of unit norm satisfy

$$4x_1^2 + x_2^2 = 1.$$

That is the ellipse shown in Figure 15.1.

The same coordinate plane can carry different inner-product geometries. The weighted norm assigns a larger cost to displacement in the first coordinate direction.

The two coordinate axes remain orthogonal in this example, but this need not hold for an arbitrary SPD matrix.

For example, if

$$G = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix},$$

then

$$\langle e_1, e_2 \rangle_G = 1.$$

Thus the standard coordinate axes are **not** orthogonal in the G-inner product.

Orthogonality depends on the chosen geometry.

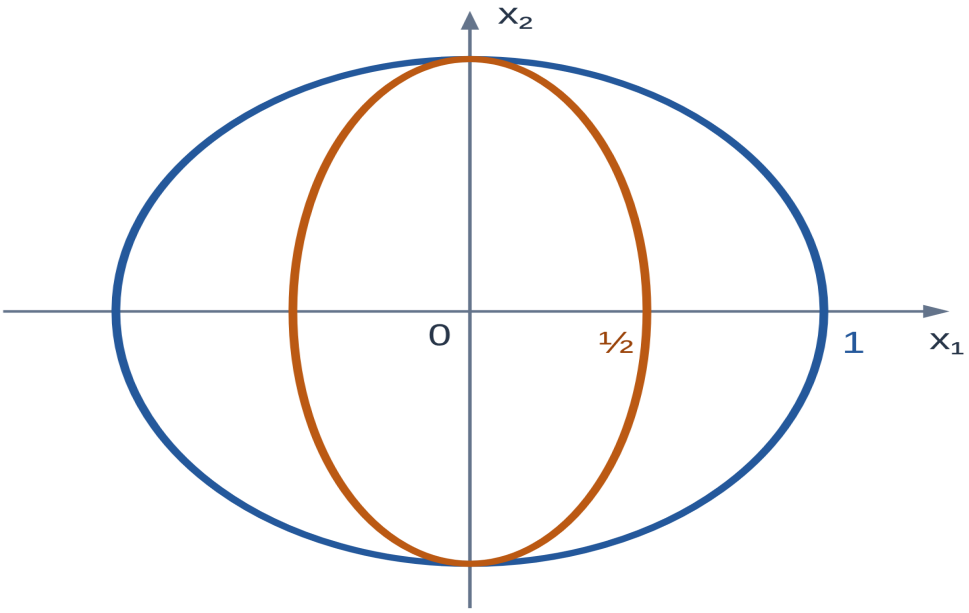


Figure 15.1: The Euclidean unit circle (blue) and the weighted unit ellipse (orange) share their vertical intercepts. On the horizontal axis their positive intercepts are 1 and 1/2.

15.4.8 Gram matrices of nonorthonormal bases

There is another reason SPD matrices arise naturally.

Let V be an inner-product space with basis

$$\mathcal{B} = (v_1, \dots, v_n).$$

Define its **Gram matrix**

$$\boxed{G_{ij} = \langle v_i, v_j \rangle.}$$

Suppose

$$x = \sum_i a_i v_i, \qquad y = \sum_j b_j v_j.$$

Then

$$\begin{aligned}\langle x, y\rangle &= \sum_{i,j} a_i b_j \langle v_i, v_j\rangle \\ &= a^T G b.\end{aligned}$$

Thus the matrix of the inner product in a possibly nonorthonormal basis is its Gram matrix.

The Gram matrix is symmetric.

It is positive definite because, for $a \neq 0$,

$$a^T G a = \left\| \sum_i a_i v_i \right\|^2 > 0.$$

The strict inequality follows from linear independence of the basis.

If the basis vectors are columns of an invertible matrix B in ordinary Euclidean coordinates, then

$$\boxed{G = B^T B.}$$

Example

Let

$$v_1 = (1, 0), \qquad v_2 = (1, 1).$$

Then

$$G = \begin{pmatrix} \langle v_1, v_1\rangle & \langle v_1, v_2\rangle \\ \langle v_2, v_1\rangle & \langle v_2, v_2\rangle \end{pmatrix} = \boxed{\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}.}$$

The matrix is SPD.

The two basis vectors are not orthogonal, which is reflected in the nonzero off-diagonal entries.

Connection to numerical methods

Gram matrices occur whenever we represent geometric or functional-analytic information in a finite-dimensional basis.

Examples include least-squares normal equations, finite-element mass matrices, weighted projections, and generalized eigenvalue problems.

This leads to an important distinction:

An inner product is an abstract geometric structure. A Gram matrix is its representation in a chosen basis.

The matrix changes when the basis changes, even if the underlying inner product remains the same.

Checkpoint 1

Given the basis

$$v_1 = (1, 1), \qquad v_2 = (1, -1),$$

calculate its Gram matrix under the standard Euclidean inner product.

Then explain why orthonormal bases have Gram matrix I, and why a basis need not be orthonormal to have an invertible Gram matrix.

15.5 Session 2 — The Riesz representation theorem

We now reach the conceptual bridge between Modules B and C.

Module B taught us that a covector is a linear functional

$$f : V \to \mathbb{R}.$$

We also emphasized that a covector is not inherently an ordinary vector.

But once an inner product is available, we can associate vectors with functionals.

The Riesz representation theorem says that this correspondence is complete and unique.

15.5.1 Every vector defines a linear functional

Fix $a \in V$.

Define

$$f_a : V \to \mathbb{R}$$

by

$$\boxed{f_a(x) = \langle x, a \rangle.}$$

Linearity of the inner product implies that f_a is a linear functional.

Therefore,

$$f_a \in V^*.$$

This defines a map

$$\boxed{R_V : V \to V^*, \qquad R_V(a) = f_a.}$$

For real spaces, R_V is linear.

The question is whether every linear functional arises this way.

15.5.2 Finite-dimensional Riesz representation theorem

Theorem. Let V be a finite-dimensional real inner-product space.

For every linear functional

$$f : V \to \mathbb{R},$$

there exists a unique vector $a_f \in V$ such that

$$\boxed{f(x) = \langle x, a_f \rangle \qquad \forall x \in V.}$$

The vector a_f is called the **Riesz representative** of f.

Equivalently,

$$\boxed{R_V : V \to V^* \text{ is an isomorphism.}}$$

This is a fundamental identification between vectors and linear measurements.

It depends on the chosen inner product.

15.5.3 Proof using an orthonormal basis

Choose an orthonormal basis

$$(e_1, \dots, e_n)$$

of V.

Recall that orthonormality means

$$\langle e_i, e_j \rangle = \delta_{ij}.$$

Take an arbitrary functional $f \in V^*$.

Define

$$\boxed{a_f = \sum_{i=1}^{n} f(e_i) e_i.}$$

We claim that

$$f(x) = \langle x, a_f \rangle.$$

Write

$$x = \sum_{i=1}^{n} x_i e_i.$$

Then

$$f(x) = \sum_{i=1}^{n} x_i f(e_i).$$

On the other hand,

$$\begin{aligned}\langle x, a_f \rangle &= \left\langle \sum_i x_i e_i, \sum_j f(e_j) e_j \right\rangle \\ &= \sum_{i,j} x_i f(e_j) \langle e_i, e_j \rangle \\ &= \sum_i x_i f(e_i) \\ &= f(x).\end{aligned}$$

Thus a representing vector exists.

Uniqueness

Suppose both $a, b \in V$ represent the same functional:

$$\langle x, a \rangle = \langle x, b \rangle \qquad \forall x \in V.$$

Subtracting,

$$\langle x, a - b \rangle = 0 \qquad \forall x \in V.$$

Choose $x = a - b$.

Then

$$\|a - b\|^2 = 0.$$

Positive definiteness implies $a = b$.

Therefore the representing vector is unique.

This completes the proof.

15.5.4 Geometric interpretation

Suppose $f \neq 0$.

Its kernel is a hyperplane:

$$\ker f = \{x : f(x) = 0\}.$$

Using Riesz representation,

$$f(x) = \langle x, a_f \rangle.$$

Therefore,

$$x \in \ker f \iff \langle x, a_f \rangle = 0.$$

Thus

$$\boxed{\ker f = \{x : x \perp a_f\}.}$$

The Riesz representative a_f is normal to the kernel hyperplane.

The length $\|a_f\|$ describes the scale of the measurement.

Indeed, by Cauchy–Schwarz,

$$|f(x)| = |\langle x, a_f \rangle| \leq \|x\| \|a_f\|.$$

Equality is attained for a suitable nonzero multiple of a_f.

Therefore,

$$\boxed{\sup_{\|x\| \leq 1} |f(x)| = \|a_f\|.}$$

This is often written

$$\boxed{\|f\|_{V^*} = \|a_f\|_V.}$$

So Riesz representation preserves the natural norms of functionals and their representing vectors.

It is not merely an algebraic isomorphism; in the real Hilbert-space setting, it is an isometric isomorphism.

15.5.5 Connection to dual bases

Recall Module B.

Let $\mathcal{B} = (v_1, \dots, v_n)$ be a possibly nonorthonormal basis of V, with dual basis

$$(\varepsilon^1, \dots, \varepsilon^n).$$

Each dual-basis functional has a Riesz representative $a_i \in V$:

$$\varepsilon^i(x) = \langle x, a_i \rangle.$$

In particular,

$$\boxed{\langle v_j, a_i \rangle = \delta_{ij}.}$$

Thus the Riesz representatives of the dual basis form a basis **biorthogonal** to the original basis.

If the original basis is orthonormal, then $a_i = v_i$.

If it is not orthonormal, the Riesz representatives generally differ from the original basis vectors.

This reveals an important structural relationship:

The dual basis is an algebraic construction determined by coordinate extraction. Its Riesz representatives depend additionally on the inner product.

15.5.6 Riesz representation in a weighted inner product

Consider

$$V = \mathbb{R}^n$$

with inner product

$$\langle x, y \rangle_G = x^T G y,$$

where G is SPD.

Suppose the functional f is represented in standard coordinates by

$$f(x) = c^T x.$$

We want its Riesz representative a_f relative to the G-inner product.

By definition,

$$f(x) = \langle x, a_f \rangle_G.$$

Therefore,

$$c^T x = x^T G a_f \qquad \forall x.$$

This implies

$$G a_f = c.$$

Hence

$$\boxed{a_f = G^{-1} c.}$$

The formula is simple but conceptually important.

The same covector f has different representing vectors under different inner products.

Example

Let

$$f(x_1, x_2) = 2x_1 + x_2.$$

Under the Euclidean inner product,

$$a_f = \begin{pmatrix} 2 \\ 1 \end{pmatrix}.$$

Under

$$G = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix},$$

the representative becomes

$$a_f = G^{-1} \begin{pmatrix} 2 \\ 1 \end{pmatrix} = \boxed{\begin{pmatrix} 1/2 \\ 1 \end{pmatrix}.}$$

The functional has not changed.

What has changed is the geometry used to represent it.

This is one of the most useful ways to understand the Riesz theorem: **it converts covectors into vectors, but the conversion depends on geometric structure.**

15.5.7 Gradients are Riesz representatives

This idea also clarifies the concept of a gradient.

Let

$$F : \mathbb{R}^n \to \mathbb{R}$$

be differentiable.

At a point x, the derivative is a linear functional

$$DF(x) : \mathbb{R}^n \to \mathbb{R}.$$

It assigns to a direction h the first-order change in F:

$$F(x + h) = F(x) + DF(x)[h] + o(\|h\|).$$

Under the Euclidean inner product, the gradient is defined by

$$\boxed{DF(x)[h] = \langle h, \nabla F(x) \rangle.}$$

Thus $\nabla F(x)$ is the Riesz representative of the derivative functional.

The derivative and the gradient are not fundamentally the same kind of object:

$$DF(x) \in V^*, \qquad \nabla F(x) \in V.$$

The inner product identifies them.

Example

Let

$$F(x_1, x_2) = x_1^2 + 3x_2^2.$$

Then

$$DF(x)[h] = 2x_1 h_1 + 6x_2 h_2.$$

The Euclidean gradient is

$$\nabla F(x) = \begin{pmatrix} 2x_1 \\ 6x_2 \end{pmatrix}.$$

If we change the inner product to one defined by G, the gradient becomes

$$\boxed{\nabla_G F(x) = G^{-1} \nabla F(x).}$$

This is not an arbitrary modification of the gradient.

It follows directly from the Riesz representation theorem.

Computational significance

This relationship appears in optimization, preconditioned gradient methods, and numerical algorithms for variational problems.

Changing the inner product changes which vector represents the same derivative functional.

That can change the geometry of optimization steps and sometimes improve numerical behavior.

15.5.8 The infinite-dimensional Riesz theorem: a preview

The finite-dimensional theorem is a special case of a more general result.

A **Hilbert space** is a complete inner-product space.

For a real Hilbert space H, every continuous linear functional

$$f : H \to \mathbb{R}$$

has a unique representing vector $a_f \in H$ satisfying

$$\boxed{f(x) = \langle x, a_f \rangle_H \qquad \forall x \in H.}$$

Moreover,

$$\|f\| = \|a_f\|.$$

The word *continuous* is essential in infinite dimensions.

Unlike the finite-dimensional situation, not every algebraic linear functional on an infinite-dimensional normed space is necessarily continuous.

Example in a function space

Take

$$H = L^2(0, 1),$$

with inner product

$$\langle u, v \rangle = \int_0^1 u(x)v(x)\,dx.$$

Define

$$F(u) = \int_0^1 x\,u(x)\,dx.$$

This is a continuous linear functional.

Its Riesz representative is the function

$$a_F(x) = x,$$

because

$$F(u) = \langle u, a_F \rangle.$$

So a linear measurement of functions can itself be represented by a function belonging to the same Hilbert space.

This is one of the foundational ideas of functional analysis.

Why completeness matters

A deeper geometric proof of Riesz representation uses the orthogonal decomposition

$$H = \ker f \oplus^{\perp} (\ker f)^{\perp}.$$

Because a nonzero continuous functional has a closed kernel, Hilbert-space completeness ensures the existence of the required orthogonal projection.

The orthogonal complement of the kernel is one-dimensional, so the functional measures precisely one distinguished direction, with a suitable scale factor.

We will develop this geometry in Session 4.

Checkpoint 2

Explain why the same functional can have different Riesz representatives under different inner products.

Then derive the formula

$$a_f = G^{-1}c$$

for a weighted inner product.

Finally, explain why the derivative $DF(x)$ is intrinsically a functional, whereas its gradient depends on the chosen inner product.

15.6 Session 3 — Adjoint operators and the geometry of linear maps

We are now ready to answer the question that originally motivated our discussion of Riesz representation:

Why does every linear transformation between finite-dimensional inner-product spaces have an adjoint, and what does that adjoint mean?

15.6.1 Defining the adjoint

Let V, W be finite-dimensional real inner-product spaces, and let

$$T : V \to W$$

be linear.

The **adjoint** of T is the unique linear map

$$T^* : W \to V$$

such that

$$\boxed{\langle Tx, y\rangle_W = \langle x, T^* y\rangle_V}$$

for every $x \in V$ and $y \in W$.

Notice that the direction is reversed:

$$T : V \to W, \qquad T^* : W \to V.$$

But T^* is not generally an inverse of T.

Rather, it tells us how to transfer an inner-product measurement of an output back into the input space.

15.6.2 Deriving the adjoint from Riesz representation

Fix $y \in W$.

Define

$$f_y : V \to \mathbb{R}$$

by

$$f_y(x) = \langle Tx, y \rangle_W.$$

This is a linear functional on V.

By the Riesz representation theorem, there is a unique vector $z_y \in V$ such that

$$f_y(x) = \langle x, z_y \rangle_V$$

for every $x \in V$.

Therefore,

$$\langle Tx, y \rangle_W = \langle x, z_y \rangle_V.$$

Define

$$\boxed{T^* y = z_y.}$$

We obtain

$$\langle Tx, y \rangle_W = \langle x, T^* y \rangle_V.$$

This proves existence.

Uniqueness follows because the Riesz representative of each f_y is unique.

To prove that T^* is linear, observe that for any $y_1, y_2 \in W$,

$$\begin{aligned} \langle x, T^*(\alpha y_1 + \beta y_2) \rangle &= \langle Tx, \alpha y_1 + \beta y_2 \rangle \\ &= \alpha \langle Tx, y_1 \rangle + \beta \langle Tx, y_2 \rangle \\ &= \langle x, \alpha T^* y_1 + \beta T^* y_2 \rangle. \end{aligned}$$

Uniqueness of the representing vector implies

$$T^*(\alpha y_1 + \beta y_2) = \alpha T^* y_1 + \beta T^* y_2.$$

Thus T^* is linear.

The adjoint is the result of applying Riesz representation to the functionals obtained by composing output measurements with T.

15.6.3 The adjoint and the algebraic dual map

Module B introduced

$$T^\vee : W^* \to V^*, \qquad T^\vee(g) = g \circ T.$$

Module C has introduced the Riesz identifications

$$R_V : V \to V^*, \qquad R_W : W \to W^*.$$

The adjoint is related to these maps by

$$\boxed{R_V \circ T^* = T^\vee \circ R_W.}$$

Indeed, for $y \in W$,

$$\begin{aligned}(T^\vee(R_W y))(x) &= (R_W y)(Tx) \\ &= \langle Tx, y\rangle_W \\ &= \langle x, T^* y\rangle_V \\ &= (R_V(T^* y))(x).\end{aligned}$$

Hence the two functionals agree.

Equivalently,

$$\boxed{T^* = R_V^{-1} \circ T^\vee \circ R_W.}$$

The two paths in Figure 15.2 express this identity.

The Riesz maps connect vector spaces to their duals. The adjoint represents the algebraic dual map using those identifications.

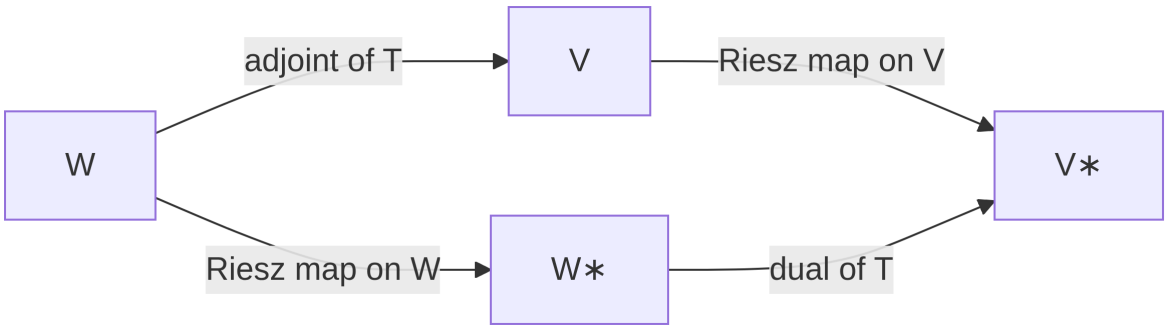


Figure 15.2: Both routes turn an output vector into the same linear functional on the input space.

In real finite-dimensional spaces, both Riesz maps are linear isomorphisms.

In complex spaces, the Riesz identification is conjugate-linear under our convention, but the adjoint remains a complex-linear transformation.

The main conceptual relationship is the same.

15.6.4 Matrix representation in orthonormal coordinates

Suppose

$$T : \mathbb{R}^n \to \mathbb{R}^m$$

is represented by A.

With standard Euclidean inner products,

$$\begin{aligned} \langle Ax, y \rangle &= (Ax)^T y \\ &= x^T A^T y \\ &= \langle x, A^T y \rangle. \end{aligned}$$

Therefore,

$$\boxed{[T^*] = A^T.}$$

For complex matrices,

$$\boxed{[T^*] = A^H = \overline{A}^T.}$$

The distinction between the algebraic dual and the adjoint now becomes explicit.

Construction	Underlying structure	Matrix in the appropriate standard coordinates
Algebraic dual $T^\vee$	Vector spaces	A^T
Inner-product adjoint T^*	Inner-product spaces	A^T over $\mathbb{R}$, A^H over $\mathbb{C}$

They coincide in real Euclidean coordinates, but not because they are conceptually identical.

15.6.5 Geometric meaning: pulling measurements backward

Suppose $x \in V$ is an input, and $Tx \in W$ is its output.

We want to measure the output in the direction $y \in W$.

The measurement is

$$\langle Tx, y \rangle_W.$$

The adjoint identity tells us that the same scalar can be obtained before applying T:

$$\boxed{\underbrace{\langle Tx, y \rangle_W}_{\text{output measurement}} = \underbrace{\langle x, T^*y \rangle_V}_{\text{input measurement}}.}$$

Thus T^*y is the input-space direction representing the output measurement associated with y.

Another interpretation involves hyperplanes.

The output condition

$$\langle w, y \rangle_W = 0$$

describes a hyperplane perpendicular to y.

Its preimage under T is characterized by

$$\langle Tx, y\rangle_W = 0.$$

By the adjoint identity,

$$\langle x, T^* y\rangle_V = 0.$$

Thus, when $T^* y \neq 0$, the preimage is the input hyperplane perpendicular to $T^* y$. If $T^* y = 0$, the preimage is all of V.

The adjoint pulls normal directions of output hyperplanes backward into the input space.

This geometric idea is useful in optimization, automatic differentiation, inverse problems, and sensitivity analysis.

15.6.6 Adjoints under nonstandard inner products

Suppose $V = \mathbb{R}^n$ and $W = \mathbb{R}^m$, with inner products

$$\langle x, z\rangle_V = x^T G_V z,$$

$$\langle y, w\rangle_W = y^T G_W w,$$

where G_V, G_W are SPD.

Let T be represented by $A \in \mathbb{R}^{m\times n}$.

We seek the matrix $A_G^\dagger$ of its adjoint.

The defining equation is

$$\langle Ax, y\rangle_W = \langle x, A_G^\dagger y\rangle_V.$$

Expanding,

$$x^T A^T G_W y = x^T G_V A_G^\dagger y.$$

Because this holds for all x, y,

$$A^T G_W = G_V A_G^\dagger.$$

Therefore,

$$\boxed{A_G^\dagger = G_V^{-1} A^T G_W.}$$

Here $A_G^\dagger$ denotes the weighted adjoint solely within this formula; many numerical linear algebra texts reserve $A^\dagger$ for the Moore–Penrose pseudoinverse. To avoid confusion, we will otherwise write $A^{*_{G_V,G_W}}$.

Thus,

$$\boxed{A^{*_{G_V,G_W}} = G_V^{-1} A^T G_W.}$$

For complex inner products with Hermitian positive definite Gram matrices, replace A^T by A^H.

This is a central formula for numerical methods using weighted inner products.

Example

Let

$$A = \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}$$

and use the same weighted inner product in the domain and codomain:

$$G = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix}.$$

Then

$$A^{*_G} = G^{-1} A^T G = \begin{pmatrix} 1 & 0 \\ 8 & 1 \end{pmatrix}.$$

The adjoint is not the ordinary transpose

$$A^T = \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix}.$$

The reason is that the geometric identification between vectors and functionals is different.

Structural lesson

The algebraic dual map is determined by T. The adjoint depends on both T and the inner products chosen on its domain and codomain.

This distinction is essential when analyzing finite-element mass matrices, generalized eigenvalue problems, and weighted optimization methods.

15.6.7 Algebraic properties of adjoints

Let $T : V \to W$, $S : W \to Z$ be linear maps between finite-dimensional inner-product spaces.

The adjoint satisfies

$$\boxed{(S \circ T)^* = T^* \circ S^*.}$$

Proof

For $x \in V, z \in Z$,

$$\begin{aligned} \langle STx, z\rangle_Z &= \langle Tx, S^*z\rangle_W \\ &= \langle x, T^*S^*z\rangle_V. \end{aligned}$$

By uniqueness of the adjoint,

$$(ST)^* = T^*S^*.$$

Other important identities include

$$\boxed{(T^*)^* = T,}$$

and, for real scalars,

$$\boxed{(\alpha T + \beta S)^* = \alpha T^* + \beta S^*.}$$

For complex scalars, conjugation appears:

$$(\alpha T + \beta S)^* = \overline{\alpha} T^* + \overline{\beta} S^*.$$

An invertible operator also satisfies

$$\boxed{(T^{-1})^* = (T^*)^{-1}.}$$

All these identities follow from the defining inner-product relationship.

15.6.8 Fundamental orthogonality relationships

We now recover the geometric versions of the annihilator identities from Module B.

Theorem. For a linear map $T : V \to W$ between finite-dimensional inner-product spaces,

$$\boxed{\begin{aligned} \ker T &= (\operatorname{im} T^*)^\perp, \\ \ker T^* &= (\operatorname{im} T)^\perp. \end{aligned}}$$

Proof of the first identity

Take $x \in V$.

Then

$$\begin{aligned} x \in \ker T &\iff Tx = 0 \\ &\iff \langle Tx, y\rangle_W = 0 \quad \forall y \in W \\ &\iff \langle x, T^*y\rangle_V = 0 \quad \forall y \in W \\ &\iff x \perp \operatorname{im} T^*. \end{aligned}$$

Thus

$$\ker T = (\operatorname{im} T^*)^\perp.$$

Applying the same argument to T^*, using $(T^*)^* = T$, gives the second identity.

Taking orthogonal complements and using finite dimensionality,

$$\boxed{\begin{aligned} \operatorname{im} T^* &= (\ker T)^\perp, \\ \operatorname{im} T &= (\ker T^*)^\perp. \end{aligned}}$$

These are the four fundamental orthogonality identities.

They give the decompositions

$$\boxed{V = \ker T \oplus^{\perp} \operatorname{im} T^*}$$

and

$$\boxed{W = \operatorname{im} T \oplus^{\perp} \ker T^*.}$$

The superscript $\perp$ indicates an orthogonal direct sum.

Comparison with Module B

Recall the algebraic identity

$$\operatorname{im} T^{\vee} = (\ker T)^0.$$

The Riesz representation theorem identifies the annihilator $(\ker T)^0$ with the orthogonal complement $(\ker T)^{\perp}$, and identifies $T^{\vee}$ with T^*.

Thus,

$$\boxed{\text{Annihilator duality} \quad \xrightarrow{\text{Riesz}} \quad \text{Orthogonality duality.}}$$

This is one of the central conceptual achievements of Module C.

A finite-versus-infinite-dimensional qualification

For a bounded operator between Hilbert spaces, we always have

$$\ker T = (\operatorname{im} T^*)^{\perp}$$

and

$$\ker T^* = (\operatorname{im} T)^{\perp}.$$

However, the images need not be closed in infinite dimensions.

Therefore the corresponding statements are

$$\boxed{\begin{aligned}(\ker T)^\perp &= \overline{\operatorname{im} T^*},\\ (\ker T^*)^\perp &= \overline{\operatorname{im} T}.\end{aligned}}$$

The closure bars are essential.

We will return to this distinction when studying functional analysis.

Checkpoint 3

Starting only from the definition of an adjoint, prove

$$\ker T^* = (\operatorname{im} T)^\perp.$$

Then explain why the identity

$$\operatorname{im} T^* = (\ker T)^\perp$$

is closely related to the annihilator theorem from Module B.

Finally, derive the weighted-adjoint formula without memorizing it.

15.7 Session 4 — Orthogonal projections and closest-point approximation

In Module A, we studied direct-sum decompositions

$$V = K \oplus U$$

and the corresponding algebraic projections.

These projections are generally not unique.

Inner-product geometry gives us a distinguished choice: the projection along the orthogonal complement.

More importantly, it explains why this choice solves an optimization problem.

15.7.1 Orthogonal complements

Let $U \subseteq V$ be a subspace of an inner-product space.

Its orthogonal complement is

$$\boxed{U^\perp = \{x \in V : \langle x, u\rangle = 0 \quad \forall u \in U\}.}$$

It is a subspace.

In finite-dimensional inner-product spaces, we have

$$\boxed{V = U \oplus^\perp U^\perp.}$$

This means every vector $x \in V$ has a unique decomposition

$$\boxed{x = u + r}$$

with

$$u \in U, \qquad r \in U^\perp.$$

The vector u is the orthogonal projection of x onto U.

15.7.2 Constructing the orthogonal decomposition

Choose an orthonormal basis

$$(q_1, \dots, q_k)$$

of U.

Define

$$u = \sum_{i=1}^{k} \langle x, q_i\rangle q_i.$$

This vector belongs to U.

Let

$$r = x - u.$$

For every basis vector q_j,

$$\begin{aligned}\langle r, q_j \rangle &= \langle x, q_j \rangle - \sum_i \langle x, q_i \rangle \langle q_i, q_j \rangle \\ &= \langle x, q_j \rangle - \langle x, q_j \rangle \\ &= 0.\end{aligned}$$

Thus r is orthogonal to every basis vector of U, hence to every vector of U.

So

$$x = u + r, \qquad u \in U, \quad r \in U^{\perp}.$$

Uniqueness follows because

$$U \cap U^{\perp} = \{0\}.$$

This proves the orthogonal decomposition theorem in finite dimensions.

15.7.3 Why orthogonal projection gives the closest point

Suppose

$$x = u + r, \qquad u \in U, \quad r \perp U.$$

Take any other $z \in U$.

Then

$$x - z = (u - z) + r.$$

Since $u - z \in U$ and $r \perp U$,

$$\langle u - z, r \rangle = 0.$$

By the Pythagorean theorem,

$$\boxed{\|x - z\|^2 = \|u - z\|^2 + \|r\|^2.}$$

The second term is fixed, while the first is nonnegative.

The minimum occurs exactly when $z = u$.

Therefore,

$$\boxed{u = \underset{z \in U}{\operatorname{argmin}} \|x - z\|.}$$

Thus the orthogonal projection is the unique closest point to x in U.

This is one of the most important applications of inner-product geometry.

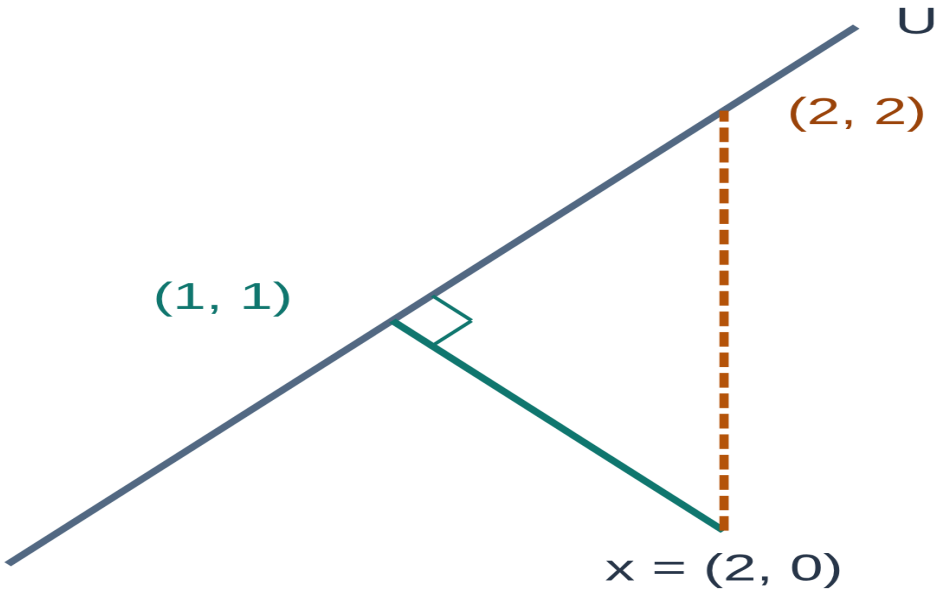


Figure 15.3: For x = (2, 0), the perpendicular to the diagonal U reaches (1, 1). Projecting along a vertical line reaches (2, 2). The two distances are $\sqrt{2}$ and 2.

In Figure 15.3, both projections map x into the same subspace U, but only the orthogonal projection $P_U x$ minimizes the distance to x.

15.7.4 Matrix of an orthogonal projection

Let $U \subseteq \mathbb{R}^n$ have an orthonormal basis consisting of columns of

$$Q \in \mathbb{R}^{n \times k}.$$

Then

$$Q^T Q = I_k.$$

The orthogonal projection onto U is

$$P_U x = \sum_{i=1}^{k} (q_i^T x) q_i.$$

In matrix notation,

$$\boxed{P_U = QQ^T.}$$

The matrix satisfies two fundamental properties:

$$\boxed{P_U^2 = P_U}$$

and

$$\boxed{P_U^T = P_U.}$$

Indeed,

$$P_U^2 = QQ^T QQ^T = Q(Q^T Q)Q^T = QQ^T,$$

and

$$P_U^T = (QQ^T)^T = QQ^T.$$

The first property says that P_U is a projection.

The second says that it is self-adjoint.

Characterization theorem

A real matrix P is an orthogonal projection if and only if

$$\boxed{P^2 = P, \qquad P^T = P.}$$

The forward implication follows from the construction above.

For the converse, suppose $P^2 = P = P^T$.

Module A gave

$$\mathbb{R}^n = \operatorname{im} P \oplus \ker P.$$

Now let $u = Px \in \operatorname{im} P$ and $k \in \ker P$.

Then

$$\langle u, k \rangle = \langle Px, k \rangle = \langle x, Pk \rangle = 0.$$

Thus the image and kernel of P are orthogonal.

Therefore P is the orthogonal projection onto its image.

15.7.5 Projection using a nonorthonormal basis

Suppose U has basis columns collected in

$$B \in \mathbb{R}^{n \times k},$$

where B has full column rank.

We seek the closest vector $Bc \in U$ to x.

The residual

$$r = x - Bc$$

must be orthogonal to U, so

$$B^T r = 0.$$

Therefore,

$$B^T(x - Bc) = 0,$$

giving

$$B^T Bc = B^T x.$$

Because the columns of B are independent, $B^T B$ is SPD and invertible.

Thus

$$c = (B^T B)^{-1} B^T x.$$

Substituting,

$$\boxed{P_U = B(B^T B)^{-1} B^T.}$$

This generalizes QQ^T to a nonorthonormal basis.

Notice that the Gram matrix $B^T B$ appears naturally.

It corrects for the fact that the basis columns are not orthonormal.

For numerical computation, one generally uses a QR factorization rather than explicitly forming the inverse of $B^T B$.

15.7.6 Orthogonal projection onto the kernel of our running matrix

Recall

$$A = \begin{pmatrix} 1 & 2 & -1 & 3 \\ 2 & 4 & 1 & 9 \\ 3 & 6 & 0 & 12 \end{pmatrix}$$

and the kernel basis matrix

$$K = \begin{pmatrix} -2 & -4 \\ 1 & 0 \\ 0 & -1 \\ 0 & 1 \end{pmatrix}.$$

The Gram matrix is

$$K^T K = \begin{pmatrix} 5 & 8 \\ 8 & 18 \end{pmatrix}.$$

Its determinant is

$$5 \cdot 18 - 8^2 = 26 > 0.$$

Therefore it is invertible, and

$$(K^T K)^{-1} = \frac{1}{26} \begin{pmatrix} 18 & -8 \\ -8 & 5 \end{pmatrix}.$$

The orthogonal projection onto $\ker A$ is

$$\boxed{P_{\ker A} = K(K^T K)^{-1} K^T.}$$

Compare this with the RREF-based projection from Module A:

$$P_{\text{RREF}} = \begin{pmatrix} 0 & -2 & 0 & -4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Both have image $\ker A$, and both satisfy

$$P^2 = P.$$

But they use different complementary subspaces.

For the RREF projection,

$$\ker P_{\text{RREF}} = \text{span}\{e_1, e_3\}.$$

For the orthogonal projection,

$$\ker P_{\ker A} = (\ker A)^\perp = \text{im}\, A^T.$$

Thus the two projections are generally different.

The RREF projection is an algebraically convenient coordinate choice.

The orthogonal projection is the geometrically distinguished choice determined by the inner product.

15.7.7 Orthogonal decompositions and quotient spaces

There is a particularly elegant connection back to Module A.

For every subspace $S \subseteq V$ in a finite-dimensional inner-product space,

$$V = S \oplus^{\perp} S^{\perp}.$$

Therefore every coset in V/S has a unique representative in $S^{\perp}$.

We obtain an isomorphism

$$\boxed{V/S \cong S^{\perp}.}$$

The quotient exists algebraically, without an inner product.

But the inner product provides a distinguished complementary subspace $S^{\perp}$.

Thus Riesz representation and orthogonal decomposition supply canonical representatives of quotient classes **relative to the chosen inner product**.

The isomorphism is not canonical if we forget the inner product.

Checkpoint 4

Prove that an orthogonal projection is self-adjoint and idempotent.

Then explain why the formula

$$B(B^T B)^{-1} B^T$$

requires the columns of B to be linearly independent.

Finally, explain how an inner product gives a distinguished realization of the quotient space V/S using $S^{\perp}$.

15.8 Session 5 — Self-adjoint operators, positive definiteness, and spectral geometry

We now investigate why symmetric and Hermitian matrices are so important in linear algebra and numerical computation.

The central reason is that self-adjointness connects the action of a linear operator with the geometry of the space.

This imposes strong structural restrictions on its eigenvalues, eigenspaces, and invariant subspaces.

15.8.1 Self-adjoint operators

Let $T : V \to V$ be a linear operator on a finite-dimensional inner-product space.

We call T **self-adjoint** if

$$\boxed{T = T^*.}$$

Equivalently,

$$\boxed{\langle Tx, y\rangle = \langle x, Ty\rangle \qquad \forall x, y \in V.}$$

The equation describes a symmetry between applying T to the first argument and applying it to the second.

In real orthonormal coordinates, self-adjointness means

$$\boxed{A = A^T.}$$

For complex matrices, it means

$$\boxed{A = A^H.}$$

Complex self-adjoint matrices are called **Hermitian**.

A subtle point about coordinates

A self-adjoint operator need not have a symmetric matrix in an arbitrary nonorthonormal basis.

Suppose the inner product has Gram matrix G.

Then

$$A^{*_G} = G^{-1}A^TG.$$

Self-adjointness means

$$A = G^{-1}A^TG.$$

Equivalently,

$$\boxed{A^TG = GA.}$$

This is the correct matrix characterization of self-adjointness in a general basis.

Thus **self-adjointness is a property of an operator relative to an inner product, not simply a property of its matrix entries**.

15.8.2 Example: a nonsymmetric matrix representing a self-adjoint operator

Consider

$$A = \begin{pmatrix} 2 & 1 \\ 2 & 3 \end{pmatrix}, \qquad G = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}.$$

Clearly,

$$A \neq A^T.$$

But

$$GA = \begin{pmatrix} 4 & 2 \\ 2 & 3 \end{pmatrix}$$

and

$$A^T G = \begin{pmatrix} 4 & 2 \\ 2 & 3 \end{pmatrix}.$$

Therefore,

$$A^T G = GA.$$

So A is self-adjoint with respect to the G-inner product, even though it is not symmetric in ordinary Euclidean coordinates.

Its eigenvalues are 1 and 4, both real.

This example is useful for generalized eigenvalue problems, where operators often become self-adjoint only after choosing a problem-dependent inner product.

15.8.3 Orthogonality of eigenvectors

Suppose T is self-adjoint and

$$Tv = \lambda v, \qquad Tw = \mu w.$$

Then

$$\begin{aligned} \lambda\langle v, w\rangle &= \langle Tv, w\rangle \\ &= \langle v, Tw\rangle \\ &= \mu\langle v, w\rangle. \end{aligned}$$

Therefore,

$$(\lambda - \mu)\langle v, w\rangle = 0.$$

If $\lambda \neq \mu$, we obtain

$$\boxed{\langle v, w\rangle = 0.}$$

Thus eigenvectors associated with distinct eigenvalues are orthogonal.

This is an important part of the geometric structure of self-adjoint operators.

For complex self-adjoint operators, the eigenvalues are real, and the same conclusion holds.

15.8.4 Why self-adjoint eigenvalues are real

Let T be self-adjoint on a complex inner-product space, and suppose

$$Tv = \lambda v, \qquad v \neq 0.$$

Then, using self-adjointness,

$$\langle Tv, v\rangle = \langle v, Tv\rangle.$$

Under our convention, the left side equals

$$\lambda \|v\|^2,$$

while the right side equals

$$\overline{\lambda}\|v\|^2.$$

Since $v \neq 0$,

$$\lambda = \overline{\lambda}.$$

Therefore,

$$\boxed{\lambda \in \mathbb{R}.}$$

This is another major advantage of Hermitian eigenvalue problems: the spectrum is real.

15.8.5 Invariant subspaces and their orthogonal complements

Recall Module A.

A subspace $U \subseteq V$ is invariant under T when

$$T(U) \subseteq U.$$

For a general operator, an invariant subspace need not have an invariant orthogonal complement.

For a self-adjoint operator, it does.

Theorem. If $T = T^*$ and U is invariant under T, then $U^\perp$ is invariant under T.

Proof

Take $x \in U^{\perp}$.

We want to show $Tx \in U^{\perp}$.

For any $u \in U$,

$$\langle Tx, u \rangle = \langle x, Tu \rangle.$$

Because U is invariant,

$$Tu \in U.$$

Since $x \in U^{\perp}$,

$$\langle x, Tu \rangle = 0.$$

Therefore,

$$\langle Tx, u \rangle = 0 \qquad \forall u \in U.$$

Thus

$$Tx \in U^{\perp}.$$

Hence

$$\boxed{T(U^{\perp}) \subseteq U^{\perp}.}$$

This leads to an orthogonal invariant decomposition

$$V = U \oplus^{\perp} U^{\perp}.$$

In a basis adapted to this decomposition, T has block-diagonal form.

This is the structural connection between self-adjointness, invariant subspaces, and spectral decomposition.

15.8.6 The spectral theorem

Finite-dimensional spectral theorem. Every self-adjoint operator T on a finite-dimensional real or complex inner-product space admits an orthonormal basis of eigenvectors.

For a real symmetric matrix,

$$\boxed{A = Q\Lambda Q^T,}$$

where Q is orthogonal and Λ is real diagonal.

For a complex Hermitian matrix,

$$\boxed{A = Q\Lambda Q^H,}$$

where Q is unitary.

The matrix Λ contains the eigenvalues, and the columns of Q are orthonormal eigenvectors.

Why is this theorem plausible?

We have already established two important facts.

First, distinct eigenspaces are orthogonal.

Second, the orthogonal complement of an invariant subspace is invariant.

Thus, once we find an eigenvector, we can separate its one-dimensional eigenspace from the rest of the space and continue studying the operator on the orthogonal complement.

This suggests an inductive construction.

Proof outline in the real case

Consider the Rayleigh quotient

$$\rho(x) = \frac{\langle Tx, x\rangle}{\langle x, x\rangle}, \qquad x \neq 0.$$

Restrict it to the unit sphere

$$\{x : \|x\| = 1\}.$$

Since the sphere is compact in finite dimensions and ρ is continuous, it attains a maximum at some unit vector v.

For symmetric T, stationarity of the quadratic form on the unit sphere implies

$$Tv = \lambda v$$

for some real λ.

Thus an eigenvector exists.

The subspace $\text{span}\{v\}$ is invariant.

Because T is self-adjoint, its orthogonal complement is also invariant.

We can restrict T to $v^{\perp}$, which has dimension one less, and repeat the argument.

By induction, we obtain an orthonormal eigenbasis of V.

The complex case follows similarly, using the real-valued Rayleigh quotient for Hermitian operators.

This outline explains why self-adjointness is crucial: it permits the invariant orthogonal decomposition needed for the induction.

15.8.7 Positive definite operators

A self-adjoint operator $T : V \to V$ is **positive definite** if

$$\boxed{\langle Tx, x\rangle > 0 \qquad \forall x \neq 0.}$$

It is **positive semidefinite** if

$$\boxed{\langle Tx, x\rangle \geq 0 \qquad \forall x.}$$

In real Euclidean coordinates, these conditions become

$$x^T Ax > 0$$

or

$$x^T Ax \geq 0.$$

For symmetric matrices, positive definiteness has a precise spectral characterization.

Theorem

Let $A = A^T \in \mathbb{R}^{n \times n}$.

Then

$$\boxed{A \text{ is positive definite } \iff \lambda_i(A) > 0 \quad \forall i.}$$

Proof

By the spectral theorem,

$$A = Q \Lambda Q^T.$$

Write $x = Qc$. Since Q is orthogonal,

$$x^T A x = c^T \Lambda c.$$

Therefore,

$$\boxed{x^T A x = \sum_{i=1}^{n} \lambda_i c_i^2.}$$

If every $\lambda_i > 0$, then this expression is positive whenever $c \neq 0$, equivalently $x \neq 0$.

Conversely, if A is positive definite, evaluate the quadratic form at any eigenvector q_i:

$$q_i^T A q_i = \lambda_i \|q_i\|^2 > 0.$$

Hence $\lambda_i > 0$.

The semidefinite analogue is

$$\boxed{A \succeq 0 \iff \lambda_i(A) \geq 0 \quad \forall i.}$$

Here $A \succeq 0$ denotes a symmetric positive semidefinite matrix.

15.8.8 Geometry of positive definiteness

Suppose A is SPD.

The quadratic form

$$E(x) = \frac{1}{2} x^T A x$$

can be interpreted as an energy.

Because A is positive definite,

$$E(x) > 0 \qquad \text{for } x \neq 0.$$

Its level sets

$$x^T A x = c$$

are ellipsoids for $c > 0$.

The eigenvectors determine their principal axes, while the eigenvalues determine how rapidly the energy increases along those directions.

In an orthonormal eigenbasis,

$$E(x) = \frac{1}{2} \sum_i \lambda_i c_i^2.$$

Thus large eigenvalues correspond to directions in which the energy increases rapidly.

Small positive eigenvalues correspond to directions in which it increases slowly.

This geometric picture is fundamental in optimization and numerical linear algebra.

15.8.9 The Rayleigh quotient and eigenvalue bounds

For a real symmetric matrix A, define

$$\boxed{\rho_A(x) = \frac{x^T A x}{x^T x}, \qquad x \neq 0.}$$

Write

$$x = \sum_i c_i q_i$$

in an orthonormal eigenbasis.

Then

$$\rho_A(x) = \frac{\sum_i \lambda_i c_i^2}{\sum_i c_i^2}.$$

The Rayleigh quotient is therefore a weighted average of the eigenvalues.

Consequently,

$$\boxed{\lambda_{\min} \leq \rho_A(x) \leq \lambda_{\max}.}$$

In particular,

$$\lambda_{\min} = \min_{x \neq 0} \rho_A(x),$$

$$\lambda_{\max} = \max_{x \neq 0} \rho_A(x).$$

This gives a variational characterization of the extreme eigenvalues.

The Rayleigh quotient is central to eigenvalue approximation, Rayleigh–Ritz methods, and the mathematical analysis of iterative eigensolvers.

15.8.10 The special operators T^*T and TT^*

Let

$$T : V \to W.$$

Consider

$$T^*T : V \to V.$$

This operator is self-adjoint:

$$(T^*T)^* = T^*(T^*)^* = T^*T.$$

Moreover,

$$\begin{aligned} \langle T^*Tx, x\rangle_V &= \langle Tx, Tx\rangle_W \\ &= \|Tx\|_W^2 \\ &\geq 0. \end{aligned}$$

Therefore,

$$\boxed{T^*T \text{ is positive semidefinite.}}$$

Similarly, TT^* is positive semidefinite.

We also have

$$\boxed{\ker(T^*T) = \ker T.}$$

To prove this, one inclusion is immediate: $Tx = 0$ implies $T^*Tx = 0$.

Conversely, if $T^*Tx = 0$, then

$$0 = \langle T^*Tx, x\rangle = \|Tx\|^2,$$

so $Tx = 0$.

Thus T^*T is positive definite precisely when T is injective.

Why this matters

The matrix A^TA plays a central role in least squares and the singular value decomposition.

Its eigenvalues are the squares of the singular values of A.

Consequently, the singular values describe how strongly A stretches vectors in orthogonal input directions.

The kernel of A corresponds to the directions associated with zero singular values.

This is a direct connection between Module A's kernel structure and Module C's positive semidefinite geometry.

Checkpoint 5

Prove that a real symmetric positive definite matrix is invertible.

Then prove

$$\ker(A^TA) = \ker A.$$

Finally, explain why the quadratic form x^TAx for SPD A can be interpreted as a geometric energy, and how its eigenvalues describe different directional scales.

15.9 Session 6 — Least squares, pseudoinverses, and numerical stability

Least squares is perhaps the clearest example of how inner-product geometry becomes a computational algorithm.

The problem is familiar:

Given

$$A \in \mathbb{R}^{m\times n}, \qquad b \in \mathbb{R}^m,$$

we want to solve

$$Ax = b.$$

But what if the system is inconsistent?

The geometry of orthogonal projection tells us what to do.

15.9.1 Formulating the least-squares problem

Instead of requiring exact equality, seek

$$\boxed{x_{\mathrm{LS}} \in \operatorname*{argmin}_{x \in \mathbb{R}^n} \|Ax - b\|_2^2.}$$

The vector Ax always belongs to the column space

$$U = \operatorname{im} A.$$

Thus minimizing $\|Ax - b\|$ is equivalent to finding the point in U closest to b.

But we already know the solution to that geometric problem.

The closest point is the orthogonal projection of b onto U.

Therefore, every least-squares minimizer satisfies

$$\boxed{Ax_{\mathrm{LS}} = P_{\operatorname{im} A} b.}$$

15.9.2 Deriving the normal equations geometrically

Define the residual

$$r = b - Ax_{\mathrm{LS}}.$$

Because Ax_{LS} is the orthogonal projection of b onto $\operatorname{im} A$,

$$\boxed{r \perp \operatorname{im} A.}$$

In particular, r is orthogonal to every column of A.

This is equivalent to

$$A^T r = 0.$$

Substituting $r = b - Ax_{\mathrm{LS}}$,

$$A^T(b - Ax_{\mathrm{LS}}) = 0.$$

Hence

$$\boxed{A^T A x_{\mathrm{LS}} = A^T b.}$$

These are the **normal equations**.

Notice how naturally the transpose appears.

It arises because the residual must belong to the orthogonal complement of the image:

$$r \in (\operatorname{im} A)^{\perp} = \ker A^T.$$

This is exactly one of the fundamental orthogonality identities from Session 3.

15.9.3 Alternative derivation using optimization

Define

$$F(x) = \frac{1}{2}\|Ax - b\|_2^2.$$

Expanding,

$$F(x) = \frac{1}{2}(Ax - b)^T (Ax - b).$$

The derivative in a direction h is

$$DF(x)[h] = (Ax - b)^T Ah.$$

Using the adjoint identity,

$$DF(x)[h] = h^T A^T (Ax - b).$$

Therefore the Euclidean gradient is

$$\boxed{\nabla F(x) = A^T (Ax - b).}$$

A minimizer must satisfy

$$\nabla F(x) = 0,$$

giving

$$A^T A x = A^T b.$$

Thus the same equations can be derived in two ways:

Geometrically, through orthogonal projection.

Analytically, through the Riesz representative of the derivative functional.

This is a particularly elegant meeting point of the ideas developed throughout the module.

15.9.4 Worked least-squares example

Let

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{pmatrix}, \qquad b = \begin{pmatrix} 1 \\ 2 \\ 4 \end{pmatrix}.$$

The system

$$Ax = b$$

requires

$$x_1 = 1, \qquad x_2 = 2, \qquad x_1 + x_2 = 4.$$

These equations are inconsistent because $1 + 2 \neq 4$.

We therefore solve the least-squares problem.

First,

$$A^T A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}.$$

Also,

$$A^T b = \begin{pmatrix} 5 \\ 6 \end{pmatrix}.$$

The normal equations become

$$\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 5 \\ 6 \end{pmatrix}.$$

The inverse is

$$(A^T A)^{-1} = \frac{1}{3} \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}.$$

Therefore,

$$\boxed{x_{\text{LS}} = \begin{pmatrix} 4/3 \\ 7/3 \end{pmatrix}.}$$

The corresponding output is

$$Ax_{\text{LS}} = \begin{pmatrix} 4/3 \\ 7/3 \\ 11/3 \end{pmatrix}.$$

The residual is

$$r = b - Ax_{\text{LS}} = \boxed{\begin{pmatrix} -1/3 \\ -1/3 \\ 1/3 \end{pmatrix}.}$$

Verify:

$$A^T r = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} -1/3 \\ -1/3 \\ 1/3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Thus the residual is orthogonal to the column space, as required.

15.9.5 When is the least-squares solution unique?

The least-squares problem always has at least one minimizer in finite dimensions.

Why?

Because the orthogonal projection

$$P_{\operatorname{im} A} b$$

always exists and belongs to $\operatorname{im} A$.

Therefore at least one x satisfies

$$Ax = P_{\operatorname{im} A} b.$$

But the minimizer need not be unique.

If x_0 is a minimizer and $k \in \ker A$, then

$$A(x_0 + k) = Ax_0.$$

Therefore $x_0 + k$ has the same residual and the same least-squares objective value.

Thus all minimizers form an affine space:

$$\boxed{x_0 + \ker A.}$$

The minimizer is unique precisely when

$$\boxed{\ker A = \{0\}.}$$

Equivalently, A must have full column rank.

In that case, $A^T A$ is SPD, so the normal equations have the unique solution

$$\boxed{x_{\text{LS}} = (A^T A)^{-1} A^T b.}$$

This formula is mathematically valid, but is not usually the preferred way to compute the solution numerically.

15.9.6 Minimum-norm solutions and the pseudoinverse

Suppose A has a nontrivial kernel.

Then there are infinitely many least-squares minimizers.

The inner product lets us select one distinguished minimizer: the one with smallest Euclidean norm.

Let x_0 be any least-squares minimizer.

Using the orthogonal decomposition

$$\mathbb{R}^n = \ker A \oplus^{\perp} (\ker A)^{\perp},$$

write

$$x_0 = k + u,$$

where

$$k \in \ker A, \qquad u \in (\ker A)^{\perp}.$$

Since $Ak = 0$,

$$Au = Ax_0.$$

Thus u is also a least-squares minimizer.

Moreover,

$$\|x_0\|^2 = \|k\|^2 + \|u\|^2.$$

Therefore,

$$\|u\| \leq \|x_0\|,$$

with equality only when $k = 0$.

So u is the unique minimum-norm least-squares solution.

We obtain the characterization:

$$\boxed{x_{\min} \in (\ker A)^{\perp}, \qquad Ax_{\min} = P_{\operatorname{im} A} b.}$$

Recall that

$$(\ker A)^{\perp} = \operatorname{im} A^T.$$

Thus the minimum-norm solution lies in the row space.

The Moore–Penrose pseudoinverse

The map sending b to this unique minimum-norm least-squares solution is represented by the **Moore–Penrose pseudoinverse**, denoted A^+:

$$\boxed{x_{\min} = A^+ b.}$$

For a full-column-rank matrix,

$$A^+ = (A^T A)^{-1} A^T.$$

For a general matrix, the pseudoinverse is most naturally understood using the singular value decomposition.

If

$$A = U \Sigma V^T$$

is an SVD, then

$$A^+ = V \Sigma^+ U^T,$$

where Σ^+ replaces each positive singular value by its reciprocal and transposes the rectangular diagonal structure, leaving zero singular values at zero.

This connects minimum-norm least squares to spectral structure.

15.9.7 Why numerical methods avoid forming normal equations unnecessarily

There is an important numerical issue.

Suppose A has full column rank, with singular values

$$\sigma_{\max}, \qquad \sigma_{\min} > 0.$$

Its 2-norm condition number is

$$\kappa_2(A) = \frac{\sigma_{\max}}{\sigma_{\min}}.$$

The eigenvalues of $A^T A$ are the squares of the singular values of A.

Therefore,

$$\boxed{\kappa_2(A^T A) = \kappa_2(A)^2.}$$

If A is already ill-conditioned, forming the normal equations can substantially worsen the conditioning of the linear system being solved.

Moreover, computing $A^T A$ explicitly introduces additional rounding errors.

For this reason, QR factorization and SVD-based methods are generally preferred for stable least-squares computation.

Geometric versus numerical perspective

The normal equations are an excellent way to understand the mathematical structure of least squares.

But they are not always the best computational algorithm.

This distinction illustrates an important principle of numerical analysis:

A mathematically correct characterization of a solution need not lead to a numerically stable way of computing it.

Understanding both the structure and its numerical realization is one of the overarching goals of our curriculum.

15.9.8 Weighted least squares

Suppose the residual space has an SPD weight matrix W.

Consider

$$\min_x \|Ax - b\|_W^2,$$

where

$$\|r\|_W^2 = r^T W r.$$

The residual at the optimum must be orthogonal to $\operatorname{im} A$ in the weighted inner product.

Thus

$$\langle Ax - b, Ah \rangle_W = 0 \qquad \forall h.$$

Expanding,

$$h^T A^T W (Ax - b) = 0 \qquad \forall h.$$

Therefore,

$$\boxed{A^T W (Ax - b) = 0.}$$

The weighted normal equations are

$$\boxed{A^T W A x = A^T W b.}$$

This is the same geometric principle, but in a different inner product.

Weighted least squares appears in statistical estimation, inverse problems, data assimilation, and numerical optimization.

It provides another example of how SPD matrices define the geometry of a numerical problem.

Checkpoint 6

Derive the normal equations in two different ways: first from orthogonal projection, then by differentiating the least-squares objective.

Prove that a least-squares minimizer is unique if and only if $\ker A = \{0\}$.

Finally, explain why forming $A^T A$ may be undesirable in floating-point arithmetic even though the normal equations are mathematically correct.

15.10 Session 7 — Functional analysis, variational methods, and numerical PDEs

We have now developed the main finite-dimensional theory.

But the same concepts become particularly powerful when the vectors are functions and the linear transformations are differential or integral operators.

This is the setting of functional analysis and much of numerical PDE theory.

Our main example will be the weak formulation of a boundary-value problem and its approximation using finite elements.

15.10.1 From finite-dimensional vectors to functions

Consider the Hilbert space

$$H = L^2(0,1)$$

with inner product

$$\boxed{\langle u, v \rangle_{L^2} = \int_0^1 u(x) v(x)\, dx.}$$

The corresponding norm is

$$\|u\|_{L^2} = \left(\int_0^1 |u(x)|^2\, dx \right)^{1/2}.$$

The elements are functions, or more precisely equivalence classes of functions equal almost everywhere.

The notions of orthogonality and projection still make sense.

For example, two functions are orthogonal when

$$\int_0^1 u(x)v(x)\,dx = 0.$$

The geometry is no longer visualized by arrows in ordinary Euclidean space, but the same inner-product identities apply.

15.10.2 Continuous dual spaces

In finite-dimensional vector spaces, every linear functional is automatically continuous.

In infinite-dimensional normed spaces, this is no longer true.

Functional analysis therefore generally studies the **continuous dual**

$$V' = \{f : V \to \mathbb{R} : f \text{ is linear and continuous}\}.$$

A linear functional f is continuous if and only if it is bounded, meaning there exists $C \geq 0$ such that

$$|f(v)| \leq C\|v\| \qquad \forall v \in V.$$

For a Hilbert space H, the Riesz representation theorem states that every element of its continuous dual has a unique representing vector.

Thus

$$\boxed{H' \cong H}$$

through the Riesz identification.

This is one reason Hilbert spaces are particularly convenient for studying linear operator equations.

15.10.3 A boundary-value problem

Consider the one-dimensional Poisson equation

$$\boxed{-u''(x) = f(x), \qquad 0 < x < 1,}$$

with homogeneous Dirichlet boundary conditions

$$u(0) = u(1) = 0.$$

We want to understand how this differential equation leads to a linear system involving an SPD matrix.

Step 1: Multiply by a test function

Choose a sufficiently smooth function v with

$$v(0) = v(1) = 0.$$

Multiply the equation by v and integrate:

$$-\int_0^1 u''(x)v(x)\,dx = \int_0^1 f(x)v(x)\,dx.$$

Step 2: Integrate by parts

Integration by parts gives

$$-\int_0^1 u''v = -\left[u'v\right]_0^1 + \int_0^1 u'v'.$$

Because v vanishes at both endpoints, the boundary term is zero.

Therefore,

$$\boxed{\int_0^1 u'(x)v'(x)\,dx = \int_0^1 f(x)v(x)\,dx.}$$

This is the **weak formulation** of the Poisson equation.

It requires fewer derivatives of u than the original differential equation.

15.10.4 The correct function-space setting

The natural space for this problem is

$$V = H_0^1(0,1),$$

the Sobolev space of functions with a weak first derivative in L^2 and zero boundary trace.

Define

$$a(u,v) = \int_0^1 u'(x)v'(x)\,dx.$$

Define the load functional

$$F(v) = \int_0^1 f(x)v(x)\,dx.$$

The weak problem becomes:

Find $u \in V$ such that

$$\boxed{a(u,v) = F(v) \qquad \forall v \in V.}$$

Assume, for example, $f \in L^2(0,1)$.

Then F is a continuous linear functional on V.

The bilinear form a is symmetric:

$$a(u,v) = a(v,u).$$

Furthermore,

$$a(u,u) = \int_0^1 |u'(x)|^2\,dx.$$

On $H_0^1(0,1)$, this expression is strictly positive for $u \neq 0$, because a function with zero weak derivative is constant, and a constant with zero boundary values must be zero.

Thus a defines an inner product on V.

Its norm,

$$\|u\|_a = \sqrt{a(u,u)},$$

is equivalent to the usual H^1 norm on $H_0^1(0,1)$, by the Poincaré inequality.

15.10.5 Riesz representation as a PDE solvability principle

The weak problem asks for $u \in V$ such that

$$a(u,v) = F(v) \qquad \forall v \in V.$$

But a is an inner product on the Hilbert space V, and F is a continuous linear functional.

Therefore the Riesz representation theorem guarantees a unique $u \in V$ satisfying

$$\boxed{F(v) = a(v,u) \qquad \forall v \in V.}$$

Because a is symmetric, this is exactly the weak formulation.

So, in this particular problem, **Riesz representation provides an existence-and-uniqueness theorem for the weak solution**.

This is a striking application: a theorem about representing linear measurements becomes a theorem about solving a differential equation.

For more general continuous, coercive bilinear forms that need not be symmetric, the corresponding existence-and-uniqueness result is the **Lax–Milgram theorem**.

That theorem is a major foundation of numerical PDE analysis.

15.10.6 The operator maps into the dual space

Recall Module B's variational viewpoint.

For each $u \in V$, define a functional

$$\mathcal{A}u \in V'$$

by

$$(\mathcal{A}u)(v) = a(u, v).$$

Thus

$$\boxed{\mathcal{A} : V \to V'.}$$

The weak problem becomes

$$\boxed{\mathcal{A}u = F.}$$

The operator does not naturally map a function to another ordinary function in this formulation.

It maps a function to a continuous linear functional.

Riesz representation allows us to identify the dual with the original Hilbert space relative to an appropriate inner product.

This is precisely the kind of connection between duality, adjoints, and PDEs that motivated our curriculum.

15.10.7 Finite-dimensional Galerkin approximation

In numerical computation, we cannot generally work with every function in an infinite-dimensional space.

Instead, choose a finite-dimensional subspace

$$V_h \subseteq V.$$

Let

$$V_h = \text{span}\{\phi_1, \dots, \phi_n\}.$$

We seek an approximation

$$u_h = \sum_{j=1}^{n} c_j \phi_j$$

satisfying

$$\boxed{a(u_h, v_h) = F(v_h) \qquad \forall v_h \in V_h.}$$

It suffices to test against the basis functions:

$$a(u_h, \phi_i) = F(\phi_i).$$

Substituting the expansion,

$$\sum_{j=1}^{n} c_j a(\phi_j, \phi_i) = F(\phi_i).$$

Define the stiffness matrix

$$\boxed{K_{ij} = a(\phi_j, \phi_i)}$$

and the load vector

$$b_i = F(\phi_i).$$

Then

$$\boxed{Kc = b.}$$

We have converted an infinite-dimensional operator equation into a finite-dimensional linear system.

15.10.8 Why the stiffness matrix is SPD

Let $c \in \mathbb{R}^n$, and define

$$v_h = \sum_i c_i \phi_i.$$

Then

$$\begin{aligned} c^T K c &= \sum_{i,j} c_i c_j a(\phi_j, \phi_i) \\ &= a(v_h, v_h). \end{aligned}$$

Since a is an inner product,

$$a(v_h, v_h) > 0$$

whenever $v_h \neq 0$.

Because the basis functions are linearly independent, $c \neq 0$ implies $v_h \neq 0$.

Therefore,

$$\boxed{c^T K c > 0 \qquad \forall c \neq 0.}$$

The matrix is symmetric because a is symmetric.

Hence K is SPD.

This is the structural origin of positive definite stiffness matrices in finite-element methods.

They inherit their geometry from the underlying variational problem.

15.10.9 A concrete finite-element example

Consider

$$-u''(x) = 1, \qquad 0 < x < 1,$$

with

$$u(0) = u(1) = 0.$$

Divide the interval into three equal elements, each of length

$$h = \frac{1}{3}.$$

Use continuous, piecewise-linear basis functions associated with the two interior nodes $x = 1/3$ and $x = 2/3$.

There are two unknown coefficients:

$$u_h = c_1\phi_1 + c_2\phi_2.$$

The stiffness matrix is

$$K = \frac{1}{h}\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} = \boxed{\begin{pmatrix} 6 & -3 \\ -3 & 6 \end{pmatrix}}.$$

For the constant forcing $f = 1$, the load vector is

$$b = \begin{pmatrix} \int_0^1 \phi_1(x)\,dx \\ \int_0^1 \phi_2(x)\,dx \end{pmatrix} = \begin{pmatrix} 1/3 \\ 1/3 \end{pmatrix}.$$

Thus

$$\begin{pmatrix} 6 & -3 \\ -3 & 6 \end{pmatrix}\begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} 1/3 \\ 1/3 \end{pmatrix}.$$

Solving gives

$$\boxed{c_1 = c_2 = \frac{1}{9}.}$$

These are the approximate solution values at the two interior nodes.

For comparison, the exact solution is

$$u(x) = \frac{1}{2}x(1-x).$$

At $x = 1/3$ and $x = 2/3$,

$$u(1/3) = u(2/3) = \frac{1}{9}.$$

In this simple one-dimensional example, the finite-element nodal values agree with the exact solution at the nodes.

The finite-element function itself is still piecewise linear, while the exact solution is quadratic.

15.10.10 Galerkin orthogonality

Let $u \in V$ be the exact weak solution and $u_h \in V_h$ the Galerkin approximation.

We have

$$a(u, v) = F(v) \qquad \forall v \in V,$$

and

$$a(u_h, v_h) = F(v_h) \qquad \forall v_h \in V_h.$$

Subtracting gives

$$\boxed{a(u - u_h, v_h) = 0 \qquad \forall v_h \in V_h.}$$

This is called **Galerkin orthogonality**.

Let

$$e = u - u_h$$

be the error.

Then the error is orthogonal to the approximation space V_h with respect to the energy inner product a.

Consequently, u_h is the orthogonal projection of u onto V_h in the energy geometry.

It follows that

$$\boxed{\|u - u_h\|_a = \min_{v_h \in V_h} \|u - v_h\|_a.}$$

Proof of the best-approximation property

Take any $v_h \in V_h$.

Write

$$u - v_h = (u - u_h) + (u_h - v_h).$$

The first term is the error e.

The second belongs to V_h.

By Galerkin orthogonality,

$$a(e, u_h - v_h) = 0.$$

Thus the Pythagorean theorem in the a-inner product gives

$$\|u - v_h\|_a^2 = \|u - u_h\|_a^2 + \|u_h - v_h\|_a^2.$$

Therefore,

$$\|u - v_h\|_a^2 \geq \|u - u_h\|_a^2.$$

Equality occurs at $v_h = u_h$.

This proves the best-approximation property.

Why this is important

We have connected an infinite-dimensional PDE problem to the same geometric principle used in ordinary least squares.

Least squares	Galerkin approximation
Approximate a vector b by an element of $\operatorname{im} A$	Approximate a function u by an element of V_h
Residual is orthogonal to the column space	Error is orthogonal to V_h in the energy inner product
Normal equations arise	Stiffness matrix equations arise
Best approximation in Euclidean norm	Best approximation in energy norm

This is a particularly clear example of mathematical transfer: the same projection geometry reappears in a much more advanced setting.

15.10.11 When the variational form is only semidefinite

A useful contrast is the Poisson problem with homogeneous Neumann conditions:

$$-u'' = f, \qquad u'(0) = u'(1) = 0.$$

Its weak form uses the same bilinear form

$$a(u, v) = \int_0^1 u'v'.$$

But now the function space does not require u to vanish at the endpoints.

Every constant function satisfies

$$a(c, v) = 0 \qquad \forall v.$$

Thus the bilinear form is positive semidefinite, not positive definite, on the full space.

The solution is determined only up to an additive constant, and a compatibility condition is necessary:

$$\boxed{\int_0^1 f(x)\,dx = 0.}$$

This follows by testing the weak equation with the constant function $v = 1$.

The quotient by constant functions removes the nonuniqueness.

This returns us to Module A's quotient-space construction, now in a PDE setting.

It also illustrates why numerical discretizations of Neumann problems produce singular positive semidefinite matrices.

Checkpoint 7

Explain why the finite-element stiffness matrix is SPD for the homogeneous Dirichlet Poisson problem.

Derive Galerkin orthogonality and use it to prove best approximation in the energy norm.

Then explain why replacing Dirichlet boundary conditions with Neumann conditions introduces a constant-function kernel and a compatibility condition.

15.11 Session 8 — Generalized eigenvalue problems and numerical spectral methods

The final session connects inner-product geometry directly to the Numerical Linear Algebra and HASC topics that motivated the curriculum.

Our main ideas are generalized self-adjointness, spectral projections, Rayleigh–Ritz approximation, and polynomial filtering.

15.11.1 Why generalized eigenvalue problems arise

In elementary linear algebra, an eigenvalue problem has the form

$$Ax = \lambda x.$$

But numerical methods for PDEs often produce a generalized eigenvalue problem

$$\boxed{Kx = \lambda Mx,}$$

where K and M are matrices arising from bilinear forms.

For example, in a finite-element discretization of

$$-u'' = \lambda u$$

with homogeneous Dirichlet conditions, one obtains

$$K_{ij} = \int_0^1 \phi_j'(x)\phi_i'(x)\,dx,$$

and

$$M_{ij} = \int_0^1 \phi_j(x)\phi_i(x)\,dx.$$

Here K is a stiffness matrix and M is a mass matrix.

Both are symmetric, and in the standard conforming Dirichlet setting they are SPD.

The matrix M is the Gram matrix of the basis with respect to the L^2 inner product.

Thus generalized eigenvalue problems arise naturally when different bilinear forms describe different aspects of the underlying function-space geometry.

15.11.2 Transforming the generalized problem

Assume M is SPD.

We can rewrite

$$Kx = \lambda Mx$$

as

$$\boxed{M^{-1}Kx = \lambda x.}$$

Define

$$T = M^{-1}K.$$

The matrix T need not be symmetric in ordinary Euclidean coordinates.

However, it is self-adjoint with respect to the M-inner product.

Let's prove this.

For $x, y \in \mathbb{R}^n$,

$$\begin{aligned}\langle Tx, y\rangle_M &= (M^{-1}Kx)^T My \\ &= x^T K^T y \\ &= x^T Ky,\end{aligned}$$

since $K = K^T$.

On the other hand,

$$\begin{aligned}\langle x, Ty\rangle_M &= x^T M(M^{-1}Ky) \\ &= x^T Ky.\end{aligned}$$

Therefore,

$$\boxed{\langle Tx, y\rangle_M = \langle x, Ty\rangle_M.}$$

Thus T is self-adjoint in the M-geometry.

This explains why generalized symmetric eigenvalue problems retain the favorable spectral properties of ordinary symmetric problems.

15.11.3 Orthogonality of generalized eigenvectors

Suppose

$$Kv_i = \lambda_i M v_i,$$

$$Kv_j = \lambda_j M v_j.$$

Multiply the first equation by v_j^T:

$$v_j^T K v_i = \lambda_i v_j^T M v_i.$$

Similarly,

$$v_i^T K v_j = \lambda_j v_i^T M v_j.$$

Because K and M are symmetric, the left-hand sides agree, and so do the M-inner products.

Thus

$$(\lambda_i - \lambda_j) v_i^T M v_j = 0.$$

If $\lambda_i \neq \lambda_j$,

$$\boxed{v_i^T M v_j = 0.}$$

Therefore generalized eigenvectors associated with distinct eigenvalues are orthogonal in the M-inner product.

They need not be orthogonal in the ordinary Euclidean inner product.

This is exactly the distinction between intrinsic geometric structure and coordinate representation developed earlier in the module.

15.11.4 The generalized Rayleigh quotient

For a nonzero vector x, define

$$\boxed{\rho_{K,M}(x) = \frac{x^T K x}{x^T M x}.}$$

Because M is SPD,

$$x^T M x > 0.$$

If x is a generalized eigenvector,

$$Kx = \lambda M x,$$

then

$$\rho_{K,M}(x) = \lambda.$$

More generally, the quotient plays the same role as the ordinary Rayleigh quotient, but in the geometry defined by M.

If K is symmetric and M is SPD, the generalized eigenvalues are real, and the extreme values satisfy variational characterizations analogous to the ordinary symmetric case.

This is particularly important in finite-element eigenvalue problems.

15.11.5 Rayleigh–Ritz approximation

Suppose $A = A^T \in \mathbb{R}^{n\times n}$.

Let $U \subseteq \mathbb{R}^n$ be a k-dimensional trial subspace with orthonormal basis columns

$$Q \in \mathbb{R}^{n\times k}, \qquad Q^TQ = I.$$

We seek an approximate eigenvector $v \in U$.

Write

$$v = Qy.$$

The Rayleigh–Ritz method constructs the smaller matrix

$$\boxed{H = Q^TAQ.}$$

We then solve

$$Hy = \theta y.$$

The scalar θ is a **Ritz value**, and

$$v = Qy$$

is a **Ritz vector**.

Where does this reduced eigenproblem come from?

We would like the residual

$$r = Av - \theta v$$

to vanish.

But because v is restricted to the trial subspace, we generally cannot achieve $r = 0$.

Instead, impose the condition

$$\boxed{r \perp U.}$$

Since the columns of Q span U, this is equivalent to

$$Q^T r = 0.$$

Substituting $v = Qy$,

$$Q^T(AQy - \theta Qy) = 0.$$

Therefore,

$$Q^T AQy = \theta Q^T Qy.$$

Since $Q^T Q = I$,

$$\boxed{Q^T AQy = \theta y.}$$

This is exactly the projected eigenvalue problem.

Rayleigh–Ritz approximation arises by requiring the eigenpair residual to be orthogonal to the trial subspace.

This is closely related to Galerkin orthogonality for PDE approximation.

Both methods project an infinite- or high-dimensional problem onto a smaller trial space and impose an orthogonality condition on the residual.

15.11.6 Generalized Rayleigh–Ritz approximation

For

$$Kx = \lambda Mx,$$

choose a trial subspace with basis columns Q.

Seek

$$v = Qy.$$

The generalized residual is

$$r = KQy - \theta MQy.$$

Impose

$$Q^T r = 0.$$

Then

$$\boxed{Q^T K Q y = \theta Q^T M Q y.}$$

Thus Rayleigh–Ritz produces a smaller generalized eigenvalue problem.

If the columns of Q are M-orthonormal, so that

$$Q^T M Q = I,$$

the equation simplifies to

$$\boxed{Q^T K Q y = \theta y.}$$

The requirement $Q^T r = 0$ means that the residual vector represents the zero functional on the trial subspace under the ordinary coordinate pairing. Equivalently, the operator residual

$$M^{-1} r$$

is M-orthogonal to the trial subspace.

This is a direct application of the weighted inner-product theory.

15.11.7 Polynomial filtering and self-adjoint spectral theory

Suppose A is Hermitian.

The spectral theorem gives

$$A = Q \Lambda Q^H.$$

For a polynomial p,

$$\boxed{p(A) = Q p(\Lambda) Q^H.}$$

To understand this, note that

$$A^k = Q\Lambda^k Q^H.$$

Therefore,

$$\begin{aligned} p(A) &= \sum_{j=0}^{d} a_j A^j \\ &= Q\left(\sum_{j=0}^{d} a_j \Lambda^j\right) Q^H \\ &= Qp(\Lambda)Q^H. \end{aligned}$$

Thus, for an eigenvector q_i,

$$p(A)q_i = p(\lambda_i)q_i.$$

A polynomial of the operator preserves the eigendirections and changes the scaling associated with each eigenvalue.

This is the fundamental mechanism of polynomial spectral filtering.

15.11.8 Why Chebyshev polynomials are useful

Chebyshev polynomials have controlled behavior on $[-1, 1]$.

The first-kind Chebyshev polynomials satisfy

$$T_0(t) = 1, \qquad T_1(t) = t,$$

and the recurrence

$$\boxed{T_{k+1}(t) = 2tT_k(t) - T_{k-1}(t).}$$

For $t \in [-1, 1]$,

$$|T_k(t)| \leq 1.$$

Outside that interval, their magnitude can grow rapidly with k.

Suppose the unwanted eigenvalues of a Hermitian matrix lie in an interval $[a, b]$.

Map that interval to $[-1, 1]$ using

$$s(t) = \frac{2t - (a+b)}{b-a}.$$

Then a filter of the form

$$p_k(t) = T_k(s(t))$$

is bounded in magnitude by 1 on the unwanted interval, while it can grow substantially at desired eigenvalues outside that interval.

After suitable scaling or normalization, this can create strong relative amplification of desired spectral components.

The operator filter is

$$p_k(A) = T_k\left(\frac{2A - (a+b)I}{b-a}\right).$$

In practice, we generally do not form this matrix polynomial explicitly.

Instead, we apply the Chebyshev recurrence to vectors or blocks of vectors.

This allows high-degree polynomial filtering using repeated matrix-vector or matrix-block products.

15.11.9 Connection to HASC and hardware-aware computing

The underlying mathematical structure explains why Chebyshev-filtered subspace iteration is effective.

A Hermitian matrix has an orthonormal eigenbasis.

Polynomial functions of that matrix act diagonally on its spectral components.

Orthogonalization keeps the evolving trial basis well-conditioned.

Rayleigh–Ritz extracts approximate eigenpairs from the trial subspace.

Residuals measure how far the approximate eigenpairs are from satisfying the original eigenproblem.

From a numerical perspective, however, we must distinguish exact arithmetic from finite-precision computation.

In floating-point arithmetic, matrix-vector products, polynomial recurrences, and orthogonalization steps introduce errors.

These errors may alter convergence behavior, especially when the desired spectral components are weak, eigenvalues are clustered, or a polynomial recurrence strongly amplifies some directions.

This creates the mathematical setting for questions in hardware-aware scientific computing: how should one balance arithmetic precision, communication costs, memory bandwidth, numerical stability, and the accuracy required for convergence?

The tools developed in Module C provide the geometric language needed to analyze those questions.

Mathematical concept	Role in an eigenvalue algorithm
Self-adjointness	Real spectrum and orthogonal spectral decomposition
Orthogonal projection	Approximation within trial subspaces
Rayleigh quotient	Eigenvalue estimates
Rayleigh–Ritz	Extraction of approximate eigenpairs
Adjoint identities	Residual orthogonality and projected equations
Positive definiteness	Weighted geometry and generalized eigenproblems
Spectral theorem	Justification of polynomial filtering
Condition numbers and singular values	Sensitivity and finite-precision behavior

The detailed convergence and stability analysis belongs to Modules D and E.

Checkpoint 8

Prove that $M^{-1}K$ is self-adjoint in the M-inner product when K is symmetric and M is SPD.

Then derive the Rayleigh–Ritz residual orthogonality condition and the associated projected eigenvalue problem.

Finally, explain why polynomial filtering is especially transparent for Hermitian matrices.

15.12 Exercises — Module C

These exercises are designed to test the six forms of understanding established in the curriculum: rigorous reasoning, geometric intuition, computation, structural connections, abstraction, and applications.

Attempt each exercise independently before consulting the solutions.

15.12.1 Exercise 1 — Inner products and Gram matrices

Let

$$G = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}.$$

Define

$$\langle x, y \rangle_G = x^T G y.$$

Prove that this is an inner product on $\mathbb{R}^2$.

Find all vectors y that are G-orthogonal to

$$x = (1, 0)^T.$$

Explain why the resulting orthogonal complement differs from the Euclidean orthogonal complement of x.

15.12.2 Exercise 2 — Riesz representation in a weighted geometry

Let

$$G = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$$

and define the functional

$$f(x_1, x_2) = 3x_1.$$

Find its Riesz representative a_f with respect to the G-inner product.

Verify explicitly that

$$f(x) = \langle x, a_f \rangle_G.$$

Calculate the norm of the functional and verify

$$\|f\|_{V^*} = \|a_f\|_G.$$

15.12.3 Exercise 3 — A weighted adjoint

Let

$$A = \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}.$$

Equip the domain and codomain with inner products represented by

$$G_V = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, \qquad G_W = \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix}.$$

Calculate the adjoint $A^{*_{G_V, G_W}}$.

Verify the defining identity for

$$x = (1, 2)^T, \qquad y = (3, 1)^T.$$

Then explain why the ordinary transpose would give the wrong answer in this geometry.

15.12.4 Exercise 4 — Orthogonal and oblique projections

Let

$$U = \operatorname{span}\{(1, 1)^T\} \subseteq \mathbb{R}^2.$$

Find the matrix of the Euclidean orthogonal projection onto U.

Project

$$x = (2, 0)^T.$$

Now construct the projection onto U along

$$W = \operatorname{span}\{(0,1)^T\}.$$

Compare the two projected vectors and their distances from x.

Verify that both matrices are idempotent but only one is symmetric.

15.12.5 Exercise 5 — Self-adjointness and positive definiteness

Consider

$$A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}.$$

Calculate its eigenvalues and orthonormal eigenvectors.

Prove that A is SPD.

Find the minimum and maximum possible values of its Rayleigh quotient.

Calculate the Rayleigh quotient of $x = (1,0)^T$.

15.12.6 Exercise 6 — Rank-deficient least squares

Let

$$A = \begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix}, \qquad b = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Find the orthogonal projection of b onto $\operatorname{im} A$.

Describe all least-squares minimizers.

Then find the unique minimizer with smallest Euclidean norm.

Explain how the kernel and its orthogonal complement determine the answer.

15.12.7 Exercise 7 — A finite-element stiffness matrix

Consider the weak problem

$$\int_0^1 u'v' = \int_0^1 v$$

for all $v \in H_0^1(0,1)$.

Use a mesh with nodes

$$0, \frac{1}{3}, \frac{2}{3}, 1$$

and continuous piecewise-linear basis functions ϕ_1, ϕ_2 corresponding to the two interior nodes.

Show that the stiffness matrix and load vector are

$$K = \begin{pmatrix} 6 & -3 \\ -3 & 6 \end{pmatrix}, \qquad b = \begin{pmatrix} 1/3 \\ 1/3 \end{pmatrix}.$$

Solve $Kc = b$.

Prove that K is SPD without relying solely on its eigenvalues.

15.12.8 Exercise 8 — Generalized eigenvalue geometry

Let

$$K = \begin{pmatrix} 2 & 0 \\ 0 & 6 \end{pmatrix}, \qquad M = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}.$$

Find the generalized eigenvalues of

$$Kx = \lambda M x.$$

Show that

$$T = M^{-1}K$$

is self-adjoint in the M-inner product.

Find an M-orthonormal eigenbasis and express the generalized Rayleigh quotient in that basis.

15.12.9 Exercise 9 — Infinite-dimensional orthogonality

Consider the Hilbert space

$$H = \ell^2$$

of square-summable real sequences.

Define

$$T : \ell^2 \to \ell^2, \qquad T(x_1, x_2, ...) = \left(x_1, \frac{x_2}{2}, \frac{x_3}{3}, ...\right).$$

Show that T is bounded, injective, and self-adjoint.

Show that the image of T is dense but not closed.

Explain why

$$\operatorname{im} T \neq (\ker T)^{\perp}$$

even though

$$\overline{\operatorname{im} T} = (\ker T)^{\perp}.$$

15.12.10 Exercise 10 — Structural synthesis

Starting from

$$T : V \to W,$$

explain the logical relationship among the following ideas:

$$\operatorname{im} T^{\vee} = (\ker T)^{0},$$

$$\operatorname{im} T^{*} = (\ker T)^{\perp},$$

$$V = \ker T \oplus^{\perp} \operatorname{im} T^{*},$$

and

$$V / \ker T \cong \operatorname{im} T.$$

Explain which results are purely algebraic and which depend on an inner product.

Finally, explain why the restriction

$$T|_{(\ker T)^{\perp}} : (\ker T)^{\perp} \to \operatorname{im} T$$

is an isomorphism in finite dimensions, and whether it must be an isometry.

15.13 Worked solutions

15.13.1 Solution 1 — Weighted geometry

The matrix G is symmetric.

Its eigenvalues are 3 and 1, both positive.

Therefore G is SPD, and

$$\langle x, y \rangle_G = x^T G y$$

is an inner product.

Now let $x = (1, 0)^T$ and $y = (y_1, y_2)^T$.

Then

$$\begin{aligned} \langle x, y \rangle_G &= \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \\ &= 2y_1 + y_2. \end{aligned}$$

Thus $x \perp_G y$ if and only if

$$2y_1 + y_2 = 0.$$

Therefore,

$$\boxed{x^{\perp_G} = \operatorname{span}\{(1,-2)^T\}.}$$

By contrast, the Euclidean orthogonal complement of x is

$$x^{\perp} = \operatorname{span}\{(0,1)^T\}.$$

The difference occurs because orthogonality depends on the chosen inner product.

15.13.2 Solution 2 — Weighted Riesz representative

The functional has coefficient column

$$c = \begin{pmatrix} 3 \\ 0 \end{pmatrix}.$$

Its Riesz representative satisfies

$$Ga_f = c.$$

Since

$$G^{-1} = \frac{1}{3}\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix},$$

we obtain

$$a_f = G^{-1}c = \boxed{\begin{pmatrix} 2 \\ -1 \end{pmatrix}.}$$

Verification:

$$Ga_f = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}\begin{pmatrix} 2 \\ -1 \end{pmatrix} = \begin{pmatrix} 3 \\ 0 \end{pmatrix}.$$

Therefore,

$$\langle x, a_f \rangle_G = x^T G a_f = 3x_1 = f(x).$$

The norm of the representative is

$$\begin{aligned}\|a_f\|_G^2 &= a_f^T G a_f \\ &= \begin{pmatrix}2 & -1\end{pmatrix}\begin{pmatrix}3\\0\end{pmatrix} \\ &= 6.\end{aligned}$$

Thus

$$\boxed{\|a_f\|_G = \sqrt{6}.}$$

By the Riesz isometry theorem,

$$\boxed{\|f\|_{V^*} = \sqrt{6}.}$$

15.13.3 Solution 3 — Weighted adjoint

We use

$$A^{*_{G_V,G_W}} = G_V^{-1} A^T G_W.$$

First,

$$A^T G_W = \begin{pmatrix}1 & 0\\2 & 1\end{pmatrix}\begin{pmatrix}1 & 0\\0 & 3\end{pmatrix} = \begin{pmatrix}1 & 0\\2 & 3\end{pmatrix}.$$

Also,

$$G_V^{-1} = \begin{pmatrix}1/2 & 0\\0 & 1\end{pmatrix}.$$

Therefore,

$$\boxed{A^{*_{G_V,G_W}} = \begin{pmatrix}1/2 & 0\\2 & 3\end{pmatrix}.}$$

Now take $x = (1,2)^T$, $y = (3,1)^T$.

We have

$$Ax = (5, 2)^T.$$

Thus

$$\langle Ax, y\rangle_{G_W} = \begin{pmatrix} 5 & 2 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix} \begin{pmatrix} 3 \\ 1 \end{pmatrix} = 21.$$

On the other side,

$$A^{*G_V, G_W} y = \begin{pmatrix} 3/2 \\ 9 \end{pmatrix}.$$

Therefore,

$$\langle x, A^{*G_V, G_W} y\rangle_{G_V} = \begin{pmatrix} 1 & 2 \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 3/2 \\ 9 \end{pmatrix} = 21.$$

The defining identity holds.

The ordinary transpose is not the correct adjoint because the inner products are not Euclidean.

15.13.4 Solution 4 — Orthogonal versus oblique projections

The subspace U has basis vector

$$u = (1, 1)^T.$$

The Euclidean orthogonal projection is

$$P_U = \frac{uu^T}{u^T u} = \boxed{\frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}.}$$

For $x = (2, 0)^T$,

$$P_U x = (1, 1)^T.$$

Its distance from x is

$$\|x - P_U x\| = \|(1, -1)\| = \sqrt{2}.$$

Now project onto U along

$$W = \text{span}\{(0, 1)^T\}.$$

Every vector decomposes as

$$(x_1, x_2) = (x_1, x_1) + (0, x_2 - x_1).$$

Therefore the oblique projection is

$$Q(x_1, x_2) = (x_1, x_1),$$

with matrix

$$\boxed{Q = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}.}$$

Then

$$Qx = (2, 2)^T.$$

Its distance from x is

$$\|x - Qx\| = 2.$$

Thus

$$\sqrt{2} < 2,$$

confirming that the orthogonal projection is closer.

Both satisfy

$$P_U^2 = P_U, \qquad Q^2 = Q.$$

But

$$P_U^T = P_U,$$

while

$$Q^T \neq Q.$$

15.13.5 Solution 5 — SPD matrix and Rayleigh quotient

For

$$A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix},$$

the characteristic polynomial is

$$\det(A - \lambda I) = (2 - \lambda)^2 - 1.$$

Its roots are

$$\boxed{\lambda_1 = 1, \qquad \lambda_2 = 3.}$$

Corresponding orthonormal eigenvectors are

$$q_1 = \frac{1}{\sqrt{2}}(1, -1)^T,$$

$$q_2 = \frac{1}{\sqrt{2}}(1, 1)^T.$$

Since A is symmetric and both eigenvalues are positive, it is SPD.

The Rayleigh quotient satisfies

$$\boxed{1 \leq \rho_A(x) \leq 3.}$$

The minimum and maximum are attained at q_1 and q_2, respectively.

For $x = (1, 0)^T$,

$$\rho_A(x) = \frac{x^T A x}{x^T x} = \boxed{2}.$$

15.13.6 Solution 6 — Rank-deficient least squares

The columns of A are identical:

$$a_1 = a_2 = (1,2)^T.$$

Thus

$$\operatorname{im} A = \operatorname{span}\{(1,2)^T\}.$$

The orthogonal projection of $b = (1,0)^T$ onto this line is

$$P_{\operatorname{im} A} b = \frac{(1,2)^T b}{(1,2)^T (1,2)} (1,2)^T.$$

Since

$$(1,2)^T b = 1, \qquad (1,2)^T (1,2) = 5,$$

we obtain

$$\boxed{P_{\operatorname{im} A} b = \begin{pmatrix} 1/5 \\ 2/5 \end{pmatrix}.}$$

But

$$Ax = (x_1 + x_2)(1,2)^T.$$

Therefore every least-squares minimizer satisfies

$$x_1 + x_2 = \frac{1}{5}.$$

The full solution family is

$$\boxed{x = \begin{pmatrix} 1/10 \\ 1/10 \end{pmatrix} + t \begin{pmatrix} 1 \\ -1 \end{pmatrix}.}$$

The kernel is

$$\ker A = \operatorname{span}\{(1,-1)^T\}.$$

Its orthogonal complement is

$$(\ker A)^\perp = \operatorname{span}\{(1,1)^T\}.$$

The minimum-norm solution must lie in this orthogonal complement.

Therefore its two coordinates must be equal.

Together with $x_1 + x_2 = 1/5$, this gives

$$\boxed{x_{\min} = \begin{pmatrix} 1/10 \\ 1/10 \end{pmatrix}.}$$

15.13.7 Solution 7 — Finite-element stiffness matrix

For continuous piecewise-linear hat functions on a uniform mesh with spacing $h = 1/3$, each interior basis function has derivative magnitude $1/h = 3$ on its two adjacent elements.

Therefore,

$$a(\phi_i, \phi_i) = 2h \left(\frac{1}{h}\right)^2 = \frac{2}{h} = 6.$$

The two neighboring basis functions overlap on one element, where their derivatives have opposite signs.

Thus

$$a(\phi_1, \phi_2) = -h \left(\frac{1}{h}\right)^2 = -\frac{1}{h} = -3.$$

Hence

$$\boxed{K = \begin{pmatrix} 6 & -3 \\ -3 & 6 \end{pmatrix}.}$$

Each hat function has area $h = 1/3$, so for forcing $f = 1$,

$$\boxed{b = \begin{pmatrix} 1/3 \\ 1/3 \end{pmatrix}.}$$

Solving gives

$$\boxed{c = \begin{pmatrix} 1/9 \\ 1/9 \end{pmatrix}.}$$

To prove that K is SPD structurally, let

$$v_h = c_1\phi_1 + c_2\phi_2.$$

Then

$$c^T K c = \int_0^1 |v_h'(x)|^2 \, dx.$$

This is nonnegative.

If it equals zero, $v_h' = 0$ almost everywhere, so v_h is constant.

Because v_h vanishes at both endpoints, it must be identically zero.

Independence of the basis functions implies $c = 0$.

Therefore $c^T K c > 0$ for every nonzero c, proving positive definiteness.

15.13.8 Solution 8 — Generalized eigenvalues

The generalized eigenvalue equation is

$$\begin{pmatrix} 2 & 0 \\ 0 & 6 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \lambda \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}.$$

The two equations are

$$2x_1 = 2\lambda x_1,$$

$$6x_2 = \lambda x_2.$$

Therefore,

$$\boxed{\lambda_1 = 1, \qquad \lambda_2 = 6.}$$

We have

$$T = M^{-1}K = \begin{pmatrix} 1 & 0 \\ 0 & 6 \end{pmatrix}.$$

It is self-adjoint in the M-inner product because

$$T^T M = MT = K.$$

An M-orthonormal eigenbasis is

$$q_1 = \frac{1}{\sqrt{2}}(1,0)^T, \qquad q_2 = (0,1)^T.$$

Indeed,

$$q_i^T M q_j = \delta_{ij}.$$

For $x = c_1 q_1 + c_2 q_2$,

$$x^T M x = c_1^2 + c_2^2,$$

and

$$x^T K x = c_1^2 + 6c_2^2.$$

Thus

$$\boxed{\rho_{K,M}(x) = \frac{c_1^2 + 6c_2^2}{c_1^2 + c_2^2}.}$$

This is a weighted average of the generalized eigenvalues.

15.13.9 Solution 9 — Dense but nonclosed range

For $x \in \ell^2$,

$$\|Tx\|_{\ell^2}^2 = \sum_{n=1}^{\infty} \frac{|x_n|^2}{n^2} \leq \sum_{n=1}^{\infty} |x_n|^2 = \|x\|_{\ell^2}^2.$$

Thus T is bounded, with $\|T\| \leq 1$.

It is injective because

$$Tx = 0 \implies x_n/n = 0 \quad \forall n,$$

so $x = 0$.

It is self-adjoint because

$$\langle Tx, y\rangle = \sum_{n=1}^{\infty} \frac{x_n y_n}{n} = \langle x, Ty\rangle.$$

Every finitely supported sequence belongs to the image: if y has only finitely many nonzero entries, choose $x_n = ny_n$.

Since finitely supported sequences are dense in ℓ^2, the image is dense.

However, consider

$$y = \left(1, \frac{1}{2}, \frac{1}{3}, ...\right).$$

We have

$$\sum_{n=1}^{\infty} \frac{1}{n^2} < \infty,$$

so $y \in \ell^2$.

If $Tx = y$, then

$$x_n = ny_n = 1$$

for every n.

But $(1, 1, ...) \notin \ell^2$.

Therefore $y \notin \operatorname{im} T$.

Its finite truncations do belong to the image and converge to y, so the image is not closed.

Since $\ker T = \{0\}$,

$$(\ker T)^\perp = \ell^2.$$

But

$$\operatorname{im} T \neq \ell^2.$$

Hence

$$\boxed{\operatorname{im} T \neq (\ker T)^\perp, \qquad \overline{\operatorname{im} T} = (\ker T)^\perp.}$$

This demonstrates why closures matter in infinite-dimensional operator theory.

15.13.10 Solution 10 — Structural synthesis

The identity

$$\operatorname{im} T^\vee = (\ker T)^0$$

is purely algebraic.

It concerns the algebraic dual map and annihilator, neither of which requires an inner product.

Once inner products are chosen, the Riesz representation theorem identifies the dual spaces with the original spaces.

Under these identifications,

$$T^\vee$$

corresponds to the adjoint T^*, and

$$(\ker T)^0$$

corresponds to

$$(\ker T)^\perp.$$

Thus

$$\boxed{\operatorname{im} T^* = (\ker T)^\perp}$$

in finite dimensions.

The orthogonal decomposition theorem gives

$$\boxed{V = \ker T \oplus^\perp (\ker T)^\perp = \ker T \oplus^\perp \operatorname{im} T^*.}$$

Independently, the first isomorphism theorem gives

$$\boxed{V/\ker T \cong \operatorname{im} T.}$$

The inner product allows us to identify the quotient with the orthogonal complement of the kernel:

$$V/\ker T \cong (\ker T)^\perp.$$

Therefore,

$$\boxed{T|_{(\ker T)^\perp} : (\ker T)^\perp \to \operatorname{im} T}$$

is an isomorphism.

It is injective because its kernel is

$$(\ker T)^\perp \cap \ker T = \{0\}.$$

It is surjective because every $v \in V$ decomposes as

$$v = k + u, \qquad k \in \ker T, \quad u \in (\ker T)^\perp,$$

and

$$Tv = Tu.$$

However, the restriction need not be an isometry.

It preserves vector-space structure but may change lengths.

For example, multiplication by 2 on $\mathbb{R}$ is an isomorphism whose kernel is zero, but

$$\|2x\| = 2\|x\|.$$

Thus the structural isomorphism theorem does not imply preservation of geometric distances.

15.14 Computational laboratory — MATLAB

This laboratory is designed to make the mathematical relationships concrete.

Each experiment corresponds to one or more sessions. The calculations use small matrices for which the theoretical results can also be verified by hand.

The code is intended to run as a MATLAB script.

```
%% Module C - Inner-Product Geometry
clear; clc;

%% 1. Riesz representation in a weighted inner product
G = [2 1;
     1 2];

% Functional f(x) = 3*x1
c = [3; 0];

% Riesz representative: G*a = c
a = G \ c;

x = [1; 2];

fprintf('Riesz representative:\n');
disp(a);

fprintf('Functional value: %.8f\n', c.'*x);
fprintf('Inner-product value: %.8f\n', x.'*G*a);
```

```
fprintf('Functional norm: %.8f\n', ...
    sqrt(c.'*(G\c)));

fprintf('Representative norm: %.8f\n', ...
    sqrt(a.'*G*a));


%% 2. Adjoint with weighted inner products
A = [1 2;
     0 1];

Gv = diag([2, 1]);
Gw = diag([1, 3]);

% Adjoint matrix: Gv^{-1} * A^T * Gw
A_adj = Gv \ (A.'*Gw);

x = [1; 2];
y = [3; 1];

left  = (A*x).'*Gw*y;
right = x.'*Gv*(A_adj*y);

fprintf('Weighted adjoint error: %.3e\n', ...
    abs(left-right));


%% 3. Orthogonal and oblique projections
u = [1; 1];

P_orth = (u*u.')/(u.'*u);

% Projection onto span(u) along span(e2)
P_oblique = [1 0;
             1 0];

x = [2; 0];

fprintf('Orthogonal projection:\n');
disp(P_orth*x);

fprintf('Oblique projection:\n');
```

```
disp(P_oblique*x);

fprintf('Orthogonal distance: %.8f\n', ...
    norm(x-P_orth*x));

fprintf('Oblique distance: %.8f\n', ...
    norm(x-P_oblique*x));

fprintf('||P_orth^2-P_orth|| = %.3e\n', ...
    norm(P_orth^2-P_orth));

fprintf('||P_orth^T-P_orth|| = %.3e\n', ...
    norm(P_orth.'-P_orth));


%% 4. Orthogonal projection onto a null space
A = [1 2 -1 3;
     2 4  1 9;
     3 6  0 12];

K = [-2 -4;
      1  0;
      0 -1;
      0  1];

% Orthogonal projector onto ker(A)
P_kernel = K * ((K.'*K)\K.');

fprintf('||P_kernel^2-P_kernel|| = %.3e\n', ...
    norm(P_kernel^2-P_kernel));

fprintf('||P_kernel^T-P_kernel|| = %.3e\n', ...
    norm(P_kernel.'-P_kernel));

fprintf('||A*P_kernel|| = %.3e\n', ...
    norm(A*P_kernel));


%% 5. SPD matrices and spectral decomposition
Aspd = [2 1;
        1 2];
```

```
[Q,Lambda] = eig(Aspd);

fprintf('Eigenvalues of SPD matrix:\n');
disp(diag(Lambda));

fprintf('||A-Q*Lambda*Q^T|| = %.3e\n', ...
    norm(Aspd-Q*Lambda*Q.'));

x = [1; 0];

rho = (x.'*Aspd*x)/(x.'*x);
fprintf('Rayleigh quotient: %.8f\n', rho);

% Cholesky factorization (upper triangular R)
R = chol(Aspd);

fprintf('||A-R^T*R|| = %.3e\n', ...
    norm(Aspd-R.'*R));


%% 6. Least squares and normal equations
Als = [1 0;
       0 1;
       1 1];

b = [1; 2; 4];

% MATLAB backslash uses a suitable numerical solver
xls = Als \ b;

r = b-Als*xls;

fprintf('Least-squares solution:\n');
disp(xls);

fprintf('Residual:\n');
disp(r);

fprintf('||A^T*r|| = %.3e\n', ...
    norm(Als.'*r));

% Compare against the normal-equation solution
```

```
x_normal = (Als.'*Als)\(Als.'*b);

fprintf('||x_ls-x_normal|| = %.3e\n', ...
    norm(xls-x_normal));


%% 7. Rank-deficient minimum-norm least squares
Arank = [1 1;
         2 2];

brank = [1; 0];

x_min = pinv(Arank)*brank;

fprintf('Minimum-norm solution:\n');
disp(x_min);

fprintf('Kernel orthogonality: %.3e\n', ...
    abs([1 -1]*x_min));


%% 8. Finite-element stiffness matrix
Kfe = [6 -3;
      -3  6];

bfe = [1/3; 1/3];

cfe = Kfe\bfe;

fprintf('FE nodal values:\n');
disp(cfe);

fprintf('Smallest stiffness eigenvalue: %.8f\n', ...
    min(eig(Kfe)));


%% 9. Generalized eigenproblem
Kgen = diag([2, 6]);
Mgen = diag([2, 1]);

[V,D] = eig(Kgen,Mgen);
```

```
fprintf('Generalized eigenvalues:\n');
disp(diag(D));

T = Mgen\Kgen;

fprintf('M-self-adjointness residual: %.3e\n', ...
    norm(T.'*Mgen-Mgen*T));


%% 10. Rayleigh-Ritz in a one-dimensional subspace
q = [1; 1]/sqrt(3);  % M-normalized for M=diag(2,1)

theta = q.'*Kgen*q;

r = Kgen*q-theta*Mgen*q;

fprintf('Ritz value: %.8f\n', theta);
fprintf('Galerkin residual orthogonality: %.3e\n', ...
    abs(q.'*r));
```

15.14.1 Interpretation of the laboratory

The most important observations are not the individual numerical answers, but the relationships they verify.

Experiment	Mathematical structure to recognize
Riesz representation	$f(x) = \langle x, a_f \rangle_G$, with $a_f = G^{-1}c$
Weighted adjoint	$\langle Ax, y \rangle_{G_W} = \langle x, A^{*G_V, G_W} y \rangle_{G_V}$
Orthogonal projection	Idempotence, self-adjointness, and best approximation
Kernel projection	$\operatorname{im} P = \ker A$, with $P^2 = P = P^T$
SPD matrix	Positive eigenvalues, Cholesky factorization, and spectral decomposition
Least squares	The residual belongs to $\ker A^T$
Minimum-norm solution	The selected solution belongs to $(\ker A)^\perp$
Finite elements	A coercive energy form leads to an SPD stiffness matrix
Generalized eigenproblem	Self-adjointness in a weighted inner product
Rayleigh–Ritz	The approximate eigenpair residual annihilates the trial subspace

For larger numerical experiments, avoid using explicit matrix inverses merely because they appear in mathematical formulas.

For example, solve

$$Ga = c$$

using a suitable factorization instead of forming G^{-1}.

Similarly, solve least-squares problems using QR or SVD-based methods rather than explicitly forming normal equations when conditioning is a concern.

This distinction between **the mathematical characterization of a solution** and **a stable computational implementation** is a recurring theme of numerical linear algebra.

15.15 Final synthesis — What Module C establishes

Module C has added geometric structure to the algebraic framework developed in Modules A and B.

The central relationship is

Algebraic dual space V^*
$\updownarrow$ Riesz representation
Inner-product space V

Through this identification, the dual map becomes the adjoint, and annihilators correspond to orthogonal complements.

This leads to orthogonal decompositions and the numerical algorithms that exploit them.

15.15.1 The conceptual chain

Inner products

$\langle x, y\rangle$ — Lengths, angles, orthogonality

Riesz representation and adjoints

$f(x) = \langle x, a_f\rangle$ — Covectors represented by vectors

Orthogonal decompositions and projections

$V = U \oplus^{\perp} U^{\perp}$ — Best approximation and least squares

Self-adjointness and positive definiteness

$T = T^*, \qquad \langle Tx, x\rangle > 0$ — Real spectrum, orthogonal eigenspaces, and energy geometry

Spectral and variational numerical methods

Least squares · Galerkin approximation · Rayleigh–Ritz · Chebyshev filtering

15.15.2 Important relationships to retain

Concept	Main result	Interpretation
Inner product	$\langle x, y\rangle_G = x^T G y$	Defines geometry
Gram matrix	$G_{ij} = \langle v_i, v_j\rangle$	Coordinate representation of geometry
Riesz theorem	$f(x) = \langle x, a_f\rangle$	Represents functionals by vectors
Adjoint	$\langle Tx, y\rangle = \langle x, T^* y\rangle$	Pulls output measurements backward
Weighted adjoint	$A^{*G_V, G_W} = G_V^{-1} A^T G_W$	Adjoint in non-Euclidean geometry
Fundamental subspaces	$\operatorname{im} T^* = (\ker T)^\perp$	Algebraic duality becomes orthogonality
Orthogonal projection	$P^2 = P = P^*$	Closest-point approximation
Self-adjointness	$T = T^*$	Symmetry relative to the inner product
Positive definiteness	$\langle Tx, x\rangle > 0$	Strictly positive quadratic energy
Normal equations	$A^T(Ax - b) = 0$	Least-squares residual orthogonality
Galerkin orthogonality	$a(u - u_h, v_h) = 0$	Best approximation in an energy norm
Generalized eigenproblem	$Kx = \lambda M x$	Spectral theory in a weighted geometry
Rayleigh–Ritz	$Q^T A Q y = \theta y$	Projected eigenvalue approximation

15.15.3 Final mastery checklist

Before beginning Module D, aim to be able to reconstruct the principal results independently and explain how they connect.

- ☐ Prove Cauchy–Schwarz and explain how an inner product induces norm and distance.
- ☐ Construct Gram matrices and explain why SPD matrices define inner products.
- ☐ Prove finite-dimensional Riesz representation and calculate weighted Riesz representatives.
- ☐ Derive adjoints from Riesz representation and compute weighted adjoints.
- ☐ Prove the fundamental orthogonality identities involving kernels, images, and adjoints.
- ☐ Construct orthogonal projectors, prove their best-approximation property, and distinguish them from oblique projections.
- ☐ Explain self-adjointness, the spectral theorem, positive definiteness, and the Rayleigh quotient.
- ☐ Derive least-squares normal equations and the minimum-norm solution characterization.
- ☐ Explain the role of Riesz representation, energy norms, and Galerkin orthogonality in a PDE approximation.
- ☐ Derive the Rayleigh–Ritz equations and explain the connection to generalized eigenproblems and polynomial filtering.

15.16 Transition to Module D — Spectral theory

Module C has introduced the essential geometry of self-adjoint operators and the spectral theorem.

Module D will build on these results to study spectral theory systematically.

In particular, we will investigate invariant subspaces, spectral decomposition, the Rayleigh quotient, Rayleigh–Ritz approximation, singular value decomposition, spectral projections, and the behavior of eigenvalues and eigenspaces under perturbations.

The central transition is from understanding the geometry of an operator to understanding how its spectrum organizes the entire space.

For a Hermitian matrix,

$$A = Q\Lambda Q^H.$$

This decomposition means that the action of A can be understood through independent orthogonal spectral directions.

That leads directly to methods for identifying selected eigenvalues and eigenspaces.

In particular, a polynomial filter satisfies

$$p(A) = Qp(\Lambda)Q^H.$$

Thus the same spectral decomposition that explains the geometry of a self-adjoint operator also explains the mechanism of Chebyshev-filtered subspace iteration.

The subsequent numerical questions concern how accurately the desired invariant subspaces can be approximated, how residuals and spectral gaps control approximation quality, and how finite-precision operations affect convergence.

The main achievement of Module C is to understand that orthogonality, adjoints, projections, positive definiteness, and spectral geometry are not isolated constructions. They arise when the algebraic theory of linear maps and duality is enriched by an inner product.

That is the foundation for the spectral and numerical methods developed in Modules D and E.

16 Module D — Spectral Theory

Eigenvalues, invariant subspaces, spectral decompositions, Rayleigh quotients, Rayleigh–Ritz approximation, singular values, and spectral projections

Advanced undergraduate · 8 study sessions · Approximately 18–24 hours

Prerequisites	Modules A–C: linear maps, kernels, direct sums, duality, inner products, adjoints, orthogonal projections, and positive definiteness
Primary objective	Understand how a linear operator organizes a space into invariant components and how those components can be approximated numerically
Applications	Numerical eigensolvers, low-rank approximation, PDE eigenvalue problems, spectral filtering, and scientific computing
Next module	Module E: Numerical Eigenvalue Methods, Conditioning, and Stability

16.1 Introduction — Understanding an operator through its spectrum

In Module A, we studied a linear transformation through its kernel, image, and the ways its domain can be decomposed.

In Module B, we developed algebraic duality, linear measurements, transpose maps, and annihilators.

In Module C, an inner product allowed us to study geometry: orthogonality, adjoints, projections, self-adjointness, and positive definiteness.

Module D brings these ideas together by asking how a linear operator acts on distinguished directions and subspaces.

For a square matrix A, an eigenvector $v \neq 0$ satisfies

$$Av = \lambda v.$$

Along such a direction, an otherwise complicated linear transformation reduces to multiplication by a scalar.

But individual eigenvectors are only the beginning. In numerical linear algebra, we often want to approximate an entire invariant subspace, not merely a single eigenvector.

The central questions are:

- When can a linear operator be decomposed into simpler invariant components?
- What makes the spectral theory of Hermitian matrices especially favorable?
- How do eigenvalues and eigenspaces relate to optimization and approximation?
- Why are residuals and spectral gaps important when computing eigenvectors numerically?
- How does the singular value decomposition extend spectral geometry to rectangular matrices?
- Why do polynomial filters, especially Chebyshev filters, reveal desired eigenspaces?

The mathematical centerpiece will be the spectral decomposition of a Hermitian matrix:

$$\boxed{A = Q\Lambda Q^*.}$$

Here Q is unitary, its columns are orthonormal eigenvectors, and Λ contains the eigenvalues.

This representation gives meaning to matrix functions:

$$\boxed{p(A) = Qp(\Lambda)Q^*.}$$

It is also the algebraic foundation of Chebyshev-filtered subspace iteration.

16.2 Our six perspectives

Perspective	Questions we will answer
Rigorous	Under what assumptions do spectral decompositions exist, and what can be proved about them?
Intuitive and geometric	How does a transformation stretch, rotate, mix, or preserve different subspaces?
Computational	How do we approximate eigenvalues, eigenvectors, invariant subspaces, and singular values?
Structural	How do kernels, direct sums, projections, adjoints, and quotient operators appear in spectral theory?
Universal	How do invariant decompositions and spectral projectors relate to algebraic functional calculus and module structure?
Applied	How does spectral theory underpin numerical eigensolvers, finite elements, PDEs, and hardware-aware computation?

16.3 Study roadmap

Session	Topic	Main result
1	Eigenvalues, eigenspaces, and invariant subspaces	Diagonalizability and invariant decompositions
2	Spectral theorem and matrix functions	$A = Q\Lambda Q^*$, $f(A)$, spectral projections
3	Geometry of spectral subspaces	Principal angles, spectral gaps, and invariant-subspace residuals
4	Rayleigh quotients and Rayleigh–Ritz	Min–max principle, Ritz values, residual bounds
5	Singular value decomposition	SVD, pseudoinverse, and optimal low-rank approximation
6	The bridge to eigenvalue algorithms	Power iteration, subspace iteration, and Ritz extraction
7	Polynomial and Chebyshev filtering	Spectral amplification and subspace convergence
8	Functional analysis and structural generalizations	Compact operators, PDE spectra, and functional calculus

We will concentrate on finite-dimensional real and complex inner-product spaces. The infinite-dimensional theory will be introduced carefully in Session 8.

Notation: A^* means conjugate transpose, reducing to A^T for real matrices. All norms $\|\cdot\|_2$ are Euclidean vector norms or induced matrix 2-norms unless stated otherwise.

16.4 Session 1 — Eigenvalues, eigenspaces, and invariant subspaces

16.4.1 From arbitrary transformations to operators

An arbitrary linear map may have different domain and codomain:

$$T : V \to W.$$

To discuss ordinary eigenvalues, we normally require a linear **operator**:

$$\boxed{T : V \to V.}$$

Why?

An eigenvector equation compares Tv with λv:

$$Tv = \lambda v.$$

Both vectors must belong to the same space.

For a rectangular map $V \to W$, this equation is not intrinsically defined without an additional identification between the spaces.

Rectangular matrices instead admit singular value decompositions, which we will study in Session 5.

16.4.2 Eigenvalues as failures of invertibility

Let $T : V \to V$ be linear.

A scalar λ is an **eigenvalue** if there exists $v \neq 0$ such that

$$Tv = \lambda v.$$

Equivalently,

$$(T - \lambda I)v = 0$$

for some nonzero v.

Thus

$$\boxed{\lambda \text{ is an eigenvalue } \iff \ker(T - \lambda I) \neq \{0\}.}$$

The corresponding **eigenspace** is

$$\boxed{E_\lambda = \ker(T - \lambda I).}$$

This connects spectral theory directly to the kernels studied in Module A.

For finite-dimensional V,

$$\lambda \text{ is an eigenvalue } \iff T - \lambda I \text{ is not invertible.}$$

If A is a matrix representing T, then

$$\boxed{\det(A - \lambda I) = 0.}$$

The determinant expression is a computational method for small matrices. The deeper definition is that $T - \lambda I$ has a nontrivial kernel.

Zero eigenvalues and singular matrices

A matrix can certainly have eigenvalues without being invertible.

In fact,

$$\boxed{0 \text{ is an eigenvalue of } A \iff A \text{ is singular.}}$$

Its zero eigenspace is exactly the null space:

$$E_0 = \ker A.$$

This means that the spectral description of a singular matrix includes its lost directions.

16.4.3 Real versus complex eigenvalues

Consider the real rotation matrix

$$R_\theta = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}.$$

For a rotation angle other than a multiple of π, there is no nonzero real vector whose image remains on its original line.

Thus there are no real eigenvectors.

However, over $\mathbb{C}$, the eigenvalues are

$$\boxed{\lambda_\pm = e^{\pm i\theta}.}$$

For example,

$$R_{\pi/2} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

has eigenvalues i and $-i$.

One corresponding eigenvector for i is

$$v_+ = \begin{pmatrix} 1 \\ -i \end{pmatrix}.$$

Indeed,

$$R_{\pi/2} v_+ = \begin{pmatrix} i \\ 1 \end{pmatrix} = i \begin{pmatrix} 1 \\ -i \end{pmatrix}.$$

So an operator's eigenstructure can depend on the underlying scalar field.

For an $n \times n$ complex matrix, the characteristic polynomial has n roots counting algebraic multiplicity.

But having n eigenvalues counting multiplicity does **not** imply that the matrix has n independent eigenvectors.

That requires diagonalizability.

16.4.4 Algebraic and geometric multiplicity

For an eigenvalue λ, define:

The **algebraic multiplicity** a_λ is its multiplicity as a root of the characteristic polynomial.

The **geometric multiplicity** is

$$\boxed{g_\lambda = \dim E_\lambda.}$$

We always have

$$\boxed{1 \leq g_\lambda \leq a_\lambda}$$

for an eigenvalue of a finite-dimensional matrix.

The inequality can be strict.

Example: a defective matrix

Consider

$$J = \begin{pmatrix} 2 & 1 \\ 0 & 2 \end{pmatrix}.$$

Its characteristic polynomial is

$$\det(J - \lambda I) = (2 - \lambda)^2.$$

Thus $\lambda = 2$ has algebraic multiplicity 2.

But

$$J - 2I = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

Its kernel is

$$\boxed{E_2 = \text{span}\{(1, 0)^T\}.}$$

Therefore,

$$g_2 = 1 < 2 = a_2.$$

The matrix has only one linearly independent eigenvector.

It cannot be diagonalized.

A geometric interpretation

The matrix acts as

$$J(x, y) = (2x + y, 2y).$$

The first coordinate contains an additional contribution from the second coordinate.

Thus the transformation is not merely independent scaling along two eigenvector directions.

The off-diagonal entry represents coupling that cannot be removed by choosing an eigenbasis.

This is the beginning of Jordan theory.

We do not need the full Jordan canonical form for your numerical linear algebra objectives, but this example is useful for understanding why self-adjointness is such a strong assumption.

16.4.5 Invariant subspaces

A subspace $U \subseteq V$ is **invariant under** T if

$$\boxed{T(U) \subseteq U.}$$

This means that once a vector belongs to U, applying T keeps it inside U.

Every eigenspace is invariant.

Indeed, if $v \in E_\lambda$, then

$$Tv = \lambda v \in E_\lambda.$$

But an invariant subspace need not be spanned by eigenvectors.

Our rotation example illustrates this over $\mathbb{R}$: the entire real plane is invariant, but a nontrivial rotation through an angle other than 0 or π has no real eigenvectors.

Why invariant subspaces matter

Suppose $U \subseteq V$ is invariant.

Then T restricts to an operator

$$T|_U : U \to U.$$

We can study its action on U independently of what happens elsewhere in the space.

This creates the possibility of reducing a large problem to a smaller problem.

Numerical eigenvalue algorithms exploit exactly this idea.

Instead of finding every eigenvector, they may approximate a low-dimensional subspace containing the eigenvectors of interest.

16.4.6 Adapted bases and block matrices

Suppose $\dim U = k$, and choose a basis

$$(u_1, \dots, u_k)$$

of U.

Extend it to a basis

$$(u_1, \dots, u_k, w_1, \dots, w_{n-k})$$

of V.

Because U is invariant, each Tu_j is a linear combination of the u_i.

Therefore, the matrix of T in this basis has the block form

$$\boxed{[T] = \begin{pmatrix} B & C \\ 0 & D \end{pmatrix}.}$$

The lower-left block is zero because vectors in U cannot be mapped outside U.

If the complement $W = \operatorname{span}\{w_1, \dots, w_{n-k}\}$ is also invariant, then $C = 0$, yielding

$$\boxed{[T] = \begin{pmatrix} B & 0 \\ 0 & D \end{pmatrix}.}$$

This is a block-diagonal decomposition.

The distinction matters: invariance of only one summand guarantees block triangularity, not block diagonality.

The quotient operator

As we saw in Module B, an invariant subspace also lets us define

$$\overline{T} : V/U \to V/U, \qquad \overline{T}(v+U) = Tv + U.$$

The block B represents the restriction $T|_U$.

The block D represents the induced operator on the quotient, in a compatible quotient basis.

Thus block triangularization connects invariant subspaces, quotient spaces, and spectral theory.

16.4.7 Diagonalizability as an invariant direct sum

An operator T is diagonalizable when V has a basis of eigenvectors.

Equivalently, its eigenspaces provide a direct-sum decomposition:

$$\boxed{V = \bigoplus_{\lambda \in \sigma(T)} E_\lambda,}$$

where $\sigma(T)$ is the set of distinct eigenvalues of T, over the chosen field.

If A is diagonalizable, there is an invertible matrix X such that

$$\boxed{A = X\Lambda X^{-1}.}$$

The columns of X are eigenvectors.

The inverse X^{-1} converts vectors into their eigenbasis coordinates.

Recall Module B: the rows of X^{-1} are the corresponding dual-basis functionals.

Thus diagonalization naturally combines eigenvectors and duality.

Why this matters numerically

The eigenvector matrix X need not be well-conditioned.

Even when a matrix is diagonalizable, its eigenvectors may be nearly linearly dependent.

Consequently, a spectral decomposition

$$A = X\Lambda X^{-1}$$

can be highly sensitive to perturbations.

For a Hermitian matrix, we can choose $X = Q$ unitary, giving

$$Q^{-1} = Q^*.$$

This eliminates the conditioning problem associated with a nonorthogonal eigenvector basis.

It is one reason Hermitian eigenvalue problems are generally more favorable numerically.

Checkpoint 1

Prove that eigenspaces associated with distinct eigenvalues are linearly independent as a family of subspaces.

Then explain why diagonalizability is equivalent to the direct sum of all eigenspaces spanning V.

Finally, distinguish carefully between:

$$\text{invertible}, \qquad \text{diagonalizable}, \qquad \text{normal}, \qquad \text{self-adjoint}.$$

These are different properties.

16.5 Session 2 — The spectral theorem and spectral projections

We now specialize to operators with particularly favorable geometry.

16.5.1 Self-adjoint and normal operators

Recall from Module C:

An operator A on an inner-product space is **self-adjoint** when

$$\boxed{A = A^*.}$$

A broader condition is **normality**:

$$\boxed{AA^* = A^*A.}$$

Every self-adjoint operator is normal.

Unitary operators are also normal, because

$$U^*U = UU^* = I.$$

But a normal operator need not be self-adjoint.

For example, a nontrivial planar rotation is orthogonal and therefore normal, but is generally not symmetric.

The complex spectral theorem

A complex matrix A is normal if and only if it is unitarily diagonalizable:

$$\boxed{A \text{ normal } \iff A = Q\Lambda Q^*}$$

for a unitary Q and diagonal Λ.

Its eigenvalues may be complex.

For a self-adjoint matrix, all eigenvalues are real.

For a unitary matrix, all eigenvalues have modulus one.

These spectral restrictions reflect the geometry that each operator preserves.

16.5.2 The finite-dimensional self-adjoint spectral theorem

Theorem. If $A = A^*$ on a finite-dimensional real or complex inner-product space, there exists an orthonormal eigenbasis.

In matrix form,

$$\boxed{A = Q\Lambda Q^*,}$$

where $Q^*Q = I$ and Λ is real diagonal.

Let us prove the real case carefully.

Step 1: Existence of an eigenvector

Consider the Rayleigh quotient

$$\rho_A(x) = \frac{x^T Ax}{x^T x}$$

for $x \neq 0$.

On the unit sphere,

$$\rho_A(x) = x^T Ax.$$

The unit sphere is compact in finite dimensions, and the quadratic form is continuous.

Therefore, there is a unit vector q maximizing ρ_A.

Set

$$\lambda = q^T Aq.$$

Because q maximizes the Rayleigh quotient, its derivative must vanish in every direction h.

For

$$\phi(t) = \frac{(q+th)^T A(q+th)}{(q+th)^T(q+th)},$$

differentiating at $t = 0$ gives

$$\phi'(0) = 2h^T(Aq - \lambda q).$$

Since $\phi'(0) = 0$ for every h,

$$Aq - \lambda q = 0.$$

Thus

$$\boxed{Aq = \lambda q.}$$

We have found an eigenvector.

Step 2: The orthogonal complement is invariant

Let

$$U = \text{span}\{q\}.$$

The subspace U is invariant.

Since A is self-adjoint, Module C showed that $U^\perp$ is also invariant.

Indeed, for $x \in U^\perp$,

$$\langle Ax, q\rangle = \langle x, Aq\rangle = \lambda\langle x, q\rangle = 0.$$

Therefore $Ax \in U^\perp$.

Step 3: Induction on dimension

The restriction

$$A|_{U^\perp} : U^\perp \to U^\perp$$

is still self-adjoint.

The subspace $U^\perp$ has dimension $n - 1$.

Applying the same argument inductively gives an orthonormal eigenbasis of $U^\perp$.

Adding q produces an orthonormal eigenbasis of the entire space.

This proves the spectral theorem.

For complex Hermitian matrices, an analogous variational argument works, using complex variations and the reality of the Rayleigh quotient.

16.5.3 Spectral decomposition into eigenspaces

Suppose A is Hermitian and its distinct eigenvalues are

$$\mu_1, \dots, \mu_s.$$

Let E_{μ_j} be the eigenspace corresponding to μ_j.

The spectral theorem gives

$$\boxed{V = E_{\mu_1} \oplus^{\perp} \cdots \oplus^{\perp} E_{\mu_s}.}$$

This is an **orthogonal invariant direct-sum decomposition**.

Each vector has a unique expansion

$$x = x_1 + \cdots + x_s, \qquad x_j \in E_{\mu_j}.$$

The operator acts independently on the components:

$$Ax = \mu_1 x_1 + \cdots + \mu_s x_s.$$

Thus an apparently complicated linear operator becomes scalar multiplication on each spectral subspace.

This is the geometric heart of the spectral theorem.

16.5.4 Spectral projections

Let P_j denote the orthogonal projection onto E_{μ_j}.

Then

$$\boxed{P_j^2 = P_j = P_j^*.}$$

Because distinct eigenspaces are orthogonal,

$$P_i P_j = 0 \qquad (i \neq j).$$

Furthermore,

$$\boxed{I = \sum_{j=1}^{s} P_j.}$$

Every vector is the sum of its spectral components.

The operator itself can be reconstructed as

$$\boxed{A = \sum_{j=1}^{s} \mu_j P_j.}$$

Proof

For any $x \in V$, the orthogonal eigenspace decomposition gives

$$x = \sum_j P_j x.$$

Applying A,

$$Ax = \sum_j A(P_j x).$$

Since $P_j x \in E_{\mu_j}$,

$$A(P_j x) = \mu_j P_j x.$$

Thus

$$Ax = \sum_j \mu_j P_j x.$$

Since this holds for every x, the operator identity follows.

Spectral projections commute with the operator

We also have

$$\boxed{AP_j = P_j A = \mu_j P_j.}$$

This means that extracting a spectral component and then applying the operator gives the same result as applying the operator first and then extracting that component.

Spectral projections are especially well matched to the operator's structure.

16.5.5 A running spectral example

Consider

$$\boxed{A = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 9 \end{pmatrix}.}$$

Its eigenvalues are

$$\boxed{1, \quad 3, \quad 9.}$$

An orthonormal eigenbasis is

$$q_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix},$$

$$q_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix},$$

$$q_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

Define

$$Q = (q_1 \; q_2 \; q_3).$$

Then

$$A = Q \begin{pmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 9 \end{pmatrix} Q^T.$$

The rank-one spectral projectors are

$$P_1 = q_1 q_1^T = \frac{1}{2} \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

$$P_2 = q_2 q_2^T = \frac{1}{2} \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

and

$$P_3 = q_3 q_3^T = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

They satisfy

$$P_1 + P_2 + P_3 = I$$

and

$$\boxed{A = P_1 + 3P_2 + 9P_3.}$$

Geometric meaning

The vector space is decomposed into three orthogonal directions.

The operator multiplies the first direction by 1, the second by 3, and the third by 9.

Every input vector is decomposed into those directions, independently scaled, and then recombined.

This is the geometry of a symmetric positive definite transformation.

16.5.6 Matrix functions and polynomial functional calculus

Suppose

$$p(t) = a_0 + a_1 t + \cdots + a_d t^d.$$

Define

$$p(A) = a_0 I + a_1 A + \cdots + a_d A^d.$$

If

$$A = Q \Lambda Q^*,$$

then

$$A^k = Q \Lambda^k Q^*.$$

Hence

$$\boxed{p(A) = Q p(\Lambda) Q^*.}$$

Equivalently,

$$\boxed{p(A) = \sum_{j=1}^{s} p(\mu_j) P_j.}$$

Thus applying a polynomial to an operator changes the scaling factor associated with each eigenspace from μ_j to $p(\mu_j)$.

Example

Take

$$p(t) = \frac{9-t}{8}.$$

Then

$$p(1) = 1, \qquad p(3) = \frac{3}{4}, \qquad p(9) = 0.$$

Therefore,

$$\boxed{p(A) = P_1 + \frac{3}{4}P_2.}$$

The eigenspace corresponding to eigenvalue 9 is completely eliminated.

The eigenspace corresponding to eigenvalue 3 is attenuated.

The eigenspace corresponding to eigenvalue 1 is preserved.

This is a simple spectral filter.

16.5.7 Spectral projections as polynomials

There is a useful converse.

Suppose the distinct eigenvalues of a diagonalizable operator are

$$\mu_1, \dots, \mu_s.$$

For each j, define the Lagrange interpolation polynomial

$$\ell_j(t) = \prod_{k \neq j} \frac{t - \mu_k}{\mu_j - \mu_k}.$$

It satisfies

$$\ell_j(\mu_i) = \delta_{ij}.$$

Therefore,

$$\ell_j(A) = \sum_i \ell_j(\mu_i) P_i = P_j.$$

Thus

$$\boxed{P_j = \prod_{k \neq j} \frac{A - \mu_k I}{\mu_j - \mu_k}.}$$

This is a striking connection:

The spectral projector onto an eigenspace can itself be expressed as a polynomial of the operator.

It follows that the projector is intrinsic to the operator, without depending on a particular orthonormal eigenbasis inside a repeated eigenspace.

This construction also connects spectral theory with polynomial interpolation, an important topic in numerical analysis.

However, explicitly evaluating these polynomials can be numerically unstable when eigenvalues are closely clustered. The formula is primarily a structural identity, not necessarily a recommended numerical algorithm.

16.5.8 Matrix functions beyond polynomials

For a Hermitian matrix, we can define a much larger class of functions by

$$\boxed{f(A) = Q f(\Lambda) Q^*,}$$

where

$$f(\Lambda) = \mathrm{diag}(f(\lambda_1), \dots, f(\lambda_n)).$$

Examples include:

$$e^A = Q e^{\Lambda} Q^*,$$

$$A^{1/2} = Q \Lambda^{1/2} Q^*$$

for positive semidefinite A, and

$$A^{-1} = Q\Lambda^{-1}Q^*$$

when A is invertible.

For finite-dimensional Hermitian matrices, it is enough that f be defined on the finite spectrum of A.

This is an elementary form of **functional calculus**: we apply scalar functions to operators by applying them to the spectral values.

In infinite-dimensional operator theory, functional calculus becomes a central construction.

16.5.9 Structural connection: projectors and algebra

The spectral decomposition expresses an operator through mutually orthogonal idempotents:

$$A = \sum_j \mu_j P_j.$$

The identities

$$P_i P_j = \delta_{ij} P_i, \qquad \sum_j P_j = I$$

are algebraic.

Self-adjointness adds the geometric property

$$P_j^* = P_j.$$

This distinction illustrates a recurring theme of our curriculum.

The existence of direct sums and algebraic projections belongs to structural linear algebra.

Orthogonality and self-adjoint projections belong to inner-product geometry.

The spectral theorem combines both.

Checkpoint 2

Prove directly that the spectral projectors of a Hermitian matrix are mutually orthogonal idempotents and that their sum is the identity.

Then derive

$$p(A) = \sum_j p(\mu_j) P_j.$$

Finally, explain why polynomial filtering changes spectral components without changing their eigenspaces.

16.6 Session 3 — The geometry of spectral subspaces

Individual eigenvectors are useful, but many important numerical algorithms seek an entire invariant subspace.

This is particularly natural when eigenvalues are clustered or repeated.

If several eigenvalues lie close together, their individual eigenvectors may be difficult to distinguish accurately, while the subspace spanned by them can still be approximated reliably.

We therefore need a way to measure the distance between subspaces.

16.6.1 A subspace is more fundamental than a chosen basis

Suppose $Q \in \mathbb{R}^{n \times k}$ has orthonormal columns spanning a subspace U.

Then

$$Q^T Q = I_k.$$

But Q is not unique.

For any orthogonal $R \in \mathbb{R}^{k \times k}$,

$$\widetilde{Q} = QR$$

also has orthonormal columns, and

$$\operatorname{im} \widetilde{Q} = \operatorname{im} Q.$$

Therefore, an algorithm tracking an invariant subspace should not mistake a change of basis inside that subspace for a change in the subspace itself.

This is important in subspace iteration and Rayleigh–Ritz methods.

A basis-independent representation

The orthogonal projector onto U is

$$\boxed{P_U = QQ^T.}$$

Replacing Q by QR gives

$$\widetilde{Q}\widetilde{Q}^T = QRR^TQ^T = QQ^T.$$

Thus the projector depends only on U, not on its chosen orthonormal basis.

Spectral subspaces can therefore be studied through their projectors.

16.6.2 Measuring the angle between two lines

Consider two unit vectors $u, v \in \mathbb{R}^n$.

The acute angle between their one-dimensional spans is

$$\boxed{\theta = \arccos |u^T v|, \qquad 0 \le \theta \le \frac{\pi}{2}.}$$

The absolute value matters because u and $-u$ span the same line.

We want to measure the angle between subspaces, not between arbitrarily oriented basis vectors.

The sine of this angle is

$$\boxed{\sin\theta = \|(I - uu^T)v\|_2.}$$

Indeed,

$$v = (u^T v)u + (I - uu^T)v$$

is an orthogonal decomposition.

Since $\|v\| = 1$,

$$\|(I - uu^T)v\|^2 = 1 - |u^T v|^2 = \sin^2\theta.$$

Thus the angle measures how much of one unit vector lies outside the other line.

16.6.3 Principal angles between subspaces

For two k-dimensional subspaces $U, W \subseteq \mathbb{R}^n$, choose orthonormal basis matrices Q_U, Q_W.

Compute the singular values of

$$Q_U^T Q_W.$$

Write them in descending order:

$$\sigma_1 \geq \cdots \geq \sigma_k.$$

The **principal angles** are defined by

$$\boxed{\cos\theta_i = \sigma_i(Q_U^T Q_W), \qquad 0 \leq \theta_1 \leq \cdots \leq \theta_k \leq \frac{\pi}{2}.}$$

This generalizes the angle between two lines.

The smallest principal angle describes the closest alignment between directions in the two subspaces.

The largest principal angle describes their greatest mismatch.

These angles are independent of the orthonormal bases chosen.

Why? Changing the bases multiplies $Q_U^T Q_W$ on the left and right by orthogonal matrices, which do not change its singular values.

A useful error measure

For subspaces of equal dimension,

$$\boxed{\|\sin\Theta(U,W)\|_2 = \sin\theta_{\max}.}$$

Here $\sin\Theta$ denotes the diagonal matrix of sines of the principal angles.

One equivalent expression is

$$\boxed{\sin\theta_{\max} = \|(I - P_U)Q_W\|_2.}$$

This is useful in numerical algorithms because Q_W is typically an orthonormal basis produced by a computation.

We can compare it with a desired invariant subspace U without requiring the basis vectors to align individually.

For equal-dimensional subspaces, another identity is

$$\boxed{\|P_U - P_W\|_2 = \sin\theta_{\max}.}$$

This provides a basis-independent metric for comparing subspaces.

16.6.4 An illustrative two-dimensional example

Let

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix}.$$

Its eigenvectors are e_1, e_2.

Suppose we want to approximate the eigenspace

$$U = \operatorname{span}\{e_1\}.$$

Take a unit vector

$$x(\theta) = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}, \qquad 0 \le \theta \le \frac{\pi}{2}.$$

The trial subspace is

$$W_\theta = \operatorname{span}\{x(\theta)\}.$$

The angle between U and W_θ is exactly θ.

Its projector is

$$P_\theta = x(\theta)x(\theta)^T.$$

The subspace error is

$$\boxed{\|P_\theta - P_U\|_2 = \sin\theta.}$$

This illustrates why subspace error is not quite the same thing as ordinary vector error.

The vectors $x(\theta)$ and $-x(\theta)$ represent the same subspace, even though their Euclidean distance can be large.

Explore the example

Trial angle	Subspace error	Rayleigh quotient	Residual norm
0°	0.000	1.000	0.000
42°	0.669	2.343	1.492
45°	0.707	2.500	1.500
90°	1.000	4.000	0.000

Notice that the residual vanishes at both 0° and 90°: both directions are exact eigenvectors, but only the first is the intended target. A small residual must therefore be interpreted together with the approximate eigenvalue and spectral separation.

The formulas behind this visualization will be derived in Session 4.

16.6.5 What is an approximately invariant subspace?

Let $A = A^* \in \mathbb{C}^{n\times n}$.

Suppose $Q \in \mathbb{C}^{n\times k}$ has orthonormal columns spanning a trial subspace U.

The subspace is invariant precisely when

$$A(U) \subseteq U.$$

Equivalently, every vector AQy must belong to $\operatorname{im} Q$.

The orthogonal projector onto U is

$$P = QQ^*.$$

A vector belongs to U exactly when its component outside U vanishes.

Therefore,

$$\boxed{U \text{ is invariant } \iff (I - P)AQ = 0.}$$

This motivates defining the **invariance residual**

$$\boxed{R_U = (I - QQ^*)AQ.}$$

It measures how much A sends the trial subspace outside itself.

Connection to Rayleigh–Ritz

Define the compressed matrix

$$H = Q^*AQ.$$

Then

$$\begin{aligned} AQ - QH &= AQ - QQ^*AQ \\ &= (I - QQ^*)AQ. \end{aligned}$$

Thus

$$\boxed{R_U = AQ - QH.}$$

When $R_U = 0$, the trial subspace is exactly invariant.

When $\|R_U\|$ is small, it is approximately invariant.

This is a key quantity in subspace iteration and numerical eigenvalue methods.

16.6.6 Why spectral gaps matter

Suppose a Hermitian matrix has eigenvalues

$$\lambda_1 \leq \cdots \leq \lambda_n.$$

We are interested in a group of eigenvalues separated from the rest.

For example, perhaps we want the eigenspace associated with λ_1, and

$$\lambda_2 - \lambda_1$$

is appreciably positive.

This difference is a **spectral gap**.

More generally, for two disjoint spectral sets Λ_1, Λ_2, define their separation by

$$\boxed{\operatorname{gap}(\Lambda_1, \Lambda_2) = \min_{\lambda \in \Lambda_1,\ \mu \in \Lambda_2} |\lambda - \mu|.}$$

Why is a gap useful?

Because a vector with a small residual can have components in several nearby eigenspaces.

If the unwanted eigenvalues are well separated from the approximate eigenvalue, those unwanted components can be bounded using the residual.

We will derive a precise version of this statement in the next session.

Structural connection to perturbation theory

For a Hermitian matrix, an isolated cluster of eigenvalues determines a spectral subspace.

Small perturbations of the matrix often lead to small changes in that subspace, provided the perturbation is small relative to the separation from the rest of the spectrum.

This is the setting of classical invariant-subspace perturbation results, including the Davis–Kahan sin Θ theorem.

The important conceptual ingredients are already present:

spectral projector + subspace angle + spectral gap.

We will use these ideas in numerical error analysis rather than proving the full perturbation theorem here.

Checkpoint 3

Explain why comparing two orthonormal basis matrices entry by entry is generally not a reliable way to measure the difference between their subspaces.

Prove

$$U \text{ invariant under } A \iff (I - QQ^*)AQ = 0.$$

Then explain the roles of the invariance residual and spectral separation when assessing a numerically computed invariant subspace.

16.7 Session 4 — Rayleigh quotients, variational principles, and Rayleigh–Ritz

We now connect spectral theory to optimization.

One of the most important facts about Hermitian matrices is that their eigenvalues can be characterized through variational problems.

This allows us to approximate eigenvalues using subspaces rather than characteristic polynomials.

16.7.1 The Rayleigh quotient

Let $A = A^* \in \mathbb{C}^{n\times n}$.

For $x \neq 0$, define

$$\boxed{\rho_A(x) = \frac{x^*Ax}{x^*x}.}$$

Because A is Hermitian, the numerator is real.

Suppose

$$Aq_i = \lambda_i q_i,$$

where the q_i form an orthonormal eigenbasis.

Write

$$x = \sum_{i=1}^{n} c_i q_i.$$

Then

$$x^*Ax = \sum_i \lambda_i |c_i|^2$$

and

$$x^*x = \sum_i |c_i|^2.$$

Therefore,

$$\boxed{\rho_A(x) = \frac{\sum_i \lambda_i |c_i|^2}{\sum_i |c_i|^2}.}$$

The Rayleigh quotient is a weighted average of the eigenvalues.

Hence

$$\boxed{\lambda_{\min} \leq \rho_A(x) \leq \lambda_{\max}.}$$

This provides a geometric interpretation of eigenvalues as extreme values of a quadratic form on the unit sphere.

16.7.2 The min–max principle

Order the eigenvalues as

$$\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n,$$

counting multiplicity.

The **Courant–Fischer min–max theorem** states that

$$\boxed{\lambda_k = \min_{\substack{S \subseteq \mathbb{C}^n \\ \dim S = k}} \max_{0 \neq x \in S} \rho_A(x).}$$

There is also the dual characterization

$$\boxed{\lambda_k = \max_{\substack{S \subseteq \mathbb{C}^n \\ \dim S = n-k+1}} \min_{0 \neq x \in S} \rho_A(x).}$$

These formulas characterize every eigenvalue through subspaces and the Rayleigh quotient.

Proof of the first characterization

Let

$$E_k = \operatorname{span}\{q_1, \ldots, q_k\}.$$

Every nonzero $x \in E_k$ satisfies

$$\rho_A(x) \leq \lambda_k.$$

Equality is attained at $x = q_k$.

Thus

$$\max_{0 \neq x \in E_k} \rho_A(x) = \lambda_k,$$

giving

$$\min_{\dim S=k} \max_{x\in S\setminus\{0\}} \rho_A(x) \leq \lambda_k.$$

For the reverse inequality, let S be any k-dimensional subspace.

Consider

$$F_k = \operatorname{span}\{q_k, \ldots, q_n\}.$$

This has dimension $n-k+1$.

Since

$$\dim S + \dim F_k = n+1 > n,$$

we must have

$$S \cap F_k \neq \{0\}.$$

Choose a nonzero $x \in S \cap F_k$.

Because $x \in F_k$, its spectral expansion contains only eigenvectors with eigenvalues at least λ_k.

Therefore,

$$\rho_A(x) \geq \lambda_k.$$

Thus

$$\max_{0\neq y\in S} \rho_A(y) \geq \lambda_k.$$

Since this holds for every k-dimensional S,

$$\min_{\dim S=k} \max_{y\in S\setminus\{0\}} \rho_A(y) \geq \lambda_k.$$

Combining both inequalities proves the result.

The second characterization follows by a symmetric argument.

Why this theorem matters

The min–max principle describes eigenvalues without referring to a particular eigenbasis.

It is invariant under changes of coordinates and extends naturally to variational eigenvalue problems.

It also explains why approximating an operator on a smaller subspace produces meaningful eigenvalue estimates.

16.7.3 Rayleigh–Ritz approximation

Let $U \subseteq \mathbb{C}^n$ be a k-dimensional trial subspace.

Choose an orthonormal basis matrix

$$Q \in \mathbb{C}^{n\times k}, \qquad Q^*Q = I.$$

We seek approximate eigenpairs of A inside U.

A vector $u \in U$ can be written as

$$u = Qy.$$

Define the compressed matrix

$$\boxed{H = Q^*AQ.}$$

Since A is Hermitian,

$$H^* = H.$$

Thus H is a smaller Hermitian matrix.

Solve

$$\boxed{Hy = \theta y.}$$

Then

$$u = Qy$$

is a **Ritz vector**, and θ is a **Ritz value**.

If y is normalized, so is u.

16.7.4 Deriving the Ritz equations from residual orthogonality

Define the eigenpair residual

$$\boxed{r = Au - \theta u.}$$

For an exact eigenpair, $r = 0$.

But an approximate eigenvector constrained to U generally cannot satisfy the exact equation.

Rayleigh–Ritz instead imposes

$$\boxed{r \perp U.}$$

This means

$$Q^* r = 0.$$

Substitute $u = Qy$:

$$Q^*(AQy - \theta Qy) = 0.$$

Therefore,

$$Q^* AQy = \theta Q^* Qy.$$

Since $Q^*Q = I$,

$$\boxed{Hy = \theta y.}$$

Thus the reduced eigenvalue problem is precisely the condition that the full-space residual annihilate the trial subspace.

This is closely related to Galerkin orthogonality from Module C.

A variational interpretation

The Ritz values are the stationary values of the Rayleigh quotient restricted to the trial subspace.

Indeed,

$$\rho_A(Qy) = \frac{y^* H y}{y^* y}.$$

Thus approximating eigenvalues in U is equivalent to studying the Rayleigh quotient of the smaller Hermitian matrix H.

16.7.5 Ritz value interlacing

Let the eigenvalues of A be

$$\lambda_1 \leq \cdots \leq \lambda_n$$

and the Ritz values of a k-dimensional trial subspace be

$$\theta_1 \leq \cdots \leq \theta_k.$$

Then

$$\boxed{\lambda_i \leq \theta_i \leq \lambda_{i+n-k}, \qquad i = 1, \ldots, k.}$$

This is a form of eigenvalue interlacing.

It follows from the min–max principle, since the Ritz variational problem is restricted to subspaces contained in the trial space.

In particular, for the smallest Ritz value,

$$\boxed{\theta_1 \geq \lambda_1.}$$

For the largest Ritz value,

$$\boxed{\theta_k \leq \lambda_n.}$$

The Ritz values remain within the spectral interval of the original Hermitian matrix.

This is useful both theoretically and as a numerical consistency check.

16.7.6 Worked Rayleigh–Ritz example

Consider the diagonal Hermitian matrix

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 9 \end{pmatrix}.$$

Let

$$q_1 = \frac{1}{\sqrt{2}}(e_1 + e_2), \qquad q_2 = e_3.$$

These form an orthonormal basis of the two-dimensional trial subspace

$$U = \operatorname{span}\{q_1, q_2\}.$$

Set

$$Q = (q_1 \; q_2).$$

The reduced matrix is

$$H = Q^T A Q.$$

Because q_1 combines the eigenvalues 1 and 3 equally,

$$q_1^T A q_1 = \frac{1+3}{2} = 2.$$

Also,

$$q_2^T A q_2 = 9.$$

The cross terms are zero.

Therefore,

$$\boxed{H = \begin{pmatrix} 2 & 0 \\ 0 & 9 \end{pmatrix}.}$$

The Ritz values are 2 and 9.

The first Ritz vector is q_1.

Its residual is

$$\begin{aligned} r_1 &= Aq_1 - 2q_1 \\ &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 3 \\ 0 \end{pmatrix} - \frac{2}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \\ &= \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix}. \end{aligned}$$

Thus

$$\boxed{\|r_1\|_2 = 1.}$$

The residual is orthogonal to the trial subspace.

The second Ritz vector e_3 is an exact eigenvector, with zero residual.

This example illustrates that a Ritz value need not be an exact eigenvalue of A.

The Ritz value 2 is an approximation obtained from a trial direction mixing the exact eigendirections for eigenvalues 1 and 3.

16.7.7 Residuals give eigenvalue inclusion bounds

Suppose $A = A^*$ and u is a unit vector.

For any scalar θ, define

$$r = Au - \theta u.$$

Expand u in an orthonormal eigenbasis:

$$u = \sum_i c_i q_i.$$

Then

$$r = \sum_i c_i(\lambda_i - \theta)q_i.$$

Therefore,

$$\|r\|^2 = \sum_i |c_i|^2|\lambda_i - \theta|^2.$$

Let

$$d = \min_i |\lambda_i - \theta|.$$

Since every term satisfies $|\lambda_i - \theta| \geq d$,

$$\|r\|^2 \geq d^2 \sum_i |c_i|^2 = d^2.$$

Hence

$$\boxed{\operatorname{dist}(\theta, \sigma(A)) \leq \|r\|_2.}$$

This tells us that if the residual is small, then θ is close to at least one eigenvalue.

The conclusion requires Hermitian or normal structure for this particularly clean estimate.

For strongly nonnormal matrices, small residuals can occur far from the spectrum, which motivates the concept of pseudospectra.

16.7.8 Residuals and eigenvector accuracy

A small residual alone does not identify which eigenvector is being approximated.

We also need spectral separation.

Suppose A has eigenvectors q_i, and we are interested in a spectral subspace

$$U_J = \operatorname{span}\{q_j : j \in J\}.$$

Let P_J be its orthogonal projector.

For a unit approximate vector u and scalar θ, define

$$r = Au - \theta u.$$

Assume

$$\delta = \min_{i \notin J} |\lambda_i - \theta| > 0.$$

Then

$$\boxed{\|(I - P_J)u\|_2 \leq \frac{\|r\|_2}{\delta}.}$$

Proof

Write

$$u = \sum_i c_i q_i.$$

The component outside the target subspace is

$$(I - P_J)u = \sum_{i \notin J} c_i q_i.$$

Therefore,

$$\|(I - P_J)u\|^2 = \sum_{i \notin J} |c_i|^2.$$

Meanwhile,

$$\|r\|^2 = \sum_i |\lambda_i - \theta|^2 |c_i|^2.$$

For $i \notin J$, the assumed separation gives

$$|\lambda_i - \theta| \geq \delta.$$

Hence

$$\|r\|^2 \geq \delta^2 \sum_{i \notin J} |c_i|^2 = \delta^2 \|(I - P_J)u\|^2.$$

Taking square roots proves the result.

This is an elementary but powerful residual-to-subspace-error estimate.

It expresses a central principle:

Residual accuracy becomes subspace accuracy when the desired spectral components are separated from the unwanted ones.

16.7.9 Why eigenvalue errors can be smaller than eigenvector errors

Return to the two-dimensional example

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix}, \qquad x(\theta) = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}.$$

The Rayleigh quotient is

$$\begin{aligned} \rho_A(x) &= \cos^2\theta + 4\sin^2\theta \\ &= 1 + 3\sin^2\theta. \end{aligned}$$

Thus its error relative to eigenvalue 1 is

$$\boxed{\rho_A(x) - 1 = 3\sin^2\theta.}$$

The eigenvector subspace error is

$$\sin\theta.$$

So the Rayleigh quotient error is quadratic in the subspace error.

Now calculate the residual:

$$r = Ax - \rho_A(x)x.$$

Using $\rho_A(x) = 1 + 3\sin^2\theta$, one obtains

$$r = 3\sin\theta\cos\theta\begin{pmatrix}-\sin\theta\\ \cos\theta\end{pmatrix}.$$

Therefore,

$$\boxed{\|r\|_2 = 3|\sin\theta\cos\theta|.}$$

For small θ,

$$\sin\theta \approx \theta, \qquad \cos\theta \approx 1.$$

Hence

$$\|r\| \approx 3\theta,$$

while

$$\rho_A(x) - 1 \approx 3\theta^2.$$

This explains why eigenvalue approximations can converge faster than the associated eigenvectors.

It is also important when interpreting numerical convergence plots: eigenvalue errors, residual norms, and subspace angles are related but not interchangeable quantities.

Checkpoint 4

Prove the eigenvalue inclusion bound

$$\operatorname{dist}(\theta, \sigma(A)) \leq \|Au - \theta u\|$$

for a unit vector u and Hermitian A.

Then derive the Rayleigh–Ritz equations from residual orthogonality.

Finally, explain why spectral separation is needed to convert a small residual into a bound on a particular eigenspace error.

16.8 Session 5 — Singular value decomposition: spectral geometry for rectangular matrices

Ordinary eigenvalues describe operators of the form

$$A : V \to V.$$

But many numerical problems involve rectangular matrices:

$$A : \mathbb{R}^n \to \mathbb{R}^m.$$

How can we analyze their geometry?

The answer is the **singular value decomposition**, or SVD.

It is among the most important constructions in numerical linear algebra because it connects kernels, images, orthogonality, conditioning, approximation, and least squares.

16.8.1 From A to the self-adjoint matrix A^*A

For a rectangular matrix $A \in \mathbb{C}^{m\times n}$, the product

$$A^*A \in \mathbb{C}^{n\times n}$$

is square and Hermitian:

$$(A^*A)^* = A^*A.$$

It is also positive semidefinite:

$$x^*A^*Ax = \|Ax\|_2^2 \geq 0.$$

Therefore the spectral theorem applies.

Let

$$A^*Av_i = \sigma_i^2 v_i,$$

where the v_i form an orthonormal basis and

$$\sigma_i \geq 0.$$

The numbers σ_i are the **singular values** of A.

They are the nonnegative square roots of the eigenvalues of A^*A.

The corresponding v_i are right singular vectors.

16.8.2 Constructing the left singular vectors

Suppose $\sigma_i > 0$.

Define

$$\boxed{u_i = \frac{Av_i}{\sigma_i}.}$$

Then

$$\begin{aligned} u_i^* u_j &= \frac{v_i^* A^* A v_j}{\sigma_i \sigma_j} \\ &= \frac{\sigma_j^2 v_i^* v_j}{\sigma_i \sigma_j} \\ &= \delta_{ij}. \end{aligned}$$

Thus the left singular vectors corresponding to positive singular values are orthonormal.

Moreover,

$$\boxed{Av_i = \sigma_i u_i.}$$

Applying A^*,

$$A^* u_i = \frac{A^* A v_i}{\sigma_i} = \sigma_i v_i.$$

So

$$\boxed{A^* u_i = \sigma_i v_i.}$$

These paired equations are fundamental.

The transformation A maps a right singular direction v_i to the corresponding left singular direction u_i, scaled by σ_i.

Its adjoint maps u_i back to v_i, scaled by the same factor.

16.8.3 Statement of the SVD theorem

Every matrix

$$A \in \mathbb{C}^{m \times n}$$

admits a factorization

$$\boxed{A = U \Sigma V^*,}$$

where:

- $U \in \mathbb{C}^{m \times m}$ is unitary.
- $V \in \mathbb{C}^{n \times n}$ is unitary.
- $\Sigma \in \mathbb{R}^{m \times n}$ is rectangular diagonal, with nonnegative singular values.

Conventionally,

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0,$$

where

$$r = \operatorname{rank} A.$$

All remaining singular values are zero.

Completing the construction

We have already constructed orthonormal right singular vectors from the spectral theorem for A^*A.

For every positive singular value, we constructed a corresponding left singular vector.

These left singular vectors span $\operatorname{im} A$.

Extend them to an orthonormal basis of $\mathbb{C}^m$.

The resulting unitary matrices U, V give

$$A = U\Sigma V^*.$$

This proves the existence of the SVD.

16.8.4 Geometric meaning: orthogonal directions and stretching factors

Consider the action of A on a unit sphere in the input space.

The SVD says we can understand this action in three steps:

$$\boxed{x \xrightarrow{V^*} \text{right singular coordinates} \xrightarrow{\Sigma} \text{scaled coordinates} \xrightarrow{U} Ax.}$$

The unitary transformations preserve lengths and angles.

The diagonal matrix Σ performs independent scaling along orthogonal singular directions.

Thus a unit sphere is mapped into an ellipsoid, possibly of lower dimension.

The singular values describe the principal semiaxis lengths of that ellipsoid.

Input unit circle

Output ellipse

Schematic of an SVD with $\sigma_1 > \sigma_2 > 0$. The orthogonal singular directions describe how the transformation stretches the input sphere.

This geometric interpretation is useful for understanding conditioning.

If one singular value is extremely small, the transformation nearly collapses one direction.

Recovering the input component in that direction from a noisy output can then be highly sensitive.

16.8.5 The four fundamental subspaces from the SVD

Let $A \in \mathbb{C}^{m\times n}$ have rank r.

Partition

$$V = (V_r\ V_0), \qquad U = (U_r\ U_0),$$

where V_r, U_r contain the singular vectors associated with positive singular values.

Then

$$\boxed{\begin{aligned} \operatorname{im} A^* &= \operatorname{im} V_r, \\ \ker A &= \operatorname{im} V_0, \\ \operatorname{im} A &= \operatorname{im} U_r, \\ \ker A^* &= \operatorname{im} U_0. \end{aligned}}$$

Thus the SVD provides **orthonormal bases for all four fundamental subspaces**.

The orthogonal decompositions are

$$\mathbb{C}^n = \operatorname{im} A^* \oplus^{\perp} \ker A$$

and

$$\mathbb{C}^m = \operatorname{im} A \oplus^{\perp} \ker A^*.$$

These are precisely the fundamental subspace decompositions derived abstractly in Modules B and C.

The SVD makes them computationally explicit.

16.8.6 The SVD and the pseudoinverse

If

$$A = U\Sigma V^*,$$

define

$$\boxed{A^+ = V\Sigma^+ U^*,}$$

where each positive singular value σ_i is replaced by $1/\sigma_i$, and zero singular values remain zero.

The minimum-norm least-squares solution is

$$\boxed{x_{\min} = A^+ b.}$$

Using the singular vectors,

$$\boxed{A^+ b = \sum_{i=1}^{r} \frac{u_i^* b}{\sigma_i} v_i.}$$

The formula is especially revealing.

The coefficient $u_i^* b$ measures the output data along the left singular direction u_i.

Dividing by σ_i reconstructs the corresponding input coefficient.

If σ_i is very small, even a small measurement error in $u_i^* b$ can be amplified.

This is the mechanism behind ill-conditioning of inverse problems.

16.8.7 Singular values and the condition number

For a matrix A with full column rank,

$$\|A\|_2 = \sigma_1.$$

Moreover,

$$\min_{\|x\|=1} \|Ax\| = \sigma_n > 0.$$

Thus the 2-norm condition number is

$$\boxed{\kappa_2(A) = \frac{\sigma_1}{\sigma_n}.}$$

For an invertible square matrix, this agrees with

$$\kappa_2(A) = \|A\|_2 \|A^{-1}\|_2.$$

A large condition number means that some directions are stretched much more strongly than others.

When solving inverse problems, reconstructing components along weakly stretched directions may strongly amplify perturbations.

For rank-deficient matrices, the ordinary inverse condition number is infinite. When working on the subspace on which A is injective, the ratio of the largest to the smallest positive singular value describes the conditioning of that restricted map.

16.8.8 Optimal low-rank approximation

The SVD also solves an important approximation problem.

Suppose

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^*.$$

For $k < r$, define the truncated SVD

$$\boxed{A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^*.}$$

This matrix has rank at most k.

The **Eckart–Young–Mirsky theorem** states that

$$\boxed{\min_{\operatorname{rank} B \le k} \|A - B\|_2 = \|A - A_k\|_2 = \sigma_{k+1}.}$$

And in the Frobenius norm,

$$\boxed{\min_{\operatorname{rank} B \le k} \|A - B\|_F = \left(\sum_{i=k+1}^{r} \sigma_i^2 \right)^{1/2}.}$$

Thus the truncated SVD is an optimal rank-k approximation in both norms.

Proof in the spectral norm

The upper bound is immediate.

Because

$$A - A_k = \sum_{i=k+1}^{r} \sigma_i u_i v_i^*,$$

its spectral norm is σ_{k+1}.

For the lower bound, let B have rank at most k.

Consider the $(k+1)$-dimensional subspace

$$S = \operatorname{span}\{v_1, \dots, v_{k+1}\}.$$

The restriction $B|_S$ cannot be injective because its image has dimension at most k.

Therefore there is a unit vector $x \in S$ with

$$Bx = 0.$$

Then

$$\|(A - B)x\| = \|Ax\|.$$

Because x belongs to the span of the first $k + 1$ right singular vectors,

$$\|Ax\| \geq \sigma_{k+1}.$$

Hence

$$\|A - B\|_2 \geq \sigma_{k+1}.$$

Since A_k attains this bound, it is optimal.

The Frobenius-norm version follows from the corresponding variational characterization of the sum of squared singular values.

Applied significance

Optimal low-rank approximation appears in:

- Principal component analysis and dimensionality reduction.
- Data compression.
- Reduced-order modeling.
- Approximation of large matrices and operators.
- Scientific computing with low-rank structure.

The deeper principle is that an operator may act strongly on only a few important directions.

A low-rank approximation retains those dominant directions and discards less significant ones.

That is conceptually related to invariant-subspace approximation, although singular subspaces and eigenspaces are not always the same.

16.8.9 A rectangular example

Consider

$$A = \begin{pmatrix} 3 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

Its singular values are

$$\sigma_1 = 3, \qquad \sigma_2 = 1.$$

The best rank-one approximation is

$$\boxed{A_1 = \begin{pmatrix} 3 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}.}$$

The error is

$$\boxed{\|A - A_1\|_2 = 1.}$$

No rank-one matrix can approximate A more accurately in spectral norm.

Checkpoint 5

Construct an SVD starting from the spectral decomposition of A^*A.

Use it to prove that

$$\ker(A^*A) = \ker A.$$

Then explain why the smallest nonzero singular values are especially important for the conditioning of inverse problems.

Finally, reconstruct the proof of the spectral-norm optimality of the truncated SVD.

16.9 Session 6 — From spectral theory to eigenvalue algorithms

We now move toward numerical methods.

The goal of this session is not yet a complete analysis of numerical stability. That is the purpose of Module E.

Instead, we will understand how eigenvalue algorithms exploit the spectral decompositions developed in the preceding sessions.

16.9.1 Why not simply solve the characteristic polynomial?

For a small matrix, eigenvalues can be found by solving

$$\det(A - \lambda I) = 0.$$

But for a large matrix, this is usually not an effective numerical strategy.

Computing characteristic polynomial coefficients can be poorly conditioned, and solving the resulting polynomial is not the standard stable approach to a large eigenvalue problem.

Instead, numerical algorithms exploit matrix-vector products, invariant subspaces, factorizations, and orthogonal transformations.

The spectral theorem helps us explain why these methods work.

16.9.2 Power iteration

Let $A \in \mathbb{R}^{n\times n}$ be symmetric, with orthonormal eigenvectors q_i.

Order its eigenvalues by decreasing magnitude:

$$|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|.$$

Suppose the initial vector is

$$x_0 = \sum_{i=1}^{n} c_i q_i, \qquad c_1 \neq 0.$$

Power iteration repeatedly applies A and normalizes:

$$\boxed{x_{k+1} = \frac{Ax_k}{\|Ax_k\|_2}.}$$

Ignoring normalization temporarily,

$$A^k x_0 = \sum_{i=1}^{n} c_i \lambda_i^k q_i.$$

Factor out $c_1 \lambda_1^k$:

$$A^k x_0 = c_1 \lambda_1^k \left(q_1 + \sum_{i=2}^{n} \frac{c_i}{c_1} \left(\frac{\lambda_i}{\lambda_1} \right)^k q_i \right).$$

Because

$$\left| \frac{\lambda_i}{\lambda_1} \right| < 1$$

for $i \geq 2$, all unwanted components decay relative to the dominant eigendirection.

Therefore,

$$\boxed{\text{span}\{x_k\} \longrightarrow \text{span}\{q_1\}.}$$

We formulate convergence in terms of spans because the normalized vector may alternate sign if $\lambda_1 < 0$.

Convergence factor

The asymptotic rate is governed by

$$\boxed{\left| \frac{\lambda_2}{\lambda_1} \right|.}$$

If this ratio is close to one, convergence can be slow.

If it is much smaller than one, convergence is rapid.

Example

Let

$$A = \operatorname{diag}(9, 3, 1), \qquad x_0 = (1, 1, 1)^T.$$

Then

$$A^k x_0 = (9^k, 3^k, 1)^T.$$

Dividing by 9^k,

$$\frac{A^k x_0}{9^k} = \left(1, \left(\frac{1}{3}\right)^k, \left(\frac{1}{9}\right)^k\right)^T.$$

The unwanted components decay geometrically.

Failure modes

Power iteration can fail to isolate a unique eigenvector if:

- The initial vector has no component in the dominant eigendirection.
- Multiple eigenvalues have equal largest magnitude.
- The dominant eigenvalue is not sufficiently separated, causing very slow convergence.

For a general nonnormal matrix, convergence behavior can be more complicated because the eigenvectors need not be orthogonal and the matrix may not even be diagonalizable.

16.9.3 Shift-and-invert iteration

What if we want an eigenvalue near a specified target μ, rather than the eigenvalue of largest magnitude?

Consider

$$B_\mu = (A - \mu I)^{-1},$$

assuming μ is not an eigenvalue.

If

$$Aq_i = \lambda_i q_i,$$

then

$$(A - \mu I)q_i = (\lambda_i - \mu)q_i.$$

Thus

$$\boxed{B_\mu q_i = \frac{1}{\lambda_i - \mu} q_i.}$$

Eigenvalues of A close to μ become eigenvalues of B_μ with large magnitude.

Applying power iteration to B_μ therefore tends to select the eigendirection whose eigenvalue is nearest to the shift.

In practice, we do not form B_μ.

Instead, each iteration solves a linear system

$$\boxed{(A - \mu I)y_k = x_k.}$$

This connects eigenvalue approximation to linear system solvers and matrix factorizations.

16.9.4 Rayleigh quotient iteration

A more adaptive method updates the shift using the current Rayleigh quotient:

$$\mu_k = \rho_A(x_k).$$

The next step solves

$$(A - \mu_k I)y_k = x_k$$

and normalizes y_k.

This is called **Rayleigh quotient iteration**.

For a real symmetric or complex Hermitian matrix, under suitable local assumptions near a simple eigenpair, it has cubic local convergence.

That is significantly faster than ordinary power iteration.

However, each step requires solving a shifted linear system.

This illustrates a recurring numerical tradeoff:

Faster convergence per iteration may require more expensive operations.

16.9.5 Subspace iteration

Power iteration seeks one eigendirection.

Subspace iteration seeks an invariant subspace of dimension k.

Suppose $X_0 \in \mathbb{C}^{n\times k}$ has linearly independent columns.

A basic iteration is

$$Y_{j+1} = AX_j,$$

followed by orthonormalization:

$$\boxed{Y_{j+1} = X_{j+1}R_{j+1}, \qquad X_{j+1}^* X_{j+1} = I.}$$

The columns of X_j span the current trial subspace.

The QR factorization prevents the basis vectors from collapsing toward the same dominant eigenvector.

It does not change the span generated by Y_{j+1}, provided Y_{j+1} has full column rank.

Why subspace iteration converges

Assume A is Hermitian and that

$$|\lambda_1| \geq \cdots \geq |\lambda_k| > |\lambda_{k+1}| \geq \cdots \geq |\lambda_n|.$$

Let

$$U_* = \text{span}\{q_1, \ldots, q_k\}$$

be the desired invariant subspace.

Suppose the starting trial subspace has sufficient overlap with U_*: specifically, its projection onto U_* has full rank k.

Repeated application of A scales desired spectral components by the first k eigenvalues and unwanted components by the remaining eigenvalues.

Under the gap assumption, the unwanted components decay relative to the desired components.

The characteristic convergence factor is

$$\boxed{\left|\frac{\lambda_{k+1}}{\lambda_k}\right|}.$$

More precisely, this ratio controls a bound on the tangent of the largest principal angle to the desired subspace.

The subspace may converge even though its basis vectors rotate or change sign from one iteration to the next.

This is another reason principal angles and projectors are preferable to entrywise basis comparisons.

16.9.6 Rayleigh–Ritz extraction during subspace iteration

After constructing an orthonormal basis X_k, form

$$H_k = X_k^* A X_k.$$

Solve the reduced eigenvalue problem

$$H_k y = \theta y.$$

The approximate eigenvector is

$$u = X_k y.$$

The residual is

$$r = Au - \theta u.$$

Thus one iteration cycle naturally consists of two complementary operations:

Subspace improvement: Apply A or a spectral filter to move the trial subspace toward the desired invariant subspace.

Ritz extraction: Solve a projected eigenvalue problem to identify approximate eigenpairs within the current trial subspace.

This division of labor is fundamental in modern iterative eigenvalue methods.

16.9.7 Orthogonal iteration and the QR algorithm

There is also a deep relationship between subspace iteration and the QR algorithm.

In orthogonal iteration, one updates a full orthonormal basis Q_k by

$$AQ_k = Q_{k+1}R_{k+1}.$$

If Q_k is square and unitary, the iteration is closely related to repeated QR factorizations and unitary similarity transformations.

The QR algorithm starts with $A_0 = A$, computes

$$A_k = Q_k R_k,$$

and then forms

$$\boxed{A_{k+1} = R_k Q_k.}$$

Because $R_k = Q_k^* A_k$,

$$A_{k+1} = Q_k^* A_k Q_k.$$

Thus each step preserves the eigenvalues.

For Hermitian matrices, appropriately shifted QR iterations converge toward diagonal form under standard assumptions.

For general matrices, the corresponding target is an upper triangular Schur form over $\mathbb{C}$.

We will examine the numerical details in Module E.

16.9.8 The role of orthogonality in numerical stability

Orthonormalization is not merely aesthetic.

Without it, repeated multiplication may make different trial vectors nearly parallel, causing severe loss of linear independence in floating-point arithmetic.

QR factorization restores a numerically useful basis for the evolving subspace.

This is closely tied to the geometry of the Stiefel and Grassmann manifolds:

- The **Stiefel manifold** consists of matrices with orthonormal columns.
- The **Grassmannian** consists of subspaces of a fixed dimension.

Subspace iteration is most fundamentally an algorithm evolving a point on a Grassmannian, while orthonormal basis matrices provide convenient computational representations.

This is a useful structural perspective for understanding eigenvalue algorithms and their software implementations.

Checkpoint 6

Derive the convergence mechanism of power iteration from the spectral decomposition.

Explain why the largest-magnitude eigenvalue is selected, rather than necessarily the largest algebraic eigenvalue.

Then describe why subspace iteration needs orthonormalization and why its convergence should be measured through subspace angles rather than differences between basis matrices.

16.10 Session 7 — Polynomial filtering and Chebyshev-filtered subspace iteration

This session brings together the main spectral ideas of Module D.

We will first understand polynomial filtering as a consequence of the spectral theorem, then show why Chebyshev polynomials are particularly effective for separating desired and unwanted spectral components.

Finally, we will derive a subspace convergence estimate explaining the mechanism behind filtered subspace iteration.

16.10.1 Polynomial filtering revisited

Let $A = A^*$ have spectral decomposition

$$A = Q\Lambda Q^*.$$

For a polynomial p,

$$p(A) = Qp(\Lambda)Q^*.$$

If

$$x = \sum_i c_i q_i,$$

then

$$\boxed{p(A)x = \sum_i c_i p(\lambda_i) q_i.}$$

Thus polynomial filtering acts independently on every eigendirection.

Suppose we want to retain eigenvectors corresponding to a selected spectral region Λ_{wanted}, while suppressing components associated with $\Lambda_{\text{unwanted}}$.

We want a polynomial with the property that

$$|p(\lambda)|$$

is relatively large on the desired eigenvalues and relatively small on the unwanted eigenvalues.

The important quantity is not necessarily the absolute size of p, but the **ratio** between its values on the wanted and unwanted spectral regions.

16.10.2 Why use a polynomial at all?

If we could construct the exact spectral projector onto the desired invariant subspace, we could apply that projector directly.

For an isolated set of eigenvalues, the desired projector is

$$P_{\text{wanted}} = \sum_{\lambda_i \in \Lambda_{\text{wanted}}} q_i q_i^*.$$

But constructing P_{wanted} directly usually requires knowing the very eigenvectors we are trying to compute.

A polynomial filter provides an approximation using only applications of A.

Because

$$p(A) = a_0 I + a_1 A + \cdots + a_d A^d,$$

we can apply $p(A)$ to vectors through matrix-vector products.

For sparse matrices, this can be much cheaper than constructing a full spectral decomposition.

The basic idea is to approximate the action of a spectral projector without explicitly forming it.

16.10.3 Chebyshev polynomials

The Chebyshev polynomials of the first kind satisfy

$$\boxed{T_k(\cos\theta) = \cos(k\theta).}$$

The first few are

$$T_0(t) = 1,$$

$$T_1(t) = t,$$

$$T_2(t) = 2t^2 - 1,$$

$$T_3(t) = 4t^3 - 3t.$$

They satisfy the three-term recurrence

$$\boxed{T_{k+1}(t) = 2tT_k(t) - T_{k-1}(t).}$$

Behavior on $[-1, 1]$

If $t \in [-1, 1]$, we can write $t = \cos\theta$.

Then

$$T_k(t) = \cos(k\theta).$$

Therefore,

$$\boxed{|T_k(t)| \leq 1 \qquad \text{for } t \in [-1, 1].}$$

This is an exceptionally useful property.

No matter how large the degree becomes, the polynomial remains bounded by one on this interval.

Behavior outside $[-1, 1]$

For $t > 1$, write

$$t = \cosh\alpha$$

with $\alpha > 0$.

Then

$$\boxed{T_k(t) = \cosh(k\alpha).}$$

Equivalently,

$$T_k(t) = \frac{1}{2}\left[\left(t + \sqrt{t^2 - 1}\right)^k + \left(t - \sqrt{t^2 - 1}\right)^k\right].$$

The magnitude grows rapidly with k.

For $t < -1$,

$$T_k(t) = (-1)^k T_k(-t).$$

Thus

\boxed{
|T_k(t)|>1

outside $[-1, 1]$, with growth becoming stronger as the degree increases.

This combination of bounded behavior inside an interval and rapid growth outside it is exactly what we need for spectral filtering.

16.10.4 The minimax property

Chebyshev polynomials are not merely convenient because of their recurrence.

They also have an extremal approximation property.

Among monic polynomials of degree $d \geq 1$, the polynomial

$$2^{1-d} T_d(t)$$

minimizes the maximum absolute value on $[-1, 1]$.

In other words,

$$\boxed{\min_{\substack{p \text{ monic} \\ \deg p = d}} \max_{t \in [-1,1]} |p(t)| = 2^{1-d}.}$$

This is a version of the Chebyshev minimax theorem.

The reason is the alternating behavior of T_d, which reaches extrema of equal magnitude at $d + 1$ points.

A polynomial of lower degree cannot cancel that alternating pattern everywhere.

This extremal behavior explains why Chebyshev polynomials repeatedly appear in approximation theory and numerical algorithms.

We will use the boundedness and growth properties directly, without requiring the full minimax theorem in every calculation.

16.10.5 Mapping an unwanted spectral interval to $[-1, 1]$

Suppose the unwanted eigenvalues of a Hermitian matrix lie in

$$[a, b], \qquad a < b.$$

Define the affine transformation

$$\boxed{s(t) = \frac{2t - (a + b)}{b - a}.}$$

It satisfies

$$s(a) = -1, \qquad s(b) = 1.$$

Therefore,

$$s([a, b]) = [-1, 1].$$

Now define

$$\boxed{p_d(t) = T_d(s(t)).}$$

For every unwanted eigenvalue $\lambda \in [a, b]$,

$$|p_d(\lambda)| \leq 1.$$

If a desired eigenvalue lies outside $[a, b]$, then $|s(\lambda)| > 1$, so $|p_d(\lambda)|$ can be much larger than one.

Thus the filter creates relative amplification of desired spectral components.

Normalization

Suppose $\lambda_\star$ is a desired reference eigenvalue outside $[a, b]$.

We can define

$$\boxed{\widehat{p}_d(t) = \frac{T_d(s(t))}{T_d(s(\lambda_\star))}.}$$

Then

$$\widehat{p}_d(\lambda_\star) = 1.$$

On the unwanted interval,

$$\boxed{|\widehat{p}_d(t)| \leq \frac{1}{|T_d(s(\lambda_\star))|}.}$$

As the degree increases, the denominator grows rapidly, giving stronger suppression of unwanted components.

For multiple desired eigenvalues, we must ensure that the polynomial is not too small at any desired eigenvalue. A filter designed around only one reference value may otherwise attenuate some desired components.

16.10.6 Explicit example: filtering the spectrum $\{1, 3, 9\}$

Return to our running matrix

$$A = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 9 \end{pmatrix},$$

with eigenvalues $1, 3, 9$.

Suppose we want the eigenspace associated with $\lambda = 1$.

We regard the interval

$$[3, 9]$$

as unwanted.

The affine transformation mapping this interval onto $[-1, 1]$ is

$$\boxed{s(t) = \frac{t-6}{3}.}$$

Then

$$s(1) = -\frac{5}{3}, \qquad s(3) = -1, \qquad s(9) = 1.$$

Consider the degree-three Chebyshev polynomial

$$T_3(t) = 4t^3 - 3t.$$

At the desired eigenvalue,

$$\begin{aligned} T_3\left(-\frac{5}{3}\right) &= 4\left(-\frac{5}{3}\right)^3 - 3\left(-\frac{5}{3}\right) \\ &= -\frac{500}{27} + 5 \\ &= -\frac{365}{27}. \end{aligned}$$

At the unwanted eigenvalues,

$$T_3(-1) = -1, \qquad T_3(1) = 1.$$

Normalize:

$$\boxed{\widehat{p}_3(t) = \frac{T_3((t-6)/3)}{T_3(-5/3)}.}$$

Then

$$\widehat{p}_3(1) = 1,$$

$$\widehat{p}_3(3) = \frac{27}{365},$$

$$\widehat{p}_3(9) = -\frac{27}{365}.$$

Thus

$$\boxed{\max_{\lambda \in \{3,9\}} |\widehat{p}_3(\lambda)| = \frac{27}{365} \approx 0.074.}$$

The degree-three filter suppresses each unwanted spectral component to approximately 7.4% of its original magnitude relative to the desired component.

Compare filter degrees

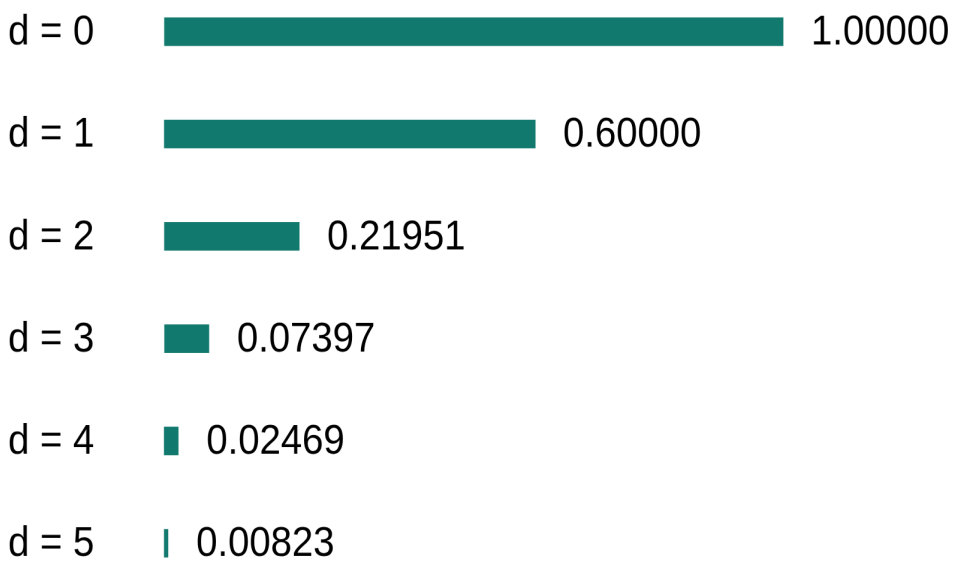


Figure 16.1: Maximum unwanted response on [3, 9] when the filter is normalized to one at the target eigenvalue 1.

Figure 16.1 shows the rapid reduction of the unwanted-to-desired response ratio as the polynomial degree increases.

The values arise from

$$\frac{1}{|T_d(-5/3)|}.$$

Acting on a vector

Let

$$x = q_1 + q_2 + q_3.$$

Then

$$\widehat{p}_3(A)x = q_1 + \frac{27}{365}q_2 - \frac{27}{365}q_3.$$

The desired component is preserved.

The unwanted components are strongly attenuated.

After normalization, the resulting vector is much closer to the eigendirection q_1.

This is the basic mechanism of Chebyshev-filtered iteration.

16.10.7 Applying the Chebyshev recurrence to a matrix

Define the rescaled matrix

$$\boxed{B = s(A) = \frac{2A - (a+b)I}{b-a}.}$$

We want to compute

$$T_d(B)X$$

for a vector or block X, without forming $T_d(B)$ explicitly.

Set

$$Y_0 = X,$$

$$Y_1 = BX.$$

Then use the recurrence

$$\boxed{Y_{j+1} = 2BY_j - Y_{j-1}.}$$

After d steps,

$$Y_d = T_d(B)X.$$

If a normalized filter is required, divide by the appropriate normalization factor.

This requires repeated applications of A to vectors or blocks, along with vector additions and scalar multiplications.

For sparse matrices, each recurrence step can often be implemented using a sparse matrix-vector or matrix-block product.

The three-term recurrence avoids explicitly forming powers of A, which would generally be computationally expensive and could destroy sparsity.

16.10.8 A rigorous subspace convergence estimate

We can now explain the convergence mechanism for filtered subspace iteration.

Suppose A is Hermitian and has the spectral decomposition

$$A = \begin{pmatrix} Q_1 & Q_2 \end{pmatrix} \begin{pmatrix} \Lambda_1 & 0 \\ 0 & \Lambda_2 \end{pmatrix} \begin{pmatrix} Q_1 & Q_2 \end{pmatrix}^*.$$

Here:

- The columns of $Q_1 \in \mathbb{C}^{n\times k}$ span the desired invariant subspace.
- The columns of Q_2 span its orthogonal complement.
- Λ_1 contains the desired eigenvalues.
- Λ_2 contains the unwanted eigenvalues.

Let the starting block $X \in \mathbb{C}^{n\times k}$ have full column rank.

Write its spectral coordinates as

$$\boxed{X = Q_1 C + Q_2 D,}$$

where $C \in \mathbb{C}^{k\times k}$ and $D \in \mathbb{C}^{(n-k)\times k}$.

Assume C is invertible.

This is the algebraic condition that the trial subspace has full-dimensional overlap with the desired subspace.

Representing the trial subspace as a graph

Because C is invertible,

$$X = (Q_1 + Q_2 D C^{-1}) C.$$

Multiplication on the right by an invertible matrix does not change the column space.

Therefore,

$$\operatorname{im} X = \operatorname{im}(Q_1 + Q_2 M),$$

where

$$\boxed{M = D C^{-1}.}$$

The matrix M describes how much the trial subspace tilts away from the desired invariant subspace.

In fact,

$$\boxed{\|M\|_2 = \tan \theta_{\max},}$$

where $\theta_{\max}$ is the largest principal angle between the two k-dimensional subspaces.

Apply a polynomial filter

Now compute

$$Y = p(A) X.$$

Using the spectral decomposition,

$$Y = Q_1 p(\Lambda_1) C + Q_2 p(\Lambda_2) D.$$

Assume that $p(\Lambda_1)$ is invertible, so that the filter does not annihilate a desired eigendirection.

The new graph matrix is

$$\boxed{M_{\text{new}} = p(\Lambda_2) M \, p(\Lambda_1)^{-1}.}$$

Therefore,

$$\|M_{\text{new}}\|_2 \leq \|p(\Lambda_2)\|_2 \|M\|_2 \|p(\Lambda_1)^{-1}\|_2.$$

Since these are diagonal spectral matrices,

$$\|p(\Lambda_2)\|_2 = \max_{\lambda \in \Lambda_{\text{unwanted}}} |p(\lambda)|,$$

while

$$\|p(\Lambda_1)^{-1}\|_2 = \frac{1}{\min_{\lambda \in \Lambda_{\text{wanted}}} |p(\lambda)|}.$$

Hence

$$\boxed{\tan \theta_{\max}^{\text{new}} \leq \gamma_p \, \tan \theta_{\max}^{\text{old}},}$$

where

$$\boxed{\gamma_p = \frac{\max_{\lambda \in \Lambda_{\text{unwanted}}} |p(\lambda)|}{\min_{\lambda \in \Lambda_{\text{wanted}}} |p(\lambda)|}.}$$

This is the central exact-arithmetic estimate behind polynomial-filtered subspace iteration.

If

$$\gamma_p < 1,$$

the trial subspace moves closer to the desired invariant subspace, as measured by its largest principal angle.

What makes Chebyshev filtering effective?

The purpose of Chebyshev filtering is to construct a polynomial with a small γ_p, while keeping its evaluation computationally practical.

The mathematical problem is therefore a polynomial approximation and spectral separation problem.

The computational problem is to evaluate the filter accurately and efficiently.

This is precisely where spectral theory, numerical analysis, and hardware-aware implementation intersect.

16.10.9 Finite-precision and hardware-aware considerations

The estimate above assumes exact arithmetic.

Real computations introduce additional effects.

A computed matrix-vector product can be modeled schematically as

$$\mathrm{fl}(Ax) = Ax + \delta,$$

where δ represents the error of the operation.

Similarly, a computed Chebyshev recurrence does not satisfy the exact polynomial recurrence.

If

$$\widehat{Y}_{j+1} = 2B\widehat{Y}_j - \widehat{Y}_{j-1} + E_j,$$

then the matrices E_j represent perturbations due to arithmetic, communication, compression, or approximate matrix products.

A convergence analysis must account for how these perturbations propagate through the recurrence.

The simple exact-arithmetic inequality

$$\tan\theta_{\mathrm{new}} \le \gamma_p \tan\theta_{\mathrm{old}}$$

then requires additional error terms.

Several numerical issues arise:

Spectral bounds. The unwanted interval $[a, b]$ must adequately contain the unwanted spectrum. An eigenvalue lying unexpectedly outside the interval may be amplified instead of suppressed.

Polynomial degree. Increasing the degree improves spectral separation in exact arithmetic, but also increases the number of matrix operations and opportunities for rounding error.

Scaling. Chebyshev polynomials can grow rapidly outside $[-1, 1]$. Careful scaling may be needed to avoid overflow or loss of numerical range.

Orthogonalization. A filtered block still needs a numerically reliable basis. QR factorization or other orthogonalization procedures control loss of linear independence.

Low precision. Lower-precision arithmetic can reduce cost on some hardware, but the resulting perturbations may limit attainable subspace accuracy.

Residuals. A small recurrence change does not automatically imply a small eigenpair residual. Residual evaluation provides a separate measure of the quality of the computed Ritz pairs.

For your HASC seminar, these are the mathematical questions that connect the exact spectral theory to implementations using inexact products, reduced precision, and hardware-dependent performance characteristics.

The detailed error recurrences, stability conditions, and performance tradeoffs belong to Module E.

Checkpoint 7

Derive the mapping

$$s(t) = \frac{2t - (a + b)}{b - a}$$

and explain why it is appropriate for Chebyshev filtering.

Then reconstruct the filtered-subspace inequality

$$\tan \theta_{\text{new}} \leq \gamma_p \tan \theta_{\text{old}}.$$

Finally, explain why the convergence factor must involve both the maximum unwanted filter response and the minimum desired filter response.

16.11 Session 8 — Spectral theory beyond matrices

We end the module by connecting finite-dimensional spectral theory to functional analysis, PDEs, and structural algebra.

These connections are important because they reveal which ideas are truly general and which depend on special assumptions such as finite dimensionality or self-adjointness.

16.11.1 Spectrum in infinite-dimensional spaces

For a finite-dimensional complex matrix, λ belongs to the spectrum precisely when

$$A - \lambda I$$

is not invertible.

In infinite-dimensional spaces, the same broad definition remains useful, but invertibility must be understood in the appropriate operator-theoretic sense.

For a bounded operator $T : X \to X$ on a complex Banach space, the spectrum is

$$\boxed{\sigma(T) = \{\lambda \in \mathbb{C} : T - \lambda I \text{ has no bounded inverse defined on all of } X\}.}$$

The spectrum may contain values that are not eigenvalues.

This is a major difference from finite-dimensional linear algebra.

16.11.2 A self-adjoint operator with no eigenvalues

Consider the Hilbert space

$$H = L^2(0, 1).$$

Define the multiplication operator

$$\boxed{(Mf)(x) = xf(x).}$$

This operator is bounded and self-adjoint.

Indeed,

$$\langle Mf, g\rangle = \int_0^1 xf(x)\overline{g(x)}\,dx = \langle f, Mg\rangle.$$

Its spectrum is

$$\boxed{\sigma(M) = [0, 1].}$$

But it has no eigenvalues.

To see why, suppose

$$Mf = \lambda f.$$

Then

$$(x - \lambda)f(x) = 0$$

for almost every $x \in (0, 1)$.

Thus f can be nonzero only on the set $\{x = \lambda\}$, which has measure zero.

Therefore $f = 0$ as an element of $L^2(0, 1)$.

So no nonzero eigenfunction exists.

This example shows that the finite-dimensional picture of spectral theory cannot always be extended by simply replacing matrices with infinite-dimensional operators.

16.11.3 Compact self-adjoint operators

An operator $T : H \to H$ on a Hilbert space is **compact** if it maps bounded sets into relatively compact sets—that is, sets whose closures are compact.

Compact self-adjoint operators have spectral properties resembling those of finite-dimensional Hermitian matrices.

For such an operator:

- Every nonzero spectral value is an eigenvalue.
- Nonzero eigenvalues have finite multiplicity.
- The nonzero spectrum can accumulate only at zero.
- The Hilbert space admits an orthonormal basis consisting of eigenvectors, including a basis for the kernel when necessary.

A typical example is

$$T : \ell^2 \to \ell^2, \qquad T(x_1, x_2, ...) = \left(x_1, \frac{x_2}{2}, \frac{x_3}{3}, ...\right).$$

Its eigenvectors are the standard coordinate sequences, with eigenvalues

$$1, \frac{1}{2}, \frac{1}{3}, \dots .$$

The eigenvalues approach zero.

Zero belongs to the spectrum but is not an eigenvalue, because the operator is injective.

This example is particularly useful for connecting spectral theory with numerical conditioning and compact inverse problems.

16.11.4 Differential operators and PDE eigenvalue problems

Consider the eigenvalue problem

$$\boxed{-u''(x) = \lambda u(x), \qquad 0 < x < 1,}$$

with boundary conditions

$$u(0) = u(1) = 0.$$

We seek nonzero solutions u.

For positive λ, write

$$\lambda = \omega^2.$$

The general solution is

$$u(x) = A \sin(\omega x) + B \cos(\omega x).$$

The condition $u(0) = 0$ gives $B = 0$.

The condition $u(1) = 0$ then requires

$$\sin \omega = 0.$$

For a nonzero solution,

$$\omega = n\pi, \qquad n = 1, 2, \ldots .$$

Therefore,

$$\boxed{\lambda_n = n^2\pi^2.}$$

The corresponding normalized eigenfunctions are

$$\boxed{q_n(x) = \sqrt{2}\sin(n\pi x).}$$

They form an orthonormal basis of $L^2(0,1)$.

Thus the differential operator

$$L = -\frac{d^2}{dx^2}$$

with Dirichlet boundary conditions has a spectral decomposition involving infinitely many eigenfunctions.

A careful operator-theoretic formulation regards L as an unbounded self-adjoint operator on $L^2(0,1)$, with a suitable dense domain such as

$$D(L) = H^2(0,1) \cap H_0^1(0,1).$$

The eigenvalues grow without bound.

Its inverse, defined on $L^2(0,1)$, is compact.

This is an important bridge between finite-dimensional spectral theory, compact operators, and PDEs.

16.11.5 Variational formulation of the PDE eigenproblem

Multiply

$$-u'' = \lambda u$$

by a test function $v \in H_0^1(0,1)$, and integrate by parts:

$$\int_0^1 u'(x)v'(x)\,dx = \lambda \int_0^1 u(x)v(x)\,dx.$$

Define

$$a(u,v) = \int_0^1 u'v'$$

and

$$m(u,v) = \int_0^1 uv.$$

Then the weak eigenvalue problem is

$$\boxed{a(u,v) = \lambda m(u,v) \qquad \forall v \in H_0^1(0,1).}$$

Its finite-element discretization gives

$$\boxed{Kc = \lambda Mc,}$$

where K and M are stiffness and mass matrices.

This is exactly the generalized eigenvalue structure introduced in Module C.

The finite-dimensional spectral theory therefore arises as an approximation of the spectral theory of a differential operator.

16.11.6 The min–max principle in PDE approximation

For the Dirichlet Laplacian, the Rayleigh quotient is

$$\boxed{\rho(u) = \frac{\int_0^1 |u'(x)|^2 \, dx}{\int_0^1 |u(x)|^2 \, dx}.}$$

The smallest eigenvalue can be characterized by

$$\boxed{\lambda_1 = \min_{u \in H_0^1(0,1) \setminus \{0\}} \rho(u).}$$

More generally, a min–max principle characterizes the higher eigenvalues using finite-dimensional trial subspaces of $H_0^1(0,1)$.

Now restrict the variational problem to a conforming finite-element space

$$V_h \subseteq H_0^1(0,1).$$

Because the minimization is performed over a smaller set,

$$\boxed{\lambda_{1,h} \geq \lambda_1.}$$

The corresponding upper-bound property extends to higher eigenvalues under the standard min–max ordering, provided the trial space has sufficient dimension.

As the finite-element spaces approximate the full function space, the computed eigenvalues can converge to the true PDE eigenvalues.

This provides a conceptual reason for eigenvalue convergence in variational numerical methods.

It also explains why Rayleigh–Ritz is so important in both finite-dimensional numerical linear algebra and infinite-dimensional spectral approximation.

16.11.7 Spectral theory as algebra: a module viewpoint

We can now revisit an abstract question from the beginning of the curriculum.

Why are invariant subspaces so fundamental?

There is an elegant answer from abstract algebra.

Let V be a vector space over $\mathbb{F}$, and let

$$T : V \to V$$

be a linear operator.

We can make V into a module over the polynomial ring $\mathbb{F}[t]$.

Define the action of a polynomial

$$p(t) = a_0 + a_1 t + \cdots + a_d t^d$$

on $v \in V$ by

$$\boxed{p(t) \cdot v = p(T)v.}$$

In particular,

$$t \cdot v = Tv.$$

Now ask: what are the submodules of this $\mathbb{F}[t]$-module?

A subspace $U \subseteq V$ is a submodule precisely when it is preserved by multiplication by t, meaning

$$T(U) \subseteq U.$$

Therefore,

$$\boxed{\text{Invariant subspaces of } T = \mathbb{F}[t]\text{-submodules of } V.}$$

This is a genuinely useful universal perspective.

It connects linear operator theory with module theory.

Eigenspaces as algebraic kernels

The eigenspace

$$E_\lambda = \ker(T - \lambda I)$$

consists of vectors annihilated by the polynomial $t - \lambda$:

$$(t - \lambda) \cdot v = 0.$$

Generalized eigenspaces consist of vectors annihilated by higher powers:

$$\ker(T - \lambda I)^k.$$

Thus Jordan theory can be understood in terms of the polynomial-ring module structure.

Diagonalizability and decomposition

A diagonalizable operator is one for which the module decomposes into a direct sum of eigenspaces:

$$V = \bigoplus_\lambda E_\lambda.$$

Over a field where the minimal polynomial splits, diagonalizability is equivalent to the minimal polynomial having no repeated roots.

For a Jordan block

$$J = \begin{pmatrix} 2 & 1 \\ 0 & 2 \end{pmatrix},$$

the minimal polynomial is

$$(t - 2)^2.$$

The repeated factor reflects the coupling that prevents diagonalization.

For a self-adjoint operator, the spectral theorem provides an especially strong decomposition: an orthogonal direct sum of eigenspaces.

Exact sequences return

If $U \subseteq V$ is invariant, the sequence

$$0 \longrightarrow U \longrightarrow V \longrightarrow V/U \longrightarrow 0$$

is not only an exact sequence of vector spaces.

It is also an exact sequence of $\mathbb{F}[t]$-modules, because the operator induces a well-defined action on the quotient.

A vector-space complement of U need not be invariant.

Therefore, the sequence may split as vector spaces without splitting as $\mathbb{F}[t]$-modules.

This is a structural explanation of why invariant complements are more restrictive than ordinary complements.

It also explains why Jordan blocks represent genuinely nontrivial extensions of eigenspace behavior.

Why this viewpoint is valuable

This perspective unifies several ideas:

- Linear operators become actions of polynomial rings.
- Invariant subspaces become submodules.
- Eigenspaces become kernels of polynomial actions.
- Spectral decompositions become module decompositions.
- Polynomial functional calculus becomes the ring action itself.

You do not need to study the full structure theorem for finitely generated modules over principal ideal domains before learning numerical eigenvalue methods.

But it is useful to recognize that the same mathematics has an abstract algebraic interpretation.

The Hilbert-space setting adds geometric requirements—orthogonality, completeness, and self-adjointness—that cannot be recovered from the module structure alone.

Checkpoint 8

Explain why the spectral theorem for finite-dimensional Hermitian matrices does not extend verbatim to all bounded self-adjoint operators on infinite-dimensional Hilbert spaces.

Then derive the generalized finite-element eigenproblem

$$Kc = \lambda Mc$$

from the weak PDE eigenvalue formulation.

Finally, prove that a subspace is invariant under T if and only if it is a submodule of V under the action $p(t) \cdot v = p(T)v$.

16.12 Exercises — Module D

The exercises are organized to develop the same six perspectives used throughout the curriculum: rigorous theory, geometry, computation, structural connections, universal generalization, and applications.

For the proof-based exercises, try writing a complete argument before consulting the solutions.

16.12.1 Exercise 1 — Diagonalizability and field dependence

Consider

$$R = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

and

$$J = \begin{pmatrix} 2 & 1 \\ 0 & 2 \end{pmatrix}.$$

Determine the eigenvalues and eigenspaces of both matrices over $\mathbb{R}$ and $\mathbb{C}$.

Which matrices are diagonalizable over each field?

Explain why the reasons for failure of real diagonalizability are different in these two examples.

16.12.2 Exercise 2 — An invariant subspace without an invariant complement

Let

$$J = \begin{pmatrix} 2 & 1 \\ 0 & 2 \end{pmatrix}, \qquad U = \operatorname{span}\{e_1\}.$$

Prove that U is invariant under J.

Show that U has many complementary vector subspaces.

However, prove that **none of its one-dimensional complements is invariant under** J.

Interpret the result using the distinction between ordinary vector-space direct sums and decompositions into invariant subspaces.

16.12.3 Exercise 3 — Spectral projectors

Consider

$$A = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 9 \end{pmatrix}.$$

Construct the spectral projector onto the eigenspace associated with eigenvalue 1 in two ways:

1. Using the corresponding normalized eigenvector.
2. Using the Lagrange interpolation polynomial

$$\ell_1(t) = \frac{(t-3)(t-9)}{(1-3)(1-9)}.$$

Verify that the two formulas agree.

Then calculate $p(A)$ for

$$p(t) = \frac{9-t}{8}.$$

16.12.4 Exercise 4 — Principal angles

Let

$$U = \operatorname{span}\{e_1, e_2\} \subseteq \mathbb{R}^3$$

and

$$W = \operatorname{span}\left\{\cos\phi\, e_1 + \sin\phi\, e_3,\ e_2\right\},$$

where $0 \le \phi \le \pi/2$.

Calculate the two principal angles between U and W.

Determine

$$\|P_U - P_W\|_2.$$

Explain why the answer is independent of the particular orthonormal bases chosen.

16.12.5 Exercise 5 — Rayleigh–Ritz and residuals

Let

$$A = \operatorname{diag}(1, 3, 9)$$

and

$$Q = \begin{pmatrix} 1/\sqrt{2} & 0 \\ 1/\sqrt{2} & 0 \\ 0 & 1 \end{pmatrix}.$$

Construct the reduced matrix

$$H = Q^T A Q.$$

Find its Ritz values and Ritz vectors.

Calculate the residual norm of each Ritz pair.

Verify the eigenvalue inclusion bound

$$\operatorname{dist}(\theta, \sigma(A)) \le \|r\|_2.$$

16.12.6 Exercise 6 — Eigenvalue versus eigenvector accuracy

Consider

$$A = \operatorname{diag}(1, 4)$$

and

$$x(\theta) = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}.$$

Derive

$$\rho_A(x) = 1 + 3\sin^2\theta.$$

Calculate the residual

$$r = Ax - \rho_A(x)x$$

and its norm.

Show that the Rayleigh quotient error relative to eigenvalue 1 is quadratic in $\sin\theta$, while the residual norm is asymptotically linear in $\sin\theta$ near $\theta = 0$.

16.12.7 Exercise 7 — SVD and minimum-norm least squares

Consider

$$A = \begin{pmatrix} 3 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

Find an SVD of A.

Determine its pseudoinverse A^{+}.

For

$$b = (6, 2, 5)^T,$$

calculate the least-squares minimizer and the residual.

Find the best rank-one approximation of A in spectral norm, and give its error.

16.12.8 Exercise 8 — Power iteration

Let

$$A = \mathrm{diag}(9, 3, 1), \qquad x_0 = (1, 1, 1)^T.$$

Find $A^k x_0$.

Determine the ratios of the second and third spectral coefficients to the first after k iterations.

Explain the convergence rate.

What happens if the initial vector instead belongs to

$$\mathrm{span}\{e_2, e_3\}?$$

16.12.9 Exercise 9 — A Chebyshev filter

Let

$$A = \mathrm{diag}(1, 3, 9).$$

We want to retain the eigenspace associated with eigenvalue 1.

Take the unwanted spectral interval $[3, 9]$.

Construct the affine map taking this interval onto $[-1, 1]$.

Then construct the normalized degree-three Chebyshev filter satisfying $p_3(1) = 1$.

Calculate

$$p_3(3), \qquad p_3(9).$$

Apply the filter to

$$x = (1, 1, 1)^T.$$

Explain the result in spectral terms.

16.12.10 Exercise 10 — A residual-based subspace estimate

Let

$$A = \mathrm{diag}(1, 4)$$

and

$$x(\theta) = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}.$$

Let

$$U = \mathrm{span}\{e_1\}.$$

Using the Rayleigh quotient $\rho_A(x)$, show that

$$\boxed{\|(I - P_U)x\| \le \frac{\|Ax - \rho_A(x)x\|}{|4 - \rho_A(x)|}}$$

whenever $\rho_A(x) \neq 4$.

Determine the value of the left-hand side and discuss when the bound is informative.

16.12.11 Exercise 11 — Spectral theory of a differential operator

Consider

$$-u'' = \lambda u$$

on $(0, 1)$, with

$$u(0) = u(1) = 0.$$

Derive the eigenvalues and eigenfunctions.

Normalize the eigenfunctions in $L^2(0,1)$.

Write down the Rayleigh quotient associated with the problem.

Explain why restricting the variational eigenproblem to a finite-dimensional conforming subspace produces approximations of the eigenvalues from above.

16.12.12 Exercise 12 — Universal structure

Let $T : V \to V$ be linear and define an action of $\mathbb{F}[t]$ on V by

$$p(t) \cdot v = p(T)v.$$

Prove that a vector subspace $U \subseteq V$ is a submodule if and only if U is invariant under T.

Explain why an invariant subspace permits a quotient operator on V/U.

Finally, use the Jordan block from Exercise 2 to explain why a short exact sequence of operator modules need not split, even when the corresponding sequence of vector spaces does.

16.13 Worked solutions

16.13.1 Solution 1 — Diagonalizability

For

$$R = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix},$$

the characteristic polynomial is

$$\lambda^2 + 1.$$

Thus it has no real eigenvalues.

Over $\mathbb{C}$, the eigenvalues are

$$\lambda_1 = i, \qquad \lambda_2 = -i.$$

Corresponding eigenvectors can be chosen as

$$v_1 = (1, -i)^T, \qquad v_2 = (1, i)^T.$$

They are linearly independent over $\mathbb{C}$, so R is diagonalizable over $\mathbb{C}$, but not over $\mathbb{R}$.

For

$$J = \begin{pmatrix} 2 & 1 \\ 0 & 2 \end{pmatrix},$$

the only eigenvalue is 2, with algebraic multiplicity two.

The eigenspace is

$$E_2 = \ker(J - 2I) = \operatorname{span}\{e_1\}.$$

Thus J has only one independent eigenvector, even over $\mathbb{C}$.

It is not diagonalizable over either field.

The rotation matrix lacks real eigenvalues because its characteristic polynomial does not split over $\mathbb{R}$.

The Jordan block is defective because its eigenspace has insufficient dimension, even over a field containing all its eigenvalues.

16.13.2 Solution 2 — No invariant complement

We have

$$Je_1 = 2e_1,$$

so $U = \operatorname{span}\{e_1\}$ is invariant.

Every line

$$W = \operatorname{span}\{(a, b)^T\}$$

with $b \neq 0$ is a complement of U.

Suppose such a complement were invariant.

Because it is one-dimensional, its nonzero generator would have to be an eigenvector of J:

$$J(a,b)^T = \lambda(a,b)^T.$$

But the only eigenvectors of J belong to $\mathrm{span}\{e_1\}$, which requires $b = 0$.

This contradicts the requirement $b \neq 0$.

Therefore U has vector-space complements but no invariant complement.

The short exact sequence involving U, V, and V/U splits as vector spaces, but not as modules under the action of J.

16.13.3 Solution 3 — Spectral projectors

The normalized eigenvector for eigenvalue 1 is

$$q_1 = \frac{1}{\sqrt{2}}(1,-1,0)^T.$$

Therefore,

$$\boxed{P_1 = q_1 q_1^T = \frac{1}{2}\begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}.}$$

The Lagrange polynomial is

$$\ell_1(t) = \frac{(t-3)(t-9)}{16}.$$

It satisfies

$$\ell_1(1) = 1, \qquad \ell_1(3) = 0, \qquad \ell_1(9) = 0.$$

Therefore, by spectral functional calculus,

$$\ell_1(A) = P_1.$$

For

$$p(t) = \frac{9-t}{8},$$

the spectral values are

$$p(1) = 1, \qquad p(3) = \frac{3}{4}, \qquad p(9) = 0.$$

Hence

$$p(A) = P_1 + \frac{3}{4} P_2.$$

Using the explicit projectors,

$$\boxed{p(A) = \begin{pmatrix} 7/8 & -1/8 & 0 \\ -1/8 & 7/8 & 0 \\ 0 & 0 & 0 \end{pmatrix}.}$$

16.13.4 Solution 4 — Principal angles

Choose

$$Q_U = (e_1 \; e_2)$$

and

$$Q_W = (\cos\phi \, e_1 + \sin\phi \, e_3 \;\; e_2).$$

Then

$$Q_U^T Q_W = \begin{pmatrix} \cos\phi & 0 \\ 0 & 1 \end{pmatrix}.$$

Its singular values are

$$1, \qquad \cos\phi.$$

Thus the principal angles are

$$\boxed{0, \quad \phi.}$$

The largest principal angle is ϕ.

Therefore,

$$\boxed{\|P_U - P_W\|_2 = \sin\phi.}$$

Changing either orthonormal basis multiplies $Q_U^T Q_W$ by orthogonal matrices on the left or right, leaving its singular values unchanged.

Thus the principal angles depend only on the subspaces.

16.13.5 Solution 5 — Rayleigh–Ritz

The reduced matrix is

$$H = \begin{pmatrix} 2 & 0 \\ 0 & 9 \end{pmatrix}.$$

The Ritz values are

$$\boxed{\theta_1 = 2, \qquad \theta_2 = 9.}$$

Corresponding Ritz vectors are

$$u_1 = \frac{1}{\sqrt{2}}(1,1,0)^T, \qquad u_2 = e_3.$$

For the first pair,

$$r_1 = Au_1 - 2u_1 = \frac{1}{\sqrt{2}}(-1,1,0)^T.$$

Thus

$$\boxed{\|r_1\|_2 = 1.}$$

For the second pair,

$$r_2 = Au_2 - 9u_2 = 0.$$

The spectrum of A is $\{1,3,9\}$.

Therefore,

$$\operatorname{dist}(2, \sigma(A)) = 1 = \|r_1\|.$$

And

$$\operatorname{dist}(9, \sigma(A)) = 0 = \|r_2\|.$$

Both satisfy the residual inclusion bound.

16.13.6 Solution 6 — Quadratic eigenvalue error

For

$$x = (\cos\theta, \sin\theta)^T,$$

the Rayleigh quotient is

$$\rho_A(x) = \cos^2\theta + 4\sin^2\theta.$$

Since $\cos^2\theta + \sin^2\theta = 1$,

$$\boxed{\rho_A(x) = 1 + 3\sin^2\theta.}$$

Therefore,

$$\boxed{\rho_A(x) - 1 = 3\sin^2\theta.}$$

The residual is

$$r = 3\sin\theta\cos\theta \begin{pmatrix} -\sin\theta \\ \cos\theta \end{pmatrix}.$$

Hence

$$\boxed{\|r\|_2 = 3|\sin\theta\cos\theta|.}$$

Near $\theta = 0$,

$$\sin\theta \sim \theta, \qquad \cos\theta \sim 1.$$

Thus

$$\rho_A(x) - 1 \sim 3\theta^2,$$

while

$$\|r\| \sim 3|\theta|.$$

The eigenvalue error is second-order in the angular error, while the residual is first-order.

16.13.7 Solution 7 — SVD and pseudoinverse

The matrix

$$A = \begin{pmatrix} 3 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}$$

is already in rectangular diagonal form.

We may take

$$U = I_3, \qquad V = I_2,$$

and

$$\Sigma = A.$$

The singular values are

$$\sigma_1 = 3, \qquad \sigma_2 = 1.$$

Its pseudoinverse is

$$\boxed{A^+ = \begin{pmatrix} 1/3 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}.}$$

For $b = (6, 2, 5)^T$,

$$x_{\mathrm{LS}} = A^{+} b = \boxed{\begin{pmatrix} 2 \\ 2 \end{pmatrix}.}$$

Then

$$A x_{\mathrm{LS}} = \begin{pmatrix} 6 \\ 2 \\ 0 \end{pmatrix},$$

so

$$\boxed{r = b - A x_{\mathrm{LS}} = \begin{pmatrix} 0 \\ 0 \\ 5 \end{pmatrix}.}$$

The best rank-one approximation is

$$\boxed{A_1 = \begin{pmatrix} 3 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}.}$$

Its spectral-norm error is

$$\boxed{\|A - A_1\|_2 = \sigma_2 = 1.}$$

16.13.8 Solution 8 — Power iteration

Since A is diagonal,

$$\boxed{A^k x_0 = (9^k, 3^k, 1)^T.}$$

The ratio of the second spectral coefficient to the first is

$$\frac{3^k}{9^k} = \boxed{\left(\frac{1}{3}\right)^k}.$$

The ratio of the third to the first is

$$\frac{1}{9^k} = \boxed{\left(\frac{1}{9}\right)^k}.$$

Thus the subspace approaches $\operatorname{span}\{e_1\}$, with asymptotic convergence factor $1/3$.

If the initial vector has no e_1 component, applying A cannot create one, because the eigenspaces are invariant.

Therefore power iteration cannot converge to $\operatorname{span}\{e_1\}$ from an initial vector entirely in $\operatorname{span}\{e_2, e_3\}$.

If the e_2 component is nonzero, it instead tends toward $\operatorname{span}\{e_2\}$.

16.13.9 Solution 9 — Chebyshev filter

The unwanted interval is $[3, 9]$.

The affine spectral map is

$$\boxed{s(t) = \frac{t-6}{3}.}$$

The degree-three Chebyshev polynomial is

$$T_3(z) = 4z^3 - 3z.$$

At the desired eigenvalue 1,

$$T_3(s(1)) = T_3(-5/3) = -\frac{365}{27}.$$

Thus the normalized filter is

$$\boxed{p_3(t) = -\frac{27}{365} T_3\left(\frac{t-6}{3}\right).}$$

We obtain

$$\boxed{p_3(1) = 1, \qquad p_3(3) = \frac{27}{365}, \qquad p_3(9) = -\frac{27}{365}.}$$

Since A is diagonal,

$$\boxed{p_3(A)(1,1,1)^T = \begin{pmatrix} 1 \\ 27/365 \\ -27/365 \end{pmatrix}.}$$

The desired spectral component is preserved, while each unwanted component is reduced to about 7.4% of its original magnitude.

16.13.10 Solution 10 — Residual and angular error

The target projector is

$$P_U = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}.$$

Thus

$$(I - P_U)x = \begin{pmatrix} 0 \\ \sin\theta \end{pmatrix},$$

so

$$\boxed{\|(I - P_U)x\| = |\sin\theta|.}$$

The unwanted eigenvalue is 4, and the separation from the Rayleigh quotient is

$$\delta = |4 - \rho_A(x)|.$$

The residual bound gives

$$|\sin\theta| \leq \frac{3|\sin\theta\cos\theta|}{|4 - (1 + 3\sin^2\theta)|}.$$

For $0 \leq \theta < \pi/2$, the denominator is

$$3\cos^2\theta.$$

Therefore the right-hand side becomes

$$\frac{\sin\theta}{\cos\theta} = \tan\theta.$$

Hence

$$\boxed{\sin\theta \le \tan\theta.}$$

The bound is sharp to first order near $\theta = 0$, where $\sin\theta \approx \tan\theta$.

It becomes less informative as $\theta \to \pi/2$, because the approximate eigenvalue approaches the unwanted eigenvalue 4 and the separation δ tends to zero.

16.13.11 Solution 11 — Dirichlet Laplacian

The eigenvalue equation is

$$-u'' = \lambda u.$$

A nonzero solution satisfying the Dirichlet boundary conditions exists precisely for

$$\boxed{\lambda_n = n^2\pi^2, \qquad n = 1, 2, \dots.}$$

Corresponding eigenfunctions are

$$u_n(x) = \sin(n\pi x).$$

Since

$$\int_0^1 \sin^2(n\pi x)\,dx = \frac{1}{2},$$

the normalized eigenfunctions are

$$\boxed{q_n(x) = \sqrt{2}\sin(n\pi x).}$$

The Rayleigh quotient is

$$\boxed{\rho(u) = \frac{\int_0^1 |u'(x)|^2\,dx}{\int_0^1 |u(x)|^2\,dx}.}$$

The min–max principle characterizes the eigenvalues through trial subspaces.

When we restrict the admissible subspaces to a conforming finite-dimensional space $V_h \subseteq H_0^1(0,1)$, we minimize over fewer possibilities.

Therefore the computed Ritz eigenvalues cannot be smaller than the corresponding exact eigenvalues:

$$\boxed{\lambda_{k,h} \geq \lambda_k}$$

for the indices represented by the discrete problem.

16.13.12 Solution 12 — Operator modules

The action of $\mathbb{F}[t]$ is defined by

$$p(t) \cdot v = p(T)v.$$

A vector subspace U is a submodule if it is closed under the action of all polynomials.

If U is a submodule, it is in particular closed under multiplication by t:

$$t \cdot u = Tu \in U.$$

Thus U is T-invariant.

Conversely, if $T(U) \subseteq U$, then induction gives

$$T^k(U) \subseteq U$$

for all k.

Consequently,

$$p(T)(U) \subseteq U$$

for every polynomial p.

Thus U is a submodule.

For the quotient, define

$$\overline{T}(v+U) = Tv + U.$$

This is well-defined because U is invariant.

For the Jordan block J from Exercise 2, $U = \operatorname{span}\{e_1\}$ is invariant but has no invariant complement.

Therefore,

$$0 \to U \to V \to V/U \to 0$$

splits as vector spaces but not as $\mathbb{F}[t]$-modules.

An operator-compatible direct-sum decomposition is more restrictive than an arbitrary vector-space decomposition.

16.14 Computational laboratory — MATLAB

The laboratory is designed to connect the theoretical material to numerical computation.

Each experiment demonstrates a different aspect of spectral theory.

The examples are deliberately small or moderately sized so that you can compare numerical output with analytical predictions.

One numerical correction to the illustrative Chebyshev attenuation chart in Session 7: the exact degree-four and degree-five bounds are respectively $81/3281 \approx 0.024688$ and $243/29525 \approx 0.008230$.

```
%% Module D - Spectral Theory
clear; clc; close all;

%% 1. Spectral theorem and orthogonal eigenbasis
A = [2 1 0;
     1 2 0;
     0 0 9];

[Q,Lambda] = eig(A);

% Sort eigenvalues and reorder eigenvectors
```

```
[lambda,idx] = sort(diag(Lambda));
Q = Q(:,idx);
Lambda = diag(lambda);

disp('Eigenvalues:');
disp(lambda);

fprintf('Orthogonality error: %.3e\n', ...
    norm(Q.'*Q-eye(3)));

fprintf('Spectral reconstruction error: %.3e\n', ...
    norm(A-Q*Lambda*Q.'));


%% 2. Spectral projectors
P1 = Q(:,1)*Q(:,1).';
P2 = Q(:,2)*Q(:,2).';
P3 = Q(:,3)*Q(:,3).';

fprintf('Projector sum error: %.3e\n', ...
    norm(P1+P2+P3-eye(3)));

fprintf('Spectral decomposition error: %.3e\n', ...
    norm(A-(lambda(1)*P1 ...
           +lambda(2)*P2+lambda(3)*P3)));

% Lagrange polynomial projector onto eigenvalue 1
I = eye(3);
P1poly = ((A-3*I)*(A-9*I))/16;

fprintf('Polynomial projector error: %.3e\n', ...
    norm(P1poly-P1));


%% 3. Principal angles between two planes
phi = 20*pi/180;

e1 = [1;0;0];
e2 = [0;1;0];
e3 = [0;0;1];

Qu = [e1, e2];
```

```
Qw = [cos(phi)*e1+sin(phi)*e3, e2];

s = svd(Qu.'*Qw);
s = min(1,max(0,s));  % Numerical safety

principal_angles = acos(s);

disp('Principal angles in degrees:');
disp(sort(principal_angles)*180/pi);

Pu = Qu*Qu.';
Pw = Qw*Qw.';

fprintf('||Pu-Pw||: %.8f\n', norm(Pu-Pw));
fprintf('sin(phi):  %.8f\n', sin(phi));


%% 4. Rayleigh-Ritz and eigenpair residuals
B = diag([1,3,9]);

Qtrial = [(e1+e2)/sqrt(2), e3];

H = Qtrial.'*B*Qtrial;
[Y,D] = eig(H);

theta = diag(D);
RitzVectors = Qtrial*Y;

disp('Ritz values:');
disp(theta);

for j = 1:length(theta)
    r = B*RitzVectors(:,j) ...
        -theta(j)*RitzVectors(:,j);

    fprintf('Ritz residual %d: %.8f\n', ...
        j,norm(r));
end


%% 5. Singular value decomposition
Arect = [3 0;
```

```
         0 1;
         0 0];

[U,S,V] = svd(Arect);

fprintf('SVD reconstruction error: %.3e\n', ...
    norm(Arect-U*S*V.'));

disp('Singular values:');
disp(svd(Arect));

% Minimum-norm least squares
b = [6;2;5];
x_min = pinv(Arect)*b;

disp('Least-squares solution:');
disp(x_min);

disp('Residual:');
disp(b-Arect*x_min);

% Best rank-one approximation
A1 = U(:,1)*S(1,1)*V(:,1).';

fprintf('Best rank-one error: %.8f\n', ...
    norm(Arect-A1));


%% 6. Power iteration
Apow = diag([9,3,1]);

x = ones(3,1);
x = x/norm(x);

for j = 1:12
    x = Apow*x;
    x = x/norm(x);
end

disp('Power iteration direction:');
disp(x);
```

```
rho = x.'*Apow*x;

fprintf('Rayleigh quotient: %.10f\n', rho);


%% 7. Subspace iteration: principal-angle convergence
rng(7);

n = 40;
k = 3;

[Vbig,~] = qr(randn(n),0);

spectrum = [9;8;7;linspace(2,0.05,n-3).'];

Abig = Vbig*diag(spectrum)*Vbig.';

Qtarget = Vbig(:,1:k);

[X,~] = qr(randn(n,k),0);

iterations = 15;
subspace_error = zeros(iterations+1,1);

for j = 1:iterations+1
    singular_cosines = svd(Qtarget.'*X);
    sigma_min = min(singular_cosines);

    subspace_error(j) = ...
        sqrt(max(0,1-sigma_min^2));

    if j <= iterations
        Y = Abig*X;
        [X,~] = qr(Y,0);
    end
end

figure;
semilogy(0:iterations, ...
    max(subspace_error,eps),'-o');
xlabel('Subspace iteration');
ylabel('sin(largest principal angle)');
```

```
title('Subspace iteration convergence');
grid on;


%% 8. Chebyshev filtering
Acheb = diag([1,3,9]);
x = ones(3,1);

% Map unwanted spectrum [3,9] to [-1,1]
Bcheb = (Acheb-6*eye(3))/3;

degree = 3;

Yprev = x;
Ycurr = Bcheb*x;

for j = 2:degree
    Ynext = 2*Bcheb*Ycurr-Yprev;
    Yprev = Ycurr;
    Ycurr = Ynext;
end

% T_3(-5/3) = -365/27
normalization = -365/27;

y_filtered = Ycurr/normalization;

disp('Chebyshev filtered vector:');
disp(y_filtered);

disp('Expected: [1;27/365;-27/365]');


%% 9. Generalized eigenproblem from FEM
Kfe = [6 -3;
      -3 6];

Mfe = (1/18)*[4 1;
              1 4];

[Vfe,Dfe] = eig(Kfe,Mfe);
```

```
lambda_fem = sort(diag(Dfe));

disp('FEM generalized eigenvalues:');
disp(lambda_fem);

disp('Exact first two eigenvalues:');
disp(([1;2]*pi).^2);
```

16.14.1 What to investigate

Experiment A — Spectral projectors

Verify

$$P_i^2 = P_i, \qquad P_i P_j = 0 \ (i \neq j),$$

and

$$\sum_i P_i = I.$$

Then replace the symmetric matrix by a diagonalizable nonsymmetric matrix.

Construct projectors using its right eigenvectors and dual left eigenvectors.

Investigate which algebraic identities still hold and which geometric identities no longer hold.

Experiment B — Subspace convergence

In the subspace iteration experiment, the desired eigenvalues are $9, 8, 7$, while the largest unwanted eigenvalue is 2.

Thus the expected exact-arithmetic convergence factor is related to

$$\frac{2}{7}.$$

Change the unwanted spectral values so that the largest becomes 6.8.

What happens to the convergence rate?

Then replace ordinary multiplication by a polynomial filter designed to separate the desired spectral region from the unwanted region.

Compare the principal-angle convergence histories.

Experiment C — Chebyshev filtering

Increase the filter degree from 3 to 5, 8, and 12.

Track the ratio of the unwanted components to the desired component after filtering.

Compare the measured reduction with the theoretical bound

$$\gamma_d = \frac{1}{|T_d(-5/3)|}.$$

At high degrees, investigate whether normalization is necessary to keep intermediate values in a convenient numerical range.

Experiment D — Generalized eigenvalues

The finite-element generalized eigenproblem approximates the continuous Dirichlet Laplacian.

Compare the first two discrete eigenvalues with

$$\lambda_1 = \pi^2, \qquad \lambda_2 = 4\pi^2.$$

Then refine the mesh by increasing the number of interior nodes.

Observe how the generalized eigenvalues approach the continuous ones.

This experiment connects abstract spectral theory to numerical PDE approximation.

16.15 Synthesis — How the concepts fit together

Module D brings together the algebraic, geometric, and numerical structures from the first three modules.

The central chain of ideas can be summarized as follows.

Linear operator

$$T : V \to V$$

Kernels, invariant subspaces, quotient operators

Spectral structure

$$V = \bigoplus_{\lambda} E_{\lambda}$$

Independent invariant components when diagonalizable

Add self-adjointness and inner-product geometry

Orthogonal spectral decomposition

$$A = \sum_{\lambda} \lambda P_{\lambda}$$

Orthogonal projectors and an orthonormal eigenbasis

Variational approximation

$$\rho_A(x), \qquad Q^* A Q$$

Rayleigh quotients, Rayleigh–Ritz, residuals, and spectral gaps

Numerical invariant-subspace methods

$$X_{k+1} \approx p(A) X_k$$

Subspace iteration, Chebyshev filtering, and residual-based accuracy assessment

16.15.1 Relationships worth retaining

Concept	Central theorem or formula	Why it matters
Eigenspace	$E_\lambda = \ker(A - \lambda I)$	Connects spectral theory to kernels
Invariant subspace	$A(U) \subseteq U$	Allows restriction and quotient operators
Diagonalizability	$A = X \Lambda X^{-1}$	Decomposes an operator into scalar actions
Spectral theorem	$A = Q \Lambda Q^*$	Orthogonal decomposition for normal matrices
Spectral projector	$P_j^2 = P_j$, $AP_j = \lambda_j P_j$	Extracts invariant spectral components

Concept	Central theorem or formula	Why it matters
Functional calculus	$p(A) = Qp(\Lambda)Q^*$	Explains polynomial filters
Principal angles	$\cos\theta_i = \sigma_i(Q_U^* Q_W)$	Measures distance between subspaces
Invariance residual	$(I - QQ^*)AQ$	Measures failure of a trial subspace to be invariant
Rayleigh quotient	$x^*Ax/(x^*x)$	Gives variational eigenvalue estimates
Min–max principle	$\lambda_k = \min_{\dim S=k} \max_{x\in S} \rho_A(x)$	Characterizes ordered eigenvalues
Rayleigh–Ritz	$Q^*AQy = \theta y$	Computes eigenpair approximations within a trial subspace
SVD	$A = U\Sigma V^*$	Spectral geometry for rectangular matrices
Best rank-k approximation	$\|A - A_k\|_2 = \sigma_{k+1}$	Optimal dimensional reduction
Power iteration	$x_{j+1} \propto Ax_j$	Amplifies dominant spectral components
Subspace iteration	$\operatorname{im} X_{j+1} = \operatorname{im}(AX_j)$	Approximates dominant invariant subspaces
Chebyshev filtering	$\|T_d(t)\| \le 1$ on $[-1, 1]$	Suppresses unwanted spectral components
Filtered convergence factor	$\gamma_p = \max_{\text{unwanted}}\|p\| / \min_{\text{wanted}}\|p\|$	Predicts improvement in subspace angle

16.16 Supplementary proof — Why normal matrices are unitarily diagonalizable

We proved the spectral theorem for self-adjoint matrices. The more general normal-matrix theorem is worth understanding because it clarifies exactly which hypothesis guarantees an orthonormal eigenbasis.

16.16.1 Step 1: Schur triangularization

Every complex square matrix A has a **Schur decomposition**

$$\boxed{A = QRQ^*,}$$

where Q is unitary and R is upper triangular.

One way to prove this is by induction.

Because the characteristic polynomial has a complex root, A has an eigenvector q_1.

Normalize it and extend it to an orthonormal basis.

In that basis, the first column of A has the form

$$\begin{pmatrix} \lambda \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Thus the matrix has block upper-triangular form

$$\begin{pmatrix} \lambda & * \\ 0 & B \end{pmatrix}.$$

Apply the same argument inductively to B, using a unitary basis change within the orthogonal complement of q_1.

This produces an upper-triangular matrix R.

16.16.2 Step 2: A normal upper-triangular matrix is diagonal

Suppose R is upper triangular and normal:

$$R^*R = RR^*.$$

Normality implies

$$\|Rx\| = \|R^*x\|$$

for every x.

Apply this to e_1.

Because R is upper triangular, its first column has only one potentially nonzero entry:

$$Re_1 = \begin{pmatrix} r_{11} \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Hence

$$\|Re_1\|^2 = |r_{11}|^2.$$

But $R^* e_1$ contains the conjugates of the first-row entries:

$$R^* e_1 = \begin{pmatrix} \overline{r_{11}} \\ \overline{r_{12}} \\ \vdots \\ \overline{r_{1n}} \end{pmatrix}.$$

Thus

$$\|R^* e_1\|^2 = |r_{11}|^2 + \sum_{j=2}^{n} |r_{1j}|^2.$$

Equality of the norms implies

$$\sum_{j=2}^{n} |r_{1j}|^2 = 0.$$

Therefore,

$$r_{12} = \cdots = r_{1n} = 0.$$

The first row has no off-diagonal entries.

The remaining lower-right block is again upper triangular and normal.

Repeating the argument inductively shows that R is diagonal.

Therefore,

$$\boxed{A = Q\Lambda Q^*.}$$

Conversely, if $A = Q\Lambda Q^*$ for unitary Q and diagonal Λ, then

$$AA^* = Q\Lambda\Lambda^*Q^*$$

and

$$A^*A = Q\Lambda^*\Lambda Q^*.$$

Because diagonal matrices commute,

$$AA^* = A^*A.$$

Thus the equivalence is proved:

$$\boxed{A \text{ is normal } \iff A \text{ is unitarily diagonalizable.}}$$

This theorem shows precisely why nonnormal diagonalizable matrices behave differently from Hermitian matrices: an arbitrary eigenvector basis need not be orthogonal, while a normal matrix always admits an orthonormal one.

16.17 Module D — Final mastery assessment

Before proceeding to Module E, the goal is not to memorize every formula. It is to be able to derive the central results, recognize their assumptions, and apply them to unfamiliar examples.

- ☐ Explain eigenvalues through kernels and distinguish invertibility, diagonalizability, normality, and self-adjointness.
- ☐ Prove the finite-dimensional self-adjoint spectral theorem and understand the normal spectral theorem.
- ☐ Construct spectral projectors and derive polynomial functional calculus.
- ☐ Calculate principal angles and explain the meaning of an invariant-subspace residual.
- ☐ Prove the min–max principle and derive the Rayleigh–Ritz equations.
- ☐ Prove eigenvalue residual inclusion and explain residual-to-subspace-error estimates.
- ☐ Construct the SVD, pseudoinverse, and optimal truncated SVD.
- ☐ Explain the convergence mechanisms of power iteration and subspace iteration.
- ☐ Construct and evaluate a Chebyshev filter, and derive its subspace convergence factor.
- ☐ Connect spectral theory to PDE eigenproblems, compact operators, and the abstract algebra of invariant subspaces.

16.17.1 Recommended consolidation exercise

For the final synthesis, take one Hermitian matrix such as

$$A = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 9 \end{pmatrix}$$

and reconstruct the following chain without consulting the notes:

Find eigenvalues and eigenspaces
↓
Construct the orthogonal spectral projectors
↓
Express A through its projectors
↓
Construct a polynomial filter $p(A)$
↓
Apply the filter to a trial vector or subspace
↓
Compute Ritz values and Ritz vectors
↓
Evaluate residuals and subspace angles
↓
Explain convergence using spectral separation.

If you can perform and explain this complete sequence, including why each step is mathematically justified, you have achieved the central objective of Module D.

16.18 Transition to Module E — Numerical methods and stability

Module D has established the exact mathematical structures underlying eigenvalue algorithms.

Module E will investigate what happens when we try to realize those structures computationally.

The focus will shift from the question

> Why does the spectral algorithm work in exact arithmetic?

to the more numerical question:

> Under what assumptions, and with what accuracy and cost, can we expect it to work in floating-point arithmetic?

The principal subjects will include power and inverse iteration, block subspace iteration, Krylov methods, Rayleigh–Ritz and numerical residuals, Chebyshev recurrence evaluation, conditioning, backward stability, perturbation theory, and precision-dependent convergence.

In the context of hardware-aware scientific computing, a central issue will be the distinction between **mathematical convergence mechanisms** and **implementation-dependent numerical behavior**.

For example, an exact-arithmetic Chebyshev filter may reduce unwanted spectral components by a theoretically predictable factor. But an implementation using inexact matrix products and reduced precision also introduces errors that may accumulate, be amplified, or eventually create an accuracy floor.

The study of those interactions is where spectral theory becomes numerical analysis.

The central achievement of Module D is a unified understanding of spectral decomposition, invariant-subspace geometry, variational eigenvalue approximation, and polynomial spectral filtering. Together, these concepts provide the mathematical foundation for studying and analyzing modern numerical eigenvalue algorithms—and, more specifically, Chebyshev-filtered subspace iteration.

17 Module E — Numerical Eigenvalue Methods, Conditioning, and Stability

17.1 From a spectral decomposition to a numerical claim

Suppose a program returns a unit vector x and a number θ. You can multiply by A, but you do not know any of its eigenvectors. What calculation would justify the statement "this is an accurate approximation to the second eigenvector"? Computing $\|Ax - \theta x\|$ is a first step. You still need to identify the eigenvalue, establish separation from competing directions, and know how accurately you evaluated the residual itself. Each of the eight sessions develops one part of that argument.

Starting point in Review II. Module A (§4.11) supplied invariant subspaces and direct sums. Module B (§5.7) explained how a functional pulls back through a map. Module C (§§6.7, 6.11) supplied orthogonal projections and the mass-matrix inner product. Module D (§§7.5–7.10) established the spectral theorem, principal angles, Rayleigh–Ritz, and polynomial filtering. Its §7.18 asks what accuracy and cost survive implementation. Here a spectral theorem becomes both a way to design an algorithm and a way to audit its output.

Running example and conventions. Work over a finite-dimensional real or complex inner-product space, using orthonormal coordinates unless specified. A star denotes conjugate transpose. Write $\|\cdot\|$ for the Euclidean or induced spectral norm, $\|\cdot\|_F$ for the Frobenius norm. The Rayleigh quotient is $\rho_A(x) = x^*Ax/(x^*x)$. Reuse the matrix from §7.5.5,

$$A_0 = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 9 \end{pmatrix}, \quad q_1 = (1, -1, 0)^T/\sqrt{2}, \quad q_2 = (1, 1, 0)^T/\sqrt{2}, \quad q_3 = e_3.$$

The corresponding eigenvalues are $1, 3, 9$. These subscripts stay attached to these vectors. For general results we state the ordering separately, since largest magnitude, largest algebraic value and closest to a shift are three different targets.

Session	Question to reconstruct at the board
1. Arithmetic	Which nearby problem explains a computed answer?
2. Perturbations	What can a small residual certify?

Session	Question to reconstruct at the board
3. Vector iteration	Which component survives repeated application?
4. Block iteration	What contracts when individual columns can rotate?
5. Krylov projection	What do we gain by keeping earlier powers?
6. Polynomial filters	How do spectral separation and recurrence errors interact?
7. Mixed precision	Which operation limits the verified accuracy?
8. Case study	Do the measured histories agree with the prediction?

Read Sessions 1–2 together, derive Sessions 3–4 first for A_0, and work through Session 6 once with scalars before interpreting a matrix experiment. The laboratory is a single MATLAB script collected after the exercises.

17.2 Session 1 — Floating-point arithmetic, conditioning, and stability

17.2.1 What one rounding step changes

In binary precision p a normalized positive number has form $(1.b_2 \dots b_p)_2 2^e$ with a bounded exponent. Adjacent numbers in $[1, 2)$ are 2^{1-p} apart. Rounding to nearest therefore gives unit roundoff $u = 2^{-p}$. In binary32, $p = 24$; in binary64, $p = 53$. MATLAB's spacing at one is $2u$. The exponent controls range independently of the precision. For a correctly rounded real operation with exact result in the normal range and no overflow,

$$\mathrm{fl}(a \circ b) = (a \circ b)(1 + \delta), \quad |\delta| \le u.$$

Indeed half an ulp in the binade starting at 2^e is $2^e u$, while the result has magnitude at least 2^e. Zero is exact. Near underflow one needs an absolute-error term. A fused multiply-add rounds the combined expression once. Complex arithmetic implemented with real operations has different operation counts.

Hand experiment. With $p = 3$ and ties to even, $1 + 1/8$ rounds to 1. Thus $(1 + 1/8) - 1$ evaluates to zero while $(1 - 1) + 1/8$ evaluates exactly. The final subtraction in the first expression is itself exact. It exposes the loss in the preceding sum. This is the same difficulty that arises when subtracting two large nearly equal terms in an eigenpair residual.

17.2.2 From dot products to a matrix product

For $ku < 1$ write $\gamma_k = ku/(1-ku)$. A product of at most k factors $1+\delta_j$ differs from one by at most γ_k: its upper bound is $(1+u)^k \le 1/(1-ku)$ by binomial expansion and the geometric series, and its lower bound is at least $1-ku$. Unroll a sequential real dot product. Each exact summand $a_j x_j$ is multiplied by the rounding factor from its product and factors from subsequent additions, at most n in total when the first product initializes the sum. Thus

$$|\,\mathrm{fl}(a^T x) - a^T x| \le \gamma_n \sum_j |a_j x_j|, \qquad |\,\mathrm{fl}(Ax) - Ax| \le \gamma_n |A|\,|x|.$$

Absolute values and the second inequality are entrywise. Equivalently, for this multiply there exists ΔA with $|\Delta A| \le \gamma_n |A|$ giving the computed result exactly as $(A+\Delta A)x$. The perturbation can change with the input and with each multiply. For sparse rows one can replace n by their maximum number of summed products under the same arithmetic model. Cancellation can make the *relative output error* large even when the componentwise backward error is small.

17.2.3 Problem sensitivity versus the error of a method

For differentiable f the local absolute condition number in specified norms is $\|Df(d)\|$. For $d, f(d) \neq 0$, the relative one is $\|Df(d)\|\|d\|/\|f(d)\|$. This follows by maximizing the first-order change $Df(d)h$ over a ball of input perturbations. Specifying which data may change is part of defining the problem.

A forward error measures distance from the exact output. A backward error is the least size of a permitted perturbation for which the computed output is exact. Backward stability means this size is small on the scale of arithmetic error under stated hypotheses. For $Ax = b$ with nonsingular A, fixed A, and $r = b - A\hat{x}$, the exact perturbed right side is $b - r$. Hence

$$x - \hat{x} = A^{-1} r, \qquad \frac{\|x - \hat{x}\|}{\|x\|} \le \kappa_2(A) \frac{\|r\|}{\|b\|}, \quad \kappa_2(A) = \|A\|\|A^{-1}\| = \sigma_{\max}/\sigma_{\min}.$$

Here $\|b\| \le \|A\|\|x\|$ supplies the last factor. For $A = \mathrm{diag}(1, \varepsilon)$, $b = e_1$ and $\hat{x} = (1,1)^T$, the relative residual is ε and the relative forward error is one. The SVD from Module D explains this amplification by the long axis of the inverse image ellipsoid. In iterative refinement a more accurate residual helps recover the correction. The eigenvalue analogue requires spectral separation, developed next.

Checkpoint. For $a = (1, -1)$, $x = (1, 1+\varepsilon)^T$, explain why the dot-product bound divided by $|a^T x|$ grows like $2\gamma_2/|\varepsilon|$. Which two terms cancel?

17.3 Session 2 — Eigenpair residuals and perturbations

17.3.1 A residual as the price of exactness

For a unit vector x, set $r = Ax - \theta x$. A matrix E makes (θ, x) exact for $A + E$ precisely when $Ex = -r$. Consequently $\|E\| \geq \|r\|$, with equality for $E = -rx^*$. This is an exact, normwise, absolute backward error allowing arbitrary matrix perturbations. For a nonzero unnormalized vector divide the residual norm by $\|x\|$. Normalizing further by $\|A\|$ gives a relative perturbation in A when $\|A\| > 0$.

If $A = A^*$ and $\theta = \rho_A(x)$, then $x^*r = 0$. Now $E = -rx^* - xr^*$ is Hermitian and has the same optimal norm. On the orthonormal pair $x, r/\|r\|$ its matrix has zero diagonal and off-diagonals $-\|r\|$, and it vanishes on their orthogonal complement. The case $r = 0$ takes $E = 0$. This construction preserves self-adjointness. Additional requirements such as a fixed sparsity pattern change the backward-error problem.

17.3.2 Locating an eigenvalue and locating a direction

Let $A = A^* = U\Lambda U^*$, $x = \sum_i c_i u_i$, $\|x\| = 1$. Pythagoras gives the identity

$$\|r\|^2 = \sum_i |\lambda_i - \theta|^2 |c_i|^2.$$

Since $\sum_i |c_i|^2 = 1$, some eigenvalue lies within $\|r\|$ of θ. To identify a target set I, let $P_I = \sum_{i \in I} u_i u_i^*$ and $\delta = \min_{i \notin I} |\lambda_i - \theta| > 0$. Discarding the target terms gives

$$\sin \angle(x, \operatorname{ran} P_I) = \|(I - P_I)x\| \leq \|r\|/\delta.$$

Knowing which eigenvalues are excluded is substantive information. An exact eigenvector of the wrong eigenvalue has zero residual. Eigenvectors inside a multiple eigenspace are interchangeable, so a particular basis there cannot be certified from the equation.

For a simple target λ_j, expansion of the Rayleigh quotient gives $\rho_A(x) - \lambda_j = \sum_{i \neq j} (\lambda_i - \lambda_j)|c_i|^2$. Its absolute value is bounded by $\max_i |\lambda_i - \lambda_j| \sin^2 \angle(x, u_j)$. Thus the value can settle to many digits while its direction still moves appreciably. For $x = \cos t\, q_2 + \sin t\, q_3$, the value is $3 + 6 \sin^2 t$ and the residual norm is $6 |\sin t \cos t|$.

17.3.3 Weyl and subspace separation

For two Hermitian matrices A and $A+E$, their eigenvalues in increasing order satisfy $|\lambda_i(A+E)-\lambda_i(A)| \le \|E\|$. **Proof.** For every unit vector the Rayleigh quotients differ by at most $\|E\|$. Taking the maximum over vectors in an i-dimensional subspace and then the minimum over such subspaces preserves both inequalities. Apply Module D's min–max theorem. This is an absolute bound. Relative sensitivity may be large near zero.

Here is a residual version of the subspace separation argument behind Davis–Kahan. Let $X^*X = I$, $H = X^*AX$, and $R = AX - XH$. Rotate X by a unitary matrix to diagonalize $H = \operatorname{diag}(\theta_j)$ in its new basis. If U_2 spans the exact unwanted eigenvectors with diagonal matrix Λ_2, then $S = U_2^*X$ satisfies

$$\Lambda_2 S - SH = U_2^* R.$$

If every $|\lambda_i(\Lambda_2) - \theta_j| \ge \delta > 0$, division entry by entry and summation of squares gives $\|S\|_F \le \|R\|_F/\delta$. The singular values of S are the sines of the principal angles when the target and trial dimensions agree. This proves a Frobenius sin-theta bound and therefore also bounds the largest sine. Entrywise division alone does not prove the same constant for a spectral-norm right side. For a perturbed invariant subspace of $A+E$, $R = -(I-XX^*)EX$; the denominator must still separate the perturbed Ritz spectrum from the unwanted spectrum of A.

Small-gap example. For $\operatorname{diag}(1, 1+g)$ add a real symmetric off-diagonal ε. The diagonalizing rotation has $|\tan(2t)| = 2|\varepsilon|/g$. Taking $\varepsilon = g/2$ rotates the axes by $\pi/8$ even as $g \to 0$. Appending a third eigenvalue 9 leaves the first two coordinates as an exact invariant plane. A well-determined plane may contain poorly determined individual directions.

17.3.4 Why nonnormality changes the certificate

If $A = V\Lambda V^{-1}$ and μ is an eigenvalue of $A+E$, then $\min_i |\mu - \lambda_i| \le \kappa_2(V)\|E\|$ (Bauer–Fike). For μ outside the spectrum, apply $(\mu I - A)^{-1}$ to Ex for a unit perturbed eigenvector x. The resolvent has norm at most $\kappa_2(V)/\min_i |\mu - \lambda_i|$, giving the claim. A normal matrix admits unitary V. For a simple eigenvalue, differentiating $Ax = \lambda x$ and testing with a left eigenvector y^* gives $\dot\lambda = y^*\dot{A}x/(y^*x)$. This is Module B's dual measurement in action. Nearly orthogonal left and right eigenvectors amplify perturbations.

For the Jordan block $J = \left(\begin{smallmatrix} 0 & 1 \\ 0 & 0 \end{smallmatrix}\right)$, adding ε in position $(2,1)$ produces eigenvalues $\pm\sqrt{\varepsilon}$. Even the order of the displacement changes at a defective eigenvalue. More generally the ε-pseudospectrum consists of the possible eigenvalues under $\|E\| \le \varepsilon$. The rank-one construction identifies it exactly with those θ admitting a unit vector of residual at most ε.

Checkpoint. If a normalized Hermitian Ritz vector has residual 10^{-8} and every unwanted eigenvalue is at least 10^{-3} away from its Ritz value, what is certified? Some eigenvalue is within 10^{-8} and the sine to the identified target space is at most 10^{-5}.

17.4 Session 3 — Vector iterations as spectral transformations

17.4.1 Power iteration and its rate

Given a unit x_0, form $y = Ax_k$, $x_{k+1} = y/\|y\|$ whenever $y \neq 0$. Normalization controls the scale and preserves the line generated by $A^{k+1}x_0$. For Hermitian A, order $|\lambda_1| > |\lambda_2| \geq \cdots$ and write $x_0 = \sum c_i u_i$ with $c_1 \neq 0$. Divide the squared norm of the unwanted components by that of the desired component:

$$\tan^2 \angle(x_k, u_1) = \sum_{i>1} \left|\frac{c_i}{c_1}\right|^2 \left|\frac{\lambda_i}{\lambda_1}\right|^{2k}.$$

Consequently the tangent decreases at least as fast as the bound $(|\lambda_2/\lambda_1|)^k \tan\angle(x_0, u_1)$. Equality holds if all active unwanted components have modulus $|\lambda_2|$. The Rayleigh quotient error is quadratic in the sine by Session 2. A negative dominant eigenvalue reverses sign each iteration. Monitor the line or its residual. A zero starting coefficient stays zero in exact arithmetic. Equal dominant moduli can prevent convergence to a line, as $\operatorname{diag}(1, -1)$ shows.

For A_0, start at $(q_2 + q_3)/\sqrt{2}$. After two multiplications the tangent to q_3 is $1/9$, and the Rayleigh quotient is $(3 + 9 \cdot 81)/82$. Compare the error $6/82$ with the angle $\arctan(1/9)$. In an optimization interpretation, ρ_A on the unit sphere has tangential gradient $2(Ax - \rho_A(x)x)$ in real coordinates. A stationary point is an eigenvector. Power iteration selects by modulus, so on an indefinite matrix it need not maximize the quotient.

17.4.2 Inverse iteration and Rayleigh quotient iteration

Choose a real shift σ outside the spectrum. Solve $(A - \sigma I)y = x_k$ and normalize y. In the eigenbasis the component multiplier is $1/(\lambda_i - \sigma)$. For a unique closest eigenvalue λ_t, nonzero starting overlap, and second-closest distance d_2, the tangent bound contracts per step by $|\lambda_t - \sigma|/d_2$. A fixed shift permits reuse of a factorization. For A_0, $\sigma = 2.8$ gives distances $1.8, 0.2, 6.2$. The target is q_2 and the worst-case factor is $1/9$. Near-singularity is the source of selectivity. It also calls for stable shifted solves and residual checks on the original eigenproblem. A solve residual $s = x_k - (A - \sigma I)\hat{y}$ changes the inverse application by $-(A - \sigma I)^{-1}s$.

Rayleigh quotient iteration (RQI) recomputes $\sigma = \rho_A(x_k)$ each time. Assume A Hermitian, λ_t simple, gap $g > 0$, and x_k close enough that $|\rho_A(x_k) - \lambda_t| < g/2$. Put $t = \angle(x_k, u_t)$ and $D = \max_i |\lambda_i - \lambda_t|$. When the shifted solve exists, the same component calculation gives

$$\tan t_{\text{new}} \leq \frac{|\lambda_t - \rho_A(x_k)|}{\min_{i \neq t} |\lambda_i - \rho_A(x_k)|} \tan t \leq (2D/g) \sin^2 t \, \tan t.$$

This proves local cubic convergence of the direction. Each changing shift can require a new factorization. Check convergence before solving: at an eigenvector the shifted matrix is singular. At an interior eigenvalue cancellation can also give a singular shift away from an eigenvector. For $\operatorname{diag}(-1, 0, 1)$, the equal mixture of the outer directions has quotient zero. A practical safeguard detects a failed solve and selects a perturbed shift. The cubic estimate makes no global claim about which eigenpair a distant start selects.

Checkpoint. Explain why a smaller shift distance helps inverse iteration in exact arithmetic and simultaneously increases the potential amplification of a linear-solve residual.

17.5 Session 4 — Block iteration, principal angles, and extraction

17.5.1 Why a basis needs attention

To find a p-dimensional invariant space, start with $X_0^* X_0 = I_p$, form $Y = AX_k$, and take X_{k+1} as the thin Q in $Y = QR$. Provided Y has full column rank, R is invertible and the range is unchanged by this operation. Independent power iterations tend to align all columns with the strongest direction. A QR step restores a basis with which the weaker directions can still be represented. A unitary change of basis $X \mapsto XW$ leaves the orthogonal projector XX^* unchanged.

Write $A = [U_1, U_2] \operatorname{diag}(\Lambda_1, \Lambda_2)[U_1, U_2]^*$ for Hermitian A, with p target columns in U_1. Put $C = U_1^* X_0$, $D = U_2^* X_0$. Assume C invertible and every target eigenvalue nonzero. In eigencoordinates a basis of the iterated space can be written

$$\begin{pmatrix} \Lambda_1^k C \\ \Lambda_2^k D \end{pmatrix} (\Lambda_1^k C)^{-1} = \begin{pmatrix} I \\ Z_k \end{pmatrix}, \qquad Z_k = \Lambda_2^k D C^{-1} \Lambda_1^{-k}.$$

This is the graph of a map from the target to its orthogonal complement. A vector on it has tangent $\|Z_k v\|/\|v\|$ to the target. Maximization therefore gives $\tan\theta_{\max} = \|Z_k\|$. In particular

$$\tan\theta_{\max,k} \le \left(\frac{\max_{\text{out}} |\lambda|}{\min_{\text{in}} |\lambda|} \right)^k \|DC^{-1}\|.$$

Separation is needed across the boundary of the target. Multiplicities wholly inside it cause no difficulty. Taking a block larger than the requested number can move that boundary beyond a cluster; then interpret the bound for the enlarged space and extract the requested Ritz pairs.

17.5.2 Rayleigh–Ritz and finite precision

With orthonormal X, compress to $H = X^* A X$. Solve $Hw_j = \theta_j w_j$ with orthonormal w_j, form $v_j = Xw_j$. Then $r_j = Av_j - \theta_j v_j$ satisfies $X^* r_j = 0$ by substitution. Conversely this orthogonality for $v = Xw$ is precisely the projected eigenproblem. Thus the defect annihilates

every trial-space test direction (Module B), or equivalently has zero orthogonal projection there (Module C). For nonorthonormal full-rank Y the corresponding problem is $Y^*AYw = \theta Y^*Yw$. An ill-conditioned Gram matrix squares the basis condition number. Orthogonalize first.

For A_0, use $X = [q_2, (q_1 + q_3)/\sqrt{2}]$. The compression is $\mathrm{diag}(3, 5)$, yet 5 is not an eigenvalue of A_0. Its residual is $4(q_3 - q_1)/\sqrt{2}$, perpendicular to both columns. One exact Ritz pair can coexist with a poor second pair. Sorted Ritz values alone do not verify a block.

Householder QR uses unitary reflections $I - 2vv^*/(v^*v)$ to zero entries below a pivot without changing exact lengths. In floating point it is a standard stable choice for a dense block. Gram–Schmidt subtracts projections onto earlier columns. If this leaves a tiny vector, errors made in subtracting large components are magnified upon normalization. A second orthogonalization pass often repairs the defect. If the input block has lost numerical rank, reorthogonalization cannot recover information already lost. Monitor $\|I - X^*X\|$, the size and conditioning of the small triangular factor, and the direct residual. Computing small angles through $\arccos(\sigma_{\min}(U_1^*X))$ loses digits near one. Prefer $\|(I - U_1U_1^*)X\|$ or $\|U_2^*X\|$ for the largest sine.

Checkpoint. For trial columns $q_3, (q_2 + q_1)/\sqrt{2}$ and target $\mathrm{span}(q_3, q_2)$, calculate the tangent after one power step. It is $1/3$, even though the first column is already exact.

17.6 Session 5 — Krylov spaces and projected eigenproblems

17.6.1 Keeping the history

For $v \neq 0$, $\mathcal{K}_m(A, v) = \mathrm{span}(v, Av, \dots, A^{m-1}v) = \{p(A)v : \deg p < m\}$. Power iteration keeps one polynomial. Krylov projection searches the span of all these candidates. If $A^m v$ lies in the earlier span then applying A to any spanning vector stays inside and the space is invariant. Algebraically, $p \mapsto p(A)v$ maps the polynomial ring onto the cyclic space generated by v. Its kernel is the ideal generated by the least-degree monic polynomial annihilating v. Polynomial division shows that its degree equals the eventual Krylov dimension. This is the first isomorphism theorem of Module A in the language of Module F2.

17.6.2 Arnoldi, Lanczos, and the residual shortcut

Normalize $v_1 = v/\|v\|$. Given $V_j = [v_1, \dots, v_j]$, project Av_j onto the basis, subtract that projection and normalize the remainder:

$$h_{ij} = v_i^* A v_j, \quad w = Av_j - \sum_{i=1}^{j} h_{ij} v_i, \quad h_{j+1,j} = \|w\|, \quad v_{j+1} = w/h_{j+1,j}.$$

A zero norm gives an invariant space in exact arithmetic. Stacking the identities gives

$$AV_m = V_m H_m + h_{m+1,m} v_{m+1} e_m^*, \qquad H_m = V_m^* A V_m.$$

H_m is upper Hessenberg because Av_j lies in the first $j+1$ basis vectors. For a unit eigenvector $H_m y = \theta y$, its lifted residual has norm $|h_{m+1,m}|\,|e_m^* y|$. Only the last coefficient and the last subdiagonal enter.

For Hermitian A, Hermitian symmetry of H_m forces all entries above its first superdiagonal to vanish as well. Writing its real diagonal as α_j and nonnegative subdiagonal as β_j gives the Lanczos recurrence

$$\beta_j v_{j+1} = Av_j - \alpha_j v_j - \beta_{j-1} v_{j-1}, \quad \beta_0 = 0.$$

In exact arithmetic earlier projections vanish by the Hessenberg identity and symmetry. In finite precision tiny components along converged Ritz directions can grow back, causing duplicate approximate eigenvalues. Full or selective reorthogonalization trades additional work and communication for control of this effect. The direct residual checks both the recurrence shortcut and the projected solve.

Two steps by hand. Start A_0 at e_1. Then $Ae_1 = 2e_1 + e_2$, so $\alpha_1 = 2$, $\beta_1 = 1$, $v_2 = e_2$. Next $Ae_2 = e_1 + 2e_2$, yielding $\alpha_2 = 2$, $\beta_2 = 0$. The projected problem has eigenvalues $1, 3$. The missing 9 cannot be recovered from this start because every polynomial in A preserves this coordinate plane.

A sparse Arnoldi step needs one matrix application and projections against earlier vectors. For m steps, storing the full basis costs order nm and orthogonalization costs order nm^2. Short-recurrence Lanczos can generate coefficients with a few vectors, but recovering Ritz vectors later requires stored or regenerated basis information. Restarting bounds storage by retaining selected approximate spectral information before building a new space. Discarding too much can erase progress near a cluster. Block iteration has fixed width and can exploit matrix–block products. A degree-d filtered block also belongs to a block Krylov space, but evaluates a prescribed polynomial without storing every power.

Checkpoint. What evidence distinguishes exact invariance from numerical breakdown? A tiny computed β_j is a clue. Directly test $\|AV_j - V_j H_j\|$ and $\|V_j^* V_j - I\|$ at the accuracy required by the application.

17.7 Session 6 — Designing and evaluating a Chebyshev filter

17.7.1 The design problem

Suppose the unwanted eigenvalues of a Hermitian operator lie in $[a, b]$ and all desired eigenvalues lie above b. Let $c = (a+b)/2$, $h = (b-a)/2 > 0$, and $B = (A - cI)/h$. The unwanted interval maps to $[-1, 1]$. A filter should be bounded there and large at every target. The same map works for a target below a using the parity of the polynomial. An interior band generally needs a different filter design: two exterior tails cannot both be mapped to this single unwanted interval by one affine map without also including the interior target.

Define $T_j(\cos t) = \cos(jt)$. The addition formula gives $T_0 = 1$, $T_1 = z$, $T_{j+1} = 2zT_j - T_{j-1}$. Thus $|T_j| \le 1$ on $[-1, 1]$, and $T_j(-z) = (-1)^j T_j(z)$. For $z > 1$ substitute $z = \cosh s$ into the same recurrence to obtain $T_j(z) = \cosh(js)$, $s = \operatorname{arcosh} z$. Growth is exponential for fixed exterior z. It slows as the target approaches the cutoff. Taking higher powers z^j also amplifies exterior components, but misses the extremal property below.

Normalized minimax theorem. Fix $z_* > 1$, $d \ge 1$. Among real polynomials p of degree at most d satisfying $p(z_*) = 1$, the least possible maximum of $|p|$ on $[-1, 1]$ is $1/T_d(z_*)$. The candidate is $p = T_d/T_d(z_*)$. To prove optimality, suppose a competitor has strictly smaller maximum. The difference $T_d - T_d(z_*)p$ alternates in sign at $\cos(j\pi/d)$ for $j = 0, \dots, d$. It has d distinct interior zeros and another zero at z_*. A nonzero polynomial of degree at most d cannot have $d + 1$ roots. This optimality concerns uniform suppression of an interval with a specified exterior normalization. It does not optimize wall time or rounding errors.

17.7.2 Subspace convergence and degree choice

The spectral theorem gives $T_d(B)u_i = T_d((\lambda_i - c)/h)u_i$. Apply the graph proof from Session 4 with each eigenvalue replaced by its filter value. For full initial target overlap,

$$\tan \theta_{\max,k} \le \rho_d^k \tan \theta_{\max,0}, \qquad \rho_d = \frac{\max_{\text{out}} |T_d((\lambda - c)/h)|}{\min_{\text{in}} |T_d((\lambda - c)/h)|}.$$

If the nearest target maps to $z_* > 1$ and all unwanted eigenvalues lie in the design interval, $\rho_d \le \operatorname{sech}(d \operatorname{arcosh} z_*)$. For a desired one-block contraction $\tau < 1$ choose $d \ge \operatorname{arcosh}(1/\tau)/\operatorname{arcosh} z_*$. Round upward. This counts polynomial degree, hence block products, per application. It is a sufficient interval estimate. The finite set of actual unwanted eigenvalues can give a smaller ratio.

Running matrix. Seek q_3 of A_0. For $[1, 3]$, $B = A_0 - 2I$ has eigenvalues $-1, 1, 7$. A degree-two filter has values $1, 1, 97$. Starting at $(q_2 + q_3)/\sqrt{2}$, two ordinary power steps leave tangent $1/9$. One degree-two filter also takes two products and leaves tangent $1/97$. Both calculations start with the same angle and cost the same number of applications. Differences in QR, memory access and coefficient operations enter an actual timing comparison.

17.7.3 Evaluating a polynomial without forming powers

Use $Y_0 = X$, $Y_1 = BX$, $Y_{j+1} = 2BY_j - Y_{j-1}$. Induction gives $Y_j = T_j(B)X$. Only successive blocks are needed. The product $BY = (AY - cY)/h$ can be implemented as an operator action. Forming B once can change cancellation, rounding and memory requirements. An exact-arithmetic scalar normalization common to the entire block leaves the resulting space unchanged.

To control range, choose an exterior reference $z_s > 1$, typically near the largest mapped target, and set $W_j = Y_j/T_j(z_s)$. Define $s_j = T_{j-1}(z_s)/T_j(z_s)$. Then

$$s_1 = 1/z_s, \quad s_{j+1} = 1/(2z_s - s_j), \quad W_0 = X, \quad W_1 = s_1 BX, \quad W_{j+1} = 2s_{j+1}BW_j - s_j s_{j+1} W_{j-1}.$$

This follows by dividing the recurrence by $T_{j+1}(z_s)$. It avoids computing the huge normalization factor explicitly. Common scaling controls overflow but leaves the ratio between the strongest and weakest wanted components intact. Large degree can make the filtered block nearly rank deficient before QR. Compare singular values of the small R factor, shorten the polynomial or orthogonalize between complete shorter filters. Inserting arbitrary independent normalizations into an unmodified three-term recurrence changes the polynomial.

17.7.4 What a local error becomes

For the unscaled recurrence define a local additive defect F_j in each computed step, incorporating whatever product and update errors occurred:

$$\widehat{Y}_0 = X, \quad \widehat{Y}_1 = BX + F_1, \quad \widehat{Y}_j = 2B\widehat{Y}_{j-1} - \widehat{Y}_{j-2} + F_j, \quad j \geq 2.$$

Let $U_0(z) = 1$, $U_1(z) = 2z$, $U_{k+1} = 2zU_k - U_{k-1}$ be the Chebyshev polynomials of the second kind (here U_k is a polynomial). Subtract the exact recurrence. Induction, using $U_0 = 1$ for the newest defect, gives the exact error representation

$$\widehat{Y}_d - Y_d = \sum_{j=1}^{d} U_{d-j}(B) F_j.$$

This identity assumes a fixed exact B and exact initial block. Errors in storing or applying B can be included in F_j. It does not presume independent random signs. Projection onto any eigendirection reduces the propagation to scalar weights. For $z = \cos t$, $U_k(z) = \sin((k+1)t)/\sin t$, bounded by $k+1$ including the limiting endpoints (expand the sine quotient as a sum of unit complex exponentials). Consequently for the unwanted spectral projector P_2,

$$\|P_2(\widehat{Y}_d - Y_d)\| \leq \sum_{j=1}^{d} (d - j + 1)\|P_2 F_j\|.$$

For $z = \cosh s > 1$, the corresponding weight is $\sinh((k+1)s)/\sinh s$. An early error in a desired component can grow very large in absolute norm. Whether it rotates the range is a separate question. For instance, at $d = 3$ a defect in the first product contributes $(4B^2 - I)F_1$, while a defect in the last update contributes F_3 directly.

17.7.5 Turning application error into angle error

In the target/complement coordinates write the input block as $\binom{C}{D}$, and write one computed filter output as $\binom{P_1C+G_1}{P_2'D+G_2}$, where P_1, P_2' are diagonal target and unwanted filter values, and G_i are the total filter errors in those coordinates. Put $\alpha = \min|\operatorname{diag} P_1|$, $\beta = \max|\operatorname{diag} P_2'|$. Right-multiplying by C^{-1} shows that the graph of the output has tangent bounded by

$$t_{\text{new}} \le \frac{\beta t + \|G_2C^{-1}\|}{\alpha - \|G_1C^{-1}\|}, \qquad t = \|DC^{-1}\|.$$

Indeed the smallest singular value of the top block after multiplication is at least the denominator, which must be positive. The unwanted numerator is bounded by the triangle inequality. This inequality explains why a global absolute error bound alone is a poor predictor of angle accuracy. Unwanted leakage adds a new direction and poor target overlap can amplify it.

If the resulting step bounds yield **uniform** constants $t_{k+1} \le qt_k + \eta$ with $0 \le q < 1$, summing a geometric progression gives $t_k \le q^k t_0 + \eta(1-q^k)/(1-q)$. The second term predicts a possible stagnation scale. It is an upper envelope, never a proof that every run stagnates there. Errors can cancel or decrease as the residual shrinks. A verified residual and a gap estimate are needed to diagnose a real run.

17.7.6 Bounds that miss part of the spectrum

If an unwanted eigenvalue falls just outside the designed interval, its multiplier becomes $\cosh(d \operatorname{arcosh}|z|)$ in magnitude. Recompute the *actual* ratio ρ_d: convergence can slow and an unwanted direction can even dominate. Conversely an overly wide interval can bring the nearest target closer to 1, requiring higher degree. Obtain bounds from analysis of the operator, enclosing intervals such as Gershgorin bounds, or preliminary iterations with explicit margins. An interior Ritz estimate of an extreme eigenvalue alone is generally not an outer enclosure, as min–max already shows.

Checkpoint. With $[a, b] = [1, 5]$ and target 8, $z_* = 2.5$. Show $T_3(2.5) = 55$. If an unwanted 6 was missed, its mapped value is 1.5, giving $T_3(1.5) = 9$. The ratio estimate changes from $1/55$ to $9/55$ when this is the only missed eigenvalue.

17.8 Session 7 — Mixed precision and hardware-aware computation

17.8.1 An error budget by operation

Operation	Where accuracy enters	Measurement or design choice
Store and apply A	Quantization perturbs the operator before accumulation begins	Retain a reference action for verification
Evaluate the filter	Every recurrence defect is propagated by a different polynomial	Select degree, scaling, product and update precisions separately
QR	Small singular directions of the input may be obscured	Check orthogonality and inspect the triangular factor
Rayleigh–Ritz	Projection errors perturb a small Hermitian problem	Form and solve the compression accurately, check its symmetry
Stop and report	Cancellation in $AX - X\Theta$ may hide the true residual	Reapply the original operator in a specified verification precision

If storage gives $\widetilde{A} = A + E_s$ and arithmetic returns $\widetilde{A}X + F$, the product error relative to the original data is $E_sX + F$. Higher-precision accumulation can reduce F but does not remove E_sX. If X itself is quantized, expand $(A + E_s)(X + E_x)$ to see the additional terms $AE_x + E_sE_x$. In a distributed computation rounding communicated entries contributes to E_x, possibly only on boundary data. A uniform normwise model may overestimate that effect. Floating-point format, product precision, accumulation precision and communication format must be specified separately.

Verification bound. If the computed verification residual is $\hat{r} = r + e$ with $\|e\| \le \epsilon_v$, then $\|r\| \le \|\hat{r}\| + \epsilon_v$. A matvec bound from Session 1 plus bounds for the scale and subtraction can supply ϵ_v under a known arithmetic model. A double-precision direct residual is a practical independent diagnostic when the filter ran in single. Calling it a rigorous enclosure additionally requires controlling its own evaluation and norm rounding. Increasing the verification precision can reveal an existing error without changing the iterate.

A scalar identity behind residual-based methods. For a polynomial p and approximate eigenpair (x, θ), the factor theorem gives $p(z) - p(\theta) = (z - \theta)s_\theta(z)$. Therefore

$$p(A)x = p(\theta)x + s_\theta(A)r, \qquad r = Ax - \theta x.$$

For $p(z) = 2z^2 - 1$, $s_\theta(z) = 2(z + \theta)$. Near convergence the argument of the operator in the correction is small. If a product error is proportional to the norm of its input, the error in applying an operator to a small residual may shrink with convergence. An accurately evaluated residual and a controlled correction recurrence are essential. When the input residual is already corrupted at the scale of $u\|A\|$, the argument loses force. This identity provides a way into the seminar paper's residual-based recurrence. The ordinary recurrence in the lab is its comparison

case. For the actual algorithm and its numerical evidence see Kodali, Ramakrishnan and Motamarri, *Residual-based Chebyshev filtered subspace iteration*.

17.8.2 Cost of reaching an accuracy target

A dense matrix–block product with an $n \times p$ real block costs approximately $2n^2p$ scalar floating-point operations. A sparse product costs approximately $2\,\mathrm{nnz}(A)p$. A thin dense QR costs order np^2 and the compressed eigensolve costs order p^3. A degree-d filter takes d matrix–block products before extraction. These counts give no sustained bandwidth, occupancy or collective latency. For an illustrative machine model with peak rate F and bandwidth B_w, moving S bytes and doing W operations requires at least $\max(W/F, S/B_w)$ time for the modeled resources. Sparse indices, cache reuse, conversions, synchronization and contention affect actual time. A block may reuse matrix entries across columns. Distributed inner products require reductions, whereas stencil products often require neighbor exchanges.

For a fair comparison specify a target cluster, a common residual tolerance on the original operator, an orthogonality threshold, starting block and stopping rule. Time setup, transfers, filtering, QR, extraction and verification; include all repetitions needed to reach tolerance. Record failures to meet tolerance. A low-precision run with faster individual products can take longer to deliver an acceptable subspace, or never meet the criterion. Timing the dense teaching example says little about a sparse device implementation.

Checkpoint. A single-precision filter and double-precision QR produce $\|I - X^*X\| \approx 10^{-15}$ but a verified residual of 10^{-6}. These diagnostics can coexist: one measures internal geometry of the basis, the other tests its relation to A.

17.9 Session 8 — A controlled experiment from prediction to verification

17.9.1 Construct a problem with known answers

Take $n = 32$ and the orthonormal discrete cosine basis U. Its zeroth column is constant $1/\sqrt{n}$, and for row $i = 0, \dots, n-1$ and column $j = 1, \dots, n-1$ its entry is $\sqrt{2/n}\cos(\pi(i+1/2)j/n)$. Set $A = U\Lambda U^T$ with 29 equally spaced unwanted eigenvalues from 1 to 5 and targets $8, 9, 10$. The dense rotation is deliberate: rounding a diagonal example can preserve exact zeros and hide leakage between directions. The exact mathematical construction and its stored floating-point realization differ slightly.

For each target eigenvector take its equal mixture with a different unwanted eigenvector. Pair 8 with 5, 9 with 1, and 10 with the second-smallest eigenvalue. These three vectors are orthonormal. Each angle is initially $\pi/4$. For the pair determining the worst angle the power multiplier ratio is $5/8$ and the degree-three Chebyshev ratio is $1/55$. All other ratios are smaller.

The graph estimate is attained, so the predicted largest tangents are **exactly** $(5/8)^k$ and 55^{-k} in mathematical arithmetic. Convert a tangent t to a sine by $t/\sqrt{1+t^2}$ before comparing with the lab's sine diagnostic.

Each outer step applies the chosen operator, orthogonalizes, solves the projected eigenproblem, and checks the residuals in the original coordinates. The lab retains the QR basis for the next step; rotating it to the Ritz basis gives the same next range in exact arithmetic. It records the largest column residual, largest sine, maximum sorted Ritz-value error and orthogonality defect. Only the synthetic example supplies the exact target basis and exact target values. In applications, identification and separation must come from other information.

17.9.2 A measured history

The numbers below were obtained by independently executing the real-valued case study in NumPy with the same cosine construction, polynomial and precision split used in the MATLAB script. Both variants use binary64 QR, extraction and verification. The binary32 variant stores the mapped matrix in single and performs the filter recurrence in single. Last digits and stagnation levels depend on implementation.

Outer steps	Predicted tangent	Largest sine, double filter	Largest sine, single filter	Max residual, double filter	Max residual, single filter
1	$1.82\,10^{-2}$	$1.82\,10^{-2}$	$1.82\,10^{-2}$	$8.08\,10^{-2}$	$8.08\,10^{-2}$
4	$1.09\,10^{-7}$	$1.09\,10^{-7}$	$1.23\,10^{-7}$	$3.28\,10^{-7}$	$4.36\,10^{-7}$
8	$1.19\,10^{-14}$	$1.58\,10^{-14}$	$5.56\,10^{-8}$	$3.61\,10^{-14}$	$3.75\,10^{-7}$
10	$3.95\,10^{-18}$	$9.38\,10^{-15}$	$6.90\,10^{-8}$	$4.61\,10^{-15}$	$4.02\,10^{-7}$

At step one the largest angle belongs to the $8, 5$ pair, while the largest residual can belong to a different pair: a residual weights the angle by the local eigenvalue separation. At step eight the double run meets an absolute residual tolerance of 10^{-12}; the displayed single run does not. The experiment combines storage and arithmetic errors. To isolate storage, round B to binary32 once and convert it back to binary64 for all products. To probe communication errors separately, an experiment needs an explicit communication pattern and quantization model.

17.9.3 Transfer to a variational PDE problem

In §6.11, $Kx = \lambda Mx$ came from testing $-u'' = \lambda u$ with finite-element basis functions. Assume $K = K^*$, $M = M^* > 0$ and $M = LL^*$. Set $y = L^*x$ and $C = L^{-1}KL^{-*}$. Then $Cy = \lambda y$ is Hermitian, $\|y\|^2 = x^*Mx$, and

$$Cy - \theta y = L^{-1}(Kx - \theta Mx).$$

For an M-normalized approximate eigenvector, $\|Kx - \theta Mx\|_{M^{-1}}$ is precisely the Euclidean residual norm of the transformed problem. Solve triangular systems to apply the transformations. An M-orthonormal trial basis X gives the small matrix X^*KX and the Ritz residual satisfies $X^*(KX - MXH) = 0$. These are the Galerkin test equations again. A verified algebraic residual certifies the *discrete* eigenproblem. Errors due to the mesh and the choice of trial functions require an additional approximation argument.

Checkpoint. Why can a change of finite-element basis alter the Euclidean norm of a residual without changing the mathematical approximation? The residual represents a functional; its coordinates and the mass Gram matrix transform together. The dual norm accounts for both changes.

17.10 Exercises and worked solutions

Try each problem before reading its solution. Problems 4, 6, 8 and 11 form a short proof-reconstruction route into the seminar.

17.10.1 1. Which data are allowed to move?

For $A = \operatorname{diag}(1, 10^{-6})$, $b = e_1$ and $\hat{x} = (1, 1)^T$, find the minimum absolute normwise perturbations when (i) only b may move, (ii) only A may move. Compare with the relative forward error.

Solution. The residual is $(0, -10^{-6})^T$. The new right side is forced to be $A\hat{x}$, so its distance is 10^{-6}. For a matrix change $E\hat{x} = r$, the lower bound is $\|r\|/\|\hat{x}\| = 10^{-6}/\sqrt{2}$. It is attained by $E = r\hat{x}^*/\|\hat{x}\|^2$. The exact $x = e_1$ gives relative forward error 1. Backward error has meaning only after specifying the allowed data and norm.

17.10.2 2. A Rayleigh quotient can conceal an inaccurate vector

For A_0 take $x = (q_2 + 0.01q_3)/\sqrt{1.0001}$. Compute the quotient error relative to 3 in absolute units, the residual norm and the sine to q_2.

Solution. With $t = 0.01$, $\rho = (3+9t^2)/(1+t^2)$, so the absolute error is $6t^2/(1+t^2) \approx 0.000600$. The two residual coordinates in the eigenbasis are $(3-\rho)/\sqrt{1+t^2}$ and $(9-\rho)t/\sqrt{1+t^2}$. Squaring and adding gives $6t/(1+t^2) \approx 0.059994$ for the norm. The sine is $t/\sqrt{1+t^2} \approx 0.0099995$. To call the quotient error relative, one would divide it by 3.

17.10.3 3. A backward error with symmetry

For $A = \mathrm{diag}(1, 3)$, $x = (1, 1)^T/\sqrt{2}$, find an optimal Hermitian perturbation for its Rayleigh quotient.

Solution. $\theta = 2$, $r = (-1, 1)^T/\sqrt{2}$, $\|r\| = 1$. Expanding $-rx^* - xr^*$ gives $\mathrm{diag}(1, -1)$. The corrected matrix is $2I$, so $(2, x)$ is exact. Any admissible E has $\|E\| \geq \|Ex\| = 1$, proving optimality. The perturbation merges two simple eigenvalues; the backward-error condition imposes no constraint on multiplicities.

17.10.4 4. Fixed versus adaptive inverse iteration

In a two-dimensional Hermitian eigenspace mixture take $x = (u_1 + tu_2)/\sqrt{1 + t^2}$, real $t \neq 0$, and distinct eigenvalues. Derive exactly the new signed coefficient ratio under RQI.

Solution. $\rho = (\lambda_1 + t^2\lambda_2)/(1 + t^2)$. The new ratio equals $t(\lambda_1 - \rho)/(\lambda_2 - \rho) = -t^3$. Thus the tangent cubes even with the sign change. A fixed shift produces $t(\lambda_1 - \sigma)/(\lambda_2 - \sigma)$ instead. Starting at $t = 1$ gives a two-cycle in the line for RQI, illustrating the locality of the convergence statement.

17.10.5 5. A block that cannot recover its target

For A_0 target $\mathrm{span}(q_2, q_3)$ and start with $[q_1, q_3]$. Predict the residuals and the sine of the largest target angle after any number of power steps.

Solution. Multiplication and QR preserve the starting invariant plane. Both Ritz residuals are zero, whereas q_1 is orthogonal to the target plane and the largest sine is one. In target coordinates the overlap matrix has rank one. The invertibility hypothesis in the graph argument fails. This is why stopping on residuals requires an eigenvalue selection or cluster identification rule as well.

17.10.6 6. From an approximate Arnoldi relation to a trustworthy residual

Suppose $AV = VH + hve_m^* + F$, $\|v\| = 1$, and $Hy = \theta y + s$ with $\|y\| = 1$. Bound the lifted residual and explain the role of $\|Vy\|$.

Solution. Substitution gives $(A-\theta I)Vy = hve_m^*y+Vs+Fy$. By the reverse triangle inequality its norm differs from $|h|\,|e_m^*y|$ by at most $\|V\|\|s\|+\|F\|$. To report the residual for a unit vector divide by $\|Vy\|$. If $\|V^*V - I\| \leq \epsilon < 1$ then $\sqrt{1-\epsilon} \leq \|Vy\| \leq \sqrt{1+\epsilon}$. A shortcut reporting zero can coexist with a nonzero direct residual when the recurrence has a defect.

17.10.7 7. Designing a degree

For an unwanted interval $[1, 5]$, a nearest target at 8, and a desired one-step angle contraction 10^{-3}, find the smallest degree certified by the interval bound.

Solution. $z_* = (8 - 3)/2 = 2.5$. Successive values $T_j(2.5)$ are $1, 2.5, 11.5, 55, 263.5, 1262.5$ by the recurrence. Since the requirement is $T_d(2.5) \geq 1000$, the first sufficient degree is five. Degrees lower than five might meet the goal on a particular starting block if the most weakly amplified target component carries no error, but the interval certificate is uniform.

17.10.8 8. Endpoint accumulation

Consider the scalar recurrence with $B = 1$, exact initial value, and the same additive defect f at every computed step starting with step one. Find the exact error after d steps. Compare it with a single defect only at step one.

Solution. $U_k(1) = k + 1$ by induction. Summation gives $f \sum_{j=1}^{d} (d - j + 1) = fd(d + 1)/2$. A single first defect gives df. The polynomial endpoint satisfies $|T_d(1)| = 1$, yet perturbations in recurrence evaluation can accumulate quadratically with degree. Using a uniform bound $\|F_j\| \leq |f|$ attains equality in this example.

17.10.9 9. Verification with a mass matrix

Let $M = \mathrm{diag}(4, 1)$, $K = \mathrm{diag}(4, 3)$, $x = (1/2, 1)^T/\sqrt{2}$. Determine the generalized Rayleigh quotient and the dual residual norm.

Solution. $x^*Mx = 1$, $x^*Kx = 2$. Thus $r = Kx - 2Mx = (-2, 1)^T/\sqrt{2}$. The Euclidean norm is $\sqrt{5/2}$, whereas $r^*M^{-1}r = (1 + 1)/2 = 1$. Under $L = \mathrm{diag}(2, 1)$ the transformed matrix is $\mathrm{diag}(1, 3)$, $y = (1, 1)^T/\sqrt{2}$, and its Euclidean residual norm is exactly one.

17.10.10 10. Distinguishing a measured floor from a theorem

Suppose $t_{k+1} \leq 0.1t_k + 10^{-7}$. Can one prove that t_k can never fall below 10^{-7}? For $t_0 = 1$ bound t_8.

Solution. The zero sequence after step zero obeys the inequality, so it supplies no lower bound. Repeated substitution gives $t_8 \leq 10^{-8} + 10^{-7}(1 - 10^{-8})/0.9 \approx 1.21\,10^{-7}$. A measured history that levels off there is consistent with this model. A smaller measured value is equally consistent.

17.10.11 11. A filter written in terms of its residual

With $p(z) = 4z^3 - 3z$, derive a correction formula involving $r = Ax - \theta x$. Which products would you evaluate accurately if the requested tolerance is much smaller than the filter's working unit roundoff?

Solution. Polynomial division gives $s_\theta(z) = 4(z^2 + \theta z + \theta^2) - 3$, hence $p(A)x = p(\theta)x + [4(A^2 + \theta A + \theta^2 I) - 3I]r$. Compute the initial residual using the original operator in verification precision and accumulate the correction accurately. Products on the correction may tolerate lower precision since their absolute errors can scale with $\|r\|$. This conclusion requires bounds on the correction operator, conversion errors and dynamic range. A small $\|r\|$ does not authorize treating its *relative* error as small automatically.

17.10.12 12. Design an equal-work comparison

For the case study, compare one degree-three application with three ordinary power steps from the same start. Which conclusion follows before any timing experiment?

Solution. Both use three matrix–block products. The respective largest tangents are $1/55 \approx 0.01818$ and $(5/8)^3 = 125/512 \approx 0.24414$. The filter gives the smaller angle for this spectrum, block and work metric. Three ordinary outer steps may perform more QR and extraction operations. Timing must specify their frequency, data movement, precision, matrix layout and verification tolerance. Neither theoretical ratio determines the faster implementation on a given machine.

17.11 Computational laboratory — MATLAB

Copy the complete block into a script named **module_e_lab.m** and run it. It uses base MATLAB with local functions at the end of a script. It produces two convergence plots, histories for the three block variants and a direct check of the Arnoldi residual formula. All reported residuals are absolute norms for unit vectors. The fixed iteration count deliberately continues past practical stopping points to expose finite-precision behavior.

```
%% Experiment 1: an exact two-mode prediction from Review II
A0 = [2 1 0; 1 2 0; 0 0 9];
q2 = [1;1;0]/sqrt(2);
q3 = [0;0;1];
x = (q2+q3)/sqrt(2);
fprintf('Power iteration: k, tangent, predicted tangent, residual\n');
for k = 0:8
    theta = x'*(A0*x);
```

```
    tangent = abs(q2'*x)/abs(q3'*x);
    fprintf('%2d  %.4e  %.4e  %.4e\n',k,tangent,3^(-k), ...
            norm(A0*x-theta*x));
    x = A0*x;
    x = x/norm(x);
end

%% Experiment 2: a known cluster and explicit orthogonal complement
n = 32;
row = (0:n-1)';
freq = 0:n-1;
U = sqrt(2/n)*cos((pi/n)*(row+0.5)*freq);
U(:,1) = U(:,1)/sqrt(2);
lambda = [linspace(1,5,n-3),8,9,10]';
A = U*diag(lambda)*U';
A = (A+A')/2;
target = U(:,end-2:end);
outside = U(:,1:end-3);
X0 = (target+U(:,[n-3,1,2]))/sqrt(2);
B = (A-3*eye(n))/2;
steps = 10;
% Each row of a history stores [max residual, largest sine,
% max Ritz-value error, orthogonality defect].
powerHist = blockHistory(A,A,X0,outside,lambda(end-2:end),steps,1);
doubleHist = blockHistory(A,B,X0,outside,lambda(end-2:end),steps,3);
singleHist = blockHistory(A,single(B),X0,outside,lambda(end-2:end),steps,3);

k = (0:steps)';
tPower = (5/8).^k;
tCheb = (1/55).^k;
disp(table(k,doubleHist(:,1),singleHist(:,1),doubleHist(:,2), ...
    singleHist(:,2),doubleHist(:,3),singleHist(:,3), ...
    'VariableNames',{'k','res64','res32','sin64','sin32','val64','val32'}));
fprintf('Maximum orthogonality defects (power, double, mixed):\n');
disp(max([powerHist(:,4),doubleHist(:,4),singleHist(:,4)],[],1));
figure;
subplot(1,2,1);
semilogy(k,powerHist(:,2),'o-',k,doubleHist(:,2),'s-', ...
    k,singleHist(:,2),'^-',k,tPower./hypot(1,tPower),'k--', ...
    k,tCheb./hypot(1,tCheb),'k:');
xlabel('Outer steps'); ylabel('Largest sine'); grid on;
legend('Power','Cheb double','Cheb mixed','Power theory','Cheb theory');
```

```
subplot(1,2,2);
% Use product count to expose differences in cost per outer step.
semilogy(k,powerHist(:,1),'o-',3*k,doubleHist(:,1),'s-', ...
    3*k,singleHist(:,1),'^-');
xlabel('Filter or power products (extraction excluded)');
ylabel('Maximum verified Ritz residual'); grid on;
legend('Power','Cheb double','Cheb mixed');

%% Experiment 3: Arnoldi applied to a symmetric problem
% Two passes of modified Gram-Schmidt control loss of orthogonality.
% Hermitian symmetry should make the Hessenberg projection tridiagonal.
start = U*ones(n,1);
[V,H,tail] = arnoldiProjection(A,start,12);
[S,D] = eig(H);
ritz = diag(D);
[~,j] = max(real(ritz));
y = S(:,j)/norm(S(:,j));
v = V*y;
shortcut = tail*abs(y(end));
direct = norm(A*v-ritz(j)*v);
fprintf('Arnoldi shortcut %.6e, direct %.6e, basis defect %.3e\n', ...
    shortcut,direct,norm(V'*V-eye(size(H))));
fprintf('Entries above first superdiagonal, Frobenius norm %.3e\n', ...
    norm(triu(H,2),'fro'));

function hist = blockHistory(A,op,X,outside,wanted,steps,degree)
    [X,~] = qr(X,0);
    p = size(X,2);
    hist = zeros(steps+1,4);
    for k = 0:steps
        H = X'*(A*X);
        [W,D] = eig((H+H')/2);
        [theta,order] = sort(diag(D));
        Z = X*W(:,order);
        residual = A*Z-Z*diag(theta);
        hist(k+1,:) = [max(vecnorm(residual)),norm(outside'*Z), ...
                       max(abs(theta-wanted)),norm(Z'*Z-eye(p))];
        if k == steps
            break
        end
        previous = cast(X,'like',op);
        current = op*previous;
```

```
        for j = 2:degree
            following = 2*(op*current)-previous;
            previous = current;
            current = following;
        end
        [X,~] = qr(double(current),0);
    end
end

function [V,H,tail] = arnoldiProjection(A,start,m)
    n = size(A,1);
    V = zeros(n,m+1);
    H = zeros(m+1,m);
    V(:,1) = start/norm(start);
    % A small tail signals possible numerical invariance. Keep its actual
    % norm in the reported estimate, even when no new vector is formed.
    tol = 10*eps*norm(A,'fro');
    for j = 1:m
        w = A*V(:,j);
        for pass = 1:2
            for i = 1:j
                coeff = V(:,i)'*w;
                H(i,j) = H(i,j)+coeff;
                w = w-coeff*V(:,i);
            end
        end
        tail = norm(w);
        H(j+1,j) = tail;
        if tail <= tol || j == m
            break
        end
        V(:,j+1) = w/tail;
    end
    H = H(1:j,1:j);
    V = V(:,1:j);
end
```

Read the output. In the left plot compare the early measured sines with their respective exact curves. In the right plot compare methods at equal product counts. Examine value error and residual separately once the value reaches its rounding scale. For Arnoldi inspect the difference between the shortcut and direct residual, and compare the off-tridiagonal entries with $\|A\|$. The Arnoldi experiment uses full reorthogonalization on a symmetric matrix to

reveal the Lanczos structure; a short-recurrence implementation has different storage and orthogonality behavior.

Two changes to try. First replace the input to the mixed run with double-precision products on a once-rounded matrix by passing **double(single(B))**. This isolates the effect of storing the mapped operator. Second, change the interval upper bound from 5 to 4, recompute both c and h, and replace $1/55$ in the prediction with the largest actual unwanted multiplier divided by the smallest actual target multiplier. Check which part of the earlier estimate depended on enclosing the entire unwanted spectrum.

17.12 Synthesis — What makes the computation reliable?

Module A's direct sum provides the coordinates $\binom{C}{D}$ in which algorithmic convergence becomes a graph contraction. Module B explains both why left eigenvectors measure perturbations and why Ritz tests annihilate the trial-space residual. Module C makes these tests computable through inner products and makes "close" precise through orthogonal projection. Module D diagonalizes a polynomial of a Hermitian operator. Module E uses all four results twice: first to choose the algorithm, then to decide what its output warrants. The hypotheses in the middle column below are as important as the formulas.

Quantity or result	Hypotheses	Interpretation
$\gamma_n \lvert A\rvert\lvert x\rvert$	Sequential real matvec, $nu < 1$, normal-range model	Componentwise absolute error envelope
$\min \lVert E\rVert = \lVert r\rVert$	Unit x, unrestricted matrix perturbation	Price of making the approximate pair exact
$\operatorname{dist}(\theta, \sigma(A)) \le \lVert r\rVert$	Hermitian or normal A, unit vector	Existence of a nearby eigenvalue
$\lVert (I-P)x\rVert \le \lVert r\rVert/\delta$	Hermitian A, unwanted spectrum separated from θ	Target-space angle certificate
$\tan t_k \le \rho^k \tan t_0$	Self-adjoint filter, full overlap, $\rho = \beta/\alpha < 1$	Block convergence across a cluster boundary
$X^*(AX - XH) = 0$	$X^*X = I$, $H = X^*AX$	Galerkin orthogonality
$\lVert r_{\text{Ritz}}\rVert = h_{m+1,m}\lvert e_m^* y\rvert$	Exact Arnoldi relation, normalized projected eigenpair	Cheap residual estimate
$\sum_{j=1}^{d} U_{d-j}(B)F_j$	Unscaled recurrence, fixed B, exact initialization	Total error from local filter defects
$\lVert L^{-1}(Kx - \theta Mx)\rVert$	$M = LL^* > 0$, $x^*Mx = 1$	Residual in the correct variational geometry

17.12.1 Mastery questions

1. Given a computed eigenpair, can you construct the optimal perturbation and specify exactly which structure it preserves?
2. Given a cluster, can you explain why an orthonormal basis may change substantially even when the projector barely moves?
3. Starting from a spectral decomposition, can you derive the tangent contraction for power, shifted inverse and filtered block iteration with their respective hypotheses?
4. Can you derive an Arnoldi residual using only the last subdiagonal, then identify the two numerical defects that compromise it?
5. Given spectral bounds and a requested contraction, can you map the interval, choose a degree and diagnose a missed unwanted eigenvalue?
6. Can you propagate a single early recurrence defect, then explain why an absolute filter error need not equal an angle error?
7. Can you design a mixed-precision comparison that separates storage error, arithmetic error, verification error and time to tolerance?

17.12.2 Where to go next

For HASC. Read the seminar recurrence with three quantities on paper: the object being multiplied at each step, the norm in which its product error is bounded, and the factor that transfers this error into the unwanted space. Compare an error proportional to the norm of a full iterate with one proportional to a residual correction. Track which steps require access to the higher-precision operator. In an experiment, maintain a separate column for measured time and another for the verified final residual. This lets you judge a performance claim at a stated mathematical accuracy.

For functional analysis. Orthogonal projections, Rayleigh quotients and bounded polynomials of a bounded self-adjoint operator extend to Hilbert space. A compact self-adjoint operator has an eigen-expansion with possible accumulation at zero, so a uniform positive gap cannot be assumed along its whole spectrum. Differential operators can be unbounded and require domains and boundary conditions. A useful next step is to prove spectral convergence of finite-element approximations using compact solution operators, distinguishing discretization error from iteration error.

For structural algebra. The polynomial action $p \mapsto p(A)$ and the cyclic quotient by an annihilating ideal explain exact Krylov termination independently of the coordinate basis. An inner product, an error norm and a rounding model add information that an algebraic quotient does not encode. Under a badly conditioned nonunitary change of coordinates the abstract operator is isomorphic while measured sensitivity can change substantially.

Foundation established in principle. After reconstructing the proofs and reproducing the experiment, you have the tools to justify selection, contraction, extraction and verification for

finite-dimensional Hermitian problems. **Further study** includes sharper clustered-subspace bounds, implicit restarting and locking, interior filter design, communication-avoiding QR, sparse generalized eigensolvers with inexact solves, and precision-dependent convergence proofs for a specific residual-based recurrence. Each needs assumptions beyond the elementary graph and residual estimates proved here.

Part III

Review III

18 Module A — Structural Linear Algebra: Linear Maps, Decomposition, and Quotient Structures

Three sensors report three numbers. The third always reports the sum of the first two. Four input settings control the sensors. Given a reading, can you tell whether it is possible? If it is possible, what exactly can you recover about the settings? These are different questions. The first concerns a subspace of the **output** space. The second concerns whole families of points in the **input** space. We will calculate both and discover a linear bijection between the possible readings and the families of indistinguishable settings.

Work over a field F (usually $\mathbb{R}$ or $\mathbb{C}$). Vectors in matrices are columns. Write $\mathcal{P}_n$ for polynomials of degree at most n, including zero. Finite dimension is assumed for dimension counts unless stated otherwise. The arguments about fibers, quotients and factorization will work without that assumption.

18.1 Session 1 — When can a list of numbers describe a polynomial?

A quadratic can be stored by its three coefficients or by its values at three distinct points. Why can either representation work, and why would four arbitrary readings be too many? To answer independently of any particular encoding, begin with the operations the encoding should respect.

18.1.1 From linear combinations to coordinates

A **vector space** is an abelian group $(V, +)$ with scalar multiplication $F \times V \to V$ satisfying $1v = v$, $(ab)v = a(bv)$, $(a + b)v = av + bv$, and $a(v + w) = av + aw$. Thus finite sums $\sum_i a_i v_i$ make sense, including the empty sum 0. Examples include matrices of fixed shape with entrywise operations, functions $X \to F$ with pointwise operations, and polynomials with formal addition and scalar multiplication. A matrix unit E_{ij} has one entry equal to 1 and all others zero. The matrix units form a basis of $F^{m\times n}$ because $M = \sum_{ij} M_{ij}E_{ij}$ and comparison of entries forces every coefficient.

A subset $U \subseteq V$ is a **subspace** if it contains 0 and contains $au + bv$ for every $u, v \in U$ and $a, b \in F$. Necessity follows from the operations within U. For sufficiency take $(a, b) = (1, 1), (-1, 0), (c, 0)$ to obtain closure under addition, inverses and scaling. The identities between operations are inherited from V. For instance the functions vanishing at a specified point form a subspace. Functions taking value 1 there fail the zero test.

Given an ordered list $B = (v_1, \dots, v_n)$, consider its **synthesis map**

$$S_B : F^n \to V, \qquad a \longmapsto \sum_{i=1}^{n} a_i v_i.$$

Its range, the **span**, is a subspace by distributivity. Every subspace containing the list contains the entire span. The list is **linearly independent** if $S_B(a) = 0$ forces $a = 0$. Subtract two representations of the same vector. Their difference synthesizes zero, so independence is precisely uniqueness wherever representations exist. A **basis** is an independent spanning list. Thus a basis makes S_B bijective, and the coordinate column $[v]_B = S_B^{-1}(v)$ is defined for every v. Independence concerns the list, including repetitions. A space admitting a finite spanning list is finite-dimensional.

18.1.2 Why do all bases have the same size?

Theorem 18.1. ***Finite exchange.*** *Let $s_1, \dots, s_n$ span V and let $u_1, \dots, u_k$ be independent. Then $k \leq n$, and k of the spanning vectors can be replaced by the u_i while preserving the span.*

Proof. Suppose $u_1, \dots, u_j$ have been inserted and the other entries are surviving generators. Expand u_{j+1} in this spanning list. Some surviving s has a nonzero coefficient, since otherwise u_{j+1} is a combination of its predecessors. Solve the expansion for that s by dividing by its coefficient. Replacing s by u_{j+1} therefore preserves the span. Starting with $j = 0$, the process takes one surviving entry at each step. If all n entries have been replaced, their span is V and an additional independent u is impossible. □

Delete any redundant vector from a finite spanning list until independence holds. This produces a basis, including the empty basis for the zero space. Applying exchange to two bases in both orders shows their lengths agree. This integer is $\dim V$. To **extend** an independent list, scan a spanning list for V, adjoining each vector outside the span accumulated so far. Each addition preserves independence (a dependence with nonzero new coefficient would solve for the new vector). The final list spans because it contains or generates each scanned entry. A subspace of an n-dimensional space has a basis as well. Repeatedly choose a vector outside the span within that subspace. Exchange bounds the number of choices by n.

In $\mathcal{P}_2(\mathbb{R})$, the bases $(1, t, t^2)$ and $(1, t-1, (t-1)^2)$ give t^2 the coordinates $(0, 0, 1)^T$ and $(1, 2, 1)^T$. Expanding in powers of t proves the second list independent by successively comparing leading

terms, and the same triangular calculation expresses the monomials in its span. There is no coordinate triple until the encoding is specified.

Checkpoint. Evaluation at three distinct real points is injective on $\mathcal{P}_2$ because a nonzero quadratic has at most two distinct roots. Can you construct the polynomial for arbitrary readings? For points s_i, form $\ell_i(t) = \prod_{j \neq i}(t - s_j)/(s_i - s_j)$. Evaluate $\sum_i y_i \ell_i$ at each point. A fourth reading must equal the value of this already determined polynomial.

18.2 Session 2 — How do we transport a calculation between bases?

A **linear map** $T : V \to W$ obeys $T(av+bw) = aT(v)+bT(w)$. It sends zero to zero, preserves differences, and by induction preserves every finite linear combination. The domain tells us which combinations may be formed. The codomain specifies the target in which attainable outputs are compared.

Prescribing images. Choose a basis $B = (b_j)$ of V and any $w_j \in W$. There is exactly one linear map with $Tb_j = w_j$. Indeed, define $T(\sum_j a_j b_j) = \sum_j a_j w_j$. Uniqueness of coordinates makes this a function, addition and scaling of coordinates give linearity, and linearity forces this formula for any proposed map. A redundant generating family does not allow arbitrary assignments. Every relation among generators must also hold among their proposed images.

With an ordered target basis C, put $A = S_C^{-1} T S_B$. This map between coordinate spaces has columns $[Tb_j]_C$, since $S_B e_j = b_j$. Thus $[Tv]_C = A[v]_B$. For $L : W \to Z$ with target basis H, inserting $S_C S_C^{-1}$ gives

$$[LT]_{H \leftarrow B} = [L]_{H \leftarrow C} [T]_{C \leftarrow B}.$$

Matrix multiplication composes descriptions whose middle coordinates agree. A bijective linear map is an **isomorphism**. Its inverse is linear since $T(au + bv) = aTu + bTv$ gives the unique preimage of the right-hand side. Two finite-dimensional spaces are isomorphic exactly when their dimensions agree. An isomorphism carries bases to bases, and matching two bases defines a bijection by the preceding construction.

18.2.1 Change the encoding, then differentiate

Let $P = S_B^{-1} S_{B'}$ and $Q = S_C^{-1} S_{C'}$. Their columns express new basis vectors in old coordinates. The direction matters. Starting with new coordinates, convert the input, apply the old matrix, and convert the result back:

$$[T]_{C' \leftarrow B'} = Q^{-1} A P.$$

For an endomorphism $V \to V$ with the same basis on both sides, $Q = P$, giving **similarity**. A rectangular map allows independent source and target changes. Rank survives both changes

because the coordinate maps are bijective. For an endomorphism the eigenvalue equation also survives similarity, since $Ax = \lambda x$ becomes $(P^{-1}AP)(P^{-1}x) = \lambda P^{-1}x$.

Consider $D : \mathcal{P}_2(\mathbb{R}) \to \mathcal{P}_1(\mathbb{R})$, $Dp = p'$. In monomial bases,

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}.$$

Take $B' = (1, t-1, (t-1)^2)$ and keep the monomial target. Differentiating the new basis gives columns $(0,0)^T, (1,0)^T, (-2,2)^T$. Alternatively multiply

$$A \begin{pmatrix} 1 & -1 & 1 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & -2 \\ 0 & 0 & 2 \end{pmatrix}.$$

Applying this to $(1,2,1)^T$ gives $(0,2)^T$, precisely $D(t^2) = 2t$. If the target is shifted as well, the last column becomes $(0,2)^T$ in $(1, t-1)$. The output polynomial is the same in either encoding.

Checkpoint. For the ordinary matrix $\mathrm{diag}(2,3)$ use the new basis (e_2, e_1) on both sides. Its matrix becomes $\mathrm{diag}(3,2)$. Swapping only the input basis would instead give columns $(0,3)^T, (2,0)^T$. Which represents similarity?

18.3 Session 3 — What can one reading determine?

Define $K = \ker T = \{x : Tx = 0\}$ and $I = \operatorname{im} T = \{Tx : x \in V\}$. Both are subspaces. For K, applying T to any linear combination of its members gives zero. For I, combine preimages before applying T. Now subtract two candidate inputs:

$$Tx = Ty \quad \Longleftrightarrow \quad T(x-y) = 0 \quad \Longleftrightarrow \quad x - y \in K.$$

Hence T is injective exactly when $K = \{0\}$ (compare each kernel vector with zero). It is surjective exactly when $I = W$, by the meaning of surjectivity. For $b \in I$ choose $Tx_p = b$. The displayed equivalence proves in both directions that

$$T^{-1}(\{b\}) = x_p + K.$$

This **fiber** is an affine space, a translate with direction space K. Translating the origin to a different solution does not change the fiber. For $b \neq 0$ the fiber excludes the zero vector. For $b \notin I$ it is empty.

The inclusion $(u,v) \mapsto (u,v,0)$ into F^3 is injective and misses e_3. The map $(u,v,w) \mapsto (u,v)$ onto F^2 is surjective and loses an entire coordinate. The map $(u,v,w) \mapsto (u,0)$ into F^2 has both a nontrivial kernel and an unattainable output. Restricting its codomain to its image changes its surjectivity.

Theorem 18.2. ***Rank–nullity.*** *If* $\dim V < \infty$*, then* $\dim V = \dim K + \dim I$*. These last two dimensions are the nullity and the rank, respectively.*

Proof. Extend a kernel basis $k_1, \dots, k_s$ to a domain basis by adjoining $h_1, \dots, h_r$. Expand any input in this basis and apply T. Only the Th_i survive, so they span I. If $\sum_i c_i Th_i = 0$, then $\sum_i c_i h_i$ belongs to K. Expressing that vector in the k's gives a relation in the full basis. All c_i vanish. Thus the Th_i form an image basis, and the domain basis has $s + r$ members. □

The proof also constructs a smaller inverse problem. On $H = \operatorname{span}(h_i)$ the restriction $T|_H : H \to I$ is bijective. Each reading has exactly one representative in H. We chose the h_i; soon we will recover the information in a reading without choosing any representative.

Checkpoint. A linear map $F^4 \to F^3$ loses at least one dimension, since its rank is at most three. When the rank is two, how many parameters describe a compatible fiber? How many independent linear constraints describe its image? The two answers are $4 - 2$ and $3 - 2$.

18.4 Session 4 — What does elimination preserve?

Return to the four settings and three sensors. Keep the matrix used again in Modules B and C:

$$A = \begin{pmatrix} 1 & 2 & -1 & 3 \\ 2 & 4 & 1 & 9 \\ 3 & 6 & 0 & 12 \end{pmatrix}.$$

The last row is the sum of the first two. Thus $b_3 = b_1 + b_2$ is necessary. Is it sufficient? Carry the right-hand side through exactly the same row operations. First subtract rows 1 and 2 from row 3, then subtract twice row 1 from row 2. Divide row 2 by three and add it to row 1. The augmented system becomes

$$\left(\begin{array}{cccc|c} 1 & 2 & 0 & 4 & \alpha \\ 0 & 0 & 1 & 1 & \beta \\ 0 & 0 & 0 & 0 & b_3 - b_1 - b_2 \end{array}\right), \qquad \beta = (b_2 - 2b_1)/3, \quad \alpha = b_1 + \beta.$$

Each elementary row operation is invertible (undo a swap, use the reciprocal scaling, or subtract the added multiple). Their product on the left is an invertible matrix E. We solved $EAx = Eb$, which is equivalent to $Ax = b$. A zero coefficient row with nonzero right-hand side is the complete obstruction here. For a compatible reading, all solutions are

$$x = (\alpha, 0, \beta, 0)^T + s(-2, 1, 0, 0)^T + t(-4, 0, -1, 1)^T.$$

In particular put $k_1 = (-2, 1, 0, 0)^T$, $k_2 = (-4, 0, -1, 1)^T$. These span the kernel, and reading coordinates 2 and 4 of a relation proves independence. For $b = (1, 2, 3)^T$, the fiber passes through e_1; for $(1, 2, 4)^T$, there is no solution.

18.4.1 Recovering the kernel and the embedded image

In **reduced row echelon form** (RREF) zero rows come last, each leading entry is 1, leading positions move right down successive nonzero rows, and each pivot is alone in its column. Elimination constructs this form by finding the leftmost available nonzero column, moving a nonzero entry into pivot position, normalizing it, clearing its column, and continuing on the remaining rows and columns. In the reduced matrix $R = EA$ the pivot columns here are e_1, e_2 in F^3. Rearrange **variables** temporarily so pivots come first. The general reduced system has the shape

$$\begin{pmatrix} I_r & G \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_p \\ x_f \end{pmatrix} = \begin{pmatrix} d \\ h \end{pmatrix}.$$

It is consistent exactly when $h = 0$. Free coordinates z parametrize the homogeneous solutions by $z \mapsto (-Gz, z)$, with inverse extraction of free coordinates. Undo the column permutation when reporting vectors. Applying this isomorphism to the standard basis gives the null-space basis, including the empty list for a trivial kernel.

Left multiplication gives $\ker R = \ker A$ because E is injective, but $\operatorname{im} R = E(\operatorname{im} A)$. Here the reduced columns span the plane $y_3 = 0$; the original columns span the different plane $y_3 = y_1 + y_2$. What transports back is **every relation among columns**: $Rc = 0$ iff $Ac = 0$. Consequently the pivot **indices** select an independent spanning family in the original matrix. Explicitly put

$$C = (a_1\ a_3) = \begin{pmatrix} 1 & -1 \\ 2 & 1 \\ 3 & 0 \end{pmatrix}, \qquad J = \begin{pmatrix} 1 & 2 & 0 & 4 \\ 0 & 0 & 1 & 1 \end{pmatrix}.$$

Then $a_2 = 2a_1$ and $a_4 = 4a_1 + a_3$, hence $A = CJ$. The two numbers Jx measure all the input information visible to the sensors. The three numbers $C(Jx)$ express that reading in the original output coordinates. The matrix sizes already record an exact rank factorization through F^2 (here and for this example $F = \mathbb{R}$).

When is a tiny pivot significant? For $A_\epsilon = \operatorname{diag}(1, \epsilon)$, $\epsilon > 0$, the algebraic rank is two, but an error δ in the second reading becomes δ/ϵ in the recovered input. The stretching factors (singular values) are 1 and ϵ for $0 < \epsilon \leq 1$. A numerical rank with absolute threshold τ counts stretches greater than τ and gives one if $\epsilon \leq \tau < 1$. Rescaling the data changes the interpretation of an absolute threshold. Modules D and E develop this distinction for arbitrary matrices.

Checkpoint. Without elimination, explain why the first two rows of J give two independent measurements and why C loses neither. Use Je_1 and Je_3 and the independence of a_1, a_3.

18.5 Session 5 — When is a decomposition a coordinate system?

Suppose two mechanisms contribute unknown vectors from subspaces U and H of V. Which totals can occur? When can the contributions be recovered from their sum? Form the **external direct sum** $U \oplus_{\text{ext}} H$ of ordered pairs with componentwise operations and the addition map $\sigma(u, h) = u + h$. Its image $U + H$ contains both spaces and is contained in every subspace containing both. Unlike the union, the sum includes mixed combinations. The union of the two coordinate axes misses $(1, 1)$.

The kernel of σ consists of $(z, -z)$ with $z \in U \cap H$. Thus the overlap measures the entire ambiguity. Rank–nullity on the pair space (concatenate the bases $(u_i, 0), (0, h_j)$) gives

$$\dim(U + H) = \dim U + \dim H - \dim(U \cap H).$$

A sum is **internally direct**, written $U \oplus H$, if σ is injective. Equivalently each element of the sum has a unique decomposition, or $U \cap H = 0$. To assert $V = U \oplus H$ one must also establish spanning.

For a finite family U_i, the same addition map on tuples has trivial kernel exactly when $\sum u_i = 0$ forces every $u_i = 0$. Equivalently $U_i \cap \sum_{j \neq i} U_j = 0$ for each i: in a relation isolate u_i, and conversely any nonzero member of such an intersection supplies a relation. The lines spanned by $(1, 0), (0, 1), (1, 1)$ intersect pairwise only at zero. Yet $(1, 0) + (0, 1) - (1, 1) = 0$. The third line lies in the **sum** of the first two.

18.5.1 A complement chooses one point per fiber

A complement H of K satisfies $V = K \oplus H$. Extending a basis of K produces one. For the sensor matrix use $H = \text{span}(e_1, e_3)$ and define $S(u, v) = (u, 0, v, 0)^T$. Since $JS = I_2$ and $\ker J = K$, every input splits as

$$x = \underbrace{x - SJx}_{\in K} + \underbrace{SJx}_{\in H}.$$

If $Sz \in K$ then $0 = JSz = z$, proving uniqueness. One could equally replace e_1 by $e_1 + k_1$. Its reading is unchanged and the new span with e_3 is another complement. For example e_1 and $e_1 + k_1$ are different representatives of the very same reading.

Theorem 18.3. ***Projections and decompositions.*** *An endomorphism P satisfies $P^2 = P$ exactly when it is the projection onto one summand along another. Its summands are* $\operatorname{im} P$ *and* $\ker P$.

Proof. For an idempotent P, write $x = Px + (x - Px)$. The second term has image zero under P, while P fixes every Py because $P(Py) = Py$. A vector both fixed and killed must vanish. Conversely if $x = u + h$ uniquely, put $P(x) = u$. Adding or scaling the components

proves linearity. Applying P again to $u + 0$ returns u. Its image and kernel are exactly the two prescribed summands. □

For the sensors SJ projects onto H along K; $I_4 - SJ$ projects onto K along H. An algebraic projection specifies a direction along which to project. Orthogonality requires an inner product and a perpendicular direction.

In Figure 18.1 the measurement is $m(x, y) = y$. A horizontal line is a whole fiber. Both transversals meet every fiber once, but the representative selected at height two depends on the transversal. The drawing models the choice S above in two dimensions.

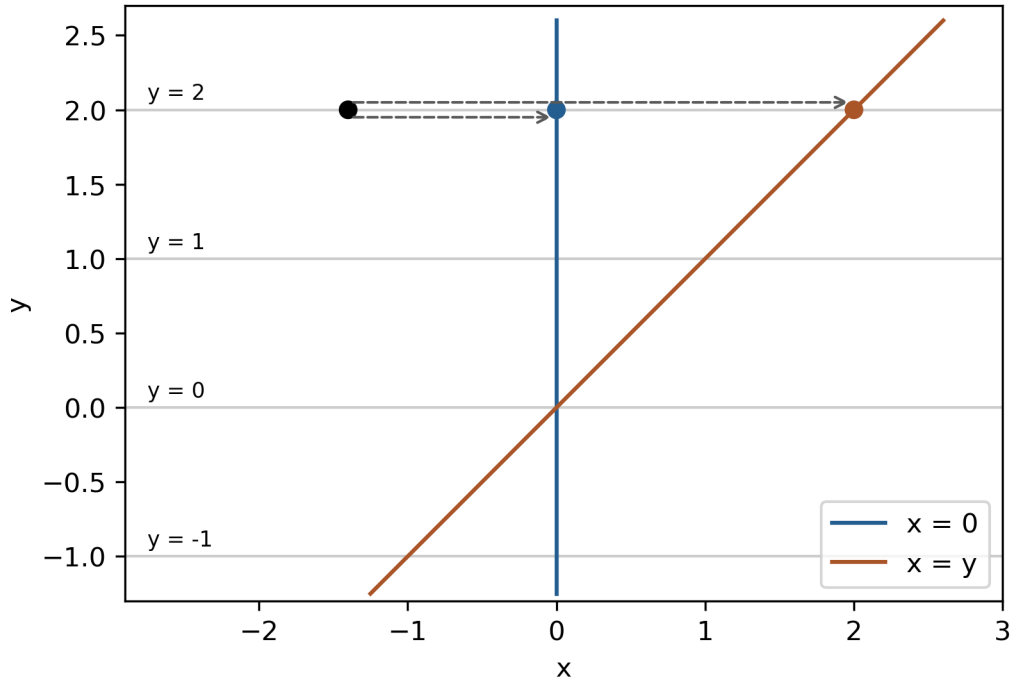


Figure 18.1: Two complements select different representatives for each height measurement. The dashed arrows lie in the kernel direction.

In three dimensions take $U = \{z = 0\}$ and $H = \operatorname{span}(1, 0, 1)$. Every (x, y, z) splits into $(x - z, y, 0) + z(1, 0, 1)$. The height determines the line component. In Figure 18.2 the dashed segment is parallel to H; projecting vertically would select a different point of the plane. Perspective alone cannot prove uniqueness. The height equation does.

Checkpoint. If you know only $\operatorname{im} P$, can you reconstruct P? Use $P(x, y, z) = (x - z, y, 0)$ and the vertical projection onto the same plane.

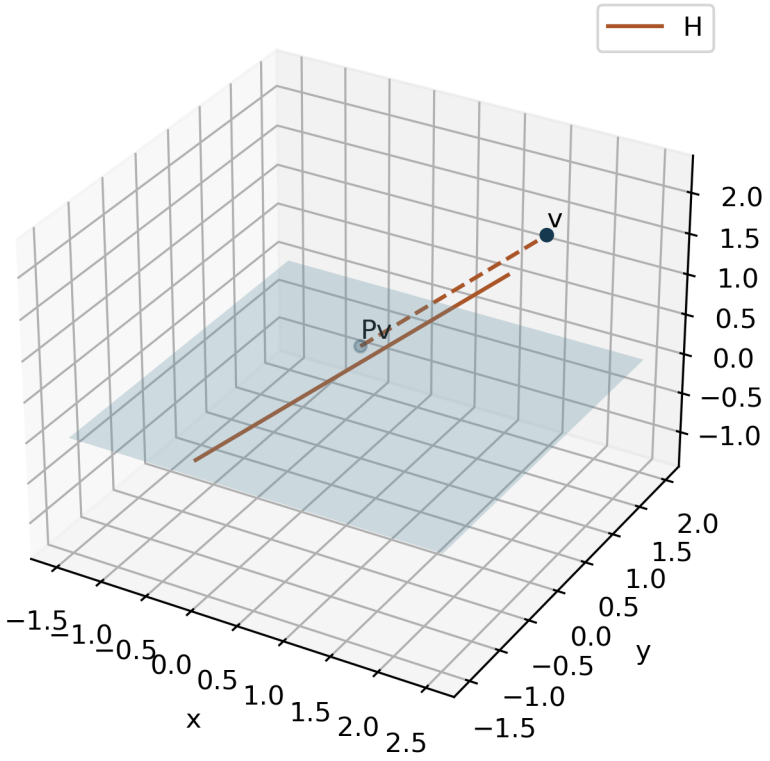


Figure 18.2: Projection of v onto the horizontal plane along the sloping line H.

18.6 Session 6 — Can we recover exactly what a measurement knows?

For $U \subseteq V$ a subspace, declare $x \sim y$ if $x - y \in U$. This is reflexive since $0 \in U$, symmetric since U contains negatives, and transitive since differences add. The class $[x] = x + U$ is the entire coset. Two cosets coincide exactly when their representatives differ by an element of U; otherwise they are disjoint, since a common point would imply that difference belongs to U.

The **quotient space** V/U consists of these classes with

$$[x] + [y] = [x + y], \qquad c[x] = [cx].$$

To check well-definedness replace x, y by $x + u, y + v$. The resulting differences $u + v$ and cu belong to U. The vector-space identities descend from those in V, the zero class is U, and $-[x] = [-x]$. The quotient map $q : x \mapsto [x]$ is linear, onto, and has kernel U. In finite dimension rank–nullity gives $\dim(V/U) = \dim V - \dim U$. More concretely, extend a basis of U by $h_1, \dots, h_r$. The classes $[h_i]$ span since all U-components vanish. Independence follows because $\sum c_i h_i \in U$ forces every $c_i = 0$.

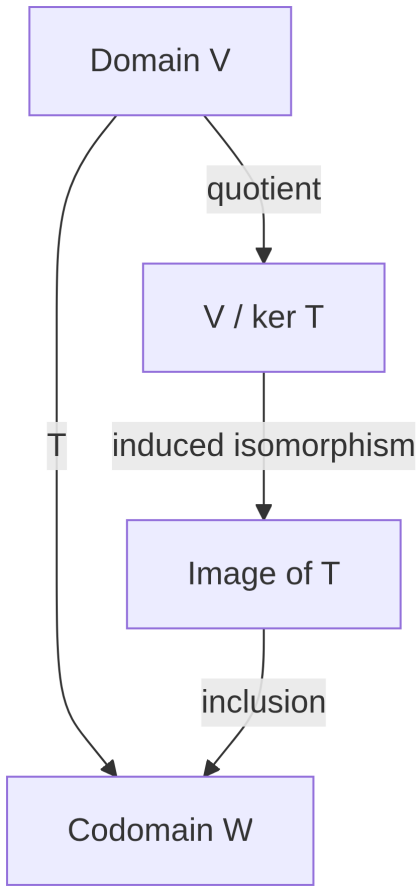
Each horizontal line in Figure 18.1 is **one vector** of the quotient by the horizontal axis. Adding two lines as classes means adding their heights. A chosen transversal draws a representative for each class. Intrinsically the zero of the quotient is the whole horizontal axis. This interpretation continues to work in dimension four, where picturing all the classes at once is harder than computing with them.

Theorem 18.4. ***First isomorphism theorem.*** *Every linear $T : V \to W$ gives an isomorphism*

$$\overline{T} : V/\ker T \longrightarrow \operatorname{im} T, \qquad [x] \longmapsto Tx.$$

Writing $i : \operatorname{im} T \hookrightarrow W$ for inclusion gives $T = i\overline{T}q$.

Proof. Two representatives of a class have difference in $\ker T$, so their outputs agree. The formulas for quotient operations and linearity of T now give linearity of $\overline{T}$. If $\overline{T}[x] = 0$, then $x \in \ker T$ and $[x] = [0]$, proving injectivity. Every element of the target is Tx for some x, proving surjectivity. Evaluate the composite on x to get Tx. Any candidate for the middle arrow must take $[x]$ to Tx because q hits every class and i is injective. This proves uniqueness. □



The two paths send each input to the same output. Passing to a class removes precisely the ambiguity; restricting the target removes unattainable readings. The middle arrow is **canonical given** T because its action is forced on every class. No inverse from $\operatorname{im} T$ to V was chosen. Its canonical inverse takes b to its *entire fiber*, viewed as a single class.

For $A = CJ$, use $([e_1], [e_3])$ in the quotient. The coordinates of $[x]$ are $(x_1 + 2x_2 + 4x_4, x_3 + x_4)^T = Jx$ since $x - SJx \in K$. With image basis (a_1, a_3), $\overline{A}$ has matrix I_2. With image coordinates (b_1, b_2) (the third entry is their sum), it has matrix

$$M = \begin{pmatrix} 1 & -1 \\ 2 & 1 \end{pmatrix}, \qquad \det M = 3.$$

Both matrices represent the same induced isomorphism. The second recovers α, β from Session 4 by solving a genuine two-by-two invertible system. The quotient description explains why this reduced inverse exists; elimination supplies its coordinates.

Checkpoint. If $U \subsetneq \ker T$, the rule $[x] \mapsto Tx$ still makes sense, but a nonzero class represented by $k \in \ker T \setminus U$ maps to zero. If U contains a vector outside $\ker T$, even well-definedness fails. Locate the exact step of the proof affected in each case.

18.7 Session 7 — Which maps survive a quotient, and which maps split it?

Suppose the stored datum is a class $[x]$. A later calculation requests $f(x)$. Can one honor that request without recovering the lost part of x?

Universal property of the quotient. For a linear $f : V \to Z$ there exists a linear $g : V/U \to Z$ with $gq = f$ if and only if $U \subseteq \ker f$. When it exists it is unique. Necessity follows by evaluating gq on U. For sufficiency set $g([x]) = f(x)$. Replacing x by $x + u$ changes $f(x)$ by $f(u) = 0$. The class operations give linearity, and surjectivity of q forces uniqueness. This is also a test for whether a proposed measurement of a class is meaningful.

Why call this a characterization? Suppose a surjection $q' : V \to Q'$ kills U and has the same factorization property. Applying the properties to q' and q gives arrows $a : V/U \to Q'$ and $b : Q' \to V/U$ with $aq = q'$ and $bq' = q$. Now $baq = q$ and $abq' = q'$. Surjectivity shows ba and ab are identities. Thus the two answers to the same mapping problem are related by a unique isomorphism respecting the quotient arrows.

There is a corresponding property on the input side. If $h : X \to V$ satisfies $Th = 0$, its values belong to K. Restricting the target gives a unique $\tilde{h} : X \to K$ with $h = j\tilde{h}$, where $j : K \hookrightarrow V$. Uniqueness follows from injectivity of j. Dually the quotient $W/\operatorname{im} T$ records the obstruction to solvability: the class of b is zero exactly when $Tx = b$ can be solved. This quotient is the **cokernel**. Any linear map out of W killing $\operatorname{im} T$ factors uniquely through it.

18.7.1 What specifies a map into or out of a direct sum?

For maps $f : U \to Z$ and $g : H \to Z$ define $L(u, h) = f(u) + g(h)$. It is linear and restricts to the prescribed maps on $(u, 0)$ and $(0, h)$. Conversely linearity forces this formula since $(u, h) = (u, 0) + (0, h)$. This is the **coproduct** property. In the reverse direction maps $f : X \to U$ and $g : X \to H$ give the unique $L(x) = (f(x), g(x))$ with those two projected components, the **product** property. For an internal sum with overlap, prescribing maps on the summands requires agreement on their intersection. The relation $(z, -z)$ from Session 5 explains exactly why.

18.7.2 Exactness and splitting

A sequence $X \xrightarrow{a} Y \xrightarrow{b} Z$ is **exact at** Y if $\operatorname{im} a = \ker b$. This implies $ba = 0$. Zero composition alone gives only the inclusion. For instance $F \to F^2 \to F$ with both arrows zero has zero composition but fails exactness at F^2.

The **short exact sequence**

$$0 \longrightarrow K \xrightarrow{j} V \xrightarrow{p} I \longrightarrow 0, \qquad p(x) = Tx,$$

encodes three facts. At K it asserts that j is injective, at V that the invisible inputs are exactly $j(K)$, and at I that p is onto. Exactness here uses the restricted target I. For the original map one can continue instead by $W \to W/I \to 0$.

A **section** of p is a linear $s : I \to V$ with $ps = \mathrm{id}_I$. Choose a basis of I, lift each basis vector to an input, and extend the assignments linearly. This proves existence in finite dimension; in arbitrary dimension the same proof uses the usual basis theorem with choice. Write

$$x = (x - spx) + s(px).$$

The first term belongs to K, and $s(I) \cap K = 0$ since $p(sy) = y$. Thus a section determines a complement. Conversely a complement H makes $p|_H$ bijective, whose inverse is a section. The map $(k, y) \mapsto j(k) + s(y)$ is a **splitting** isomorphism $K \oplus_{\mathrm{ext}} I \to V$. Every short exact sequence of vector spaces has this form after identifying the first space with its injected image. For two sections s, s' one has $p(s' - s) = 0$; adding any linear map $I \to K$ to s gives another section. This describes all choices, including when there is only one.

Over a ring a surjection need not split. In abelian groups (modules over $\mathbb{Z}$), $\mathbb{Z} \to \mathbb{Z}/2\mathbb{Z}$ has kernel $2\mathbb{Z}$. A section would send $[1]$ to an odd integer m with $2m = 0$, which is impossible. The proof by lifting a basis fails because the target has a relation $2[1] = 0$ that a lift must respect. Module F2 takes up this distinction.

Checkpoint. In the sensor problem find a section using S and M^{-1}. For compatible b, it is $s(b) = SM^{-1}(b_1, b_2)^T$. Show that replacing $s(b)$ by $s(b) + b_1 k_1$ leaves $As(b)$ unchanged.

18.8 Session 8 — Which part of a physical or numerical solution is identifiable?

18.8.1 Compatible readings, inconsistent data, and an inverse on classes

The linear functional $\ell(b) = b_3 - b_1 - b_2$ detects the complete obstruction for A. Its vanishing is sufficient because M is invertible. In fact W/I is identified with $\mathbb{R}$ by $[b] \mapsto \ell(b)$, another first isomorphism theorem. Thus two *different* quotients arise. V/K records the part of the input visible in the data. W/I records the part of the data no input can explain.

For a noisy reading $(1, 2, 4)$, the obstruction is 1. Algebra alone assigns no size to a correction. With the Euclidean inner product, $n = (-1, -1, 1)^T$ is perpendicular to I, and the corrected reading is $b' = b - n/3$. Indeed $n^T b' = 0$, and for every $y \in I$, $\|b - y\|^2 = \|b - b'\|^2 + \|b' - y\|^2$. This proves that b' is the unique closest reading. All inputs attaining it minimize

the residual. Among those, choosing the unique representative perpendicular to K minimizes input norm by the same squared-length identity. The map taking b to that representative is the Moore–Penrose pseudoinverse in Euclidean coordinates. Modules C and D derive the general construction.

18.8.2 Edge differences, cycles and conservation

Orient a path $1 \to 2 \to 3$. Its discrete gradient sends vertex potentials to edge differences:

$$G_0 p = \begin{pmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix} p.$$

Every pair of readings is attainable: choose $p_1 = 0$, then add the readings successively along the path. Every constant shift is invisible. Add a closing edge $3 \to 1$ to obtain G by adjoining row $(1, 0, -1)$. Now the edge data d must satisfy $d_1 + d_2 + d_3 = 0$ by telescoping. This condition is sufficient because the path construction produces the first two differences, forcing the third. Thus $\mathbb{R}^3/\operatorname{span}(\mathbf{1})$ is canonically isomorphic via G to the zero-sum **edge** plane. A reference potential $p_1 = 0$ selects one solution; the zero-mean convention selects another. On any graph, equal potentials across every edge are constant on each connected component, so the gradient nullity counts the components.

For an oriented edge flow f, $(G^T f)_i$ is inflow minus outflow at vertex i. On the triangle $G^T(1, 1, 1)^T = 0$: circulation around a cycle has no net source. A gradient has zero signed sum around every cycle. Conversely on a connected graph those zero sums allow potential recovery by integrating differences along any path from a fixed vertex. Two paths differ by a closed walk, which decomposes into cycles after cancellations, hence give the same potential.

The path Laplacian is

$$L = G_0^T G_0 = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{pmatrix}.$$

The identity $p^T L p = \|G_0 p\|^2$ proves $\ker L = \ker G_0$. Since its column sums vanish and its rank is two, its image is the zero-sum **vertex** plane. In $Lp = b$ the compatibility condition expresses total source balance. The gradient and Laplacian thus share a kernel while their outputs encode different quantities.

18.8.3 Which boundary condition removes an integration constant?

For $n \geq 1$ over $\mathbb{R}$, $D : \mathcal{P}_n \to \mathcal{P}_{n-1}$ kills constants. Comparing coefficients of $p' = \sum_{j=1}^{n} j a_j t^{j-1}$ proves these are its entire kernel. Termwise integration $t^j \mapsto t^{j+1}/(j+1)$ supplies a section, proving surjectivity. Thus $\mathcal{P}_n/\mathcal{P}_0 \cong \mathcal{P}_{n-1}$. The section chooses an antiderivative vanishing at

zero. Integration from a different basepoint changes this choice. Over a field of characteristic $p > 0$ the derivative of t^p vanishes, which shows precisely why the scalar assumption matters.

For $D : C^1([0,1]) \to C([0,1])$ the fundamental theorem of calculus supplies the section $g \mapsto (t \mapsto \int_0^t g(s)\,ds)$; the mean value theorem implies the same constant kernel. Restrict the domain to $u(0) = u(1) = 0$. Now injectivity holds, but the image consists precisely of g with $\int_0^1 g = 0$. One endpoint fixed the representative; the second introduced a compatibility condition. In infinite dimensions an algebraic isomorphism alone gives no norm estimate. In the sup norm differentiation can amplify oscillations: $\sin(nt)/n$ tends uniformly to zero while its derivative has supremum norm one. In the C^1 norm $\|u\|_\infty + \|u'\|_\infty$ differentiation is bounded. Module F1 additionally asks whether images are closed and whether quotients are complete. A closed subspace of a Banach space yields a Banach quotient; for a nonclosed subspace the distance to it can vanish even for a distinct coset.

18.8.4 What does a subspace algorithm actually carry between iterations?

Store a trial space as columns of a full-rank matrix B. For invertible R, $\operatorname{im}(BR) = \operatorname{im} B$ because R runs through all coefficient columns. In a thin QR factorization $B = QR$, Q has orthonormal columns and R is invertible for full column rank. Hence Q stores the same space, and $\|Qc\|_2 = \|c\|_2$. Replacing a basis can improve the behavior of coordinates without changing any candidate solution.

For a square operator H, a subspace U is **invariant** when $H(U) \subseteq U$. Begin a basis with a basis of U. Its first columns under H have no component in the remaining coordinates, giving block form $\left(\begin{smallmatrix} H_{11} & H_{12} \\ 0 & H_{22} \end{smallmatrix}\right)$. If the complement is also invariant then $H_{12} = 0$. One invariant summand alone gives no reason for that second zero. The induced action on the quotient V/U has matrix H_{22} because the upper components disappear on taking classes.

Take the symmetric matrix recurring in the spectral and numerical modules,

$$H = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 9 \end{pmatrix}.$$

The horizontal plane and vertical axis are invariant. Within the plane, $(1,-1,0)^T$ and $(1,1,0)^T$ span eigenlines of eigenvalues $1, 3$; the axis has eigenvalue 9. These three independent directions give a direct sum. If $Hv = \lambda v$, induction gives $H^j v = \lambda^j v$, hence $f(H)v = f(\lambda)v$ for a polynomial f. For this example $f(t) = (t-1)(t-3)/48$ gives the projection onto the vertical axis along the horizontal plane. This exact filter requires known eigenvalues. Chebyshev filtering in Module E uses spectral bounds to achieve approximate separation.

Starting with v, the **Krylov space** $\mathcal{K}_m(H, v) = \operatorname{span}(v, Hv, \dots, H^{m-1}v)$ contains every $f(H)v$ for $\deg f < m$. If the next power already lies in this span, applying H to its generators proves invariance. Block subspace iteration instead propagates a trial basis Q to a basis of $\operatorname{im}(HQ)$

(assuming rank is maintained). With orthonormal Q, Q^*HQ represents the projected operator QQ^*H restricted to the trial space. A reduced eigenpair $Q^*HQz = \theta z$ yields $Q^*(HQz - \theta Qz) = 0$. The residual is perpendicular to the trial space and may still be nonzero. This is the algebra behind Rayleigh–Ritz and a reason Module E asks for explicit residual checks.

Checkpoint. Start with $v = e_1$ in this example. Then $Hv = (2, 1, 0)^T$ and $\mathcal{K}_2$ is the horizontal plane. Why can no polynomial in H applied to this v recover a nonzero vertical component, even though the vertical eigenvalue has greatest magnitude?

18.9 Computational laboratory — From an inverse problem to a factorization

Run the cells in document order with Python 3, SymPy, NumPy and Matplotlib in the Jupyter environment used by Quarto. SymPy performs the algebra over exact integers and rationals. NumPy supplies plotting coordinates. Predict each printed result before running the corresponding cell.

18.9.1 A. What can row reduction certify?

```
import sympy as sp

A = sp.Matrix([[1, 2, -1, 3], [2, 4, 1, 9], [3, 6, 0, 12]])
reduced, pivots = A.rref()
K = sp.Matrix.hstack(*A.nullspace())
C = A[:, list(pivots)]
J = reduced[:len(pivots), :]
assert A*K == sp.zeros(3, 2) and K.rank() == 2
assert C.rank() == A.rank() and A.rank() + K.cols == A.cols
assert C*J == A
print("RREF:", reduced.tolist())
print("Pivot indices (one-based):", [i+1 for i in pivots])
print("Kernel basis:", [list(v) for v in K.columnspace()])
print("Image basis:", [list(v) for v in C.columnspace()])
print("Rank and nullity:", A.rank(), K.cols)
```

```
RREF: [[1, 2, 0, 4], [0, 0, 1, 1], [0, 0, 0, 0]]
Pivot indices (one-based): [1, 3]
Kernel basis: [[-2, 1, 0, 0], [-4, 0, -1, 1]]
Image basis: [[1, 2, 3], [-1, 1, 0]]
Rank and nullity: 2 2
```

The two free coordinates give two independent kernel vectors. The output basis lives in three-dimensional sensor space. Multiplying CJ recovers every original column, a certificate that the two selected columns span the image.

18.9.2 B. Which two components sum to the input?

```
S = sp.eye(4)[:, list(pivots)]
P = S*J
x = sp.Matrix([5, 2, 7, 3])
visible, invisible = P*x, (sp.eye(4)-P)*x
assert visible + invisible == x and A*invisible == sp.zeros(3, 1)
assert J*S == sp.eye(2) and P*P == P
assert sp.Matrix.hstack(K, S).det() != 0
print("Kernel component:", list(invisible))
print("Complement component:", list(visible))
```

```
Kernel component: [-16, 2, -3, 3]
Complement component: [21, 0, 10, 0]
```

The output is $(-16, 2, -3, 3)^T + (21, 0, 10, 0)^T$. The nonzero determinant checks uniqueness **for every input**: the two kernel vectors and the two complement vectors form a basis together.

18.9.3 C. Can a new representative alter the quotient coordinates?

```
a, b = sp.symbols("a b")
shifted = x + K*sp.Matrix([a, b])
assert (J*shifted-J*x).expand() == sp.zeros(2, 1)
print("Class coordinates:", list(J*x))
print("Chosen representative:", list(S*J*x))
```

```
Class coordinates: [21, 10]
Chosen representative: [21, 0, 10, 0]
```

The two parameters range over the whole fiber. The first output describes its class; the second embeds one representative back in the original domain. Multiplication by S performs the latter choice.

18.9.4 D. Where is the invertible part of a singular problem?

```
M = C[:2, :]
inclusion = sp.Matrix([[1, 0], [0, 1], [1, 1]])
assert inclusion*M*J == A
assert M.det() != 0
reading = A*x
assert M.inv()*reading[:2, :] == J*x
print("Induced matrix:", M.tolist(), "determinant:", M.det())
print("Reading:", list(reading), "recovered coordinates:", list(J*x))
```

```
Induced matrix: [[1, -1], [2, 1]] determinant: 3
Reading: [11, 52, 63] recovered coordinates: [21, 10]
```

Here the inclusion matrix converts plane coordinates (u, v) to $(u, v, u + v)$. The determinant belongs to the induced two-dimensional map. The original rectangular matrix has no determinant.

18.9.5 E. Which edge readings come from potentials?

```
G = sp.Matrix([[-1, 1, 0], [0, -1, 1], [1, 0, -1]])
p = sp.Matrix([2, 5, 4])
ones = sp.ones(3, 1)
assert G.rank() == 2 and G.nullspace() == [ones]
assert G*(p+a*ones) == G*p
assert ones.T*G == sp.zeros(1, 3)
print("Gradient:", list(G*p), "kernel basis:", list(G.nullspace()[0]))
print("Cycle sum:", sum(G*p), "rank:", G.rank())
```

```
Gradient: [3, -1, -2] kernel basis: [1, 1, 1]
Cycle sum: 0 rank: 2
```

Differences $(3, -1, -2)$ close around the cycle. The right kernel records a potential ambiguity; the left kernel imposes a compatibility condition on edge data. That both are spanned by all-ones columns in this example reflects the chosen triangle orientation.

18.9.6 F. Which basis represents the trial plane?

```
R = sp.Matrix([[2, 1], [1, 1]])
new_basis = C*R
assert R.det() != 0 and new_basis*R.inv() == C
assert sp.Matrix.hstack(C, new_basis).rank() == 2
old_coordinates = sp.Matrix([3, -2])
new_coordinates = R.inv()*old_coordinates
assert C*old_coordinates == new_basis*new_coordinates
print("New basis:", new_basis.tolist())
print("Old and new coordinates:", list(old_coordinates), list(new_coordinates))
```

```
New basis: [[1, 0], [5, 3], [6, 3]]
Old and new coordinates: [3, -2] [5, -7]
```

One multiplication proves containment of the new span in the old; the inverse multiplication proves the reverse containment. Replacing R by a singular matrix could discard a direction. This is the span-preservation test behind the QR step in subspace iteration.

18.10 Exercises — Reconstruct the arguments

18.10.1 Build an encoding

1. **Interpolation with missing information.** On $\mathcal{P}_2(\mathbb{R})$ take $E(p) = (p(0), p(1))$. Construct a basis adapted to this measurement. Compute its matrix and all polynomials yielding $(2, 5)$. Which part of your construction can be changed without changing E?
2. **When can prescriptions extend?** In F^2 assign $e_1, e_2, e_1 + e_2$ the images a, b, c in a space W. Prove the exact condition for a linear extension. Generalize to an arbitrary spanning family using its synthesis map.
3. **Dimension from measurements.** For a linear T, lift an image basis to V and append a kernel basis. Prove directly that the result is a basis and deduce rank–nullity. Where is finiteness used?

18.10.2 Diagnose solvability and decomposition

4. **Audit an elimination.** For the sensor matrix solve $Ax = (2, 1, 3)^T$, list bases of kernel and image, and test what happens if the last reading becomes 4. Explain why using pivot columns from the RREF could give a wrong consistency decision.

5. **Sum, union, and pairwise tests.** Use three distinct lines in the plane to exhibit both failures. Construct instead three lines in F^3 whose sum is direct and a plane and line in F^3 whose sum is the whole space but has no preferred perpendicularity.
6. **Count an overlap two ways.** Give a proof of the sum dimension formula using an adapted basis instead of the addition map. Apply both approaches to the planes $x = 0$ and $y = 0$ in $\mathbb{R}^3$.
7. **Recover the splitting.** For $P(x, y) = (x+2y, 0)$ find its kernel, image and complementary projector. Give another projector with the same image, and find a vector where the two projections disagree.

18.10.3 Descend, factor, and split

8. **A quotient test.** Let $U = \operatorname{span}(1, 1, 0)$ in $\mathbb{R}^3$. Does $f(x, y, z) = x - y + 2z$ descend to $\mathbb{R}^3/U$? Does $g(x, y, z) = x + y$? Give coordinates for the quotient, prove the first isomorphism theorem for your coordinate map, and determine the induced f.
9. **Two quotients in succession.** For $U \subseteq H \subseteq V$, construct a canonical isomorphism $(V/U)/(H/U) \to V/H$. For arbitrary subspaces U, H identify $(U + H)/H$ with a quotient of U. In each case exhibit the underlying map and compute its kernel.
10. **Sections under a change of coordinates.** For $p(x, y) = y$, classify all linear sections. Find a linear automorphism of the plane preserving p and the kernel pointwise that moves a chosen section. Explain why the induced quotient isomorphism survives this symmetry.
11. **Exactness from measurements.** Given a short exact sequence $0 \to U \xrightarrow{j} V \xrightarrow{p} W \to 0$, show that every scalar measurement on V vanishing on $j(U)$ is uniquely a measurement on W after p. Give a counterexample when exactness in the middle fails.

18.10.4 Apply the structure

12. **Balanced sources.** For the path Laplacian solve $Lp = (1, -2, 1)^T$. Identify the solution with zero mean and the one with $p_1 = 0$. Explain solvability for arbitrary source data.
13. **Boundary data.** Analyze $u \mapsto u''$ from $C^2([0, 1])$ to $C([0, 1])$. Construct a linear right inverse. Which homogeneous freedoms survive $u(0) = u(1) = 0$? What compatibility condition arises for $u'(0) = u'(1) = 0$?
14. **Invariant or merely projected?** Let H be the symmetric matrix in Session 8. Compare its compressions to the lines through e_1 and $(1, 1, 0)^T$. Determine the residual in each case for a unit trial vector and explain the difference.

18.11 Worked solutions

1. The polynomials $1-t, t$ give the readings $(1,0),(0,1)$. Vanishing at both points means divisibility by $t(t-1)$, so the kernel is its span. Subtract from p the polynomial $p(0)(1-t)+p(1)t$. The remainder belongs to the kernel. This proves spanning, and evaluation of a zero combination followed by comparison of quadratic coefficients proves independence. In basis $(1-t, t, t(t-1))$ the matrix is $(I_2 \;\; 0)$. The fiber is $2+3t+c\,t(t-1)$. Either lift can be altered by a kernel polynomial. The readings and the quotient classes of the lifts are fixed.

2. Linearity forces $c = a + b$. Conversely $(x, y) \mapsto xa + yb$ realizes the prescriptions if that equality holds. For a spanning family let $S : F^m \to V$ synthesize the generators and $Y : F^m \to W$ synthesize their prescribed images. Necessarily $Y = TS$ implies $\ker S \subseteq \ker Y$. Conversely define $T(Sc) = Yc$. Every input has this form, and any second representation differs by a vector killed by Y. Adding and scaling coefficient vectors proves linearity. Since S is onto this extension is unique.

3. Lift an image basis w_i to h_i. For any v, expand $Tv = \sum a_i w_i$. Then $v - \sum a_i h_i$ is in the kernel and can be expanded there. In a dependence among the lifted and kernel vectors, applying T forces all lift coefficients to vanish, then kernel independence forces the others to vanish. This constructs a basis. Counting a finite basis gives the integer formula. The basis argument itself uses finite combinations and also works with infinite algebraic bases if they are available.

4. The reduced right side has $\beta = -1$, $\alpha = 1$ and last entry zero. All solutions are $(1,0,-1,0)^T + sk_1 + tk_2$. The kernel basis is $(-2,1,0,0)^T, (-4,0,-1,1)^T$ and the image basis is $(1,2,3)^T, (-1,1,0)^T$. The changed reading violates $b_3 = b_1 + b_2$. Testing the original compatible reading against the column space of the RREF would wrongly reject it since its third entry is nonzero. One must transform the reading along with the equations.

5. Take the lines through $e_1, e_2, e_1 + e_2$. The sum of the first two generators lies outside their union and equals the third generator, exhibiting a nontrivial three-way relation. In F^3 the three coordinate lines sum directly because each coordinate in a zero sum vanishes separately. The plane $z = 0$ and the line through $(1,0,1)$ sum directly by the height argument in Session 5. In the usual real Euclidean geometry the latter line is oblique. Dotting its generator with e_1 gives 1.

6. Extend a basis d of $U \cap H$ separately to (d, u) in U and (d, h) in H. The concatenation (d, u, h) spans the sum. In any zero combination the combination of the h's is in both spaces, so it is a combination of d. Independence of (d, h) now kills all h coefficients, and independence of (d, u) kills the rest. Counting includes the common list once. For the planes use $d = e_3$, $u = e_2$, $h = e_1$. The addition map on pairs instead starts with four dimensions and has one-dimensional kernel spanned by $(e_3, -e_3)$. Both counts give three.

7. The image is the x-axis; the kernel is the line through $(-2,1)$. The map fixes its image, so $P^2 = P$. Its complementary projector sends (x, y) to $(-2y, y)$. Projection $(x, y) \mapsto (x, 0)$

onto the same axis sends e_2 to zero, while $Pe_2 = (2, 0)$. The different outputs reveal different directions along which the projection acts.

8. Put $J_0(x, y, z) = (x - y, z)$. Its kernel is exactly U, and it hits (u, v) at $(u, 0, v)$. Thus $[x, y, z] \mapsto (x-y, z)$ is independent of representative, has zero kernel and is onto; it is linear by its formula. This proves the required isomorphism. The descended measurement is $(u, v) \mapsto u + 2v$, forced by surjectivity. For g, the equivalent representatives 0 and $(1, 1, 0)$ would give 0 and 2, so no function on classes has those values.

9. Define $[x]_U \mapsto [x]_H$. Changing x by an element of U preserves its H-class. The map is onto, linear, and its kernel consists of the U-classes of members of H, exactly H/U. Apply the first isomorphism theorem. For the second claim send $u \in U$ to $[u]_H$ in $(U+H)/H$. Every $[u + h]_H = [u]_H$, and a vector of U maps to zero exactly in $U \cap H$. Hence $U/(U \cap H) \cong (U + H)/H$. Both isomorphisms are forced by the original quotient maps.

10. A section is determined by the image of 1, which must be $(c, 1)$, giving $s_c(y) = (cy, y)$. The shear $L(x, y) = (x + y, y)$ is invertible, fixes the axis pointwise, satisfies $pL = p$, and carries s_c to s_{c+1}. Thus the given data cannot distinguish a preferred section. On classes the shear induces the identity because $L(x, y) - (x, y) = (y, 0)$ lies in the kernel. The canonical identification of a class with its height therefore commutes with this symmetry.

11. For a functional λ set $\mu(w) = \lambda(v)$ using any $pv = w$. Surjectivity gives a lift, and two lifts differ in $\ker p = j(U)$, where λ vanishes. Using sums of lifts shows linearity and surjectivity forces uniqueness. For a failure take $U = 0$, $V = F^2$, $p(x, y) = x$, $W = F$. The functional y vanishes on $j(U)$ but distinguishes $(0, 0)$ and $(0, 1)$ which have equal p-images.

12. The first and third equations give $p_1 = p_2 + 1$ and $p_3 = p_2 + 1$. The middle equation follows. Thus $p = (1, 0, 1)^T + c\mathbf{1}$. Zero mean gives $c = -2/3$ and grounding vertex 1 gives $c = -1$. For any b the column sums require $\sum b_i = 0$. Conversely take $p_2 = 0$, $p_1 = b_1$, $p_3 = b_3$. The middle reading is $-b_1 - b_3 = b_2$. The two normalizations change no differences.

13. A section is $Sg(t) = \int_0^t (t - s)g(s)\,ds$. Differentiation gives $(Sg)'(t) = \int_0^t g(s)\,ds$ and $(Sg)'' = g$. The general solution is $Sg + a + bt$ since a zero second derivative gives a constant first derivative. For fixed zero endpoint values take $a = 0$ and $b = -Sg(1)$, leaving no freedom and no compatibility condition. For zero endpoint derivatives take $b = 0$ and require $\int_0^1 g = 0$, leaving a arbitrary. The same differential expression thus defines different operators with different domains.

14. For $q = e_1$ the compression is the scalar $q^T Hq = 2$ and $Hq - 2q = e_2$. The residual is perpendicular to the trial line. For $q = (1, 1, 0)^T/\sqrt{2}$, the compression is 3 and the residual vanishes. Only the second line is invariant. Diagonalizing a compression solves an exact problem on the trial space, while exactness of the lifted eigenpair is tested in the full space.

18.12 Session 9 — Which information survives a boundary condition?

18.12.1 Integrated mastery problem

A bending measurement on cubic polynomials returns $Tp = (p''(0), p''(1), p''(2))$. Work over $\mathbb{R}$. Before calculating a matrix, predict the number of independent readings. Then construct the matrix, its kernel, image, compatibility equation and all inputs for a compatible reading. Choose a complement and construct its projection. Identify the canonical quotient isomorphism and a short exact sequence. Finally suppose the experiment also requires $p(0) = p'(0) = p'(1) = 0$. Determine the new kernel and image. Decide which linear measurements on a cubic can be recovered from the original three readings.

18.12.2 Worked reconstruction

A cubic has an affine second derivative. Its first two values determine that function, and every affine function is a second derivative. Thus exactly two readings are independent. For $p = a + bt + ct^2 + dt^3$, the standard-coordinate matrix is

$$[T] = \begin{pmatrix} 0 & 0 & 2 & 0 \\ 0 & 0 & 2 & 6 \\ 0 & 0 & 2 & 12 \end{pmatrix}.$$

The kernel is $\mathcal{P}_1$. If the readings are (u, v, w), then $w = 2v - u$. Conversely this relation suffices: choose $c = u/2$ and $d = (v - u)/6$. The full fiber is

$$p(t) = a + bt + \tfrac{u}{2}t^2 + \tfrac{v-u}{6}t^3, \qquad a, b \in \mathbb{R}.$$

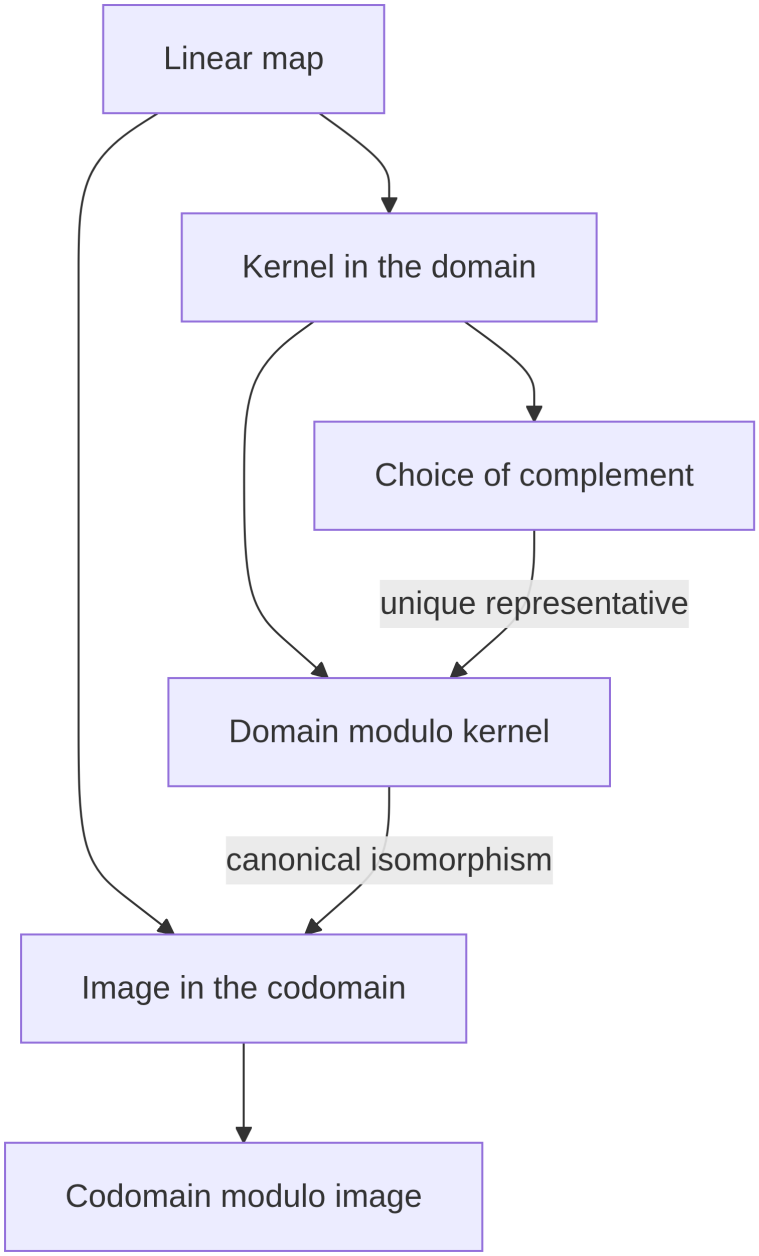
The rank–nullity count $4 = 2+2$ describes a two-dimensional ambiguity and a two-dimensional attainable plane. On the complement $\operatorname{span}(t^2, t^3)$ choose $a = b = 0$. The associated projection is $Pp = p - p(0) - p'(0)t$. It fixes the complement, annihilates $\mathcal{P}_1$ and therefore squares to itself. Using expansion about 1 would instead choose the complement of polynomials with $p(1) = p'(1) = 0$.

The canonical arrow $[p] \mapsto Tp$ takes $\mathcal{P}_3/\mathcal{P}_1$ bijectively to the plane $I = \{(u, v, 2v - u)\}$. Its inverse sends the reading to the class of the displayed cubic, independent of a, b. This realizes the exact sequence $0 \to \mathcal{P}_1 \to \mathcal{P}_3 \to I \to 0$. Surjectivity supplies a preimage, and exactness at the domain identifies precisely which preimages belong to the same class.

The first two boundary conditions choose $a = b = 0$. The third requires $2c+3d = (u+v)/2 = 0$. Thus the new domain is one-dimensional, its kernel is zero and its image is the line $(u, -u, -3u)$. For instance $p = t^2/2 - t^3/3$ gives $(1, -1, -3)$. An extra condition imposed after uniqueness was already achieved has excluded previously attainable readings.

Finally, a linear measurement f on cubics is recoverable exactly when it vanishes on 1 and t. Necessity comes from the zero reading being shared by every affine polynomial. Sufficiency follows from the universal property and $\overline{T}^{-1}$. Explicitly $f(p) = u f(t^2)/2 + (v-u) f(t^3)/6$. A functional on the attainable plane is determined uniquely. Its extension to all three-component readings can be changed by any multiple of $u - 2v + w$.

18.12.3 How the constructions fit together



In the last diagram the complement represents classes via restriction of the quotient map. The codomain quotient records the remaining obstruction. Returning to the sensors, Jx encodes a class, SJx is one lift of it, and $\ell(b)$ encodes the output obstruction. Each expression answers a different inverse question.

Construction or result	What to reconstruct	Hypotheses or choices
Synthesis and coordinates	S_B is onto for a spanning family and one-to-one for an independent family	Ordered list for a coordinate map
Exchange and dimension	Replace one surviving generator per independent vector	Finite spanning list over a field
Matrix transport	$A' = Q^{-1}AP$	Specified bases of source and target
Solvability and fibers	$b \in I$ and $T^{-1}(\{b\}) = x_p + K$	Compatible b for the fiber formula
Rank–nullity	Lift an image basis, append a kernel basis	Finite domain for integer counts
Direct sum and projection	Unique components correspond to $P^2 = P$	Algebraic; orthogonality needs more structure
Quotient and factorization	f descends exactly when $U \subseteq \ker f$	U a subspace, no dimension bound
First isomorphism	$[x] \mapsto Tx$	Any linear map, target restricted to its image
Splitting	s right-inverse to p gives $V = \ker p \oplus s(I)$	Section chosen; basis theorem over a field
Numerical rank	Count stretches above a tolerance	Inner products, scale, error model and threshold

What is canonical? Given T, its kernel, image, fibers, quotient and induced isomorphism are determined. Given subspaces in a common ambient space, their sum and intersection are determined. Given a decomposition, its component projections are determined. A kernel alone generally specifies neither a complement nor a projection. Coordinates, bases of the image, choices of representatives and matrix entries require further data. The distinction always refers to what has already been supplied.

Common error	Diagnostic
Calling every fiber a subspace	Does the proposed fiber contain the zero input?
Reading image vectors from the RREF	Which output coordinates did the row operations change?
Treating pairwise intersections as sufficient	Try isolating one summand against the sum of **all** the others.
Applying a rule to a quotient representative	Replace x by $x + u$ and check for every $u \in U$.

Common error	Diagnostic
Inferring exactness from zero composition	Can every vector killed by the next map be lifted through the preceding map?
Calling an algebraic projection orthogonal	Which inner product and which kernel determine its direction?
Equating an exact rank with a reliable numerical rank	What input perturbation would erase the smallest resolved stretch?

18.12.4 Mastery checklist

- ☐ Rebuild the exchange proof and use it to justify dimension and basis extension.
- ☐ Represent differentiation in two different ordered bases and track the direction of each coordinate conversion.
- ☐ Use elimination on a new rectangular system to produce kernel and original-image bases, a compatibility equation and the full affine fiber.
- ☐ Prove rank–nullity by both extending a kernel basis and lifting an image basis.
- ☐ Detect a failed direct sum and construct two projections with the same kernel.
- ☐ Define quotient operations, check representatives, and prove the induced isomorphism in both directions.
- ☐ Reconstruct the universal properties of kernel, quotient and finite direct sum from their mapping problems.
- ☐ Check an exact sequence term by term, construct a section and say which parts depend on choices.
- ☐ Interpret a lost direction and an unattainable output for a graph, a differential operator and a numerical trial space.
- ☐ Solve the cubic measurement problem from a blank page, then change the boundary condition and predict the new image.

18.12.5 Transition to Module B — What can be measured?

A linear measurement on W is a functional $\lambda : W \to F$. Measuring an output gives a functional on the input, $(T^\vee\lambda)(x) = \lambda(Tx)$. Thus $T^\vee : W^\vee \to V^\vee$ reverses direction. Which output measurements become invisible? Exactly those vanishing on I, the **annihilator** I°. Which input measurements can come from outputs? They must vanish on K. Conversely such a measurement descends to V/K, transfers across $\overline{T}$ to I, and in finite dimension extends to W by completing an image basis. Consequently $\ker T^\vee = I^\circ$ and $\operatorname{im} T^\vee = K^\circ$. In the sensor problem this says that every recoverable scalar quantity is a combination of the two rows of J, while ℓ is the redundant output measurement. Module B begins at that distinction.

19 Module B — Duality and Linear Measurements: From Covectors to Universal Structure

A sensor array reports Ax for an unknown input x. An engineer asks for just **one number**, $f(x)$. Can the array provide that number even when it cannot recover x? If two different formulas in the readings both give the answer, can we explain their difference? These two questions lead respectively to the image and the kernel of the dual map. We will solve them for the rectangular matrix from Module A and then recognize the same construction in graph flows, weak equations and spectral coefficients.

Conventions. Work over a field F, with finite-dimensional spaces unless indicated. $V^\vee = \mathrm{Hom}_F(V, F)$ denotes the algebraic dual, $T^\vee$ the algebraic dual map and S^0 an annihilator. A column c of functional coefficients acts by $x \mapsto c^T x$, including over $\mathbb{C}$. Superscript H denotes conjugate transpose. Reserve T^* for the inner-product adjoint. Inner products here are linear in the first variable. We use F^n for coordinate columns and write span for all finite linear combinations.

19.1 Session 1 — How much of a vector does a measurement describe?

A potential field may be stored as a function, a stress as a matrix, a displacement as a column. In each case one may ask for a scalar observation. Evaluation $p \mapsto p(1)$ on polynomials, trace $M \mapsto \sum_i M_{ii}$ on square matrices, and $f(x, y, z) = 2x - y + 3z$ on $\mathbb{R}^3$ all obey

$$f(av + bw) = af(v) + bf(w).$$

Such a map is a **linear functional**, also called a **covector**. The rule applies to the objects before choosing coordinates. Evaluation at 1, for instance, gives the same reading whether a polynomial is written in monomials or in powers of $t - 1$.

Two sensors can be combined: $(af + bg)(v) = af(v) + bg(v)$. The new rule is linear in v by distributivity. The zero functional, additive inverses and vector-space identities follow

pointwise from the field operations. Thus measurements themselves form a vector space $V^\vee$. The common operation linking the two spaces is the **evaluation pairing**

$$\langle f, v\rangle_{\text{eval}} = f(v), \qquad V^\vee \times V \to F.$$

It is linear in each argument separately, also over $\mathbb{C}$. Its arguments have different types. An inner product, introduced in Module C, takes two vectors from the same space and over $\mathbb{C}$ conjugates scalars in one argument.

19.1.1 The geometry a single reading gives you

For a nonzero f choose u with $f(u) = 1$ by rescaling a vector on which f is nonzero. Write

$$v = f(v)u + (v - f(v)u).$$

The parenthesized term lies in $\ker f$ and $Fu \cap \ker f = \{0\}$. Consequently $V = \ker f \oplus Fu$, the kernel is a hyperplane (codimension one), and $f^{-1}(c) = cu + \ker f$. In the real plane these are parallel lines, with the scalar labels recording both scale and sign. In three dimensions the level sets of $2x - y + 3z$ are parallel planes. A nonzero covector determines a kernel, but the kernel alone cannot calibrate the instrument: f and $2f$ on a real space give different readings on the same family of planes.

For a family $f_1, \dots, f_m$, equal readings on x, y mean $x - y \in \bigcap_i \ker f_i$. The family **separates points** exactly when this intersection is zero. On $\mathcal{P}_2(\mathbb{R})$, readings at $0, 1$ miss the direction $t(t-1)$. A third reading at 2 detects it. The three readings together distinguish every quadratic because a polynomial of degree at most two with three distinct roots is zero. The example distinguishes independence of instruments from sufficiency to recover an input.

19.1.2 Can all linear measurements distinguish two different vectors?

If $v \neq 0$, extend v to a basis of V and define a functional taking v to 1 and the other basis vectors to zero. Such assignments extend linearly and uniquely because basis expansions are unique. Conversely a nonzero functional takes a nonzero value somewhere, by definition. These two statements express **nondegeneracy** of evaluation on both sides. In finite dimension the extension step follows by adding independent vectors until they span. The infinite-dimensional algebraic version uses basis extension with choice.

Choose a basis $b_1, \dots, b_n$. Sending a functional to its list of readings $(f(b_1), \dots, f(b_n))$ is a linear bijection $V^\vee \to F^n$: any prescribed list extends uniquely. Hence $\dim V^\vee = n$. The proof identifies the dual with a coordinate space **after choosing the basis**. It does not yet give an intrinsic way to turn a vector into a functional. In Session 6 a change of scale will expose the missing choice.

Checkpoint. Does knowledge of the *labelled* level sets recover a functional? Yes: read off the label of the unique set containing v. Knowledge of the zero set alone identifies a nonzero functional on a real space only up to a nonzero scalar. To see this, expand v along the common kernel and one transverse vector.

19.2 Session 2 — How does an oblique grid measure coordinates?

In an ordered basis $B = (b_1, \dots, b_n)$ each vector has a unique expansion $v = \sum_j x_j b_j$. What instrument would read x_i without reporting the other coordinates?

Theorem 19.1. ***Dual basis theorem.*** *There is a unique family $(\varepsilon^1, \dots, \varepsilon^n)$ satisfying $\varepsilon^i(b_j) = \delta^i_j$. It is a basis of $V^\vee$, and*

$$v = \sum_i \varepsilon^i(v) b_i, \qquad f = \sum_i f(b_i)\varepsilon^i.$$

Proof. Define $\varepsilon^i(v) = x_i$. Uniqueness of expansion makes it well-defined, and adding or scaling expansions proves linearity. Its values on the basis are precisely the prescribed deltas, which also force uniqueness. Evaluating the proposed expansion of f at each b_j gives $f(b_j)$; agreement on a basis gives equality everywhere. A zero combination $\sum_i a_i \varepsilon^i = 0$ evaluated at b_j gives $a_j = 0$, proving independence. The first displayed expansion is the construction of the coordinate functions itself. □

Write B also for the matrix of basis columns expressed in standard coordinates. Stacking the dual basis as **rows** yields a matrix D with $DB = I$. Hence $D = B^{-1}$. If each covector is stored as a coefficient **column**, the matrix of dual basis columns is B^{-T}. The transpose here changes storage convention.

19.2.1 A grid whose axes meet at an oblique angle

Take $b_1 = (1, 0)^T$, $b_2 = (1, 1)^T$. Then

$$B = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \quad B^{-1} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}, \qquad \varepsilon^1(x, y) = x - y, \quad \varepsilon^2(x, y) = y.$$

At $v = (3, 2)^T$, the readings are $1, 2$, giving $v = b_1 + 2b_2$. The first coordinate is constant when moving along b_2 because $\varepsilon^1(b_2) = 0$. A Euclidean dot product with b_1 would give 3, the wrong coordinate. In Figure 19.1 the two labelled lines through the marked point are its two coordinate readings. Their intersection determines v. The arrows are basis **vectors**; the level sets represent **covectors** without selecting normal arrows.

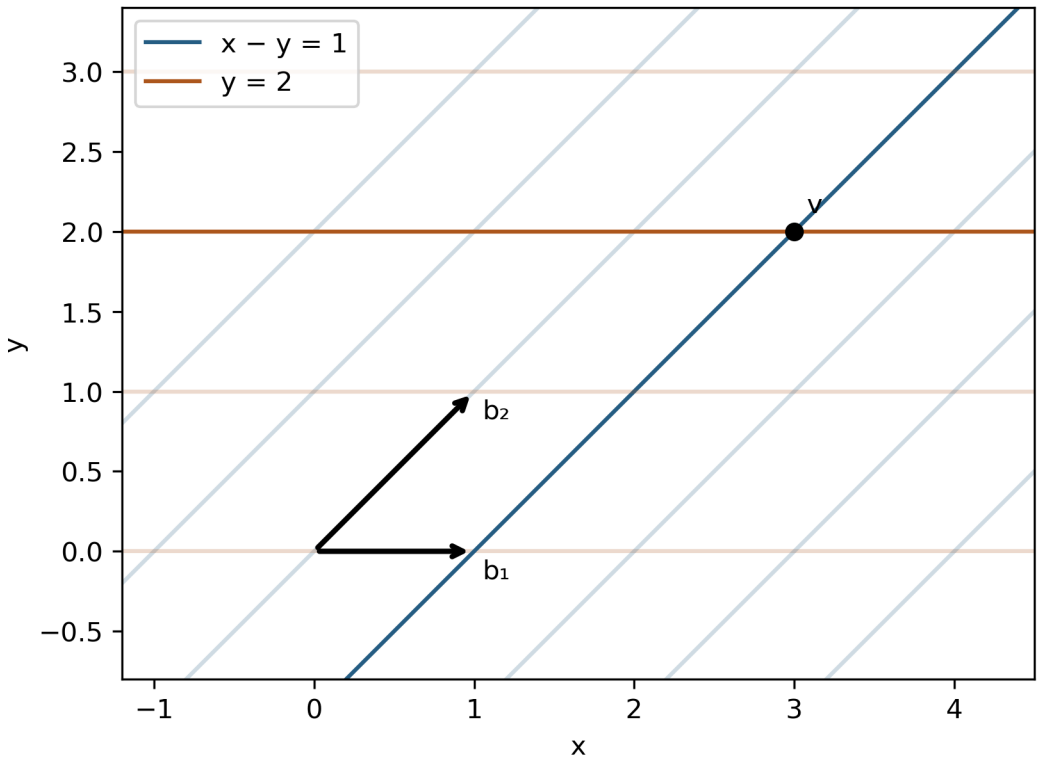


Figure 19.1: Dual coordinate readings on an oblique grid. The blue family measures $x - y$ and the orange family measures y.

19.2.2 Which quantities change oppositely?

Let $B' = BP$. The columns of P express the new basis in the old coordinates. From $Bx = B'x'$ obtain $x' = P^{-1}x$. Write $c_i = f(b_i)$ and $c'_j = f(b'_j)$. Because $b'_j = \sum_i b_i P_{ij}$, we have $c'_j = \sum_i P_{ij} c_i$, hence

$$x' = P^{-1}x, \quad c' = P^T c, \qquad (c')^T x' = c^T P P^{-1} x = c^T x.$$

For $f(x, y) = 2x - y$ in the oblique example, $c_{\text{old}} = (2, -1)^T$ in standard coordinates and $c_{\text{new}} = (2, 1)^T$ in the dual grid. Both read 4 on $(3, 2)^T$ because the vector's new coordinates are $(1, 2)^T$.

In the customary terminology, the vector's **components** transform contravariantly to the basis change and the functional's components transform covariantly. The **dual basis itself** changes inversely to the primal basis: as rows, $(BP)^{-1} = P^{-1}B^{-1}$. A separate use of *contravariant* refers to dualization reversing the direction of maps. Keeping the basis, its components, and the direction of a map distinct resolves the apparent clash of terminology.

Checkpoint. Double just b_1. The first component of a fixed vector halves, the first component of a fixed covector doubles, and the first dual basis element halves. Check all three directly by evaluation.

19.3 Session 3 — How does an output instrument become an input instrument?

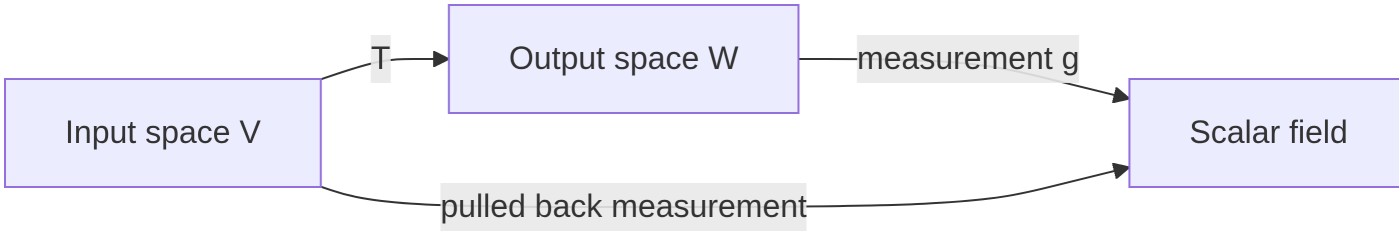
Given $T : V \to W$ and $g : W \to F$, an instrument that first applies T and then measures by g reports the scalar $g(Tv)$. Viewed from the input, it is a single linear functional. Define the **pullback**

$$T^\vee : W^\vee \to V^\vee, \qquad T^\vee g = g \circ T.$$

Linearity in v follows by composing linear maps; linearity in g follows because $(ag + bh)(Tv) = ag(Tv) + bh(Tv)$. Its defining identity is

$$\langle T^\vee g, v \rangle_{\text{eval}} = \langle g, Tv \rangle_{\text{eval}}.$$

This determines $T^\vee g$ at every v. The reversal of direction expresses where the instrument can now be used. It requires neither an inverse for T nor an inner product.



For $V \xrightarrow{T} W \xrightarrow{S} Z$ take $h \in Z^\vee$. At v, both $((ST)^\vee h)(v)$ and $(T^\vee S^\vee h)(v)$ equal $h(S(Tv))$. Thus $(ST)^\vee = T^\vee S^\vee$ and $\mathrm{id}_V^\vee = \mathrm{id}_{V^\vee}$. If T is invertible these identities applied to $T^{-1}T$ and TT^{-1} give $(T^\vee)^{-1} = (T^{-1})^\vee$. If $i : S \hookrightarrow V$ is inclusion, $i^\vee f$ is f restricted to S. For $q : V \to V/S$, $q^\vee g$ measures v by measuring its class $[v]$.

19.3.1 Deriving the transpose

Let $A = [T]_{C \leftarrow B}$, so $Tb_j = \sum_i A_{ij} d_i$ in the target basis $C = (d_i)$. With g represented by $c_i = g(d_i)$, the jth coefficient of its pullback in $B^\vee$ is

$$(T^\vee g)(b_j) = \sum_i A_{ij} c_i = (A^T c)_j.$$

Hence $[T^\vee]_{B^\vee \leftarrow C^\vee} = A^T$. It has shape $n \times m$ for an $m \times n$ primal matrix. For every coordinate column x, $(A^T c)^T x = c^T A x$. A quick consistency check: if $A' = Q^{-1} A P$, then $(A')^T = P^T A^T Q^{-T}$, exactly the change of coordinates for the two dual spaces.

Return to Module A's real sensor matrix

$$A = \begin{pmatrix} 1 & 2 & -1 & 3 \\ 2 & 4 & 1 & 9 \\ 3 & 6 & 0 & 12 \end{pmatrix}.$$

Suppose the instrument reports $g(b) = b_2 - 2b_1$. Its pulled-back row is row 2 minus twice row 1, namely $(0, 0, 3, 3)$. Thus $x_3 + x_4$ is recoverable by dividing the reported scalar by three, even if neither setting is recoverable individually. Conversely $n = (-1, -1, 1)^T$ represents an output functional whose pullback vanishes: row 3 is the sum of the first two. Adding any multiple of n to an output formula gives the same input reading. Session 4 will show that this accounts for **every** ambiguity in such a formula.

Over $\mathbb{C}$, take $Tz = iz$ and $g(z) = z$. Then $(T^\vee g)(z) = iz$, while the Euclidean Hermitian adjoint acts by $-i$. Replacing the transpose by conjugate transpose in the algebraic pullback would give a wrong value already at $z = 1$. Session 9 will derive the adjoint by specifying the additional map that represents functionals as vectors.

Checkpoint. Which source and target belong to the transpose of a 3×4 sensing matrix? An input to the transpose consists of three **weights on readings**; its four outputs are coefficients of a measurement on the settings.

19.4 Session 4 — Which questions about an unknown input can the data answer?

For a subspace $S \subseteq V$, define its **annihilator**

$$S^0 = \{f \in V^\vee : f(s) = 0 \text{ for every } s \in S\}.$$

If $f \in S^0$, moving the input by any $s \in S$ leaves the reading unchanged. Conversely constancy along every such direction implies $f(s) = f(0) = 0$. Since linear combinations of measurements vanishing on S also vanish there, S^0 is a subspace of the **dual**. It suffices to test a spanning list. If its vectors form the columns of D, the coefficient columns of S^0 solve $D^T c = 0$.

Dimension and order. Extend a basis $s_1, \dots, s_k$ of S to $s_1, \dots, s_n$ of V. In the dual expansion of f, the first k coefficients $f(s_i)$ vanish precisely for $f \in S^0$. The remaining $n - k$ dual basis vectors give a basis of S^0, proving $\dim S^0 = n - k$. For $S \subseteq U$, every functional zero on U is zero on S, so $U^0 \subseteq S^0$. Larger invisible subspaces impose more constraints on possible instruments.

Theorem 19.2. ***Recovery and redundancy.*** *For a linear $T : V \to W$ between finite-dimensional spaces,*

$$\ker T^\vee = (\operatorname{im} T)^0, \qquad \operatorname{im} T^\vee = (\ker T)^0.$$

Proof. A functional g on W has zero pullback exactly when $g(Tv) = 0$ for every v, that is, when it vanishes on the image. This equality holds in any dimension.

If $f = T^\vee g$, then $f(k) = g(0) = 0$ for $k \in \ker T$. For the converse, suppose f kills $\ker T$. Prescribe a measurement on the attainable outputs by $g_0(Tv) = f(v)$. This rule is well-defined because two inputs yielding the same output differ by a kernel vector. Adding and scaling preimages proves linearity on $\operatorname{im} T$. To obtain an instrument on all of W, extend an image basis to a basis of W, keep the values of g_0 on the image basis, and assign zero to the added vectors. The resulting linear g satisfies $T^\vee g = f$. All possible extensions agree on the image; their differences form exactly $(\operatorname{im} T)^0$. □

The second proof has two distinct steps. **Recovery on actual data is unique.** Extrapolation to unattainable data can involve a choice. Over an arbitrary algebraic vector space basis extension gives the same result under the usual choice assumption. For continuous measurements, boundedness of the recovered rule is a further question.

19.4.1 Computing all four spaces in the sensor problem

Solving $Ax = 0$ gives $x_3 = -x_4$, $x_1 = -2x_2 - 4x_4$. Write $K = \ker A$, $I = \operatorname{im} A$. Let $r = (1, 2, 0, 4)$ and $s = (0, 0, 1, 1)$ be measurement rows. In ordinary coordinates we have the exact factorization

$$A = \begin{pmatrix} 1 & -1 \\ 2 & 1 \\ 3 & 0 \end{pmatrix} \begin{pmatrix} r \\ s \end{pmatrix}.$$

The first matrix has independent columns; the second hits both coordinate axes. Hence the rank is two. A recipe for measuring rx is $b_3/3$, and for sx it is $(b_2 - 2b_1)/3$.

Space	Ambient space	Basis given in standard coordinates
K	inputs $\mathbb{R}^4$	$(-2,1,0,0)^T$, $(-4,0,-1,1)^T$
I	outputs $\mathbb{R}^3$	$(1,2,3)^T$, $(-1,1,0)^T$
$\ker A^\vee = I^0$	output dual	the row $(-1,-1,1)$
$\operatorname{im} A^\vee = K^0$	input dual	the rows r, s

To check the last line without a dimension argument, write an input measurement as (a,b,c,d). Vanishing on both kernel generators gives $b = 2a$ and $d = 4a + c$. Thus its row is precisely $ar+cs$. To check the third line, impose zero readings on the two image columns. The equations $g_1+2g_2+3g_3 = 0$ and $-g_1+g_2 = 0$ give exactly the span shown. This is the algebraic meaning of row space and left null space. In general $\operatorname{rank} T^\vee = \dim V - \dim\ker T = \operatorname{rank} T$. It follows that T is injective iff $T^\vee$ is surjective, and T is surjective iff $T^\vee$ is injective.

19.4.2 A compatibility test that is also sufficient

If $Tx = b$ is solvable then $g(b) = 0$ for every $g \in (\operatorname{im} T)^0$. Conversely, if b lies outside the image, append b to an image basis and extend to a basis of W. Assign 1 to b and 0 to every other basis element. This constructs a witness g killing the image but detecting b. Consequently

$$Ax = b \text{ solvable} \quad \Longleftrightarrow \quad c^T b = 0 \quad \text{for every } c \in \ker A^T.$$

The formula uses linear functional coefficients over any field. For the sensors take $b = (2,1,3)^T$. The witness n gives zero; the recovered coordinates are $rx = 1$, $sx = -1$. One solution is $(1,0,-1,0)^T$, and adding any member of K gives all others. Changing b_3 to 4 yields $n^T b = 1$, a certificate of impossibility.

Checkpoint. Can you recover x_3 by designing a more complicated, possibly nonlinear, function of Ax? The inputs 0 and $(-4,0,-1,1)^T$ give identical data and different answers. Even a nonlinear rule cannot distinguish them. Linearity of a *recoverable linear quantity* ensures there is a linear rule on the image.

19.5 Session 5 — Which instruments survive identification, and which survive restriction?

Let $S \subseteq V$. A quotient vector $[v] = v + S$ represents a whole family of inputs; $[v] = [w]$ iff $v - w \in S$. Its linear operations are inherited from representatives. One may lose access to v by storing only $[v]$. Alternatively one may retain ordinary inputs but perform experiments only on S. These procedures pose different measurement problems.

Theorem 19.3. ***Descent and restriction.*** *The quotient map $q : V \to V/S$ induces a canonical isomorphism $(V/S)^\vee \cong S^0$ via pullback. For finite-dimensional V, the inclusion $i : S \hookrightarrow V$ induces a canonical isomorphism $V^\vee/S^0 \cong S^\vee$ via restriction.*

Proof. $g \mapsto gq$ takes quotient measurements into S^0 and is injective since q is onto. For $f \in S^0$ define $g([v]) = f(v)$. If $w = v + s$, then $f(w) = f(v)$, so this is well-defined. Linearity follows from quotient operations. These recipes are mutual inverses and use no lift or complement.

Restriction $i^\vee f = f|_S$ has kernel S^0 by definition. Extend any functional on S by prescribing its values on an extended basis of V. Thus restriction is onto. Two extensions have the same restriction precisely when they differ in S^0. The first isomorphism theorem now gives the second canonical isomorphism $[f] \mapsto f|_S$. The class of extensions is forced even though a particular extension may vary. □

19.5.1 See both statements in the same three coordinates

Let $S = \operatorname{span}(1, 1, 0)$ and $Q(x, y, z) = (x - y, z)$. The map Q is onto, with kernel S, so it supplies coordinates (u, w) on V/S. Pulling back $\alpha u + \beta w$ gives the row $(\alpha, -\alpha, \beta)$ on V. This lists all measurements that survive forgetting displacement along S. For the second problem restrict (a, b, c) to $(t, t, 0)$ and obtain $(a + b)t$. The rows with prescribed $a + b$ form an entire coset of S^0. For example $(2, 0, 0)$ and $(1, 1, 3)$ both restrict to $2t$. Neither of those two rows descends to V/S.

19.5.2 What happens to a short exact sequence?

Exactness at the middle of a pair of consecutive arrows means that the incoming image equals the outgoing kernel. In

$$0 \longrightarrow S \xrightarrow{i} V \xrightarrow{q} V/S \longrightarrow 0$$

the endpoints express injectivity and surjectivity. The two parts of Theorem 19.3 jointly prove exactness of the sequence with reversed arrows:

$$0 \longrightarrow (V/S)^\vee \xrightarrow{q^\vee} V^\vee \xrightarrow{i^\vee} S^\vee \longrightarrow 0.$$

At the first dual space $q^\vee$ is injective; at $V^\vee$ its image is S^0, exactly the kernel of restriction; at $S^\vee$ extension gives surjectivity. Any short exact sequence of vector spaces reduces to this one by identifying the first space with its image and the last with the quotient. Returning to the sensors, dualizing $V/K \cong I$ identifies $I^\vee$ with the recoverable measurements K^0. Passing from $I^\vee$ to formulas on the full measurement apparatus W amounts to extending a functional.

Analytic boundary. In a normed space write V' for the continuous dual, consisting of those linear f with $|f(v)| \le C\|v\|$ for some C. In finite dimension all linear functionals are

continuous: in coordinates the bound follows from a finite sum and equivalence of norms. For a closed subspace S of a normed space the quotient norm is $\|[v]\| = \inf_{s\in S} \|v+s\|$. A bounded f killing S satisfies $|f(v)| = |f(v+s)| \le \|f\|\|v+s\|$. Taking the infimum proves continuity of the descended functional, even preservation of its norm since q is contractive. A nonclosed S would give only a seminorm. Extending a **bounded** scalar functional from S with its norm preserved is the Hahn–Banach theorem, a stronger assertion than algebraic basis extension. A continuous functional vanishing on a set vanishes on its closure. Session 8 gives a concrete range for which this matters.

Checkpoint. Which arrow gives a unique object on going backwards? A functional killing S gives one functional on V/S. A functional on S gives one **class** in $V^\vee/S^0$, usually containing many extensions.

19.6 Session 6 — Can a vector be recovered from all its possible readings?

Treat v itself as an instrument on instruments:

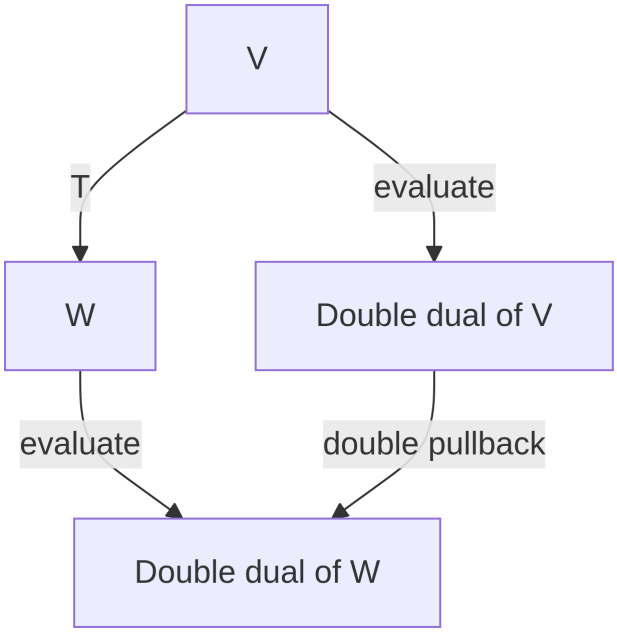
$$\iota_V : V \to V^{\vee\vee}, \qquad \iota_V(v)(f) = f(v).$$

For fixed v this is linear in f by pointwise operations. For variable v it defines a linear map because every f is linear. If $\iota_V(v) = 0$, every functional vanishes on v, so nondegeneracy gives $v = 0$.

Theorem 19.4. ***Double dual and naturality.*** *In finite dimension ι_V is an isomorphism. For any linear $T : V \to W$, one has $T^{\vee\vee}\iota_V = \iota_W T$.*

Proof. An injective map between spaces of the same finite dimension is onto, and the dual basis theorem twice gives $\dim V^{\vee\vee} = \dim V$. For a constructive inverse, given $\Phi \in V^{\vee\vee}$ take $v = \sum_i \Phi(\varepsilon^i) b_i$. Expanding f in the dual basis gives $f(v) = \sum_i f(b_i)\Phi(\varepsilon^i) = \Phi(f)$. Uniqueness of the representing vector shows its value is independent of the basis used to find it.

For naturality test both paths on an arbitrary $g \in W^\vee$. The first gives $(T^{\vee\vee}\iota_V(v))(g) = \iota_V(v)(T^\vee g) = g(Tv)$. The second gives $\iota_W(Tv)(g) = g(Tv)$. Agreement on every g proves the identity. This part has no dimension hypothesis. □



Here *natural* means that evaluating vectors on instruments commutes with **every** linear map between spaces, as the square asserts. Since one pullback reverses composition and preserves identities, dualization is called a **contravariant functor**. Two pullbacks restore the original direction. The family of evaluation maps is a **natural transformation** from the identity functor to the double-dual functor and a natural isomorphism in finite dimension. These terms in Module F2 encode the identities just proved.

What goes wrong one dual earlier? A basis B does define $J_B : V \to V^\vee$ by $b_i \mapsto \varepsilon^i$, extended linearly. In a real line with basis e, $J_e(e)(e) = 1$. Use $2e$ as basis instead. Its dual basis functional takes e to $1/2$, hence $J_{2e}(e)(e) = 1/4$. More intrinsically, a basis-free identification compatible with all changes of basis would give a nondegenerate bilinear form invariant under every automorphism. In particular over $\mathbb{R}$ invariance under $v \mapsto 2v$ would force $h(v, w) = h(2v, 2w) = 4h(v, w)$, hence $h = 0$. This illustrates why equal dimension alone supplies no preferred identification. A specified inner product supplies extra structure and changes which symmetries must be respected.

19.6.1 Can invisible directions be recovered from invisible measurements?

For $S \subseteq V$, S^{00} is first a subspace of $V^{\vee\vee}$. For $s \in S$ and $f \in S^0$, $\iota_V(s)(f) = 0$, so $\iota_V(S) \subseteq S^{00}$. In finite dimension the annihilator dimension formula gives $\dim S^{00} = n - (n - \dim S) = \dim S$, proving equality. In the original V, the vectors killed by every member of S^0 are precisely S: a vector outside S can be separated by extending a basis, as in the compatibility proof. This latter separation statement does not assert that every bidual element is evaluation in infinite dimension.

For subspaces $U, H \subseteq V$, a functional kills $U + H$ iff it kills each separately. Thus $(U+H)^0 = U^0 \cap H^0$. Also $U^0 + H^0 \subseteq (U \cap H)^0$. To prove equality in finite dimension, put $n = \dim V$. The dimension of the left side is

$$(n - \dim U) + (n - \dim H) - (n - \dim(U + H)) = n - \dim(U \cap H),$$

using the sum dimension formula twice. The containing space has the same dimension. For the coordinate axes in $\mathbb{R}^2$, the two annihilators are the complementary coordinate measurements. Killing both axes forces the zero measurement, while killing only their intersection imposes no constraint.

Checkpoint. In the double-dual square, what are the inputs of $T^{\vee\vee}$, $\iota_V(v)$ and $T^{\vee}g$? They are respectively a functional on $V^{\vee}$, a functional on V, and a vector of V. Type-checking the evaluation proof catches an attempted reversal of either arrow.

19.7 Session 7 — Who reads the coefficient of an eigenvector?

If a subspace is preserved by evolution, what happens to the instruments that ignore it? Suppose $T : V \to V$ and $T(U) \subseteq U$. For $f \in U^0$ and $u \in U$ we have $(T^{\vee} f)(u) = f(Tu) = 0$. Thus U^0 **is invariant under** $T^{\vee}$. In finite dimension the converse holds: if every $f \in U^0$ kills Tu for each $u \in U$, then $Tu \in U$ by separation.

There are two smaller evolutions. Restriction gives $T_U : U \to U$. The quotient operator $\widehat{T}[v] = [Tv]$ on V/U is well-defined because a change of representative by u changes the output by $Tu \in U$. From $qT = \widehat{T}q$, dualizing gives $T^{\vee}q^{\vee} = q^{\vee}\widehat{T}^{\vee}$. Since $q^{\vee}$ identifies the quotient dual with U^0, the dual quotient evolution is exactly $T^{\vee}|_{U^0}$ under this identification. Similarly $Ti = iT_U$ yields $i^{\vee}T^{\vee} = T_U^{\vee}i^{\vee}$: restricting a pulled-back instrument is the same as pulling back its restriction. In an adapted primal basis T has blocks $\left(\begin{smallmatrix} A & B \\ 0 & C \end{smallmatrix}\right)$. The dual coordinates killing U have first block zero, and transposition acts on their last block by C^T. This shows the quotient evolution directly in coordinates.

19.7.1 A basis of modes brings its own instruments

Assume now that T has an eigenbasis v_i over F with $Tv_i = \lambda_i v_i$. Let ℓ_i be its dual basis. Then

$$x = \sum_i \ell_i(x) v_i, \qquad (T^{\vee}\ell_i)(v_j) = \lambda_j \delta_{ij} = \lambda_i \ell_i(v_j).$$

Agreement on all basis vectors proves $T^{\vee}\ell_i = \lambda_i \ell_i$. These are **left eigenfunctionals**. Define $P_i x = v_i \ell_i(x)$. Biorthogonality gives

$$P_i P_j x = v_i \ell_i(v_j) \ell_j(x) = \delta_{ij} P_i x, \quad \sum_i P_i = I, \quad T = \sum_i \lambda_i P_i.$$

The last two identities follow by expanding x and Tx. If $X = (v_1 \ \cdots \ v_n)$ in fixed coordinates, the rows of X^{-1} are the ℓ_i. Thus $A = X\Lambda X^{-1}$ means **measure the components, scale each, then reconstruct**. Scaling an eigenvector inversely scales its dual functional, leaving P_i fixed. Within a repeated eigenspace a change of basis can alter its individual rank-one projectors. Their sum, the projector onto the entire eigenspace along all the other eigenspaces, is intrinsic to a diagonalizable operator.

For a polynomial p, repeated multiplication gives $p(T) = \sum_i p(\lambda_i)P_i$. The same measurement coefficients appear in a polynomial filter. Hermitian operators have an orthonormal eigenbasis, so the rows can then be written v_i^H and the projectors are orthogonal. An arbitrary diagonalizable matrix need not have that geometry. A matrix without a full eigenbasis does not admit the displayed rank-one eigenbasis expansion.

19.7.2 Two eigenlines close together

Take $v_1 = (1, 0)^T$, $v_2 = (1, 1)^T$, with eigenvalues $1, 3$. Then

$$A = X \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix} X^{-1} = \begin{pmatrix} 1 & 2 \\ 0 & 3 \end{pmatrix}, \quad P_1 = \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix}, \quad P_2 = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}.$$

The first coordinate in this eigenbasis is $x - y$, from Session 2. Hence $P_1(3, 2)^T = (1, 0)^T$. The displacement from the input to the projected point is parallel to the second eigenline. Orthogonal projection onto the first line gives $(3, 0)^T$ instead. Neither P_i is symmetric. Computing the $(1, 1)$ entries of AA^T and A^TA gives 5 and 1, respectively, so A is nonnormal.

Replace the second eigenvector by $(1, \delta)^T$ with $\delta > 0$. Now

$$X_\delta^{-1} = \begin{pmatrix} 1 & -1/\delta \\ 0 & 1/\delta \end{pmatrix}, \qquad \ell_1(x, y) = x - y/\delta, \quad \ell_2(x, y) = y/\delta.$$

A vertical input error η changes the two spectral coefficients by $-\eta/\delta$ and $+\eta/\delta$. The exact scalar responses can be large and cancel on reconstruction. This is sensitivity of the mathematical measurements themselves; roundoff can add further loss. Module E's bounds on polynomial filtering must account for the conditioning of a nonorthogonal eigenbasis. Module D's orthonormal Hermitian case has unitary X, which preserves Euclidean lengths.

Checkpoint. For $\delta = 1$ find a polynomial producing P_1 exactly. The degree-one polynomial $p(t) = (3-t)/2$ takes 1 to 1 and 3 to 0. Applying $p(A)$ reproduces the same oblique projector.

19.8 Session 8 — Where does a transpose arise without a dot product?

19.8.1 A graph trades edge instruments for vertex instruments

Orient the path $1 \to 2 \to 3$. Its incidence matrix with edges as rows is the discrete gradient D_0; add the edge $3 \to 1$ to obtain D:

$$D_0 = \begin{pmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix}, \qquad D = \begin{pmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \\ 1 & 0 & -1 \end{pmatrix}.$$

Vertex potentials p produce edge differences Dp. Adding a constant to all potentials changes none. Conversely, zero differences force equality along edges, hence constancy on each connected component. Edge weights $h = (a, b, c)^T$ define the functional $d \mapsto h^T d$. Evaluate:

$$a(p_2 - p_1) + b(p_3 - p_2) + c(p_1 - p_3) = (c - a)p_1 + (a - b)p_2 + (b - c)p_3.$$

The collected coefficients are $D^T h$. They measure **inflow minus outflow** at each vertex for the oriented flow h. If divergence is defined as outflow minus inflow, it is $-D^T h$ in these coordinates. The transpose arose by composition alone. A weighted inner-product adjoint would additionally depend on vertex and edge weights.

The circulation $(1, 1, 1)^T$ pulls back to zero and spans $\ker D^T$. Hence edge differences must sum to zero around the triangle. Conversely any zero-sum d arises: choose $p_1 = 0$, $p_2 = d_1$, $p_3 = d_1 + d_2$, and the closing difference is d_3. On the path there is no cycle constraint. On the triangle each vertex measurement pulled back from edges has weights summing to zero, exactly the measurements insensitive to constant potential shifts. For example $p_3 - p_1$ is the sum of the first two readings or the negative of the third. The two edge recipes differ by a circulation. This is the recovery theorem in graph coordinates. In the language of Module F2 an edge has a signed boundary at its endpoints; the boundary of a cycle is zero. Pairing an edge chain with a potential difference yields exactly the pairing of its boundary with the potential.

19.8.2 A weak equation produces a functional before it produces a vector

Suppose $a : U \times V \to \mathbb{R}$ is bilinear and the desired u must satisfy $a(u, v) = L(v)$ for every test v. Fixing u leaves a linear rule on tests. Thus the appropriate operator is $\mathcal{A} : U \to V^\vee$ with $(\mathcal{A}u)(v) = a(u, v)$; linearity in the first slot makes $\mathcal{A}$ linear. For trial basis ϕ_j and test basis ψ_i, expansion $u = \sum_j u_j \phi_j$ gives

$$K_{ij} = a(\phi_j, \psi_i), \quad b_i = L(\psi_i), \quad Ku = b.$$

The row index specifies the **test**. The result Ku lists readings of a functional on test basis vectors. Under trial basis change P and test basis change Q, this gives $K' = Q^T K P$, $u' = P^{-1}u$, $b' = Q^T b$. For a symmetric form and equal bases this is congruence $P^T K P$. A map from vectors to vectors would use a different target coordinate change.

For a concrete finite element calculation take $-u'' = 1$ on $[0,1]$ with $u(0) = u(1) = 0$. Multiplying by an endpoint-zero test and integrating by parts gives $\int u'v' = \int v$. Choose one trial and test hat ϕ with value 1 at $1/2$, zero at the endpoints, and constant slopes $2, -2$ on the half intervals. Its stiffness is $\int(\phi')^2 = 4$ and the load is $\int \phi = 1/2$. The discrete equation is $4c = 1/2$, so $u_h = \phi/8$. The residual functional $r(v) = \int v - \int u_h' v'$ vanishes on the chosen test span. On $v = t(1-t)$ it has value $1/6 - 1/8 = 1/24$, since $\int \phi' v' = 1$. Vanishing on finitely many tests has not solved the equation on all tests. Halving or doubling the hat changes load and coefficients by the coordinate rules above, while u_h stays fixed.

For the infinite test space the relevant target is a **continuous dual**. For example Cauchy–Schwarz gives $|\int u'v'| \le \|u'\|_2 \|v'\|_2$, so fixing u yields a bounded functional in a norm controlling $\|v'\|_2$. Completing suitable endpoint-zero functions gives the energy space used in Module F1. The finite matrix came from restricting both trial and test spaces. Stability and convergence as those spaces change require additional analysis.

19.8.3 A derivative is a measurement of a perturbation

Let $J(y_1, y_2) = y_1^2 + y_1 y_2$. Expanding $J(y+h)$ gives the linear part $dJ_y(h) = (2y_1 + y_2)h_1 + y_1 h_2$ and a quadratic remainder $h_1^2 + h_1 h_2$. At $(1,2)$ the differential has row $(4,1)$. If $y = Lx$ with $L = \left(\begin{smallmatrix} 1 & 2 & 0 \\ 0 & 1 & 1 \end{smallmatrix}\right)$, then $d(J \circ L)_x = dJ_{Lx} \circ L$. At a point mapping to $(1,2)$ the input sensitivity has row $(4, 9, 1)$. This is the chain rule as a pullback. Reverse-mode differentiation composes such pullbacks starting at a scalar output. A gradient **vector** appears only after representing the differential with a chosen inner product.

A related numerical identity applies when $Au = b$ and a computed $\tilde{u}$ has residual $r = b - A\tilde{u}$. If the quantity of interest has coefficient column c and $A^T z = c$, then $c^T(u - \tilde{u}) = z^T r$. This converts the residual into an exact output error for that scalar. Existence of such a z is precisely $c \in (\ker A)^0$, the same recovery condition encountered with the sensors.

19.8.4 Where the finite-dimensional argument stops

Consider $T : \ell^2(\mathbb{R}) \to \ell^2(\mathbb{R})$, $(Tx)_n = x_n/n$, on square-summable real sequences. This is bounded and injective. Its range contains all finitely supported sequences, so is dense by truncation, but misses $(1/n)$, whose only possible preimage is the nonsummable constant sequence. Every **continuous** measurement annihilating the range therefore vanishes everywhere and cannot certify that this particular datum is unattainable.

In fact $f(x) = \sum_n x_n/n$ is a bounded input functional by Cauchy–Schwarz, killing the zero kernel. If $f = g \circ T$ for a bounded g, testing on the unit sequence e_n forces $g(e_n) = 1$ for all n. On $y_N = \sum_{n=1}^{N} e_n/\sqrt{N}$, we would have $\|y_N\| = 1$ and $g(y_N) = \sqrt{N}$. Impossible. The recovered rule on the dense image is algebraically well-defined but unbounded. For Hilbert-space bounded operators the adjoint identity involves a closure, $\overline{\operatorname{im} T^*} = (\ker T)^{\perp}$. Equality without the closure needs closed range. This is one reason Module F1 distinguishes exact solvability from approximation.

Checkpoint. In the graph, finite element and sensitivity examples identify what the rows index. They index edges, tests and output components respectively. In each case a linear combination of rows describes what one can measure on the inputs.

19.9 Computational laboratory — Seven experiments in measuring and reconstructing

Use Python 3 with SymPy, NumPy and Matplotlib in the Jupyter environment used by Quarto. Run cells in document order. Algebraic certificates use exact arithmetic. The floating-point experiment at the end measures sensitivity in Euclidean coordinates.

19.9.1 A. Calibrate three instruments on a nonorthogonal basis

```
import sympy as sp

B = sp.Matrix([[1, 1, 0], [0, 1, 1], [0, 0, 1]])
dual_rows = B.inv()
print("Dual instruments by rows:", dual_rows.tolist())
print("Readings on the three basis vectors:", (dual_rows*B).tolist())
assert dual_rows*B == sp.eye(3)
```

```
Dual instruments by rows: [[1, -1, 1], [0, 1, -1], [0, 0, 1]]
Readings on the three basis vectors: [[1, 0, 0], [0, 1, 0], [0, 0, 1]]
```

The first row measures $x - y + z$, the second $y - z$, the third z. A zero off the diagonal means that moving along the corresponding other basis direction changes no reading of that instrument.

19.9.2 B. Change coordinates while preserving every evaluation

```
v = sp.Matrix(sp.symbols("x y z"))
c = sp.Matrix(sp.symbols("a b c"))
v_grid, c_grid = B.inv()*v, B.T*c
assert sp.expand(c_grid.dot(v_grid) - c.dot(v)) == 0
v_num, c_num = sp.Matrix([4, 2, 1]), sp.Matrix([2, -1, 3])
print("New vector coordinates:", list(B.inv()*v_num))
print("New covector coefficients:", list(B.T*c_num))
print("Old and new readings:", c_num.dot(v_num),
      (B.T*c_num).dot(B.inv()*v_num))
```

```
New vector coordinates: [3, 1, 1]
New covector coefficients: [2, 1, 2]
Old and new readings: 9 9
```

The printed values $9, 9$ check one case; the symbolic identity covers all vectors and instruments for this basis. Dot here denotes multiplication and summation of coefficient entries over the reals. The invariant is evaluation, independent of the basis.

19.9.3 C. Pull a three-channel instrument back to four settings

```
A = sp.Matrix([[1, 2, -1, 3], [2, 4, 1, 9], [3, 6, 0, 12]])
g = sp.Matrix([-2, 1, 0])
x = sp.Matrix(sp.symbols("x1:5"))
pullback = A.T*g
print("Coefficients on input settings:", list(pullback))
assert sp.expand(g.dot(A*x) - pullback.dot(x)) == 0
```

```
Coefficients on input settings: [0, 0, 3, 3]
```

Three output weights become four input weights. The identity checks that the two instruments agree on every settings vector, even when distinct settings give identical data.

19.9.4 D. Test both annihilator equalities as equalities of spaces

```
def basis_matrix(columns):
    return sp.Matrix.hstack(*columns)

K = basis_matrix(A.nullspace())
I = basis_matrix(A.columnspace())
left_kernel = basis_matrix(A.T.nullspace())
pullback_image = basis_matrix(A.T.columnspace())
image_ann = basis_matrix(I.T.nullspace())
kernel_ann = basis_matrix(K.T.nullspace())

def same_span(B1, B2):
    return B1.rank() == B2.rank() == B1.row_join(B2).rank()

assert same_span(image_ann, left_kernel)
assert same_span(kernel_ann, pullback_image)
assert left_kernel.T*I == sp.zeros(left_kernel.cols, I.cols)
assert pullback_image.T*K == sp.zeros(pullback_image.cols, K.cols)
print("Output obstructions:", image_ann.T.tolist())
print("Recoverable input instruments:", kernel_ann.T.tolist())
```

```
Output obstructions: [[-1, -1, 1]]
Recoverable input instruments: [[-1/4, -1/2, 1, 0], [1/4, 1/2, 0, 1]]
```

Zero products check containment. Checking ranks of both bases and their concatenation proves equality even when the returned bases differ. This is stronger than testing one candidate functional.

19.9.5 E. Calibrate a quotient instrument and change the representative

```
Q = sp.Matrix([[1, -1, 0], [0, 0, 1]])
direction = sp.Matrix([1, 1, 0])
alpha, beta, t = sp.symbols("alpha beta t")
weights = sp.Matrix([alpha, beta])
lifted = Q.T*weights
assert Q.rank() == 2 and Q.nullspace() == [direction]
assert lifted.dot(direction) == 0
assert sp.expand(lifted.dot(v+t*direction) - weights.dot(Q*v)) == 0
candidate = sp.Matrix(sp.symbols("d1:4"))
print("Pulled-back row:", list(lifted))
```

```
print("Necessary and sufficient vanishing condition:",
      sp.Eq(candidate.dot(direction), 0))
```

```
Pulled-back row: [alpha, -alpha, beta]
Necessary and sufficient vanishing condition: Eq(d1 + d2, 0)
```

Every row (d_1, d_2, d_3) satisfying the printed condition arises uniquely by taking $\alpha = d_1$, $\beta = d_3$. Thus the parameterization is exhaustive, and every parameter choice gives a functional independent of the coset representative.

19.9.6 F. Detect a circulation and reconstruct a vertex measurement

```
D = sp.Matrix([[-1, 1, 0], [0, -1, 1], [1, 0, -1]])
h = sp.Matrix([2, 3, -1])
p = sp.Matrix([1, 4, 2])
ones = sp.ones(3, 1)
assert D*ones == sp.zeros(3, 1) and D.T*ones == sp.zeros(3, 1)
assert D.rank() == 2
assert h.dot(D*p) == (D.T*h).dot(p)
assert D.T*(h+t*ones) == D.T*h
print("Edge differences:", list(D*p))
print("Vertex weights:", list(D.T*h), "sum:", sum(D.T*h))
print("Scalar reading by either route:", h.dot(D*p))
```

```
Edge differences: [3, -2, -1]
Vertex weights: [-3, -1, 4] sum: 0
Scalar reading by either route: 1
```

Zero total vertex weight makes the instrument insensitive to a constant shift. Adding a unit circulation to the edge weights changes the instrument on arbitrary edge data but agrees on every gradient. The equalities distinguish primal ambiguity from dual redundancy.

19.9.7 G. Compare an exact projector with sensitive spectral coordinates

```
H = sp.Matrix([[1, 2], [0, 3]])
values = [1, 3]
modes = [(H-lam*sp.eye(2)).nullspace()[0] for lam in values]
X = basis_matrix(modes)
L = X.inv()
projectors = [X[:, i]*L[i, :] for i in range(2)]
assert L*H == sp.diag(*values)*L
assert all(P*P == P for P in projectors)
P1, P2 = projectors
assert P1*P2 == P2*P1 == sp.zeros(2)
assert P1+P2 == sp.eye(2) and P1+3*P2 == H
assert P1.T != P1 and H.T*H != H*H.T
print("Right modes by columns:", X.tolist())
print("Dual left instruments by rows:", L.tolist())
print("Projectors:", [P.tolist() for P in projectors])
```

```
Right modes by columns: [[1, 1], [0, 1]]
Dual left instruments by rows: [[1, -1], [0, 1]]
Projectors: [[[1, -1], [0, 0]], [[0, 1], [0, 1]]]
```

Now use floating-point arithmetic on the family X_δ. A fixed vertical perturbation tests how distinguishable the two modes are as their lines approach. The final column reports the measured *change in coefficients*, a sensitivity which would be present in exact arithmetic too.

```
eta = 1e-8
print("delta       cond_2(X)    change in first coefficient")
for delta in [1., 1e-2, 1e-4, 1e-6]:
    basis = np.array([[1., 1.], [0., delta]])
    before = np.linalg.solve(basis, [1., 0.])
    after = np.linalg.solve(basis, [1., eta])
    print(f"{delta:8.1e} {np.linalg.cond(basis):12.4g} {after[0]-before[0]:22.4g}")
```

```
delta       cond_2(X)    change in first coefficient
 1.0e+00        2.618                  -1e-08
 1.0e-02          200                  -1e-06
 1.0e-04        2e+04                 -0.0001
 1.0e-06        2e+06                   -0.01
```

The predicted change is $-\eta/\delta$. A discrepancy in its last computed digits is a floating-point effect. The factor $1/\delta$ is predicted by the exact inverse and reflects intrinsic sensitivity for these modes and this perturbation.

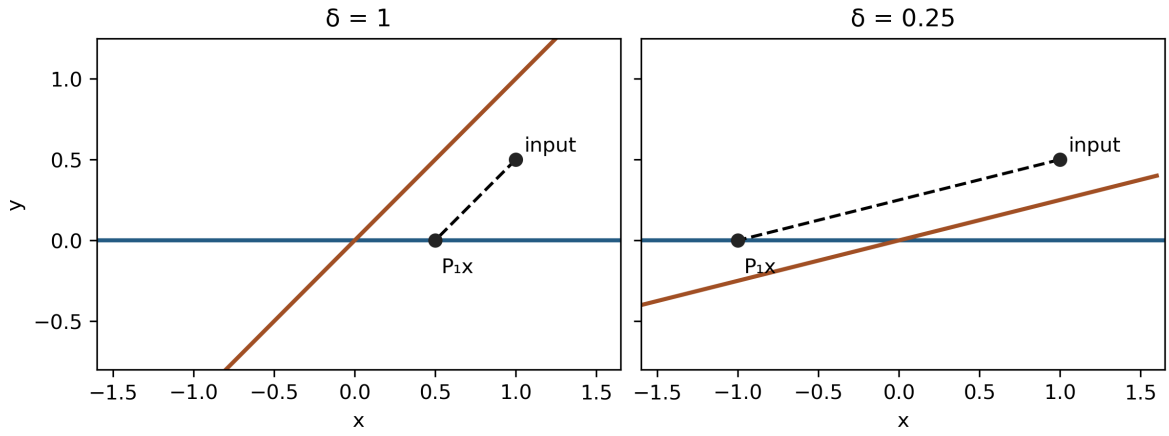


Figure 19.2: Two projections onto the same eigenline for two choices of the discarded eigenline. Dashed segments run parallel to the discarded mode.

In Figure 19.2 the input is fixed. Changing the discarded eigenline changes ℓ_1 and moves $P_1 x$ from $1/2$ to -1 on the horizontal axis. The visible horizontal range is the same for both panels, so the displacement directly illustrates the coefficient formula.

19.10 Exercises — Measurements, obstructions and representations

Try first to specify the source and target of every map. Then choose coordinates if they simplify the argument.

19.10.1 Construct and transport instruments

1. **Sampling a polynomial.** On $\mathcal{P}_2(\mathbb{R})$ take $f_1(p) = p(0)$, $f_2(p) = p'(0)$ and $f_3(p) = p(1)$. Find a polynomial basis dual to these instruments. Express $p(2)$ as a combination of the readings and explain why the recipe is unique.
2. **Reconstruct the dual basis proof.** Start with any finite basis and a list of target scalars. Construct the unique functional with these readings. Derive both the dimension of the dual and a test for whether a given family spans it. Apply the test to trace and determinant on real 2×2 matrices.
3. **A coordinate error.** For $B = \left(\begin{smallmatrix} 1 & 2 \\ 1 & 1 \end{smallmatrix}\right)$ find its dual rows. For $v = (4, 3)^T$ and $f(x, y) = 3x - y$, compute both coordinate columns in the new basis. Show the failure if you transform both by B^{-1}.
4. **Pullback and complex scalars.** For $A = \left(\begin{smallmatrix} 1 & i & 0 \\ 0 & 1 & 2 \end{smallmatrix}\right)$, $g(y) = iy_1 + y_2$, compute gA using composition. Then apply $S(y) = y_1 - y_2$ after A and compare $(SA)^\vee$ with $A^\vee S^\vee$ on an arbitrary scalar weight.

19.10.2 Reconstruct the structural proofs

5. **Build an obstruction.** Let $U = \operatorname{span}((1,1,0),(0,1,1))$. Construct U^0, test $b = (1,0,1)$ for membership in U, and prove the general codimension count using an adapted dual basis.
6. **Reconstruct a missing sensor.** For Module A's matrix determine whether $2x_1 + 4x_2 - x_3 + 7x_4$ is recoverable. If it is, give all recipes. Prove from scratch why the kernel test is necessary and sufficient for any linear map between finite-dimensional spaces.
7. **Two quotient experiments.** For $S = \operatorname{span}((1,0,1))$ in $\mathbb{R}^3$, give coordinates on V/S. Write down both arrows in the dual short exact sequence as matrices and check exactness at all three nonzero spaces.
8. **Double dual and symmetry.** Given an arbitrary functional Φ on $V^\vee$, reconstruct v with $\iota_V(v) = \Phi$ in finite dimension. Show the reconstruction commutes with $T : V \to W$. Explain why the analogous assignment of a functional to each vector by copying its coordinates changes under rescaling.
9. **Annihilators as a calculus.** Prove $(U + H)^0 = U^0 \cap H^0$. Use the bidual identification and double annihilation to deduce the intersection formula without a second dimension computation.
10. **Invariant annihilators.** Let $T(x,y) = (x + 2y, 3y)$ and $U = \operatorname{span}(e_1)$. Compute $T|_U$, the quotient operator and the action of $T^\vee$ on U^0. Is the complementary coordinate line invariant? Explain what information restriction and quotient omit.

19.10.3 Use measurements in an unfamiliar problem

11. **A graph with a free branch.** Add an edge $3 \to 4$ to the oriented triangle. Which edge measurements can be prescribed as gradients? Find all edge-weight recipes for measuring $p_4 - p_1$. Justify completeness.
12. **A nonnormal spectral decomposition.** For $H = \left(\begin{smallmatrix} 2 & 4 \\ 0 & 4 \end{smallmatrix}\right)$ find right modes, dual left modes and both projectors. Evaluate $p(H)$ for $p(t) = (4-t)/2$ and explain its range and kernel.
13. **Dual coordinates of a load.** In the one-hat problem use trial $\phi' = 3\phi$ and test $\psi' = 2\phi$ (the primes here label new basis functions). Recompute the stiffness entry, load and solution coefficient. Explain why the two factors in the stiffness change differ from the transformation for an endomorphism.
14. **An algebraic instrument with no norm bound.** On c_{00}, the finitely supported sequences with the ℓ^2 norm, let $f(x) = \sum_n x_n$. Show it is an algebraic functional, prove it is unbounded, and decide whether a continuous extension to ℓ^2 exists.
15. **Metric-dependent sensitivity.** A real differential has row $(2, -1)$. Find its representing vector in the usual inner product, and in $\langle x, y\rangle = y^T M x$ with $M = \operatorname{diag}(4,1)$. Does the directional derivative along $(1,2)$ change?

19.11 Worked solutions

1. The kernel of the first two readings consists of ct^2; the third detects c. Solve the three interpolation problems with readings e_j. The polynomials are $1-t^2$, $t-t^2$, t^2, respectively. Direct substitution gives the identity reading matrix. Every quadratic with readings a, b, c therefore equals $a(1-t^2)+b(t-t^2)+ct^2$; the difference has all three readings zero. At 2 the result is $-3a-2b+4c$. Uniqueness follows by testing any candidate recipe on these three polynomials.

2. Define $f(\sum_j x_j b_j) = \sum_j c_j x_j$. Expansion is unique and respects the vector operations; all values are forced by linearity. The list of readings is therefore a coordinate isomorphism $V^\vee \to F^n$. A finite family spans the dual iff its reading rows on the basis span $F^{1\times n}$, equivalently the measurement matrix has rank n. Trace is linear with row $(1,0,0,1)$ on the ordered matrix units $(E_{11}, E_{12}, E_{21}, E_{22})$. Its span has dimension one out of four. Determinant fails even additivity: the two diagonal matrix units have determinant zero and their sum has determinant one. It cannot be included as a member of the dual.

3. The inverse has rows $(-1,2), (1,-1)$. The new vector column is $(2,1)^T$, the new covector column $B^T(3,-1)^T = (2,5)^T$. Their evaluation is 9, agreeing with $3\cdot 4-3$. Applying B^{-1} incorrectly to the covector yields $(-5,4)^T$, which reports -6. Operationally, covector components are readings on the new basis vectors, where multiplication by B^T comes from putting each of those vectors into the instrument.

4. $g(Ax) = i(x_1+ix_2)+x_2+2x_3 = ix_1+2x_3$. Multiplying the coefficient column $(i,1)^T$ by A^T yields $(i,0,2)^T$. A conjugation of the entry i in A would incorrectly leave a coefficient 2 on x_2. Next $SA = (1, i-1, -2)$. A scalar weight c pulls back by S to $(c,-c)^T$, then by A to $(c,(i-1)c,-2c)^T$, the transpose of the composite times c. In general this calculation is evaluation of the same composed function in two groupings.

5. Annihilation imposes $a+b=0$ and $b+c=0$. The row $(1,-1,1)$ spans the solutions and takes value 2 on the proposed b, proving $b \notin U$. Generally extend a k-element subspace basis to an n-element basis. In the resulting dual basis any annihilator member has its first k coefficients zero and every remaining coefficient is free. Thus the annihilator has dimension $n-k$.

6. The requested row is $2r-s$ in Session 4. Its recipe can be $2b_3/3-(b_2-2b_1)/3$, with row $(2,-1,2)/3$. Every other recipe adds a multiple of $(-1,-1,1)$. In general if $f = gT$, then f vanishes on $\ker T$. Conversely set $h(Tv) = f(v)$: two possible preimages differ by a kernel vector, so the rule gives one value. It is linear by choosing sums and scalar multiples of preimages. Extend h linearly to a basis of the codomain completed from an image basis. All extensions pull back to f, and two recipes agree on inputs precisely when their difference annihilates the image. This proves both fundamental identities.

7. Use $Q(x,y,z)=(x-z,y)$. It is onto and its kernel is S. Choose t as coordinate of $(t,0,t)$ in S. The dual arrows have matrices

$$Q^T=\begin{pmatrix}1&0\\0&1\\-1&0\end{pmatrix},\qquad i^T=\begin{pmatrix}1&0&1\end{pmatrix}.$$

The first has independent columns. The kernel of the second consists of $(a,b,-a)^T$, precisely the first image. The last arrow is onto since $(c,0,0)^T$ maps to c. In particular zero composite alone would not have proved middle exactness; the displayed parameterization does.

8. Set $v=\sum_i \Phi(\varepsilon^i)b_i$. For any f, expand in the dual basis and apply Φ to get $\Phi(f)=\sum_i f(b_i)\Phi(\varepsilon^i)=f(v)$. Uniqueness follows since coordinate instruments distinguish vectors. For any $g\in W^\vee$, $(T^{\vee\vee}\Phi)(g)=\Phi(T^\vee g)=(T^\vee g)(v)=g(Tv)$, so its reconstructed vector is Tv. By contrast in a real line changing e to $2e$ changes the copied coefficient of e from 1 to $1/2$ and also halves its dual basis instrument. The resulting functional evaluates on e as $1/4$ instead of 1.

9. A functional killing U and H kills every $u+h$, and each subspace belongs to the sum, proving the first formula. Apply it in $V^\vee$ to U^0 and H^0. It gives $(U^0+H^0)^0=U^{00}\cap H^{00}=\iota_V(U\cap H)$. Take annihilators again. In finite dimension double annihilation recovers U^0+H^0 in the triple dual. Under the evaluation identification for $V^\vee$, the annihilator of $\iota_V(U\cap H)$ consists exactly of the f with $f(u)=0$ on $U\cap H$. This yields the intersection formula with all spaces accounted for.

10. On U, T acts as identity. Modulo U only y is observed, and it becomes $3y$. Thus the quotient acts by 3. The annihilator is spanned by the row $(0,1)$ and its pullback is $(0,3)$, the same scalar. The complementary y-axis is not invariant since $Te_2=(2,3)^T$. The off-diagonal 2 records a transfer from a chosen lift of a quotient direction into U. Neither smaller operator sees it.

11. Label edges in order $12,23,31,34$. A gradient must satisfy $d_{12}+d_{23}+d_{31}=0$ and d_{34} is unrestricted. Sufficiency follows by integrating along the tree $12,23,34$ with $p_1=0$, then checking the closing edge. Summing on that same tree path gives weights $(1,1,0,1)^T$ for p_4-p_1. All recipes add $c(1,1,1,0)^T$. To see completeness, at vertex 4 a difference of recipes with zero pullback must have zero weight on edge 34, then balance at the triangle vertices forces the three other weights equal.

12. Right modes $(1,0)^T,(2,1)^T$ have eigenvalues $2,4$. The inverse basis matrix has rows $(1,-2),(0,1)$, the left instruments. Multiplying each column by its dual row gives $P_1=\left(\begin{smallmatrix}1&-2\\0&0\end{smallmatrix}\right)$ and $P_2=\left(\begin{smallmatrix}0&2\\0&1\end{smallmatrix}\right)$. The deltas ensure each kills the other's range and fixes its own. Since $p(2)=1$ and $p(4)=0$, $p(H)=P_1$. Its range is the first eigenline, its kernel the sloping second line. The projector is oblique in the usual Euclidean geometry.

13. Bilinearity gives $K'=2\cdot 4\cdot 3=24$, and the load on the new test is $2(1/2)=1$. Thus the trial coefficient is $1/24$, representing $(1/24)(3\phi)=\phi/8$. Trial basis vectors enter the first

slot of the form, and test vectors the second. Evaluating on a rescaled test multiplies the load by the same scale. An endomorphism's output consists of vector coordinates in a target basis, which instead transform with the inverse basis matrix.

14. Finite support makes the formula a finite sum for every input, and distributivity proves linearity. The vector with first N entries $1/\sqrt{N}$ and all others zero has norm one but reading $\sqrt{N}$. Hence no uniform norm bound holds. Restriction of any continuous extension would be bounded, so no continuous extension exists. Algebraic and continuous duals already differ on this simple normed space.

15. Writing $f(x) = y^{T}Mx$ requires $My = (2,-1)^{T}$. The representing vector is $(2,-1)^{T}$ in Euclidean geometry and $(1/2,-1)^{T}$ for the specified M. Along $(1,2)^{T}$ both give 0. A direction's measured change is independent of which metric represents the measurement.

19.12 Session 9 — What can three curvature readings tell us about a cubic?

19.12.1 Integrated mastery problem

In the final problem of Module A the device $T : \mathcal{P}_3(\mathbb{R}) \to \mathbb{R}^3$ reported $Tp = (p''(0), p''(1), p''(2))$. Suppose the stored readings are the only surviving experimental data. A colleague asks for $p'(1)-p'(0)$; another asks for $p(1)-p(0)$. Decide which answer can be reconstructed. Find **all** output formulas for the recoverable quantity, characterize every possible input quantity that can be answered, and explain the exact sequence and naturality identity in this particular experiment.

19.12.2 Worked reconstruction

Write $p = a + bt + ct^2 + dt^3$. Its second derivative is affine, so a reading (u, v, w) obeys $w = 2v - u$. Conversely $c = u/2$ and $d = (v-u)/6$ reconstruct a cubic giving that reading. Thus $K = \ker T = \mathcal{P}_1$ and $I = \operatorname{im} T = \{u - 2v + w = 0\}$. This supplies the primal answer $V/K \cong I$.

For output weights (α, β, γ) expand the reading:

$$(T^{\vee}g)(p) = 2(\alpha+\beta+\gamma)c + 6(\beta+2\gamma)d.$$

The coefficients on a, b vanish. Every pair of coefficients on c, d can be obtained by choosing $\gamma = 0$ and solving for α, β. Hence the image of $T^{\vee}$ is **all** of K^0. Its kernel is the span of the output measurement $(1,-2,1)$, exactly I^0. The first question asks for $2c + 3d = (u+v)/2$, so all output recipes are

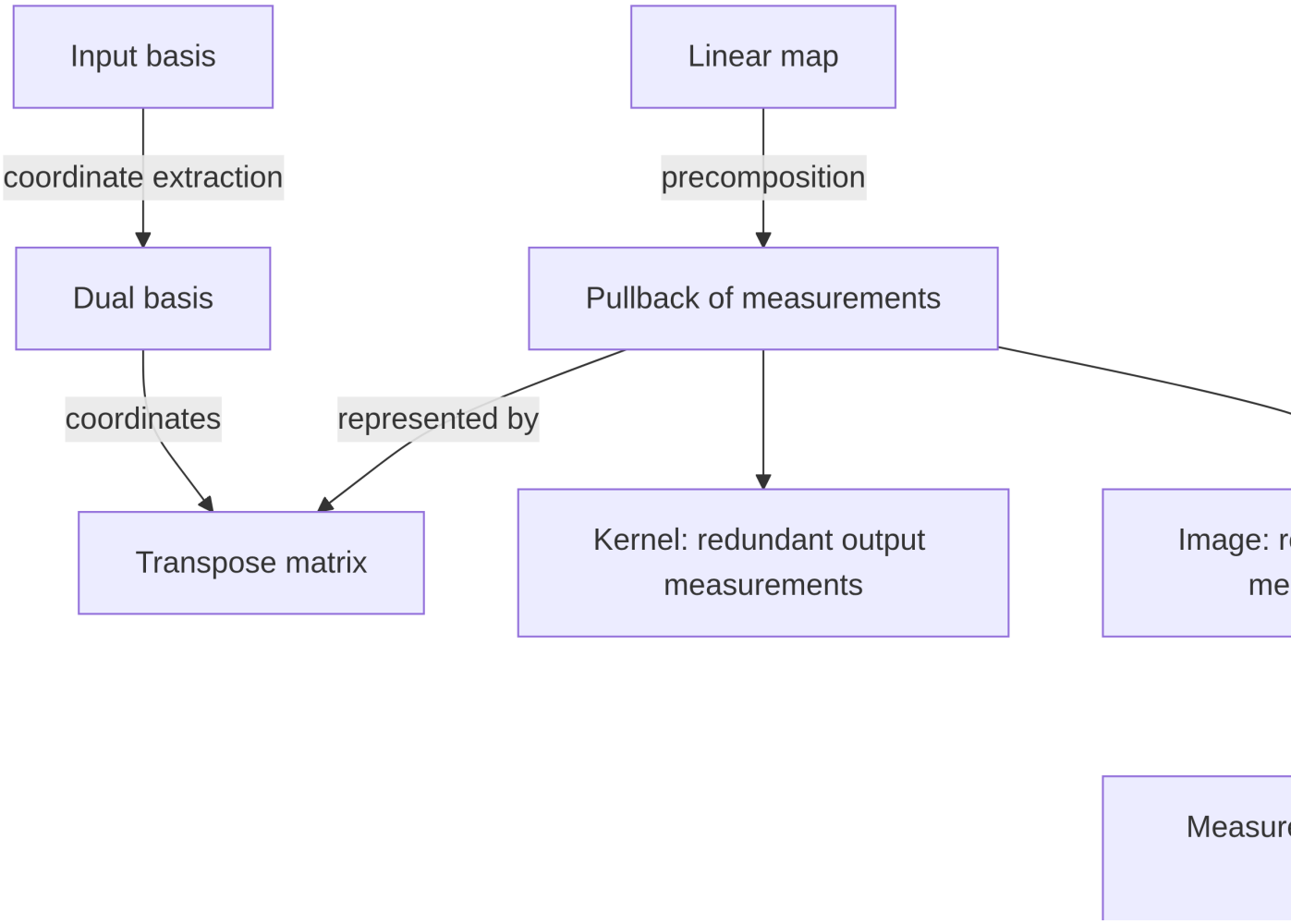
$$(1/2, 1/2, 0) + s(1,-2,1), \qquad s \in \mathbb{R}.$$

Integration gives another derivation: the integral of the affine function p'' on $[0,1]$ is the average of its endpoint values. The second question asks for $b+c+d$. Adding t changes that answer by 1 and changes none of the readings. No recipe exists.

The unique instrument on the quotient is $[p] \mapsto 2c+3d$; adding an affine polynomial changes neither coefficient. The dual exact sequence $0 \to I^\vee \to V^\vee \to K^\vee \to 0$ first embeds instruments on attainable data by composition with T restricted in codomain. The second arrow tests an instrument on affine polynomials. Being zero on both 1 and t is therefore exactly the criterion for recovery. Every instrument on K extends because its values on t^2, t^3 may be freely prescribed. The other exact sequence $0 \to I^0 \to W^\vee \to I^\vee \to 0$ explains the freedom s in the formula.

Finally test naturality on the output instrument $g(u,v,w) = v$. Sending the vector p to Tp and evaluating at g gives $p''(1) = 2c+6d$. Evaluating p on the pulled-back instrument $T^\vee g$ gives the same number. Replacing g by any output instrument repeats exactly the argument, so the equality is one of bidual elements, not just agreement on a selected sensor.

19.12.3 Conceptual map



The diagram records two ways to reach the transpose: specify how to measure on a basis, or compose measurements and then express the result in dual coordinates. At the bottom, restriction and descent explain the two exact sequences in the cubic problem. They separate the ambiguity of an input representative from the ambiguity of an output formula.

Result	Content	Assumptions that matter
Evaluation	$(f, v) \mapsto f(v)$ separates both arguments	Algebraic basis extension for separation in arbitrary dimension
Dual basis	$\varepsilon^i(b_j) = \delta^i_j$, $\dim V^\vee = \dim V$	Finite basis for the finite expansion of every functional

Result	Content	Assumptions that matter
Dual map	$g \mapsto gT$, $(ST)^\vee = T^\vee S^\vee$	Linear structure alone
Transpose	$[T^\vee] = [T]^T$	Corresponding dual bases; linear coefficient convention
Output obstruction	$\ker T^\vee = (\operatorname{im} T)^0$	Algebraic, any dimension
Input recovery	$\operatorname{im} T^\vee = (\ker T)^0$	Algebraic extension; automatic in finite dimension
Quotient dual	$(V/S)^\vee \cong S^0$	Canonical descent, any algebraic dimension
Dual exactness	$\ker i^\vee = \operatorname{im} q^\vee$, $i^\vee$ onto	Restriction and extension over a field
Bidual	$T^{\vee\vee}\iota_V = \iota_W T$	Any dimension for naturality; finite dimension suffices for bijectivity
Lattice of subspaces	Sums and intersections interchange under annihilation	Finite dimension suffices; S^{00} lives in the bidual
Spectral coordinates	$P_i x = v_i \ell_i(x)$, $p(T) = \sum_i p(\lambda_i) P_i$	An eigenbasis over the chosen field
Continuous recovery	A rule on the image must be bounded to extend continuously	Norms, topology of the range, and bounded extension

Canonical or chosen? Evaluation, pullback, restriction, annihilators, descent and the bidual map require only their stated spaces and maps. A dual basis is forced *given* a primal basis. A particular extension off a subspace generally requires a choice, while its class modulo the annihilator is determined. The projector onto a full eigenspace along the other eigenspaces of a diagonalizable operator is determined by the operator. Individual eigenvector scalings and a basis within a repeated eigenspace are choices.

19.12.4 What geometry adds to duality

Object	Source and target	Coordinate description
Vector v	Element of V; evaluates instruments through $\iota_V(v)$	Column x in a chosen basis
Covector f	Linear map $V \to F$	Coefficient column c acting by $c^T x$
Algebraic dual map $T^\vee$	$W^\vee \to V^\vee$	A^T in corresponding dual bases

Object	Source and target	Coordinate description
Inner-product adjoint T^*	$W \to V$	A^H in orthonormal coordinates, with metric factors in general

Where do the metric factors come from? In finite dimension choose inner products and set $R_V(y)(x) = \langle x, y\rangle_V$. This is a linear functional of x. Over $\mathbb{C}$, R_V is **conjugate-linear in** y. It is injective by positivity: $R_V(y) = 0$ implies $\langle y, y\rangle = 0$. In an orthonormal basis a functional with coefficient column c is represented by the vector $\overline{c}$, proving surjectivity. Equivalently, with Hermitian positive definite Gram matrix G_V the pairing is $y^H G_V x$. One solves $G_V y = \overline{c}$ for the unique representative. The real case drops conjugation.

Consequently the map $T^* = R_V^{-1} T^\vee R_W$ is linear (two conjugate-linear factors over $\mathbb{C}$). Its defining identity is

$$\langle Tx, y\rangle_W = \langle x, T^* y\rangle_V, \qquad [T^*] = G_V^{-1} A^H G_W.$$

Indeed $y^H G_W A x = (T^* y)^H G_V x$ for every x, which yields the matrix formula. In orthonormal coordinates both Grams are identities. Further, $R_V^{-1}(S^0) = S^\perp$ because vanishing on s becomes $\langle s, y\rangle_V = 0$. Here is the precise transition to Module C: specify a geometry and the same instruments acquire representing vectors, lengths and perpendicular directions. In infinite-dimensional Hilbert spaces the Riesz theorem represents the **continuous** dual.

19.12.5 Common wrong turns

Tempting step	A way to diagnose it
Copy the vector's change-of-basis rule for the instrument	Evaluate both transformed objects. A scalar reading must remain unchanged.
Conjugate entries when forming an algebraic complex pullback	Test multiplication by i on the identity functional.
Treat S^0 and $S^\perp$ as subspaces of the same space	Write their ambient spaces first, then specify a Riesz map.
Assume the recovery recipe is unique on the whole codomain	Add an image-annihilating functional and test only attainable readings.
Replace a left eigenfunctional by a transposed right eigenvector	Test on every other right eigenvector. The required values are zero.
Certify exact solvability from continuous annihilators of a dense image	Check whether the image is closed; compare the diagonal sequence example.

19.12.6 Mastery checklist

- ☐ Construct the evaluation pairing and prove both nondegeneracy statements.
- ☐ Construct a dual basis, derive both coordinate-change rules and explain why a reading is invariant.
- ☐ Derive composition reversal and the transpose rule from the values of a functional.
- ☐ Compute an annihilator and prove the two kernel–image identities, including the extension step.
- ☐ Decide whether data determine a requested quantity and give every linear output recipe.
- ☐ Distinguish a quotient measurement from a class of extensions; prove exactness at each term of the reversed sequence.
- ☐ Construct the bidual identification, verify naturality and recover a subspace by double annihilation.
- ☐ Construct left eigenfunctionals and oblique projectors, and identify what changes inside repeated eigenspaces.
- ☐ Interpret a transpose on a graph, a finite element load and an output-error functional.
- ☐ Derive the adjoint through the Riesz maps, track complex conjugation and identify the topology needed for continuous duality.

Opening question for Module C. A computation reports a residual functional r. To say that r is *small* we need to specify the size of a unit test vector. What inner product on the test space should determine that size, and which vector represents r in that geometry?

20 Module C — Inner-Product Geometry: From Duality to Orthogonality, Approximation, and Variational Methods

A sensor reports $(1, 2, 4)$ although every noise-free reading satisfies $b_3 = b_1 + b_2$. Algebra detects the inconsistency: the functional $b \mapsto b_3 - b_1 - b_2$ returns 1. What is the smallest correction that makes the data possible? A second question arises after we correct the readings: if many inputs explain them, which input has the smallest size? To answer the first question we need a geometry on **outputs**. To answer the second we need a geometry on **inputs**. There is no reason these geometries should have the same units, dimension or weights.

A second experiment replaces the sensors by a finite-element mesh. There we will have two genuinely different ways of measuring the same function: its mean-square size and the energy in its derivative. Understanding why a single discretization needs both is the route from duality to the generalized eigenproblem $Kx = \lambda Mx$.

Conventions and starting point. Work over $\mathbb{R}$ or $\mathbb{C}$. As in the regenerated Module B, $\langle x, y\rangle$ is linear in x and conjugate-linear in y. In coordinates $\langle x, y\rangle_G = y^H G x$, where H denotes conjugate transpose. In texts linear in the second argument the same scalar is written $\langle y, x\rangle_G$. Write $T^\vee g = g \circ T$ for algebraic pullback, T^* for the adjoint between specified inner-product spaces and A^H for an operation on an array. All dimension-dependent assertions below refer to finite dimension unless a function space is specified. We use the algebraic fact that $Tx = b$ is solvable precisely for $b \in \operatorname{im} T$ and then its solutions are $x_0 + \ker T$, since equal outputs mean that the difference of inputs is killed.

20.1 Session 1 — How much change is one unit?

An **inner product** is linear in its first argument, satisfies $\langle x, y\rangle = \overline{\langle y, x\rangle}$, and obeys $\langle x, x\rangle > 0$ for every $x \neq 0$. These axioms imply $\langle x, x\rangle$ is real, $\langle 0, y\rangle = 0$, and $\langle x, ay + bz\rangle = \bar{a}\langle x, y\rangle + \bar{b}\langle x, z\rangle$. Over the reals the form is symmetric and bilinear. Over the complex numbers a bilinear form cannot be positive on all nonzero vectors: replacing x by ix would change the sign of its diagonal value.

Write $\|x\| = \sqrt{\langle x, x\rangle}$ and say $x \perp y$ when $\langle x, y\rangle = 0$. Expanding the square gives

$$\|x + y\|^2 = \|x\|^2 + 2\,\mathrm{Re}\langle x, y\rangle + \|y\|^2.$$

This gives Pythagoras for perpendicular vectors. Over $\mathbb{C}$ a single Pythagorean equality only forces the real part to vanish. Testing also with iy detects the imaginary part.

Theorem 20.1. ***Cauchy–Schwarz.*** *$|\langle x, y\rangle| \le \|x\|\|y\|$, with equality exactly when x, y are linearly dependent.*

Proof. The case $y = 0$ satisfies both assertions. Otherwise write $x = \alpha y + z$ with $\alpha = \langle x, y\rangle / \|y\|^2$. The choice kills $\langle z, y\rangle$. Pythagoras then gives $\|x\|^2 = |\langle x, y\rangle|^2/\|y\|^2 + \|z\|^2$. Dropping the last term proves the bound; equality requires $z = 0$. This also finds the multiple of y closest to x: any other coefficient adds the squared length of a component along y. □

Positivity gives definiteness of the norm, and sesquilinearity gives $\|ax\| = |a|\|x\|$. Applying Cauchy–Schwarz to the cross term bounds $\|x + y\|^2$ by $(\|x\| + \|y\|)^2$, proving the triangle inequality. Thus $d(x, y) = \|x - y\|$ is a distance. For nonzero real vectors the bound permits the definition $\cos\theta = \langle x, y\rangle/(\|x\|\|y\|)$ with $0 \le \theta \le \pi$.

Can a given norm come from an inner product? Adding the expansions for $x + y$ and $x - y$ cancels the cross terms:

$$\|x + y\|^2 + \|x - y\|^2 = 2\|x\|^2 + 2\|y\|^2.$$

The parallelogram law characterizes inner-product norms. Here is why the converse works. On the underlying real space put $q(x) = \|x\|^2$ and $B(x, y) = (q(x+y) - q(x-y))/4$. It is symmetric and $B(0, y) = 0$. Applying the parallelogram law to $(x + y, z)$ and $(x - y, z)$ and subtracting gives $B(x + z, y) + B(x - z, y) = 2B(x, y)$. Taking $z = x$ gives doubling. Taking $x = (u + v)/2$, $z = (u - v)/2$ then gives additivity. Rational homogeneity follows by addition and division; continuity of the norm extends it to real scalars. Finally $B(x, x) = q(x)$. In a complex space $B(ix, iy) = B(x, y)$ since $\|iw\| = \|w\|$, hence $B(ix, y) = -B(x, iy)$. Then $B(x, y) - iB(ix, y)$ is linear in the first slot, conjugate symmetric, and has diagonal $q(x)$. Over $\mathbb{R}^2$, the 1-norm fails the law on e_1, e_2 (left side 8, right side 4).

20.1.1 Change the measuring instrument

For Hermitian $G = G^H$, $y^H G x$ is conjugate symmetric. It defines an inner product precisely when $x^H G x > 0$ for $x \neq 0$, i.e. G is **Hermitian positive definite** (HPD; SPD in the real symmetric case). A semidefinite weight can assign zero size to a nonzero direction. Our recurring metric is

$$G = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}, \qquad x^T G x = x_1^2 + x_2^2 + (x_1 + x_2)^2.$$

The coordinate axes are perpendicular in Euclidean geometry. Here they have length $\sqrt{2}$ and make angle $\pi/3$. The vectors $(1,1)$ and $(1,-1)$ are eigenvectors of G of eigenvalues $3,1$. Measuring along the corresponding Euclidean unit directions gives ellipse semiaxes $1/\sqrt{3}, 1$. Thus a highly weighted direction displaces less far on paper before using up one unit of length.

Checkpoint. Is $(1,-2)$ perpendicular to e_1 in this metric? Its first weighted component is $2-2=0$. Its ordinary first component is 1. The meaning of a right angle changes with the metric.

20.2 Session 2 — How do coordinates remember the geometry?

Suppose software stores coefficients in a basis $B=(b_1,\dots,b_n)$. Multiplying coordinate columns entry by entry measures lengths correctly only in a particular kind of basis. Expand $u=\sum_j x_j b_j$, $v=\sum_i y_i b_i$. Sesquilinearity gives

$$\langle u,v\rangle=\sum_{ij}\bar{y}_i\,\langle b_j,b_i\rangle x_j=y^H G_B x, \qquad (G_B)_{ij}=\langle b_j,b_i\rangle.$$

The order of the indices follows our convention. This **Gram matrix** is Hermitian. For any list of vectors, $x^H G_B x=\|\sum_j x_j b_j\|^2\geq 0$, with equality precisely when the combination vanishes. Thus its Gram matrix is positive definite exactly when the list is independent. An orthogonal basis gives a positive diagonal Gram matrix, an orthonormal basis gives the identity. For an orthogonal basis the coordinates are $\langle u,b_j\rangle/\|b_j\|^2$, as pairing the expansion with b_j shows.

Let the columns of $B'=BP$ specify a new basis relative to the old, so $x=Px'$. Substitution gives

$$G_{B'}=P^H G_B P.$$

This **congruence** changes the description of the same geometry. Replacing G_B by a new metric while keeping the same basis instead changes which vectors are perpendicular. Neither operation is the similarity $P^{-1}AP$ for an endomorphism.

For the oblique basis $b_1=(1,0)^T, b_2=(1,1)^T$ of Module B in Euclidean space,

$$G_B=B^TB=\begin{pmatrix}1&1\\1&2\end{pmatrix}.$$

The vector $(3,2)^T$ has coefficients $(1,2)^T$, and the Gram formula gives 13, its squared length. Ordinary coordinate squaring in this basis would give 5. If the physical geometry is instead the metric G of Session 1, the same basis has $B^TGB=\left(\begin{smallmatrix}2&3\\3&6\end{smallmatrix}\right)$ and the squared length becomes 38. The vector and its coordinates did not move.

20.2.1 Constructing coordinates in which lengths are simple

Given independent $b_1, \dots, b_n$, define successively

$$w_j = b_j - \sum_{i<j} \langle b_j, q_i \rangle q_i, \qquad q_j = w_j / \|w_j\|.$$

If the preceding q_i are orthonormal, pairing with each proves w_j is perpendicular to their span. It is nonzero because the preceding q_i span exactly the preceding b_i and b_j is independent of them. Induction proves that **Gram–Schmidt** produces an orthonormal basis preserving every initial span. In Euclidean matrix coordinates this is thin $B = QR$, where $r_{ij} = q_i^H b_j$ for $i \le j$, and the positive diagonal of R consists of the norms of the remainders. The triangular R converts trial coefficients to orthonormal coefficients. In exact arithmetic $B^H B = R^H R$. Numerical QR implementations can use orthogonal Householder transformations to avoid the loss of orthogonality that direct subtraction can suffer with nearly dependent input columns.

Checkpoint. For an independent list of functions ϕ_i on $[0, 1]$, in the L^2 inner product the same expansion gives entries $\int \phi_j \bar{\phi}_i$. This Gram matrix will reappear as the mass matrix. Which geometry would give entries involving derivatives instead?

20.3 Session 3 — Which vector represents a fixed functional?

A functional $f \in V^\vee$ already assigns a number to every vector. We now want a vector a_f with $f(x) = \langle x, a_f \rangle$. Define the **Riesz map** $R_V(a) = \langle \cdot\, , a \rangle$. Over $\mathbb{C}$, $R_V(\alpha a) = \bar{\alpha} R_V(a)$. Over $\mathbb{R}$ it is linear.

Theorem 20.2. ***Finite-dimensional Riesz representation.*** *Every linear functional has exactly one such representative. The dual norm* $\|f\|_{\text{dual}} = \sup_{\|x\| \le 1} |f(x)|$ *equals* $\|a_f\|$.

Proof. Choose an orthonormal basis q_i and set $a_f = \sum_i \overline{f(q_i)} q_i$. For $x = \sum_i x_i q_i$, pairing with a_f gives $\sum_i x_i f(q_i) = f(x)$. Two representatives would have a difference orthogonal to every x, hence to itself, so coincide. Cauchy–Schwarz gives the norm upper bound. For $a_f \neq 0$ take $x = a_f / \|a_f\|$ to attain it; for $a_f = 0$ both norms vanish. □

In arbitrary coordinates let the algebraic row be c^T, exactly as in Module B. Equating $a_f^H G x = c^T x$ for all x gives

$$G a_f = \bar{c}, \qquad \|f\|_{\text{dual}}^2 = c^T G^{-1} \bar{c}.$$

Thus a positive definite solve changes a **functional's coefficients** into a **vector's coordinates**. If the functional is instead given as $d^H x$, the right side of the solve is d. For the ith coordinate functional in a basis, take $c = e_i$. The representatives have coordinate columns

$G_B^{-1} e_i$ and satisfy $\langle b_j, a_i \rangle = \delta_{ij}$. Only for an orthonormal basis is each b_i its own coordinate functional's representative.

Consider $f(x_1, x_2) = x_1$ over the reals. Its Euclidean representative is e_1. In the metric of Session 1 its representative is $(2, -1)^T/3$ and its norm is $\sqrt{2/3}$. It gives the same reading on every input. Its kernel is still the vertical axis, and $\langle s, a_f \rangle_G = 0$ for every vertical s. The vector $a_f / \|a_f\|_G$ points in the direction of largest increase per *weighted* unit displacement. For a differentiable real scalar field this construction applied to df_x defines its gradient in that metric. In real coordinates $dF_x(h) = (\nabla F)^T h = (\nabla_G F)^T G h$ for every h, so $\nabla_G F = G^{-1} \nabla F$. An optimization method that changes metric can change its search direction even at the very same point.

Figure 20.1 shows both unit boundaries and their supporting lines for f. Each marked point maximizes the first coordinate on its own boundary. The tangent there is vertical in both drawings. The slanted radius is perpendicular to that tangent in the weighted metric.

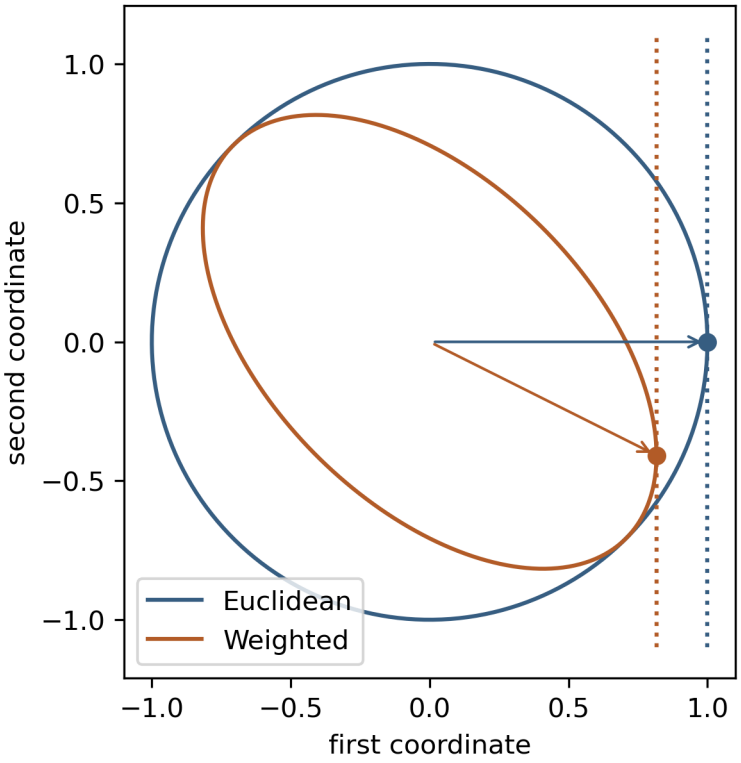


Figure 20.1: Unit boundaries and maximizers of the same coordinate functional under two metrics.

Checkpoint. With $G = I$ over $\mathbb{C}$ and $f(z_1, z_2) = iz_1 + z_2$, the representative is $(-i, 1)^T$. Check at $(i, 0)$ that the conjugation is essential. Both the measurement and inner product

give -1.

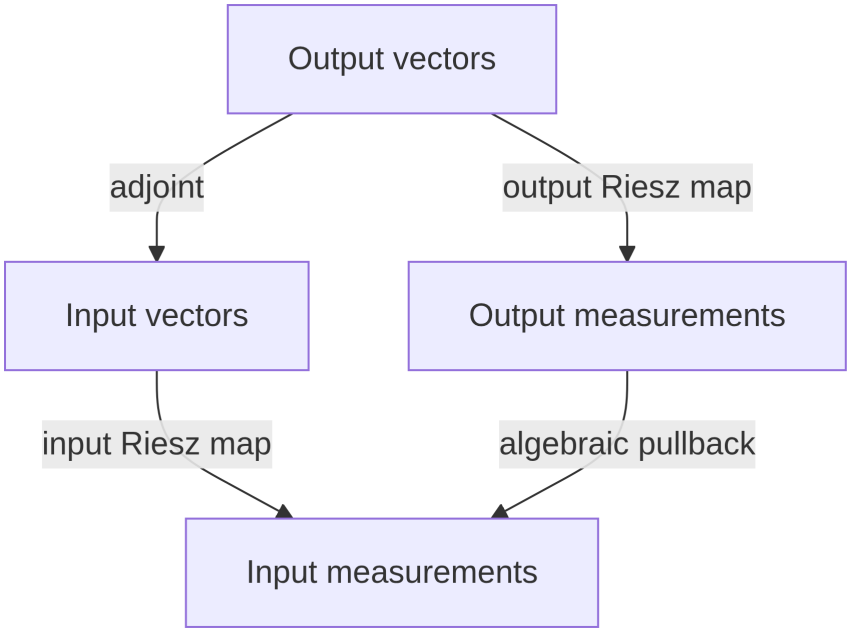
20.4 Session 4 — How should an output direction be pulled back?

Let $T : V \to W$ be linear. Module B pulled back an output *measurement* g by composing it with T. Given an output *vector* y, first decide what it measures: $R_W y$ acts by $w \mapsto \langle w, y\rangle_W$. Pull that measurement back to V and find its input representative.

Theorem 20.3. ***Adjoint.*** *In finite-dimensional inner-product spaces there is a unique linear $T^* : W \to V$ such that*

$$\langle Tx, y\rangle_W = \langle x, T^* y\rangle_V, \qquad T^* = R_V^{-1} T^\vee R_W.$$

Proof. For fixed y, the left side is a linear functional of x. By Theorem 20.2 its representing vector exists and is unique. This gives the displayed composition, since the functional is $(R_W y)\circ T$. Its dependence on y is linear: over $\mathbb{C}$ the two Riesz factors are conjugate-linear and the middle factor is linear. Equality of pairings with every x forces equality of the representing vectors, hence uniqueness of the operator. □



Composition on the lower path depends only on T. Both vertical maps depend on geometry. In coordinates equate $y^H G_W A x$ and $([T^*]y)^H G_V x$:

$$[T^*] = G_V^{-1} A^H G_W.$$

Ordinary orthonormal coordinates on both spaces give A^H, and over $\mathbb{R}$ this becomes A^T. In corresponding **dual** bases $T^\vee$ always has matrix A^T, including when A is complex. Even orthonormal coordinates do not identify the two complex operations: multiplication by i has algebraic pullback matrix $[i]$ and adjoint matrix $[-i]$.

Consider $T(x, y) = x$ with the weighted input geometry G and the usual scalar output. Then $T^*t = t(2, -1)^T/3$ while $T^\vee$ sends an output coefficient t to the functional with row $(t, 0)$ over $\mathbb{R}$. The weighted result is familiar: pulling back the identity measurement gives precisely the representer in Session 3. Changing **only the output metric** to $\langle s, t\rangle = 4\bar{t}s$ multiplies the adjoint by four. The primal map and all attainable readings stay fixed.

Adjoint identities. Conjugating the defining relation and interchanging arguments gives $\langle T^*y, x\rangle_V = \langle y, Tx\rangle_W$. Uniqueness implies $(T^*)^* = T$. For $S : W \to Z$ pair STx with z and pass first S, then T, across the pairing to obtain $(ST)^* = T^*S^*$. The identity is its own adjoint. Applying the composition rule to both inverse identities gives $(T^{-1})^* = (T^*)^{-1}$ for an isomorphism. Scalar factors conjugate: $(\alpha T)^* = \bar{\alpha}T^*$. Each middle space here carries the same inner product in both occurrences.

Checkpoint. If only the coordinates change and $A' = Q^{-1}AP$, show that the new adjoint matrix is $P^{-1}[T^*]Q$. Substitute P^HG_VP, Q^HG_WQ into the weighted formula and cancel the inverse factors. The Gram matrices repair exactly the coordinate dependence of raw conjugate transposition.

20.5 Session 5 — What makes one point in a subspace the closest?

Let S be a subspace of V. Define $S^\perp = \{z : \langle s, z\rangle = 0 \ \forall s \in S\}$. By conjugate symmetry either order tests orthogonality. By definition $R_V(S^\perp) = S^0$ in finite dimension, where S^0 consists of functionals vanishing on S. The annihilator is fixed by S. Its representing vectors move when we change the inner product.

Theorem 20.4. ***Orthogonal decomposition and best approximation.*** *In finite dimension every x has a unique expression $x = p + r$ with $p \in S$ and $r \in S^\perp$. The point $p = P_Sx$ is the unique minimizer of $\|x - s\|$ over $s \in S$.*

Proof. Choose an orthonormal basis $q_1, \dots, q_k$ of S and set $p = \sum_i \langle x, q_i\rangle q_i$. Direct testing against the basis gives $x - p \perp S$. For uniqueness subtract two decompositions: their difference in S belongs to $S^\perp$, so its squared norm is zero. For any $s \in S$ the two terms of $x - s = (x - p) + (p - s)$ are perpendicular. Consequently

$$\|x - s\|^2 = \|x - p\|^2 + \|p - s\|^2.$$

This identifies the minimum and its unique point of attainment, even for $S = 0$ or $S = V$. □

The formula proves linearity of P_S. It fixes S, kills $S^\perp$, and hence $P_S^2 = P_S$. Decompose both arguments to see $\langle P_S x, y\rangle = \langle P_S x, P_S y\rangle = \langle x, P_S y\rangle$. Conversely, any idempotent P splits V into image and kernel by $x = Px + (x - Px)$. If also $P = P^*$ then $\langle Pu, k\rangle = \langle u, Pk\rangle = 0$ for every kernel vector, so this is the orthogonal decomposition. Other complementary directions give **oblique projectors**. Also $(S^\perp)^\perp = S$: decompose $x = s + r$, and if $x \perp S^\perp$, testing against r gives $\|r\|^2 = 0$.

20.5.1 Compute the projection without first orthogonalizing

Let B have independent columns spanning S. The condition on $p = Bc$ in coordinates is $B^H G(x - Bc) = 0$. The Gram matrix is HPD, so

$$P_S = B(B^H G B)^{-1} B^H G.$$

In an orthonormal trial basis Q this becomes $QQ^H G$, since $Q^H G Q = I$. The inverse in the formula denotes a solve. Geometrically $S^\perp$ selects the minimum-length representative in every class $[x] \in V/S$: write $x = p + r$, then $x + S = r + S$ and $\|r + s\|^2 = \|r\|^2 + \|s\|^2$.

For the array from A and B, with Euclidean input geometry, a kernel basis matrix is

$$N = \begin{pmatrix} -2 & -4 \\ 1 & 0 \\ 0 & -1 \\ 0 & 1 \end{pmatrix}, \qquad N^T N = \begin{pmatrix} 5 & 8 \\ 8 & 18 \end{pmatrix}.$$

For $x = e_1$, solve $(N^T N)c = N^T e_1 = (-2, -4)^T$ to get $c = (-2, -2)^T/13$. The orthogonal kernel component is $(12, -2, 2, -2)^T/13$. Subtracting it gives $(1, 2, -2, 2)^T/13$, the minimum-length input with the same data as e_1. The RREF complement $\mathrm{span}(e_1, e_3)$ instead selects e_1 itself. Both choices explain the reading; only the perpendicular one minimizes Euclidean input norm.

20.5.2 Four subspaces and one closure

Test y on all Tx. Then $T^* y = 0$ iff $y \perp \operatorname{im} T$. Apply the same argument to T^* and use $(T^*)^* = T$. Taking double perpendiculars in finite dimension gives all four identities:

$$\ker T^* = (\operatorname{im} T)^\perp, \quad \ker T = (\operatorname{im} T^*)^\perp,$$
$$\operatorname{im} T^* = (\ker T)^\perp, \quad \operatorname{im} T = (\ker T^*)^\perp.$$

For example $R_V(\operatorname{im} T^*) = \operatorname{im} T^\vee = (\ker T)^0$ by the Riesz square. This recovers Module B's input measurement criterion in geometric form.

For a bounded operator between Hilbert spaces the first line remains exact; the images on the left of the second line acquire closures. The projection theorem in a Hilbert space applies

to **closed** subspaces and yields $S^{\perp\perp} = \overline{S}$. For example $(Tx)_j = x_j/j$ on real ℓ^2 has trivial kernel, is self-adjoint by termwise pairing, and its range contains every finite sequence. Yet $(1/j)_{j\geq 1}$ has no square-summable preimage. Its distance to the image is zero by truncation without a closest image point. Completeness of the ambient space alone cannot make this range closed.

Checkpoint. The projection $P(x, y) = (x - y, 0)$ fixes the horizontal axis. Its kernel is the diagonal line. Can it minimize Euclidean distance to the axis for all inputs? Test e_2. Then identify an inner product in which this decomposition *is* orthogonal by declaring its two spanning directions orthonormal.

20.6 Session 6 — Which transformations respect the geometry?

For an endomorphism H, the following definitions refer to the selected inner product.

Property	Defining identity	Geometric consequence
Self-adjoint	$H^* = H$	The operator may pass between arguments without changing the pairing
Skew-adjoint	$H^* = -H$	$\mathrm{Re}\langle Hx, x\rangle = 0$
Normal	$H^*H = HH^*$	$\|Hx\| = \|H^*x\|$ for all x
Unitary (orthogonal over $\mathbb{R}$)	$H^*H = HH^* = I$	Pairings, hence lengths and distances, are preserved
Positive semidefinite	Self-adjoint and $\langle Hx, x\rangle \geq 0$	Quadratic energy may vanish on nonzero directions
Positive definite	Self-adjoint and $\langle Hx, x\rangle > 0$ for $x \neq 0$	$\langle Hx, y\rangle$ defines a new inner product

Why does self-adjointness constrain eigenvectors? If $Hv = \lambda v$ with $v \neq 0$, $\lambda\|v\|^2 = \langle Hv, v\rangle = \langle v, Hv\rangle = \bar{\lambda}\|v\|^2$, so λ is real. For $Hw = \mu w$, the same calculation yields $(\lambda - \bar{\mu})\langle v, w\rangle = 0$. Distinct eigenvalues force perpendicular eigenvectors. If U is invariant and $z \perp U$, then $\langle Hz, u\rangle = \langle z, Hu\rangle = 0$ for $u \in U$, hence $U^\perp$ is invariant too. An invariant subspace and its orthogonal complement therefore give a block diagonal decomposition.

Why is there an entire orthonormal eigenbasis? On a nonzero finite-dimensional real or complex space the real function $\langle Hx, x\rangle$ attains a maximum λ on the compact unit sphere. At a maximizer v consider the quotient $\rho(v + tw) = \langle H(v + tw), v + tw\rangle/\|v + tw\|^2$ for real t. Its derivative at zero is $2\,\mathrm{Re}\langle Hv - \lambda v, w\rangle$. This must vanish for every w; in particular take $w = Hv - \lambda v$ to obtain an eigenvector equation. Restrict to $v^\perp$, which is invariant and whose restriction is again self-adjoint. Induction on dimension yields the finite-dimensional spectral

theorem. This proof links spectral decomposition to optimization; Module D develops the consequences.

In that basis, if $x = \sum c_i v_i \neq 0$,

$$\rho_H(x) = \frac{\sum_i \lambda_i |c_i|^2}{\sum_i |c_i|^2} \in [\lambda_{\min}, \lambda_{\max}].$$

The Rayleigh quotient is the coefficient of the best approximation to Hx along x. It measures directional energy per squared unit length. The weighted-average identity proves that a Hermitian matrix is positive definite precisely when all eigenvalues are positive, and semidefinite when all are nonnegative. Invertibility alone only excludes zero. For instance $\operatorname{diag}(1, -1)$ is invertible and indefinite.

20.6.1 Two sources of positive matrices

For a linear T between possibly different spaces, T^*T is self-adjoint by the composition rule and $\langle T^*Tx, x\rangle = \|Tx\|^2 \geq 0$. This proves $\ker(T^*T) = \ker T$: one inclusion follows by applying T^* to zero, the reverse by testing the quadratic form. It is positive definite precisely when T is injective. Likewise TT^* is positive definite precisely when T is onto in finite dimension. These facts determine whether normal equations have a unique solution.

A Hermitian positive definite matrix admits a **Cholesky factorization** $G = LL^H$ with lower triangular L of positive real diagonal. For existence take the positive square root from an orthonormal eigenbasis and factor $G^{1/2} = QR$ by QR with positive diagonal. Then $G = R^H R$, so $L = R^H$. Computational Cholesky obtains the triangular factor directly by matching columns, without an eigensolve. In Session 1,

$$G = \begin{pmatrix} \sqrt{2} & 0 \\ 1/\sqrt{2} & \sqrt{3/2} \end{pmatrix} \begin{pmatrix} \sqrt{2} & 1/\sqrt{2} \\ 0 & \sqrt{3/2} \end{pmatrix}.$$

Thus $\|x\|_G = \|L^H x\|_2$, and a solve with G reduces to two triangular solves. The transformation $x \mapsto L^H x$ sends the weighted ellipse to a Euclidean circle.

Finally take $M = \operatorname{diag}(2, 1)$ and $K = \left(\begin{smallmatrix} 2 & 1 \\ 1 & 2 \end{smallmatrix}\right)$. The operator $H = M^{-1}K = \left(\begin{smallmatrix} 1 & 1/2 \\ 1 & 2 \end{smallmatrix}\right)$ satisfies $MH = K = H^H M$. It is self-adjoint in M even though its standard-coordinate matrix is nonsymmetric. Its quadratic energy there is $x^H K x$. Changing the geometry can reveal a symmetry obscured by looking only at the entries of H.

Checkpoint. For the last example, which object is an operator and which records a form on two vectors? Under a basis change P, H changes by similarity and K by congruence. Verify that the identity $MH = K$ is preserved.

20.7 Session 7 — When the equations disagree, what exactly should be minimized?

Return to the sensors. The possible outputs form a subspace $I = \operatorname{im} A$. In Euclidean output geometry, the unique closest reading is $\widehat{b} = P_I b$. Every input satisfying $Ax = \widehat{b}$ is a least-squares minimizer. Thus an answer always exists in finite dimensions, but a unique **fitted output** need not come from a unique **input**.

Theorem 20.5. ***Normal equations and the minimum-norm solution.*** *The minimizers of $\|Ax-b\|_2$ are exactly the solutions of $A^H Ax = A^H b$. They form $x_0 + \ker A$. Exactly one lies in $(\ker A)^\perp$, and it is the unique minimizer of input norm among all least-squares minimizers.*

Proof. By orthogonal projection the fitted output is characterized by $b - Ax \perp \operatorname{im} A$, equivalently $A^H(b - Ax) = 0$. Conversely this condition gives a global minimum by Pythagoras. Fix one preimage of the projection; all others differ by a kernel vector. Resolve this preimage into its perpendicular and kernel components. Dropping the kernel component leaves its fitted data unchanged. Any kernel addition to the perpendicular representative increases its squared norm by the squared norm of that addition. □

For a calculus derivation let $J(x) = \|Ax-b\|^2$ and perturb by th for real t. Expansion gives

$$J(x + th) - J(x) = 2t \operatorname{Re}\left(h^H A^H (Ax - b)\right) + t^2 \|Ah\|^2.$$

Stationarity for every h, in particular $h = A^H(Ax - b)$, gives the normal equations even for complex variables. Conversely at a stationary point the remaining term is nonnegative in every direction. Full column rank makes the minimizer unique, including for inconsistent data. A small residual alone need not imply small input error: for $A = \operatorname{diag}(1, \epsilon)$ an error of one in the second setting changes the residual by only $|\epsilon|$.

20.7.1 A complete inverse problem with two projections

Use the original sensor array

$$A = \begin{pmatrix} 1 & 2 & -1 & 3 \\ 2 & 4 & 1 & 9 \\ 3 & 6 & 0 & 12 \end{pmatrix}, \quad n = (-1, -1, 1)^T, \quad b = (1, 2, 4)^T.$$

Its columns span the plane $n^T y = 0$ (the first and third are independent). Since $n^T b = 1$, projection gives $\widehat{b} = b - n/3 = (4, 7, 11)^T/3$, with residual $n/3$. Now seek a preimage supported in coordinates 1 and 3. Those columns are $(1, 2, 3)^T$ and $(-1, 1, 0)^T$; matching the third and

first readings gives $x_0 = (11, 0, -1, 0)^T/9$. The entire family is $x_0 + Nz$, with N from Session 5. Its minimum-norm member is

$$x^\dagger = (I - P_{\ker A})x_0, \qquad P_{\ker A} = N(N^TN)^{-1}N^T.$$

The output residual lies in $\ker A^T$; the selected input lies in $\operatorname{im} A^T$. These perpendicularity conditions live in different spaces. With compatible $b = (1, 2, 3)^T$ the first projection is the identity on b and the second still changes e_1 to the shorter representative from Session 5. Experiment G computes both the singular-case family and the exact minimum-norm answer.

The resulting map $b \mapsto x^\dagger$ is the **Moore–Penrose pseudoinverse** $A^\dagger$: project onto the image and invert A restricted to $(\ker A)^\perp$. This restriction is bijective onto the image by the decomposition theorem. Hence $AA^\dagger = P_{\operatorname{im} A}$ and $A^\dagger A = P_{(\ker A)^\perp}$. Their self-adjointness and their action on the two subspaces also give $AA^\dagger A = A$ and $A^\dagger AA^\dagger = A^\dagger$. With a compact singular value decomposition $A = U_r\Sigma_r V_r^H$ (orthonormal columns, positive singular values), $A^\dagger = V_r\Sigma_r^{-1}U_r^H$: read the attainable components, undo their stretch and reconstruct with no kernel component.

20.7.2 Why the equations we derive can differ from the algorithm we run

For full column rank thin QR, $A = QR$ gives $Rx = Q^Hb$, since QQ^Hb is the fitted output. For example take columns $(1, 1, 0)^T$ and $(0, 0, 1)^T$ and $b = (1, 3, 2)^T$. Normalizing the columns gives $R = \operatorname{diag}(\sqrt{2}, 1)$, $Q^Hb = (4/\sqrt{2}, 2)^T$, hence $x = (2, 2)^T$. The fitted output $(2, 2, 2)^T$ and residual $(-1, 1, 0)^T$ are perpendicular; both measurements on the residual vanish. The same matrix with $b = (1, 1, 2)^T$ fits exactly.

For full column rank the extreme eigenvalues of A^HA are $\sigma_{\max}^2, \sigma_{\min}^2$, so $\kappa_2(A^HA) = \kappa_2(A)^2$. Forming the Gram matrix can lose information about nearly parallel columns; explicitly inverting it introduces unnecessary work. Solving normal equations with Cholesky can be efficient for a well-conditioned tall matrix. QR avoids forming the Gram matrix. SVD exposes small stretches and permits a numerical rank decision for nearly singular problems. Thresholding singular values changes which directions count as resolved. In Experiment F we compare *coefficient error* as well as residual size, which can conceal the difference.

20.7.3 What if different sensors have different costs?

For HPD output weight W, minimize $(Ax - b)^HW(Ax - b)$. Repeating the perpendicularity argument in that geometry gives $A^HWAx = A^HWb$. Equivalently solve ordinary least squares for $L^HAx \approx L^Hb$ where $W = LL^H$. An input weight G changes the adjoint to $G^{-1}A^HW$. It cancels from the stationarity equations but changes minimum-input-norm selection when there is a kernel. For the sensor plane, the residual now runs along the Riesz representative $W^{-1}n$, and

$$\hat{b}_W = b - W^{-1}n\,\frac{n^Tb}{n^TW^{-1}n}.$$

For $W = \operatorname{diag}(1, 1, 4)$ this gives residual $(-4, -4, 1)^T/9$; the expensive third reading is adjusted by $1/9$ instead of $1/3$. The original compatibility functional n^T is exactly the same.

Checkpoint. For which b does changing W leave the optimal residual exactly zero? Precisely the attainable b: every positive definite norm vanishes only on the zero residual.

20.8 Session 8 — How can a differential equation ask for a closest point?

Consider the real boundary-value problem $-u'' = f$ on $(0, 1)$ with $u(0) = u(1) = 0$. A piecewise linear trial function has jumps in slope and generally no classical second derivative at the mesh nodes. Can we still test whether it approximately solves the equation? Multiply a smooth solution's equation by a test function v vanishing at both endpoints and integrate by parts:

$$\int_0^1 -u''v = -[u'v]_0^1 + \int_0^1 u'v' = \int_0^1 fv.$$

After this step only first derivatives occur. Let $V = H_0^1(0, 1)$. On this interval its elements have absolutely continuous representatives, endpoint values zero and square-integrable first derivative. Equivalently it is the completion of smooth compactly supported tests in the norm $(\|v\|_{L^2}^2 + \|v'\|_{L^2}^2)^{1/2}$. A weak derivative g satisfies $\int v\varphi' = -\int g\varphi$ for all smooth compactly supported φ. Continuous piecewise linear hats have the piecewise constant slopes as weak derivatives because interior endpoint terms cancel.

Set $a(u, v) = \int u'v'$, $m(u, v) = \int uv$ and $F(v) = \int fv$, with all integrals over $[0, 1]$. The mass product m measures amplitudes; the energy form a measures slopes. If $a(v, v) = 0$, then v is constant and its zero boundary value forces zero. More quantitatively,

$$|v(t)| = \left| \int_0^t v'(s)\, ds \right| \le \sqrt{t}\, \|v'\|_{L^2}, \qquad \|v\|_{L^2} \le \|v'\|_{L^2}/\sqrt{2}.$$

This Poincaré bound (sharpness is unnecessary) makes $\|v\|_a = \|v'\|_{L^2}$ equivalent to the complete H^1 norm on V. Thus V is Hilbert in the energy geometry. For $f \in L^2$, Cauchy–Schwarz and Poincaré give $|F(v)| \le \|f\|_{L^2}\|v\|_a/\sqrt{2}$. Hilbert-space Riesz representation of this **continuous** functional provides a unique $u \in V$ with $a(u, v) = F(v)$ for every v. This is the weak solution; smoothness sufficient to recover the classical differential equation is a separate question. Lax–Milgram extends the existence argument to bounded coercive forms, including nonsymmetric real forms for which exact orthogonal projection need not hold.

20.8.1 A mesh with two unknowns

On nodes $0, 1/3, 2/3, 1$ use the two interior hats ϕ_i with value one at node i, zero at the other nodes and linear on each element. The trial space V_h is their span; the test space is also V_h.

Seek $u_h = \sum_j c_j \phi_j$ satisfying $a(u_h, \phi_i) = F(\phi_i)$ for every test index. Thus $K_{ij} = a(\phi_j, \phi_i)$ and $b_i = F(\phi_i)$. On an element of length h, the two slopes are $-1/h, 1/h$, so the local stiffness is $\frac{1}{h}\left(\begin{smallmatrix} 1 & -1 \\ -1 & 1 \end{smallmatrix}\right)$. For $f = 1$ each adjacent half-hat contributes $h/2$ to the load. Adding adjacent contributions, with endpoint unknowns removed, gives

$$K = 3\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}, \qquad b = \tfrac{1}{3}\begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad c = \tfrac{1}{9}\begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

This is a Gram matrix in the energy product: $c^T K c = \| \sum_j c_j \phi_j \|_a^2 > 0$ for nonzero c by linear independence. The exact solution $u(t) = t(1-t)/2$ happens to match the nodal values. Its nonzero curvature between nodes ensures that the approximate function still has nonzero energy error.

Galerkin orthogonality. Subtract the discrete and continuous equations for an arbitrary $v_h \in V_h$. The common right side cancels, giving $a(u - u_h, v_h) = 0$. For any $w_h \in V_h$, split $u - w_h = (u - u_h) + (u_h - w_h)$ to obtain

$$\|u - w_h\|_a^2 = \|u - u_h\|_a^2 + \|u_h - w_h\|_a^2.$$

Thus u_h is exactly the energy-orthogonal projection of u. It can be computed without knowing u because the equation supplies its readings on tests. This reasoning uses a conforming trial space, the same test space and an energy inner product. Orthogonality in L^2 would solve a different minimization problem.

On a mesh element the interpolant's slope equals the average of u' on that element. Since u' is affine of slope -1, the integrated squared slope error is $h^3/12$. Moreover the slope error integrates to zero against every constant trial slope, so the interpolant satisfies Galerkin orthogonality and is indeed u_h. On a uniform mesh the total energy error is $h/\sqrt{12}$ even though the nodal error vanishes. This gives a check on the refinement experiment independent of the matrix solver.

What happens with Neumann data? For zero imposed endpoint derivatives, tests need not vanish at the endpoints. Constants now lie in the kernel of a. Testing with $v = 1$ forces $\int f = 0$. Conversely in one dimension, integrating $u'(t) = -\int_0^t f(s)\,ds$ gives both endpoint derivative conditions precisely when the integral is zero. Integrating again leaves a free constant. On $H^1/\mathbb{R}$ the derivative form is positive definite and the load descends exactly when it kills constants. Choosing the zero-mean representative fixes a normalization. The null direction from Module A, the compatibility test from B and the geometry from C are three aspects of the same boundary condition.

Checkpoint. Can solving $Kc = b$ more accurately remove the between-node slope error? Even an exact solve gives $h/\sqrt{12}$ on this mesh. Enlarging the trial space changes the approximation itself.

20.9 Session 9 — How does a mass matrix enter an eigenvalue problem?

A vibration mode must balance restoring energy against mass. In the real weak problem $a(u,v) = \lambda m(u,v)$ for every test v, scale cancels from the ratio $a(u,u)/m(u,u)$. Testing a finite trial expansion against basis vectors gives $Kx = \lambda Mx$, with $K_{ij} = a(\phi_j, \phi_i)$ and $M_{ij} = m(\phi_j, \phi_i)$. Integration of the products of the two linear shape functions gives the local mass matrix $\frac{h}{6}\left(\begin{smallmatrix} 2 & 1 \\ 1 & 2 \end{smallmatrix}\right)$: the diagonal integrals are $h/3$, the cross integral $h/6$. For the previous two-hat mesh,

$$M = \tfrac{1}{18}\begin{pmatrix} 4 & 1 \\ 1 & 4 \end{pmatrix}.$$

The same shapes therefore produce **two** Gram matrices. The vector $(1,1)^T$ has $Kx = 3x$ and $Mx = (5/18)x$, so its vibration eigenvalue is $54/5$, not the stiffness eigenvalue 3. The antisymmetric vector $(1,-1)^T$ gives 54. Their mass pairing vanishes. The first value approximates π^2, the lowest Dirichlet eigenvalue of $-u''$ on the unit interval.

20.9.1 A self-adjoint operator behind a Hermitian pair

For Hermitian K and HPD M, the operator $T = M^{-1}K$ is M-self-adjoint since $T^H M = MT = K$. Its Rayleigh quotient is

$$\rho_{K,M}(x) = \frac{\langle Tx, x\rangle_M}{\langle x, x\rangle_M} = \frac{x^H K x}{x^H M x}.$$

The spectral theorem in the M-geometry supplies an M-orthonormal eigenbasis and real eigenvalues, even for indefinite K. Alternatively factor $M = LL^H$ and substitute $z = L^H x$:

$$(L^{-1}KL^{-H})z = \lambda z.$$

The transformed matrix is Hermitian. Orthonormal z_i map back to $x_i = L^{-H}z_i$ with $x_i^H M x_j = \delta_{ij}$. Distinct modes obey this orthogonality directly by subtracting the two paired eigenvalue equations. Implement the transformations with triangular solves.

20.9.2 Rayleigh–Ritz asks fewer questions than the full eigenproblem

If columns of Q form an independent trial basis, choose $x = Qy \neq 0$ and require the residual of $Tx - \theta x$ to be M-orthogonal to the trial space. For any test Qz this reads

$$0 = (Qz)^H M(TQy - \theta Qy) = z^H Q^H (KQy - \theta MQy).$$

Arbitrary z gives the reduced generalized eigenproblem

$$Q^H K Q y = \theta Q^H M Q y.$$

The reduced mass is HPD by independence. If $Q^H M Q = I$, it drops from the right side. With $M = I$ we obtain ordinary Rayleigh–Ritz, or $Q^H K Q y = \theta y$ for Euclidean orthonormal Q. Two residuals have distinct coordinate types: $r = Kx - \theta Mx$ is a residual in test readings with $Q^H r = 0$; $M^{-1} r = Tx - \theta x$ is its vector representative, perpendicular in M. A zero *projected* residual only says the selected tests cannot detect the remaining defect.

For one trial column $q = e_1$ on the two-hat mesh the Ritz value is $6/(4/18) = 27$. The residual is $(0, -9/2)^T$: testing the first hat succeeds exactly but testing the other fails. In an M-orthonormal eigenbasis the quotient is a weighted average of the true eigenvalues, so even the single trial value lies in the spectral interval. The smallest Ritz value minimizes the quotient over the trial subspace; enlarging that subspace can only reduce this minimum. The derivative argument in Session 6 restricted to trial perturbations recovers the same reduced equation.

Bridge to the seminar. In subspace iteration or a Krylov method a basis can change while the trial space is fixed. Replacing Q by QP with P invertible changes the reduced pair by congruence and gives exactly the same Ritz vectors in the original space after converting coefficients. Filtering by $p(T)$ generally changes the space: each mode's amplitude is multiplied by $p(\lambda_i)$, as follows by applying powers of T to an eigenvector. A Chebyshev polynomial can be scaled to stay small on an unwanted spectral interval while growing on a separated target region. After filtering, orthonormalization restores a usable basis and Rayleigh–Ritz reads out approximate eigenpairs. In computation the Gram product, orthogonalization, and independent residual checks cost differently and can lose different amounts of accuracy. Modules D and E build the spectral and numerical arguments on this geometry.

Checkpoint. For a full-rank trial basis, a Ritz pair makes $Q^H r = 0$. When does that force $r = 0$? Certainly when Q spans the entire space; also for the corresponding Ritz vectors when the trial space is invariant under $M^{-1}K$. Invariance puts the vector residual in the trial space as well as its perpendicular.

20.10 Computational laboratory — Check the geometry as well as the equations

Use a Python 3 Jupyter kernel with NumPy, SymPy and Matplotlib. Run the cells in document order. Experiments B–E and G use exact arithmetic to check identities. The floating-point experiments report residuals, condition numbers or errors with their mathematical interpretation.

20.10.1 A. Weighted unit directions

Figure 20.1 was generated by mapping the Euclidean circle through L^{-H}. Check the unit lengths, and predict the largest Euclidean radius by minimizing the quadratic form on the

Euclidean unit circle.

```
stretches = np.linalg.eigvalsh(G)
weighted_lengths_sq = np.sum(ellipse * (G @ ellipse), axis=0)
assert np.allclose(weighted_lengths_sq, 1.)
print("Metric eigenvalues:", stretches)
print("Euclidean semiaxis lengths:", 1/np.sqrt(stretches))
print("Pairing of coordinate axes in G:", G[0, 1])
```

```
Metric eigenvalues: [1. 3.]
Euclidean semiaxis lengths: [1.         0.57735027]
Pairing of coordinate axes in G: 1.0
```

The eigenvalues are $1, 3$ and their reciprocals' square roots give the radii. The off-diagonal entry is a measured angle term, distinct from either stretch.

20.10.2 B. Re-encode the same geometry

```
import sympy as sp

metric = sp.Matrix([[2, 1], [1, 2]])
basis = sp.Matrix([[1, 1], [0, 1]])
gram = basis.T*metric*basis
v = sp.Matrix([3, 2])
coords = basis.inv()*v
assert (coords.T*gram*coords)[0] == (v.T*metric*v)[0]
print("New Gram matrix:", gram.tolist(), "coordinates:", list(coords))
print("Squared physical length:", (v.T*metric*v)[0])
```

```
New Gram matrix: [[2, 3], [3, 6]] coordinates: [1, 2]
Squared physical length: 38
```

Congruence preserves the quadratic value on the same physical vector. Reusing the old matrix with new coordinates would compute a different value.

20.10.3 C. Represent one complex measurement in two metrics

```
c = sp.Matrix([1, sp.I])
complex_metric = sp.Matrix([[2, sp.I], [-sp.I, 2]])
for weight in [sp.eye(2), complex_metric]:
    representative = weight.inv()*c.conjugate()
    assert representative.H*weight == c.T
    print("Representative:", list(representative),
          "squared dual norm:", (representative.H*weight*representative)[0].simplify())
```

```
Representative: [1, -I] squared dual norm: 2
Representative: [1/3, -I/3] squared dual norm: 2/3
```

The second matrix is HPD (leading minors $2, 3$). Equality of rows proves equality on all inputs. Compare each reported norm with the supremum on that metric's unit ball.

20.10.4 D. Pull back through different domain and codomain geometries

```
A = sp.Matrix([[1, 2, 0], [0, 1, 1]])
domain = sp.diag(1, 2, 3)
target = metric
adj = domain.inv()*A.T*target
assert adj.T*domain == target*A
for u, y in [(sp.Matrix([1, 2, 3]), sp.Matrix([1, -1])),
             (sp.Matrix([0, -1, 2]), sp.Matrix([3, 0]))]:
    assert (y.T*target*A*u)[0] == ((adj*y).T*domain*u)[0]
print("Weighted adjoint:", adj.tolist())
print("Transpose:", A.T.tolist())
```

```
Weighted adjoint: [[2, 1], [5/2, 2], [1/3, 2/3]]
Transpose: [[1, 0], [2, 1], [0, 1]]
```

The first assertion checks the identity for all inputs and output directions. The two numerical substitutions also illustrate the order of applications for a rectangular map.

20.10.5 E. Two projections onto the same plane

```
plane = sp.eye(3)[:, :2]
Porth = plane*(plane.T*plane).inv()*plane.T
Poblique = sp.Matrix([[1, 0, -1], [0, 1, 0], [0, 0, 0]])
v3 = sp.Matrix([2, 1, 2])
for projector in [Porth, Poblique]:
    assert projector**2 == projector and projector*plane == plane
    residual = v3-projector*v3
    print("Point:", list(projector*v3), "squared distance:", residual.dot(residual),
          "residual readings on plane:", list(plane.T*residual))
```

```
Point: [2, 1, 0] squared distance: 4 residual readings on plane: [0, 0]
Point: [0, 1, 0] squared distance: 8 residual readings on plane: [2, 0]
```

Both maps fix the plane. At this input the squared distances are 4 and 8. The second rule discards direction $(1, 0, 1)^T$; only the first residual has zero pairing with both plane basis vectors.

In Figure 20.2 we intersect the plane with the slice $y = 1$ through the input. The plane appears edge-on. Its perpendicular direction is vertical in this slice; the discarded oblique direction has equal horizontal and vertical components. The extra squared distance is the squared separation of the two feet.

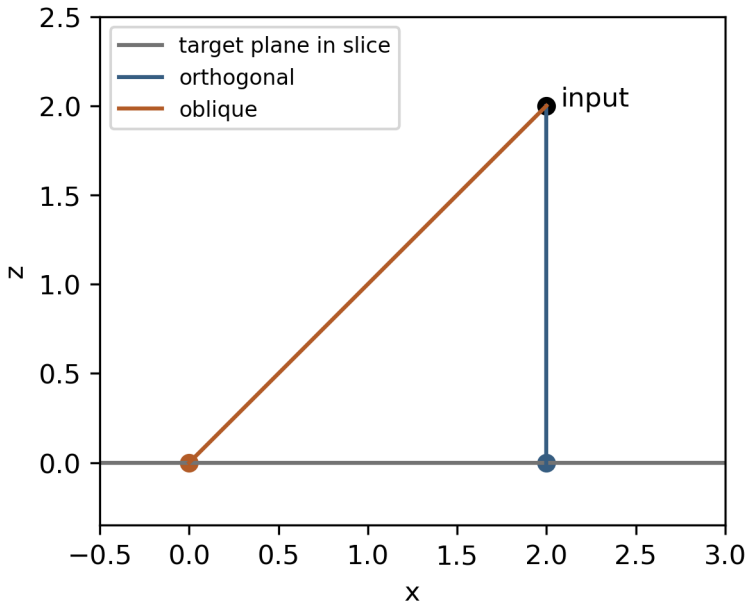


Figure 20.2: Cross-section of the two projections at y = 1. Distances are drawn at equal scale.

20.10.6 F. Compare normal equations, QR and SVD

```
def compare_solvers(design, rhs, reference):
    Q, R = np.linalg.qr(design, mode="reduced")
    estimates = {"QR": np.linalg.solve(R, Q.T @ rhs),
                 "SVD": np.linalg.pinv(design) @ rhs}
    try:
        estimates["normal"] = np.linalg.solve(design.T @ design, design.T @ rhs)
    except np.linalg.LinAlgError:
        print("Rounded normal matrix is singular.")
    print(f"cond(A) = {np.linalg.cond(design):.3e}")
    for method, estimate in estimates.items():
        err = np.linalg.norm(estimate-reference)
        res = np.linalg.norm(design @ estimate-rhs)
        print(f"{method:6s}  coefficient error {err:.3e}   residual {res:.3e}")

design = np.array([[1., 0.], [1., 0.], [0., 1.]])
compare_solvers(design, np.array([1., 3., 2.]), np.array([2., 2.]))
t = np.linspace(-1., 1., 40)
near = np.column_stack((np.ones_like(t), 1.+1e-6*t))
reference = np.array([1., -1.])
compare_solvers(near, near @ reference, reference)
```

```
cond(A) = 1.414e+00
QR      coefficient error 2.220e-16   residual 1.414e+00
SVD     coefficient error 6.661e-16   residual 1.414e+00
normal  coefficient error 0.000e+00   residual 1.414e+00
cond(A) = 3.379e+06
QR      coefficient error 3.366e-11   residual 0.000e+00
SVD     coefficient error 3.366e-11   residual 4.965e-16
normal  coefficient error 9.094e-04   residual 2.408e-09
```

For the first problem all three methods should attain residual $\sqrt{2}$, which is forced by the data. The second problem uses nearly parallel columns and consistent constructed data. The response along $(1, -1)^T$ is tiny. Compare the absolute coefficient errors even if all residuals appear small. Last digits can vary by platform; the predicted vulnerability is squaring the singular-value ratio in the normal system.

20.10.7 G. Select the minimum-norm input for incompatible data

```
sensor = sp.Matrix([[1, 2, -1, 3], [2, 4, 1, 9], [3, 6, 0, 12]])
N = sp.Matrix.hstack(*sensor.nullspace())
bad = sp.Matrix([1, 2, 4])
x0 = sp.Matrix([11, 0, -1, 0])/9
Pk = N*(N.T*N).inv()*N.T
xmin = sensor.pinv()*bad
assert xmin == (sp.eye(4)-Pk)*x0
assert N.T*xmin == sp.zeros(2, 1)
assert sensor.T*(bad-sensor*xmin) == sp.zeros(4, 1)
z = sp.Matrix(sp.symbols("s t"))
assert (sensor*(xmin+N*z)-sensor*xmin).applyfunc(sp.simplify) == sp.zeros(3, 1)
print("Minimum-norm input:", list(xmin))
print("Fitted data:", list(sensor*xmin), "residual:", list(bad-sensor*xmin))
```

```
Minimum-norm input: [1/9, 2/9, -5/18, 1/6]
Fitted data: [4/3, 7/3, 11/3] residual: [-1/3, -1/3, 1/3]
```

The normal equations alone accept the whole two-parameter family $x^\dagger + Nz$. The additional two independent conditions $N^T x^\dagger = 0$ select a single input. They characterize the pseudoinverse solution independently of the algorithm used to compute it.

20.10.8 H. Assemble the Poisson equation and measure the slope error

```
def poisson_matrices(elements):
    h = 1/elements
    n = elements+1
    stiffness = np.zeros((n, n))
    mass = np.zeros((n, n))
    load = np.zeros(n)
    for j in range(elements):
        element = slice(j, j+2)
        stiffness[element, element] += np.array([[1., -1.], [-1., 1.]])/h
        mass[element, element] += h*np.array([[2., 1.], [1., 2.]])/6
        load[element] += h/2
    return stiffness[1:-1, 1:-1], mass[1:-1, 1:-1], load[1:-1]

def poisson_solution(elements):
```

```
    K, M, load = poisson_matrices(elements)
    values = np.r_[0., np.linalg.solve(K, load), 0.]
    return np.linspace(0., 1., elements+1), values

for count in [3, 6, 12]:
    nodes, vals = poisson_solution(count)
    h = 1/count
    mids = (nodes[1:]+nodes[:-1])/2
    slopes = np.diff(vals)/h
    # Two-point Gauss integrates the squared affine slope error exactly.
    offsets = np.array([-1., 1.])*h/(2*np.sqrt(3))
    diff = .5-(mids[:, None]+offsets)-slopes[:, None]
    energy_error = np.sqrt(h/2*np.sum(diff**2))
    assert np.isclose(energy_error, h/np.sqrt(12))
    print(count, "elements; energy error", round(energy_error, 7))
```

```
3 elements; energy error 0.096225
6 elements; energy error 0.0481125
12 elements; energy error 0.0240563
```

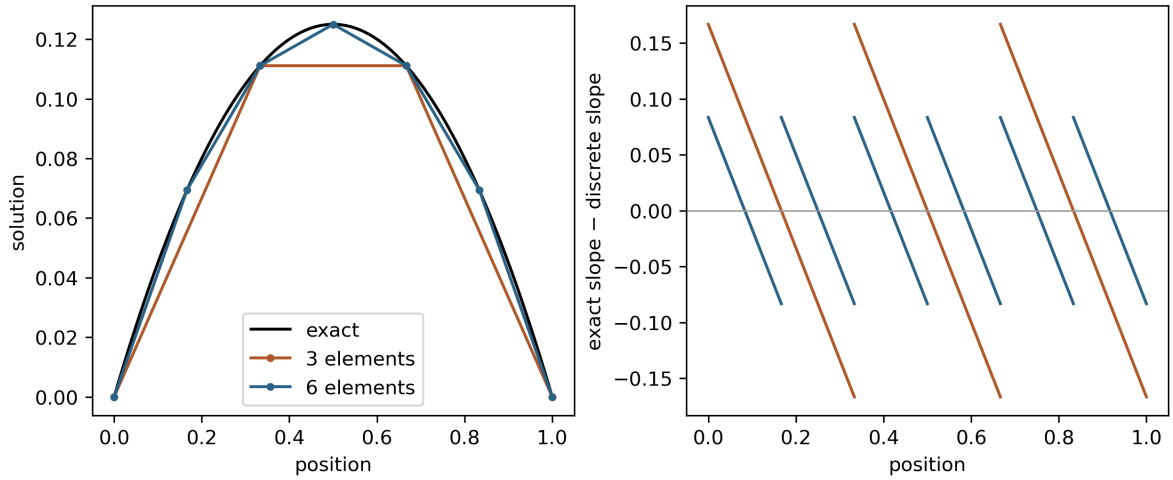


Figure 20.3: Poisson solution and its piecewise linear Galerkin approximations. The right panel shows the slope error, which persists between exact nodal values.

Each segment of the slope-error graph in Figure 20.3 integrates to zero. Halving the mesh length halves the energy error, even though nodal-value comparisons alone would already report machine-level accuracy.

20.10.9 I. Compute generalized eigenpairs and test a proper trial subspace

```
def hermitian_pair(K, M):
    L = np.linalg.cholesky(M)
    left = np.linalg.solve(L, K)
    transformed = np.linalg.solve(L, left.T).T
    vals, Z = np.linalg.eigh(transformed)
    return vals, np.linalg.solve(L.T, Z)

K, M, load = poisson_matrices(3)
vals, modes = hermitian_pair(K, M)
assert np.allclose(modes.T @ M @ modes, np.eye(2))
assert np.allclose(K @ modes, (M @ modes)*vals)
Q = np.array([[1.], [0.]])
ritz, small_modes = hermitian_pair(Q.T @ K @ Q, Q.T @ M @ Q)
xr = Q @ small_modes[:, 0]
rr = K @ xr-ritz[0]*(M @ xr)
assert np.allclose(Q.T @ rr, 0.) and np.linalg.norm(rr) > 1.
print("Generalized eigenvalues:", vals, "one-dimensional Ritz value:", ritz)
print("Residual per unit first coefficient:", rr/xr[0])
```

```
Generalized eigenvalues: [10.8 54. ] one-dimensional Ritz value: [27.]
Residual per unit first coefficient: [ 8.37382645e-16 -4.50000000e+00]
```

This routine uses real symmetric matrices, so transposes in the implementation are ordinary transposes. The returned vectors are normalized in M. Dividing the residual by the first coefficient recovers the hand calculation for the unnormalized e_1. Neither normalization alters the Ritz value.

20.11 Exercises — Reconstruct, calculate, transfer

20.11.1 Geometry from measurements

1. **Identify an inner product.** On $\mathbb{R}^2$ let $g_t(x,y) = y^T \left(\begin{smallmatrix} 1 & t \\ t & 1 \end{smallmatrix}\right) x$. For which real t is this an inner product? Determine when the coordinate axes are perpendicular. Give a nonzero null direction at each endpoint of the admissible interval.
2. **Rebuild the inequalities.** Minimize $\|x - \alpha y\|^2$ over complex α and deduce Cauchy–Schwarz, its equality case and the triangle inequality. Use the parallelogram law to test $\|\cdot\|_\infty$ on $\mathbb{R}^2$.

3. **A dual basis in an oblique grid.** In Euclidean $\mathbb{R}^2$ take $b_1 = (1,0), b_2 = (1,1)$. Find both Riesz representatives of the dual basis in physical coordinates and in basis coordinates. Explain why just normalizing the original axes cannot solve the problem.
4. **A weighted complex measurement.** For $f(x) = ix_1 + 2x_2$ and $G = \mathrm{diag}(4,1)$ find the Riesz representative, the dual norm and a maximizing unit input. Prove the existence and uniqueness statement for an arbitrary finite-dimensional functional.
5. **Adjoints without orthonormal coordinates.** Let T have standard real matrix $\mathrm{diag}(2,3)$ with Euclidean physical geometry. Use the oblique basis of problem 3 on both sides. Compute its coordinate matrix and its adjoint matrix there. Derive the general formula from the Riesz square.

20.11.2 Approximation and operator structure

6. **Detect a closest point.** Project $(2,1)$ onto the horizontal axis first in the Euclidean metric and then in $G = \left(\begin{smallmatrix} 2 & 1 \\ 1 & 2 \end{smallmatrix}\right)$. Give an oblique projector that returns a third answer. Prove the best-approximation criterion for any subspace in finite dimension.
7. **Reconstruct the four subspaces.** For $T(x,y,z) = (x+y, z)$ with Euclidean metrics compute the four spaces in the orthogonality theorem. Reprove the result in general. Identify exactly the step that gives only a closure in Hilbert space.
8. **Positive versus invertible.** For the M, K pair at the end of Session 6, verify weighted self-adjointness and find the two eigenvalues and the range of the generalized Rayleigh quotient. Give an invertible self-adjoint matrix with a nonzero vector of zero quadratic energy.
9. **Inconsistent but unique.** For $A = (1,1)^T$ and $b = (0,3)^T$ compute the minimizer in output weight $W = \mathrm{diag}(1,w)$, $w > 0$. Derive the weighted normal equations and explain what happens as w increases.
10. **A second source of nonuniqueness.** For $A = \left(\begin{smallmatrix} 1 & 1 \\ 2 & 2 \\ 0 & 0 \end{smallmatrix}\right)$, $b = (1,2,3)^T$, find every least-squares minimizer. Select the shortest in Euclidean input geometry, then the shortest in $\mathrm{diag}(4,1)$. Explain why the fitted reading does not change.
11. **A vanished normal-equation pivot.** For columns $(1,0,0)^T$ and $(1,\epsilon,0)^T$, compute the Gram determinant in exact arithmetic. Explain what can happen if $1+\epsilon^2$ rounds to one. What quantity is still present in the input available to QR?

20.11.3 From tests to modes

12. **Two variational geometries.** Assemble the one-interior-hat stiffness, mass and load for $-u'' = 1$ with homogeneous Dirichlet data on the mesh $0, 1/2, 1$. Find the Galerkin coefficient and its generalized Rayleigh quotient. Prove the energy best-approximation property for any conforming space.

13. **The missing boundary condition.** For homogeneous Neumann data determine which of $f = 1$, $f = 2t-1$ admit weak solutions. For the compatible case construct the solution family, describe the quotient and give the zero-mean representative.
14. **A Ritz value invisible to one test.** For the two-hat K, M from Session 9 choose trial vector $q = (1, 2)^T$. Find θ, the full residual r, and verify its test equation. Decide whether q is a full eigenvector.
15. **The price of a nonclosed range.** For the diagonal operator $(Tx)_j = x_j/j$ on ℓ^2, produce data for which the least-squares infimum is zero but there is no minimizer. Explain which orthogonality criterion still holds and what fails.

20.12 Worked solutions

1. The eigenvectors $(1, 1)$ and $(1, -1)$ have eigenvalues $1 + t$ and $1 - t$. Decomposition along these perpendicular directions proves positivity precisely for $-1 < t < 1$. Bilinearity and symmetry hold for every t. The coordinate pairing is t, so the axes are perpendicular exactly at $t = 0$. At $t = 1$ the difference direction is null; at $t = -1$ the sum direction is null.

2. For $y \neq 0$ choose $\alpha_0 = \langle x, y\rangle/\|y\|^2$. Then $z = x-\alpha_0 y \perp y$, and $\|x-\alpha y\|^2 = \|z\|^2+|\alpha_0-\alpha|^2\|y\|^2$, proving the minimizing claim. At $\alpha = 0$ this yields $|\langle x, y\rangle|^2 \leq \|x\|^2\|y\|^2$, equality iff $z = 0$. If $y = 0$ equality always holds. Expanding $\|x + y\|^2$ and bounding twice the real part gives the triangle inequality. For e_1, e_2 in the max norm the parallelogram left side is 2, while the right side is 4. This norm cannot be an inner-product norm.

3. The dual basis rows in physical coordinates are $(1, -1), (0, 1)$, the rows of B^{-1}. In Euclidean geometry the representing columns are their transposes. Applying B^{-1} to them gives $(2, -1)^T, (-1, 1)^T$, the columns of $(B^TB)^{-1}$. The first representative must kill b_2 and the second must kill b_1. Rescaling a basis vector can set its own length, but cannot turn its nonzero pairing with the other axis into zero.

4. $Ga_f = (-i, 2)^T$ gives $a_f = (-i/4, 2)^T$. The squared norm is $1/4 + 4 = 17/4$, so $2a_f/\sqrt{17}$ is a maximizing input of unit length. For existence expand in an orthonormal basis and use conjugated coefficients $\overline{f(q_i)}$ for the representative; pairing with each q_i returns $f(q_i)$, hence the two functionals agree by linearity everywhere. The difference of two possible representatives pairs to zero with itself, proving uniqueness.

5. Directly $A = B^{-1}\,\mathrm{diag}(2, 3)B = \left(\begin{smallmatrix} 2 & -1 \\ 0 & 3 \end{smallmatrix}\right)$. From $T^* = R_V^{-1}T^\vee R_W$, its output on y represents the functional $x \mapsto y^H G_W Ax$. Therefore $G_V[T^*]y = A^H G_W y$. Here $G_V = G_W = B^TB$ and $G_V^{-1}A^TG_V = A$. This agrees with physical self-adjointness while $A^T \neq A$. Using an ordinary transpose in this basis would represent the wrong operator.

6. The Euclidean foot is $(2, 0)$. In G, orthogonality to e_1 means $2(2 - c) + 1 = 0$, giving $(5/2, 0)$. The idempotent $(x, y) \mapsto (x - y, 0)$ gives $(1, 0)$; its discarded direction is $(1, 1)$. Generally construct p by summing the components of x on an orthonormal basis of the target.

Then $x-p \perp S$, and the squared distance to any other s increases by $\|p-s\|^2$. Conversely any minimizer must equal p by this strict increase. This proves both necessity and sufficiency.

7. The kernel is spanned by $(1,-1,0)$, and the image is $\mathbb{R}^2$. The adjoint sends (a,b) to (a,a,b), so its kernel is zero and its image is the plane $x=y$. In general $z \in \ker T^*$ iff $\langle v, T^* z\rangle = 0$ for all v iff $z \perp \operatorname{im} T$. Apply this to T^* for the second kernel identity; taking perpendiculars gives the image identities because finite-dimensional subspaces equal their double perpendiculars. In a Hilbert space the double perpendicular recovers the closure. Problem 15 shows how this changes solvability.

8. $T = M^{-1}K$ has $MT = K = K^H = T^H M$. The characteristic equation for the pair is $(2-2\lambda)(2-\lambda)-1=0$, giving $\lambda_\pm = (3 \pm \sqrt{3})/2$. The quotient ranges over the closed interval between these values, since in an M-orthonormal eigenbasis it is a convex combination with weights $|c_i|^2 / \sum |c_i|^2$; varying the two coefficients attains every weight. The Euclidean matrix $\operatorname{diag}(1,-1)$ has nonzero determinant and zero energy on $(1,1)^T$.

9. The minimized expression is $|x|^2 + w|x-3|^2$. Vanishing derivative (or weighted perpendicularity) gives $x = 3w/(1+w)$. More generally, the fitted residual is W-perpendicular to every Ah iff $h^H A^H W(Ax-b) = 0$ for every h. This is equivalent to the weighted normal equations. The column is independent, giving uniqueness despite incompatibility. As w increases the fit approaches the second reading.

10. The first two rows are compatible when $x_1 + x_2 = 1$; the unavoidable third residual is 3. Hence the full line is minimizing. Its Euclidean shortest point is $(1/2, 1/2)$ by perpendicularity to $(1,-1)$. In the weighted input norm the condition is $4x_1 - x_2 = 0$. Solving with the sum condition gives $(1/5, 4/5)$. Both map to $(1,2,0)$ because changing the representative along the kernel cannot change a fitted reading.

11. The Gram matrix has rows $(1,1), (1,1+\epsilon^2)$, so the determinant is ϵ^2. If the last diagonal rounds to one, the explicitly formed matrix is singular even for $\epsilon \neq 0$. This is consistent with squaring the condition number. QR applied to the original columns has the independent second-row entry ϵ available: subtracting the first column from the second leaves $(0,\epsilon,0)^T$ in exact arithmetic. This advantage does not remove the original input sensitivity when ϵ is tiny.

12. The hat slopes are $2, -2$, giving $K = 4$. The integrated squared hat is $1/3$, so $M = 1/3$; the area gives load $1/2$. The Galerkin coefficient is $1/8$ and the generalized quotient is 12. For any conforming subspace, subtract the test equations to obtain $a(u-u_h, v_h) = 0$ for all tests there. Testing against $u_h - w_h$ kills the cross term in the energy expansion. The squared error for w_h exceeds that for u_h by $\|u_h - w_h\|_a^2$. Positivity gives uniqueness.

13. Testing with the constant 1 excludes $f = 1$. For $2t-1$ the integral vanishes. Integration gives $u'(t) = t - t^2$, zero at both endpoints, and $u(t) = t^2/2 - t^3/3 + C$. The integral of the nonconstant part is $1/6 - 1/12 = 1/12$, hence zero mean fixes $C = -1/12$. Adding any constant gives the same weak equation. The energy norm is definite on classes modulo constants.

14. We have $Kq = (0, 9)^T$ and $Mq = (6, 9)^T/18$. Hence $\theta = 18/(24/18) = 27/2$. The residual is $(-9/2, 9/4)^T$, whose pairing with $(1, 2)$ is zero. Its second component is nonzero, so this is a Ritz vector only. An exact eigenvector would pass every test in the full two-dimensional space.

15. Set $b_j = 1/j$. Truncation to the first n entries produces $b^{(n)} = Tx^{(n)}$ for an input with n entries equal to one and the rest zero. Then $\|b - b^{(n)}\|^2 = \sum_{j>n} j^{-2} \to 0$, while no ℓ^2 input with every entry equal to one exists. The range has zero perpendicular and the adjoint has zero kernel, so the kernel identity holds. Double perpendicular gives the whole space, the closure of the range. If a minimizer existed it would have zero residual and yield the excluded preimage.

20.13 Integrated mastery problem — One instrument, two geometries, one reduced model

On the two-interior-hat mesh, an instrument reads the displacement at the first interior node: $F(v) = v(1/3)$. In nodal coordinates its row is $(1, 0)$. Compare the following operations.

1. Construct its Riesz representatives and dual norms in mass and energy geometry. Interpret the adjoint of the instrument in both metrics with ordinary scalar output.
2. Apply this point load and solve the energy equation. Restrict the calculation to the one-dimensional trial space spanned by $q = (1, 1)^T$. Derive the reduced displacement, its error and the residual load. Distinguish a residual **functional** from its energy **representative**.
3. Compute the generalized Ritz value for the same trial space. Explain why this trial space gives an exact vibration mode although it only approximates the static response. Decide which conclusions change under rescaling of q.

20.13.1 Worked reconstruction

From Session 9 take $K = \left(\begin{smallmatrix} 6 & -3 \\ -3 & 6 \end{smallmatrix}\right)$ and $M = \frac{1}{18}\left(\begin{smallmatrix} 4 & 1 \\ 1 & 4 \end{smallmatrix}\right)$. Put $d = e_1$. On the finite element space, the Riesz vectors are

$$a_K = K^{-1}d = (2, 1)^T/9, \qquad a_M = M^{-1}d = (24, -6)^T/5.$$

Their own squared norms equal their first components, since $a^H G a = d^H a$. The dual norms are therefore $\sqrt{2}/3$ and $\sqrt{24/5}$. In either geometry the adjoint of the scalar instrument sends t to $t a_G$. These are finite-dimensional mass and energy representatives. On the full L^2 space point evaluation is not even well-defined on equivalence classes modulo changes on sets of measure zero. On the one-dimensional endpoint-zero energy space it is continuous by the estimate in Session 8.

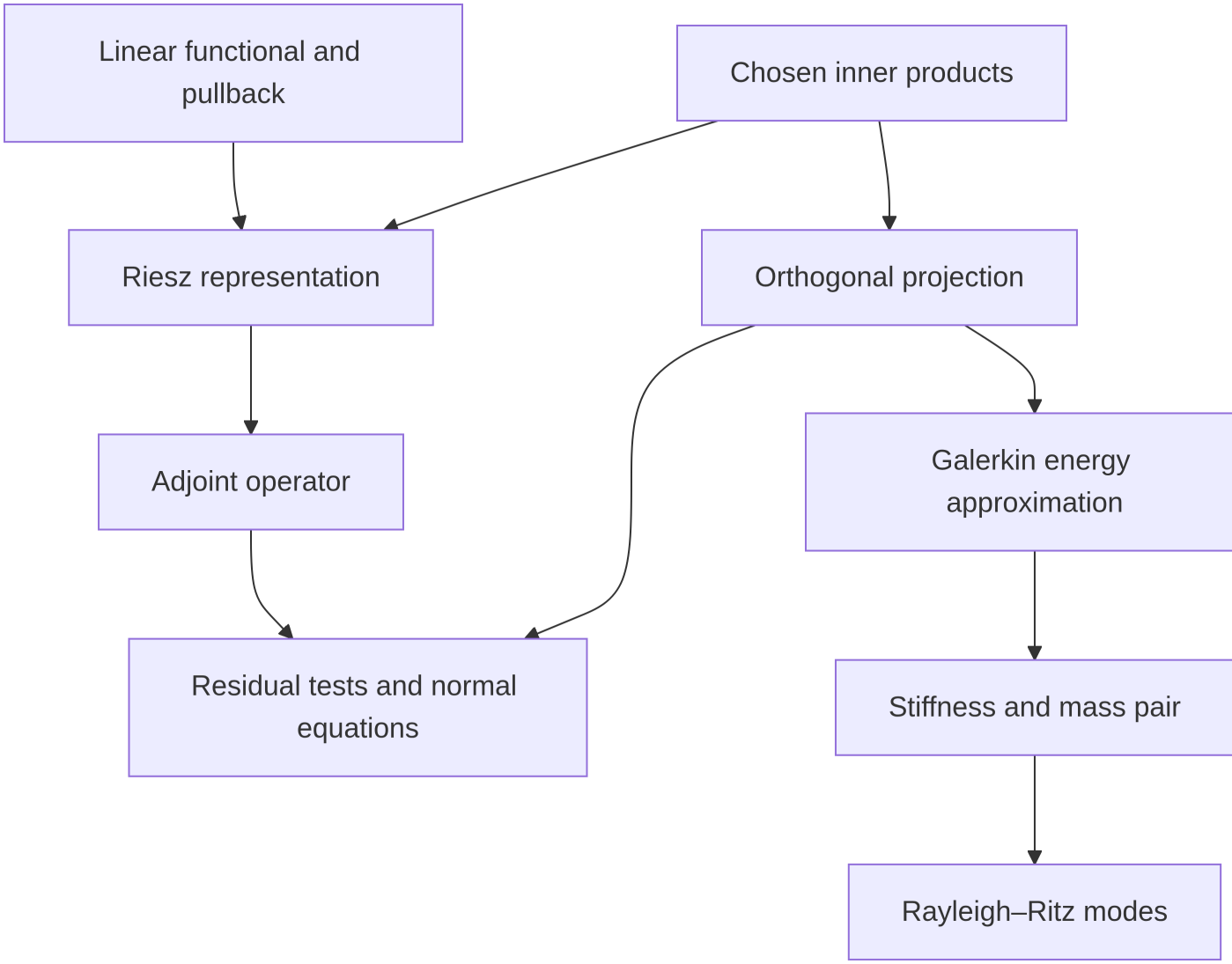
The static response solves $Ku = d$, so $u = a_K$. The reduced response is $u_Q = cq$, where $(q^T Kq)c = q^T d$. Here $Kq = 3q$, giving $c = 1/6$. Hence

$$u - u_Q = (1, -1)^T/18, \qquad d - Ku_Q = (1, -1)^T/2.$$

The residual row represented in algebraic coefficient coordinates by $d - Ku_Q$ vanishes on q. Solving with K maps that functional to the error vector. Its energy dual norm equals the error's energy norm, since this solve is Riesz representation. In fact $\|u-u_Q\|_K^2 = (u-u_Q)^T(d-Ku_Q) = 1/18$. Every other displacement on the trial line has greater energy error by the projection theorem.

For vibration, $Mq = (5/18)q$, so $Kq = (54/5)Mq$. The Ritz residual vanishes on *every* test. The one-dimensional trial space contains this exact mode, but the load also excites the antisymmetric mode: the load has $F((1, -1)^T) = 1$. This missing direction accounts for the static error. Replacing q by αq, $\alpha \neq 0$, rescales c by $1/\alpha$ and both reduced Gram entries by $|\alpha|^2$. It changes neither displacement, eigenvalue nor either residual in the original space.

20.14 Synthesis — Which structure made each conclusion possible?



Projection occurs more than once, with different ambient spaces. Least squares projects noisy **data** onto attainable outputs. Minimum norm projects an **input representative** off the kernel. Galerkin projects an unknown **function** in the energy geometry using only load measurements. Rayleigh–Ritz projects an **operator action** and tests an eigenpair residual. Specifying what is being projected and which pairing tests the residual determines the appropriate equation in each case.

Construction	Conclusion	Data or assumptions required
Algebraic complement H of S	Unique decomposition into S and H	A choice with $V = S \oplus H$
Annihilator S^0	All functionals insensitive to S	S inside V, no metric
Orthogonal complement $S^\perp$	Minimum-length representatives modulo S	Inner product; finite dimension or closed S in Hilbert space
Algebraic dual $T^\vee$	Composition $g \mapsto gT$, represented by A^T	Linear map and corresponding dual bases for coordinates
Adjoint T^*	Pullback represented on vectors	Inner products on both spaces; bounded operator in Hilbert setting
Cauchy–Schwarz and Riesz	Norm of a measurement equals length of its representative	Positive definite product; continuous dual and completeness in Hilbert Riesz
Gram change	$G' = P^H G P$	Invertible change of basis for the same geometry
Orthogonal projection	Unique best approximation, idempotent and self-adjoint	Subspace in finite dimension; closed subspace in Hilbert space
Spectral theorem used here	Orthonormal eigenbasis and real spectrum	Finite-dimensional self-adjoint operator
Least squares and pseudoinverse	Unique fitted data, unique shortest minimizing input	Finite dimensions; unique minimizer without tie-break iff injective
Galerkin energy projection	Error is energy-perpendicular to conforming trial space	Positive symmetric energy form, same trial and test space
Generalized Rayleigh–Ritz	Reduced Hermitian pair and residual test $Q^H r = 0$	$K = K^H$, M HPD, independent trial columns

What is determined? A functional and its kernel are determined by the linear rule. Its representative and its norm are determined after choosing geometry. The same dependence holds for adjoints, perpendicular complements and closest points. Once the geometry and subspace are fixed, the orthogonal projection requires no further basis choice. Bases, matrix entries and Cholesky factors are coordinate descriptions. Module F2's natural evaluation $V \to V^{\vee\vee}$ uses no metric; the Riesz identification with the single dual instead respects maps preserving the chosen geometry.

Common error	A calculation that detects it
Treating a Gram change as a change of physical length	Transform coordinates as well and compare $x^H G x$
Identifying covector coefficients and vector coordinates	Solve $Ga = \bar{c}$ and test $a^H G x = c^T x$
Calling every idempotent orthogonal	Test $P^H G = GP$ in the specified metric
Inferring a small solution error from a small residual	Check the smallest resolved stretch and any kernel component
Inferring zero PDE error from exact nodal values	Measure slopes or the energy integral between nodes
Interpreting a Ritz residual with the wrong metric	Distinguish $Q^H r = 0$ from $Q^H M r = 0$; identify whether r is a load or a vector
Claiming existence of a closest point from an infimum	In infinite dimensions check closedness of the approximating subspace

20.14.1 Mastery checklist

- ☐ Reconstruct Cauchy–Schwarz and the triangle inequality from a one-dimensional projection.
- ☐ Compute a Gram matrix, transform it under a basis change and distinguish this from changing the metric.
- ☐ Prove Riesz representation and calculate a weighted gradient and dual norm, including complex conjugation.
- ☐ Derive an adjoint from algebraic pullback and verify its defining identity between different metrics.
- ☐ Rebuild the four fundamental subspace identities and identify where closure enters.
- ☐ Construct an orthogonal projector, characterize it algebraically and prove its best-approximation property.
- ☐ Recognize weighted self-adjointness and positivity, and explain the existence of perpendicular eigenmodes.
- ☐ Derive both normal-equation proofs, describe every minimizer and select the minimum-norm one.
- ☐ Explain the difference in conditioning between QR and explicitly forming a Gram matrix.
- ☐ Derive the weak Poisson equation, assemble the stiffness and mass matrices, and prove Galerkin orthogonality.
- ☐ Derive generalized Rayleigh–Ritz and distinguish residual orthogonality from an exact eigenpair.

Transition to Module D. In Euclidean orthonormal eigen-coordinates a Hermitian operator has $A = U \Lambda U^H$, hence $p(A) = \sum_i p(\lambda_i) u_i u_i^H$. The matrices $u_i u_i^H$ project onto perpendicular modes. Summing over a repeated eigenvalue gives the projector onto its entire eigenspace,

independent of which orthonormal basis spans it. Module D asks how much of that subspace can be identified from approximate data and how a polynomial filter can amplify it before Rayleigh–Ritz extraction.

21 Module D — Spectral Theory

A vibration experiment records a mixture of modes. A single measurement gives a vector x, and applying the stiffness operator gives another vector Ax. What would let us extract a whole family of modes from such products? We need three ingredients. A decomposition tells us what a mode is. Geometry tells us how to measure the error in a proposed family. A polynomial changes the mixture in a predictable way using only applications of A. We will derive all three and use them to design a filtered subspace iteration.

We will return to the matrix shared with Module E,

$$A_0 = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 9 \end{pmatrix}.$$

The first two coordinates interact. Equal and opposite displacements in these coordinates behave independently. Finding the right coordinates turns this observation into a calculation. Later we will deliberately mix those coordinates again to test what an approximate eigensolver can certify.

Conventions. Until Session 10 all spaces are finite dimensional over $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. In orthonormal coordinates use $\langle x, y\rangle = y^*x$, linear in x, as in the notes. A star on a matrix denotes conjugate transpose. The adjoint satisfies $\langle Ax, y\rangle = \langle x, A^\dagger y\rangle$. In coordinates with domain and range Gram matrices G, H, this gives $A^\dagger = G^{-1}A^*H$. In orthonormal coordinates we write A^* for the adjoint itself. $\|\cdot\|$ denotes the Euclidean or induced operator 2-norm. $\|\cdot\|_F$ is the square root of the sum of squared entry magnitudes. An orthonormal basis matrix for a subspace has columns of unit length and $Q^*Q = I$.

Study sessions	Question to answer on paper
1–2	What do a restriction, a quotient and a dual coordinate each remember?
3–4	When do the coordinates become orthogonal, and how does a polynomial act on them?
5–7	How far is a trial space from a target, and what can one infer without knowing the target?
8	Which directions of a rectangular map are amplified or lost?

Study sessions	Question to answer on paper
9	What exactly contracts during a filtered block iteration?
10	Which parts survive a passage to functions or a change of scalar ring?

21.1 Session 1 — Finding structure that survives an application of the operator

A line $L = \mathbb{F}v$ is preserved by an endomorphism $T : V \to V$ precisely when $Tv = \lambda v$ for some scalar λ and $v \neq 0$. Preservation means $T(L) \subseteq L$ and includes collapse to zero. For a real line, a negative eigenvalue reverses direction. A rectangular matrix maps between two coordinate spaces. Identifying an input vector with a scalar multiple of the output requires extra structure.

The eigenspace $E_\lambda = \ker(T - \lambda I)$ contains zero as well as all eigenvectors for λ. In coordinates λ is an eigenvalue exactly when $\chi_A(\lambda) = 0$, where $\chi_A(t) = \det(tI - A)$. This follows because a square map has nontrivial kernel precisely when its determinant vanishes. Similarity preserves χ_A, since $tI - S^{-1}AS = S^{-1}(tI - A)S$.

A field experiment. For a real plane rotation through ϕ, $\chi(t) = t^2 - 2\cos(\phi)t + 1$. If $\sin\phi \neq 0$, no real line is invariant. Over $\mathbb{C}$, the real rotation matrix with first column $(\cos\phi, \sin\phi)^T$ sends $(1, -i)^T$ to $e^{i\phi}(1, -i)^T$. Real and imaginary parts span the rotating real plane. Conjugation supplies the other eigenpair. Thus a useful real invariant piece can have dimension two and contain no real eigenline.

21.1.1 What the off-diagonal block measures

Proposition. A subspace U is invariant if and only if in a basis starting with a basis of U the matrix has the form

$$\begin{pmatrix} B & C \\ 0 & D \end{pmatrix}.$$

Proof. The first columns are images of the basis of U. Their lower coordinates vanish precisely when every such image lies in U. The upper-left block represents $T|_U$. On the quotient, define $\bar{T}(v + U) = Tv + U$. Replacing v by $v + u$ changes Tv by $Tu \in U$, so the formula is well-defined. Modulo U, the remaining basis vectors give a basis and their images have coordinates D. The block C records components along U acquired by the chosen representatives. Its vanishing means their span is also invariant. Conversely two complementary invariant spaces give a block diagonal matrix. □

In Module A language the quotient map π satisfies $\pi T = \bar{T}\pi$. In Module B language the annihilator $U^\circ = \{\ell : \ell|_U = 0\}$ is invariant under the dual map $\ell \mapsto \ell T$ because $\ell(Tu) = 0$. Functionals in U° measure exactly those quantities that descend to V/U.

Write $g_\lambda = \dim E_\lambda$ and a_λ for the multiplicity of λ as a root of χ_A. Taking $U = E_\lambda$ gives $B = \lambda I_{g_\lambda}$, so factoring the block triangular determinant gives $g_\lambda \le a_\lambda$. The inequality can be strict. For $J = \left(\begin{smallmatrix} 2 & 1 \\ 0 & 2 \end{smallmatrix}\right)$ both restriction to $\mathbb{F}e_1$ and quotient by that line act as 2. Yet $Je_2 = 2e_2 + e_1$. An invariant complementary line would supply a second eigenvector, whereas $\ker(J - 2I) = \mathbb{F}e_1$. This *defective* matrix demonstrates exactly what is lost by recording only the two diagonal blocks. To obtain a proper invariant plane without an eigenbasis, embed J as a diagonal block in $\operatorname{diag}(J, 5)$.

21.2 Session 2 — Reading spectral components with the dual basis

When does every vector have unique spectral coordinates? Suppose $\lambda_1, \dots, \lambda_s$ are distinct and $\sum_j u_j = 0$, $u_j \in E_{\lambda_j}$. Apply $\prod_{i \ne j}(T - \lambda_i I)$. All but one term vanish, leaving $\prod_{i \ne j}(\lambda_j - \lambda_i)u_j = 0$. Hence the sum of distinct eigenspaces is direct.

Diagonalization criterion. The following are equivalent over the chosen field: $A = X\Lambda X^{-1}$ with Λ diagonal; there is a basis of eigenvectors; $V = \bigoplus_j E_{\lambda_j}$. Indeed $AX = X\Lambda$ says column by column that the columns of an invertible X are an eigenbasis. Grouping these columns by eigenvalue yields the decomposition. Conversely concatenate bases of the summands. When χ_A splits, the multiplicities sum to n, so equality $g_\lambda = a_\lambda$ for every root is equivalent to diagonalizability.

Write $X = [v_1, \dots, v_n]$ and let ℓ_i be row i of X^{-1}. The identities $\ell_i v_j = \delta_{ij}$ and $\ell_i A = \lambda_i \ell_i$ say that the dual basis measures modal amplitudes and consists of left eigenfunctionals. Its representing columns satisfy $A^* \ell_i^* = \overline{\lambda_i}\ell_i^*$. Define $D_i = v_i \ell_i$. Then

$$D_i D_j = \delta_{ij} D_i, \qquad \sum_i D_i = XX^{-1} = I, \qquad A = \sum_i \lambda_i D_i.$$

Each multiplication here just extracts a coordinate and puts it back on its own basis vector. For a repeated eigenvalue, $P_\lambda = \sum_{\lambda_i = \lambda} D_i$ is the projection onto E_λ along the sum of the other eigenspaces. This prescription uniquely specifies P_λ even if the basis inside E_λ changes.

An oblique measurement. For $B = \left(\begin{smallmatrix} 1 & 2 \\ 0 & 3 \end{smallmatrix}\right)$ choose $v_1 = e_1$, $v_2 = (1, 1)^T$. The inverse basis matrix has rows $(1, -1)$ and $(0, 1)$. Thus $P_1(x, y) = (x - y, 0)$ projects parallel to the diagonal line, and $P_3(x, y) = (y, y)$. Applying the first projection to e_2 gives $-e_1$. Orthogonal projection onto the horizontal axis would give zero. As eigenlines approach each other the coordinate measurements can grow large. The bound $\|XDX^{-1}\| \le \kappa_2(X)\|D\|$ with $\kappa_2(X) = \|X\|\|X^{-1}\|$ quantifies the role of the eigenbasis, including its scaling.

In Figure 21.1 the two projections share their image. The dashed segments show the kernels through the input point. Changing the kernel changes which part of the vector counts as its horizontal component.

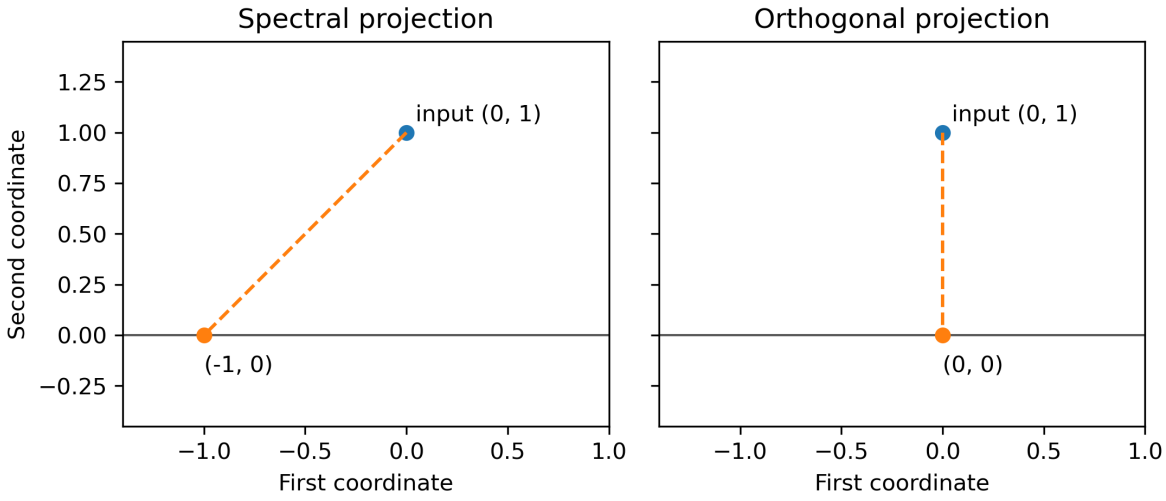


Figure 21.1: Projecting the same input onto the horizontal eigenline along two different complementary lines.

21.3 Session 3 — When can the modes be chosen orthogonal?

For generalized coordinates it costs effort to separate the two measurements above. An orthonormal eigenbasis makes coefficient extraction simply $q_i^* x$. What property of the original operator guarantees such a basis?

Hermitian spectral theorem. A self-adjoint operator on a finite-dimensional real or complex inner-product space has an orthonormal eigenbasis and real eigenvalues. Conversely a real diagonal array of eigenvalues in an orthonormal basis gives a self-adjoint operator.

Proof. Over $\mathbb{C}$ the characteristic polynomial has a root in positive dimension. If $Av = \lambda v$, then $v^* A v = \lambda \|v\|^2$ equals its conjugate, forcing $\lambda \in \mathbb{R}$. For real symmetric matrices apply this existence argument over $\mathbb{C}$; a nonzero real or imaginary part of the complex eigenvector is a real eigenvector for the same real λ. Normalize v. More generally, if U is invariant and $x \perp U$, then $\langle Ax, u\rangle = \langle x, Au\rangle = 0$ for $u \in U$. Thus $v^{\perp}$ is invariant and the restriction is self-adjoint with the inherited inner product. Induct on dimension and adjoin v to the resulting basis. Conversely $(Q\Lambda Q^*)^* = Q\Lambda Q^*$ when Λ is real diagonal. $\square$

For A_0 this produces $q_1 = (1,-1,0)^T/\sqrt{2}$, $q_2 = (1,1,0)^T/\sqrt{2}$, $q_3 = e_3$ with eigenvalues $1, 3, 9$. Orthogonality also follows directly for distinct eigenvalues by moving A across the

inner product. Within a repeated eigenspace any orthonormal basis works. A Hermitian operator is positive semidefinite when $x^*Ax \geq 0$ for all x, and positive definite when the inequality is strict for every nonzero x. Expansion in this basis shows these are exactly the nonnegative and positive spectrum conditions, respectively.

Normal spectral theorem. Over $\mathbb{C}$, $AA^* = A^*A$ if and only if $A = Q\Lambda Q^*$ for a unitary Q and complex diagonal Λ.

Proof. For any normal N, $\|Nx\|^2 = x^*N^*Nx = x^*NN^*x = \|N^*x\|^2$. The shift $N = A - \lambda I$ is normal by expansion. Hence an eigenvector v of A is also an eigenvector of A^* for $\bar{\lambda}$. Its orthogonal complement is invariant under both operators by the adjoint identity. The matrix splits orthogonally into a scalar and a block which still commutes with its adjoint. Induction beginning with a complex root of χ_A gives the unitary eigenbasis. Conversely diagonal matrices commute with their adjoints and unitary similarity preserves this identity. □

For a normal operator, components can acquire independent phases as well as independent magnitudes. For instance $\mathrm{diag}(i, 2)$ is normal, although its first eigenvalue prevents self-adjointness. The real quarter-turn is unitary over $\mathbb{C}$ and orthogonal over $\mathbb{R}$; its complex eigenvalues have modulus one. In Session 2, B has two distinct, nonorthogonal eigenlines, ruling out normality. The Jordan block lacks even an unrestricted eigenbasis.

21.4 Session 4 — Turning a scalar function into an operator

In the Hermitian or complex normal case collect orthonormal eigenvectors for λ in Q_λ. Then $P_\lambda = Q_\lambda Q_\lambda^*$ is self-adjoint. The product $Q_\lambda^* Q_\mu$ is the identity for equal groups and zero for distinct groups. Consequently $P_\lambda P_\mu = \delta_{\lambda\mu} P_\lambda$ and $\sum_\lambda P_\lambda = QQ^* = I$. The identity $A = \sum_\lambda \lambda P_\lambda$ specifies an action on every input: project into modes, scale, and sum.

Functional calculus. For any diagonalizable A and any polynomial p,

$$p(A) = \sum_\lambda p(\lambda) P_\lambda.$$

Indeed multiplication of two such expansions multiplies corresponding coefficients because the cross-products vanish. Induction gives the formula for powers, including $A^0 = I$, and linearity gives it for polynomials. Over an orthogonal decomposition, Pythagoras further gives $\|p(A)\| = \max_\lambda |p(\lambda)|$, attained on a maximizing eigenvector. For a general diagonalizable operator the same norm argument fails at its first use of orthogonality.

21.4.1 An exact mode selector

For distinct spectral points let $L_j(t) = \prod_{i \neq j} (t - \lambda_i)/(\lambda_j - \lambda_i)$. Evaluating at each eigenvalue gives $L_j(\lambda_i) = \delta_{ij}$, hence $L_j(A) = P_{\lambda_j}$. For A_0 the projector onto the vertical axis is $(A_0 -$

$I)(A_0-3I)/48$. A function f specified only on this finite spectrum therefore gives the operator $\sum_j f(\lambda_j)P_{\lambda_j}$ by polynomial interpolation. Distinct eigenvalues that receive the same value under f merge into a larger eigenspace of $f(A)$.

Choice on the spectrum	Meaning on a mode of eigenvalue λ
t^k	k discrete time steps multiply it by λ^k.
$e^{\tau t}$	The power series converges in matrix norm and yields $e^{\tau\lambda}$. Thus $x' = Ax$ has solution $e^{\tau A}x(0)$.
$1/t$ with $0 \notin \sigma(A)$	The resulting operator inverts A, verified by multiplying the expansions.
$\sqrt{t}$ for Hermitian $A \geq 0$	The nonnegative spectral square root squares to A. Any positive semidefinite square root commutes with A and restricts on each E_λ to the scalar $\sqrt{\lambda}$ by its own spectral theorem.

If A is diagonalizable its minimal polynomial, the monic polynomial of smallest degree annihilating A, is $\prod_j(t-\lambda_j)$: each factor is necessary by evaluation on eigenvectors and the product suffices. Polynomial evaluation thus depends only on remainders modulo this product. The list of values at the roots gives the Chinese remainder coordinates, and L_j reconstructs one coordinate at a time. With a defective block $J = 2I + N$, $N^2 = 0$, direct expansion gives $p(J) = p(2)I + p'(2)N$. The value at the eigenvalue alone has lost information. Exact projector formulas also presume exact spectral points. Tiny denominators for a cluster motivate selecting the entire cluster using interval information in Session 9.

21.5 Session 5 — Measuring an error when the answer is a space

Two programs can return orthogonal eigenvectors in the same repeated eigenspace. To compare their spaces, use the orthogonal projector $P_U = Q_U Q_U^*$. Replacing Q_U by $Q_U S$ for a unitary S leaves the projector unchanged. Pythagoras applied to $x-u = (I-P_U)x + (P_U x - u)$ gives $\operatorname{dist}(x, U) = \|(I-P_U)x\|$. For unit vectors x, y the angle between their lines lies in $[0, \pi/2]$ and satisfies $\cos\theta = |y^*x|$. Taking absolute value removes arbitrary signs or phases.

21.5.1 Deriving the principal directions

Choose unit vectors $u \in U$, $w \in W$ maximizing $|u^*w|$, then repeat in the respective orthogonal complements of the previously chosen vectors. The successive maxima are cosines of the **principal angles** $0 \leq \theta_1 \leq \cdots \leq \theta_s \leq \pi/2$, $s = \min(\dim U, \dim W)$. In basis coordinates this maximizes $|a^*Cb|$ where $C = Q_U^* Q_W$. For fixed b, Cauchy–Schwarz gives maximum $\|Cb\|$.

Its square is b^*C^*Cb, maximized on a top eigenvector of the positive semidefinite C^*C by orthogonal expansion. For a positive maximum choose $a = Cb/\|Cb\|$. Orthogonal eigenvectors b_i give orthogonal Cb_i, so removing each chosen pair leaves precisely the next largest stretch. Complete null directions arbitrarily. Thus $\cos\theta_i = \sigma_i(C)$, the decreasing singular values. This derivation uses only the Hermitian theorem and anticipates Session 8.

Projector distance theorem. For equal positive dimensions, $\|P_U - P_W\| = \sin\theta_s$. To see the norm, take the paired principal vectors u_i, w_i with $u_i^* w_j = c_i\delta_{ij}$, $c_i = \cos\theta_i$. Where $s_i = \sin\theta_i > 0$, write $w_i = c_i u_i + s_i z_i$. Subtraction and normalization give orthonormal z_i perpendicular to all u_j. On each resulting plane,

$$P_U - P_W = \begin{pmatrix} s_i^2 & -c_i s_i \\ -c_i s_i & -s_i^2 \end{pmatrix}.$$

Its square is $s_i^2 I$. Common directions and the external complement contribute zero. The norm of the orthogonal block sum is its largest block norm. For unequal dimensions a larger space contains a unit vector perpendicular to the smaller, even if all paired angles vanish. For instance comparing a plane with a contained line gives projector distance one.

Example. Take $U = \operatorname{span}(e_1, e_2)$ and $W = \operatorname{span}(e_1, \cos t\, e_2 + \sin t\, e_3)$ for $0 \le t \le \pi/2$. The common axis gives angle zero. The other angle is t. The worst error in projecting a unit vector from W onto U is $\sin t$, attained on the tilted direction.

21.5.2 A diagnostic when the target is unknown

For a trial frame Q, $H = Q^*AQ$ gives $AQ = QH + R$ with $R = (I - QQ^*)AQ$. Thus $R = 0$ exactly when the trial space is invariant. Under $Q \mapsto QS$ the residual becomes RS, preserving its norm. Zero residual can certify the wrong invariant space if the target was specified by a spectral window. Even for the right window, scale matters: take $A = \operatorname{diag}(1, 1+\varepsilon)$ and the diagonal line at 45°. Its residual norm is $\varepsilon/2$, which tends to zero at fixed angle. Session 7 will identify the missing denominator.

21.6 Session 6 — Finding eigenvalues by optimizing a scalar measurement

Let $A = A^*$ and $x = \sum_i c_i q_i \neq 0$. Its **Rayleigh quotient** is

$$\rho_A(x) = \frac{x^*Ax}{x^*x} = \frac{\sum_i \lambda_i |c_i|^2}{\sum_i |c_i|^2}.$$

This weighted mean is unchanged by rescaling x. It lies between the smallest and largest eigenvalues. Equality at either end requires all nonzero coefficients to belong to that endpoint

eigenspace. For A_0 and $x = q_2 \cos t + q_3 \sin t$ it is $3 + 6\sin^2 t$. Intermediate values occur even though the spectrum is discrete.

Stationarity. Vary a unit x along $x + th$ for real t. Differentiating the quotient at zero gives $2\,\mathrm{Re}(h^*(Ax - \rho_A(x)x))$. Radial derivatives vanish. The vector $r = Ax - \rho_A(x)x$ is perpendicular to x and therefore tangent to the sphere. Choosing $h = r$ proves that stationarity is equivalent to $r = 0$. Along planes joining an interior eigenvector to a lower or higher mode, the quotient curves down or up. These stationary values can therefore be saddles.

Courant–Fischer. Order by increasing algebraic value $\lambda_1 \le \cdots \le \lambda_n$. Then

$$\lambda_k = \min_{\dim S = k} \max_{0 \neq x \in S} \rho_A(x) = \max_{\dim L = n-k+1} \min_{0 \neq x \in L} \rho_A(x).$$

Proof. Any k-space meets $\mathrm{span}(q_k, \dots, q_n)$ nontrivially since their dimensions sum to $n + 1$. That direction has quotient at least λ_k. Thus all candidate maxima are at least λ_k. The first k eigenvectors span a space with maximum exactly λ_k. Dually every $(n - k + 1)$-space meets the span of the first k eigenvectors, where the quotient is at most λ_k. The space of the last $n - k + 1$ eigenvectors attains equality. □

For A_0, any plane must admit a vector of quotient at least 3 and one of quotient at most 3. The horizontal plane minimizes the worst energy, giving maximum 3; the plane spanned by q_2, q_3 maximizes the least energy, giving minimum 3. This explains how the middle eigenvalue can be characterized using extrema even though it is a saddle on the whole sphere.

Weighted geometry. If $K = K^*$ and $M = M^* > 0$, then $M^{-1}K$ is self-adjoint for $\langle x, y \rangle_M = y^* M x$. Set $z = M^{1/2}x$. The generalized quotient $x^*Kx/(x^*Mx)$ becomes the ordinary quotient of $M^{-1/2}KM^{-1/2}$ at z. This invertible change of variables preserves subspace dimensions and transfers both min–max formulas. Generalized eigenvectors $Kx = \lambda Mx$ can be chosen M-orthonormal.

21.7 Session 7 — What spectral information can a trial space certify?

In a large problem we can afford a few vectors as columns of Q. For orthonormal columns compress $A = A^*$ to $H = Q^*AQ$ and solve $Hy = \theta y$, $\|y\| = 1$. The **Ritz pair** is (θ, u) with $u = Qy$. Its residual $r = Au - \theta u$ obeys

$$Q^* r = Hy - \theta y = 0.$$

Conversely this orthogonality gives the reduced equations. It is the Galerkin condition from Module C: every test vector in the trial space measures zero error in the eigen-equation. In particular $u^* r = 0$ gives $\theta = \rho_A(u)$. The compression represents the restriction exactly if the space is invariant. Otherwise projection discards part of Au.

Ritz interlacing. For increasing algebraic order on both spectra, an m-dimensional compression has $\lambda_i \le \theta_i \le \lambda_{i+n-m}$ for $1 \le i \le m$. Indeed $\rho_H(y) = \rho_A(Qy)$. Restricting the minimizing i-spaces in Courant–Fischer to subspaces of range Q can only increase the minimum. Restricting the maximizing $(m-i+1)$-spaces can only decrease the maximum, and the unrestricted maximum for that dimension equals λ_{n-m+i}. Both bounds include multiplicities.

Residual inclusion and identification. For any unit $u = \sum c_i q_i$ and scalar θ,

$$\|Au - \theta u\|^2 = \sum_i |\lambda_i - \theta|^2 |c_i|^2 \ge \operatorname{dist}(\theta, \sigma(A))^2.$$

This gives proximity to *some* spectral value. To identify a target cluster J, assume $\delta = \min_{i \notin J} |\lambda_i - \theta| > 0$. Keeping only the outside terms gives

$$\|(I - P_J)u\| \le \|r\|/\delta.$$

For a simple identified eigenvalue the left side is the sine of the angle to its eigenline. Multiplicity makes the whole eigenspace the intrinsic target. For a block of orthonormal Ritz vectors u_j and common outside separation δ, summing the squared inequalities bounds $\|(I - P_J)U\|_F \le \|AU - U\Theta\|_F/\delta$. With equal dimensions the operator norm on the left equals $\sin\theta_{\max}$ by the principal-plane construction and is bounded by this Frobenius estimate.

What converges quadratically? For unit $u = q\cos t + w \sin t$, $w \perp q$, $\|w\| = 1$, $Aq = \lambda q$, the cross terms vanish and

$$\rho_A(u) - \lambda = (\rho_A(w) - \lambda)\sin^2 t.$$

There can be cancellation for interior eigenvalues. For two distinct modes of eigenvalues λ, μ the residual at the quotient has norm $|\mu - \lambda|\,|\sin t \cos t|$, obtained by resolving Au along u and the perpendicular direction in their plane. Near $t = 0$ the quotient error is second order while the residual is first order. The example in Session 5 has small absolute errors in both measurements simply because its gap is small.

A compression you can audit. Use $Q = [q_1, (q_2 + q_3)/\sqrt{2}]$ for A_0. The reduced matrix is $\operatorname{diag}(1, 6)$. The second Ritz vector has residual $3(q_3 - q_2)/\sqrt{2}$, perpendicular to both columns. Its norm 3 equals the distance from 6 to the nearest spectrum, and 6 sits midway between two modes. A solver could solve the reduced problem exactly and still deliver this residual.

For a full-rank trial basis Z in a generalized problem, orthogonality of $KZy - \theta MZy$ to every test column yields $Z^*KZy = \theta Z^*MZy$. The reduced mass matrix is positive definite because $Zy \ne 0$ for $y \ne 0$. A change of trial basis changes both small matrices by congruence and leaves the Ritz values unchanged.

21.8 Session 8 — Spectral geometry for maps between different spaces

For $A : \mathbb{C}^n \to \mathbb{C}^m$, the squared output length is x^*A^*Ax. Hence directions of extreme amplification can be found in the *domain*. The Hermitian spectral theorem gives orthonormal v_i with $A^*Av_i = \sigma_i^2 v_i$, positive values first in decreasing order. The kernel of A^*A equals $\ker A$ by the length identity. For each $\sigma_i > 0$ set $u_i = Av_i/\sigma_i$. Then

$$u_i^* u_j = \frac{v_i^* A^* A v_j}{\sigma_i \sigma_j} = \delta_{ij}, \qquad Av_i = \sigma_i u_i, \qquad A^* u_i = \sigma_i v_i.$$

Thus $AA^*u_i = \sigma_i^2 u_i$. These u_i span the image since all remaining v_i are killed. Complete them by an orthonormal basis of $\ker A^* = (\operatorname{im} A)^\perp$. With $r = \operatorname{rank} A$ we have proved the **singular value decomposition**

$$A = U\Sigma V^* = \sum_{i=1}^{r} \sigma_i u_i v_i^*.$$

Here U, V are square unitary matrices and Σ is m by n, diagonal in its rectangular shape. The first r right vectors span $\operatorname{im} A^* = (\ker A)^\perp$, the remaining right vectors span $\ker A$. These two domain spaces and the two range spaces give the four fundamental subspaces.

In input coordinates v_i, the unit ball maps to the solid ellipsoid with semiaxes $\sigma_i u_i$ in $\operatorname{im} A$. If A is injective the unit sphere maps to the boundary relative to this image. A nontrivial kernel allows interior points too, since kernel components consume input length without affecting the output. The longest semiaxis gives $\|A\| = \sigma_1$. For invertible A, $\kappa_2(A) = \sigma_1/\sigma_n$. For a nonzero rank-deficient map, inversion restricted to $(\ker A)^\perp$ has worst gain $1/\sigma_r$.

21.8.1 Which inverse solves inconsistent data?

Expand b in left singular coordinates. Writing $x = \sum_i \xi_i v_i$ gives

$$\|Ax - b\|^2 = \sum_{i=1}^{r} |\sigma_i \xi_i - u_i^* b|^2 + \|(I - U_r U_r^*)b\|^2.$$

The first r coordinates of any minimizer are forced. Setting all remaining coordinates to zero uniquely minimizes $\|x\|$ among them. Thus $x = A^+ b$ with $A^+ = \sum_{i=1}^{r} \sigma_i^{-1} v_i u_i^*$. The products $AA^+ = U_r U_r^*$ and $A^+A = V_r V_r^*$ are orthogonal projectors. These identities imply $AA^+A = A$, $A^+AA^+ = A^+$ and self-adjointness of both products, the Moore–Penrose conditions. Small singular values amplify measurement errors in their corresponding coordinates.

21.8.2 What can any rank budget achieve?

Eckart–Young–Mirsky, spectral and Frobenius cases. For $0 \le k < r$ truncate the expansion after k terms to obtain A_k. Its spectral error is σ_{k+1} and its squared Frobenius error is $\sum_{i>k} \sigma_i^2$. Both are minimal over all matrices of rank at most k. For $k \ge r$ the error is zero.

Proof. The residual expansion has orthogonal singular directions, giving the stated errors. For any competitor B of rank at most k, its kernel meets $\operatorname{span}(v_1, \dots, v_{k+1})$ in a unit vector x. Then $\|(A-B)x\| = \|Ax\| \ge \sigma_{k+1}$. For the second norm let P project onto $\operatorname{im} B$. Orthogonally splitting every column gives $\|A-B\|_F^2 \ge \|(I-P)A\|_F^2 = \sum_i \sigma_i^2(1-\|Pu_i\|^2)$. Each weight $\|Pu_i\|^2$ lies in $[0,1]$ and their sum over all left basis vectors is $\operatorname{tr} P \le k$. Their weighted sum against decreasing σ_i^2 is at most $\sum_{i\le k} \sigma_i^2$: move any available weight to an earlier, larger coefficient until the first k slots are filled or the weight is exhausted. Subtract this upper bound from $\|A\|_F^2$. $\square$

A rectangular calculation. For $A = \left(\begin{smallmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{smallmatrix}\right)$, right axes $(1,1,0)^T/\sqrt{2}, e_3$ map to $\sqrt{2}e_1, e_2$ in the output plane. The antisymmetric input is invisible. The minimum-norm input for (b_1, b_2) is $(b_1/2, b_1/2, b_2)$. Dropping the second mode costs exactly one in either norm. With many small trailing singular values the two errors can differ substantially: worst directional loss uses one tail value, while total squared loss counts the whole tail. This distinction drives choices of numerical rank and compression tolerance.

21.9 Session 9 — Designing a polynomial that improves a trial space

Suppose the only expensive operation available is $X \mapsto AX$. A polynomial of degree d can use d such products. How should we allocate them? In an eigenbasis $p(A) = Xp(\Lambda)X^{-1}$, so the choice of polynomial sets the change in every modal amplitude. In a normal eigenbasis the coordinate norm also measures physical length.

21.9.1 First follow a line

For this subsection order eigenvalues by decreasing **magnitude**. Let A be normal, $|\lambda_1| > |\lambda_2|$ and $x_0 = \sum_i c_i q_i$ with $c_1 \ne 0$. Repeated normalization $x_{j+1} = Ax_j/\|Ax_j\|$ represents the line of $A^{j+1}x_0$ while controlling scale. Dividing all coordinates of $A^j x_0$ by $c_1\lambda_1^j$ leaves its first coordinate one. Therefore

$$\tan \angle(x_j, q_1) = \left(\sum_{i>1} |c_i/c_1|^2 |\lambda_i/\lambda_1|^{2j} \right)^{1/2} \le |\lambda_2/\lambda_1|^j \tan \angle(x_0, q_1).$$

Zero coefficients can give faster convergence. For Hermitian A the Rayleigh error is bounded by the spectral width times the squared sine. A negative or complex dominant eigenvalue can change the phase at every step while the line converges. With zero initial overlap the missing mode stays missing. With tied magnitudes a unique direction need never emerge: $\operatorname{diag}(1,-1)$ alternates the line through $(1,1)^T$ with its perpendicular. For a diagonalizable nonnormal matrix the coordinate ratios still decay but angles require control of X and X^{-1}.

Changing which mode wins. Shift-and-invert uses $(A-\mu I)^{-1}$, $\mu \notin \sigma(A)$, whose mode factors are $(\lambda_i-\mu)^{-1}$. A solve per step targets the nearest eigenvalue if it is unique in distance and represented initially. Rayleigh quotient iteration updates $\mu=\rho_A(x)$ before each solve. Near an isolated Hermitian eigenline its shift error is $O(\sin^2 t)$ by Session 7. Provided the shifted systems are nonsingular, the unwanted-to-wanted amplification ratio is $O(\sin^2 t)$, which multiplies the old tangent to give local cubic angular convergence. Check the residual before solving at an exact eigenpair.

21.9.2 Now follow a space

Let A be normal over $\mathbb{C}$ (real symmetric also suffices), and split an orthonormal eigenbasis into Q_1, Q_2 for wanted and unwanted groups with diagonal blocks Λ_1, Λ_2. Suppose a full-rank trial matrix X has exactly $k = \dim \operatorname{range} Q_1$ columns. Write $X = Q_1 C + Q_2 D$. Full overlap means $C = Q_1^* X$ is invertible. Then the trial space is the graph

$$\{Q_1 z + Q_2 G z : z \in \mathbb{C}^k\}, \qquad G = DC^{-1}.$$

The vector over z makes an angle with the target whose tangent is $\|Gz\|/\|z\|$. Taking the supremum gives $\tan\theta_{\max} = \|G\|$, since principal directions extremize projection loss. Full overlap is precisely the exclusion of a right angle. A change of trial basis multiplies C, D on the right by the same invertible matrix, leaving G intact.

Filtered contraction theorem. Write W, U for the wanted and unwanted spectral sets. If $\alpha = \min_{\lambda\in W}|p(\lambda)| > 0$ and $\beta = \max_{\lambda\in U}|p(\lambda)|$, then

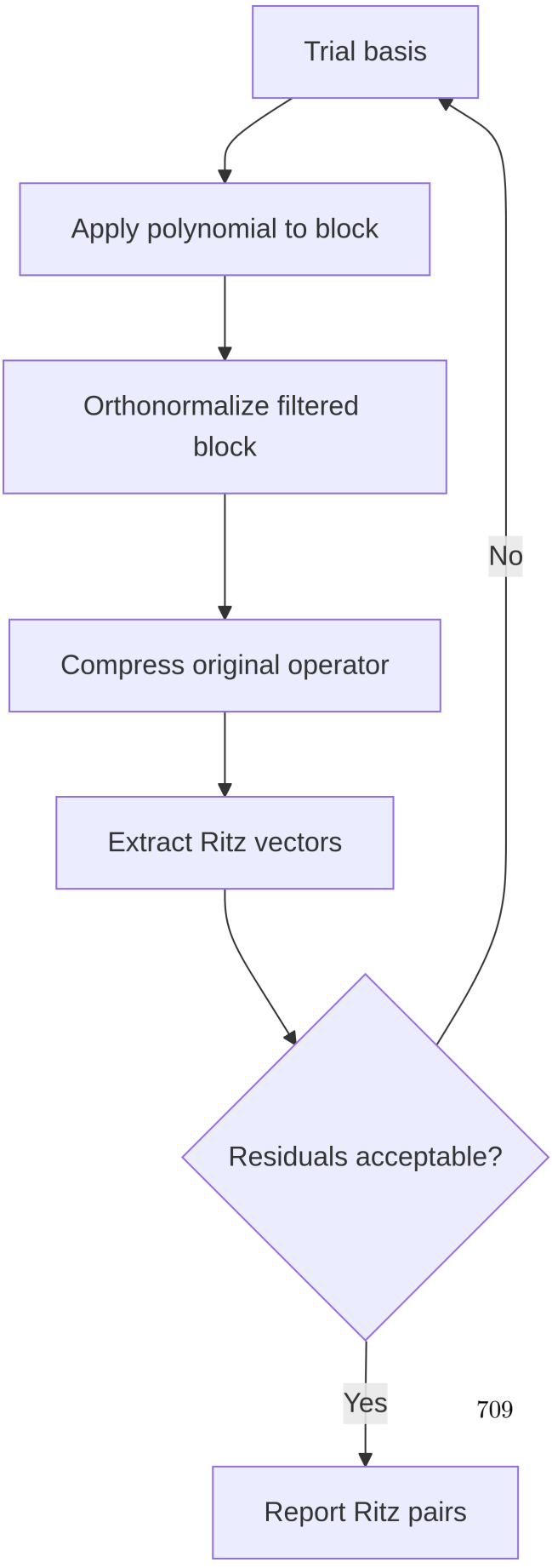
$$G_{\text{new}} = p(\Lambda_2) G_{\text{old}} p(\Lambda_1)^{-1}, \qquad \tan\theta_{\max}^{\text{new}} \le \gamma_p \tan\theta_{\max}^{\text{old}}, \qquad \gamma_p = \beta/\alpha.$$

Proof. After applying $p(A)$ the coefficient blocks become $p(\Lambda_1)C$ and $p(\Lambda_2)D$. The first is invertible by hypothesis. Multiply the new lower block by the inverse upper block and use the induced norm inequality. The norms of the two diagonal factors are exactly β and $1/\alpha$. Orthonormalizing the columns preserves their span and therefore the graph. □

For $p(t) = t$ and the first k modes in decreasing magnitude, the factor is $|\lambda_{k+1}|/|\lambda_k|$. A strict gap across the boundary suffices. Multiplicity *inside* the target is harmless. After j repetitions the bound contains γ_p^j. Neither this estimate nor Ritz extraction requires the individual columns to converge. QR orthonormalization chooses a usable frame for the next product.

Rayleigh–Ritz rotates this frame to diagonalize the compression and supplies residuals for individual approximate modes.

The diagram distinguishes changing the space from changing its coordinates. The filter is the step that changes the graph. The reduced solve produces quantities that can be tested against the original operator.


Trial basis
Apply polynomial to block
Orthonormalize filtered block
Compress original operator
Extract Ritz vectors
Residuals acceptable?
No
Yes
Report Ritz pairs

Full-basis connection. In unshifted QR iteration factor $A_j = U_j R_j$ and set $A_{j+1} = R_j U_j = U_j^* A_j U_j$. With $Z_j = U_0 \cdots U_{j-1}$ one has $A_j = Z_j^* A Z_j$ and $AZ_j = Z_{j+1} R_j$. Thus the accumulated bases follow orthogonal iteration. The triangular factor gives simultaneous iteration of their leading subspaces when the corresponding blocks have full rank.

21.9.3 From a spectral window to a Chebyshev filter

Suppose the unwanted real eigenvalues lie in $[a, b]$, $a < b$. We want a polynomial small on that entire interval and nonzero on the target. Affinely rescale by $s(t) = (2t - a - b)/(b - a)$. Define $T_0(x) = 1$, $T_1(x) = x$ and $T_{j+1}(x) = 2xT_j(x) - T_{j-1}(x)$. The cosine addition formula proves inductively that $T_j(\cos\phi) = \cos(j\phi)$. Thus $|T_j| \le 1$ throughout $[-1, 1]$. The hyperbolic addition formula similarly gives $T_j(\cosh u) = \cosh(ju)$. Parity gives $T_j(-x) = (-1)^j T_j(x)$. Beyond either endpoint the magnitude grows exponentially in degree at each fixed exterior point.

Why this shape? For real $x_* > 1$ and $d \ge 1$, among real polynomials h of degree at most d satisfying $h(x_*) = 1$, the least possible uniform norm on $[-1, 1]$ is $1/T_d(x_*)$. It is attained by $g(x) = T_d(x)/T_d(x_*)$. If a competitor were strictly smaller in norm, $g - h$ would alternate signs at the $d + 1$ extrema $\cos(j\pi/d)$, $j = 0, \ldots, d$. It would have d distinct roots between them and another at x_*, impossible for a nonzero polynomial of degree at most d. Reflection handles $x_* < -1$. This minimax statement concerns a whole interval and one prescribed external value. A polynomial tailored to a few exactly known unwanted eigenvalues can annihilate all of them.

Take $p_d(t) = T_d(s(t))$, optionally divided by its value at a desired exterior reference point. Normalization fixes the gain there and leaves γ_p unchanged. For A_0, choose unwanted $[1, 3]$ and wanted 9. Then $s(t) = t - 2$, $s(9) = 7$, and $T_2(7) = 97$. The normalized quadratic $p(t) = (2(t - 2)^2 - 1)/97$ has values $(1/97, 1/97, 1)$. For every line overlapping q_3, its tangent to q_3 therefore shrinks by *exactly* $1/97$. Two ordinary power steps could shrink the slowest unwanted-to-wanted ratio only to $(3/9)^2 = 1/9$. Both use two matrix products. If an unaccounted unwanted mode lies at 10, it maps to 8 and gains $127/97 > 1$. An interval estimate can determine which subspace an algorithm finds.

Evaluating the filter. For a block X, initialize $Y_0 = X$, $Y_1 = s(A)X = (2AX - (a + b)X)/(b - a)$. Induction using the scalar recurrence proves

$$Y_{j+1} = 2s(A)Y_j - Y_{j-1} = T_{j+1}(s(A))X.$$

This uses matrix-block products, scalar combinations and two previous blocks. Forming powers of a large sparse matrix could destroy sparsity. Dividing the final result by the reference gain normalizes the mathematical polynomial; intermediate unnormalized blocks can still grow large. At moderate degrees this recurrence illustrates the exact action cleanly. For high degree one can propagate normalized blocks, with coefficients involving ratios of successive reference values.

Which gap? An additive distance δ from competing eigenvalues converts a residual into an angle bound. A multiplicative ratio γ_p predicts improvement per filtering step. One is an accuracy certificate and the other a convergence estimate. Computing either from approximate spectral information introduces a separate estimation problem.

21.10 Session 10 — What survives beyond finite matrices?

21.10.1 Spectral values without eigenvectors

For a bounded operator on a complex Banach space (a complete normed space), $\sigma(T)$ consists of λ for which $T-\lambda I$ has no everywhere-defined bounded inverse. In finite dimensions injectivity, surjectivity and bounded invertibility coincide for an endomorphism. Infinite dimensions separate them.

On $L^2(0,1)$ let $(Tf)(x) = xf(x)$. Its norm is at most one and moving the real multiplier across the inner product proves self-adjointness. An eigenfunction would have to vanish except where $x = \lambda$, a set of measure zero, hence represents zero. For $\lambda \notin [0,1]$, multiplication by $1/(x-\lambda)$ gives a bounded inverse. For $\lambda \in [0,1]$ normalize the indicator of $(0,1) \cap (\lambda-\varepsilon, \lambda+\varepsilon)$ to obtain a unit f_ε with $\|(T-\lambda I)f_\varepsilon\| \le \varepsilon$. A bounded inverse would force $1 \le \|(T-\lambda I)^{-1}\|\varepsilon$ for all $\varepsilon > 0$. Consequently the entire interval is spectrum, with no eigenvalues. Projection onto functions supported in a subinterval plays the role of selecting a spectral band.

A compact operator sends every bounded sequence to one whose images have a convergent subsequence. On ℓ^2, $Te_j = e_j/j$ is compact: truncating its diagonal after N gives a finite-rank map with operator error $1/(N+1)$, and norm limits of finite-rank operators are compact (finite-dimensional images of the unit ball have finite nets at every scale). Its eigenvalues are $1/j$. Zero is spectral because $\|Te_j\| \to 0$, yet its kernel is trivial. Outside the closure of $\{1/j\}$ the inverse shifted diagonal is uniformly bounded. The compact self-adjoint spectral theorem gives an orthonormal eigenbasis on the closure of the range, completed by a basis of the kernel; nonzero eigenvalues have finite multiplicity and can accumulate only at zero. Module F1 develops the Hilbert-space setting and the compact-inverse route to PDE spectra.

21.10.2 From a vibrating string to a generalized eigenproblem

Seek a nonzero u satisfying $-u'' = \lambda u$, $u(0) = u(1) = 0$. Multiplication by $\bar{u}$ and integration by parts give $\int |u'|^2 = \lambda \int |u|^2$, so $\lambda > 0$: a function of zero derivative obeying the endpoints vanishes. The ODE solution is $a\sin(\sqrt{\lambda}x) + b\cos(\sqrt{\lambda}x)$. The first endpoint forces $b = 0$, the second forces $\sqrt{\lambda} = j\pi$, $j \ge 1$. Thus

$$\lambda_j = j^2\pi^2, \qquad u_j(x) = \sqrt{2}\sin(j\pi x).$$

Integrating the product of two sines verifies L^2-orthonormality. The string modes resemble the eigenvectors of A_0, but each is now a function.

Multiply the ODE by $\bar{v}$ and integrate by parts for tests vanishing at both endpoints. The weak problem seeks $u \neq 0$ in $H_0^1(0,1)$ (square-integrable functions with square-integrable weak derivative and zero endpoint traces) such that

$$\int_0^1 u'\overline{v'}\,dx = \lambda \int_0^1 u\bar{v}\,dx \quad \text{for every } v \in H_0^1(0,1).$$

Taking trial and test functions in the same finite-dimensional space with basis ϕ_j gives $Kc = \lambda Mc$ with $K_{ij} = \int \phi_j'\overline{\phi_i'}$, $M_{ij} = \int \phi_j\bar{\phi}_i$. M records the physical inner product between basis functions. On a uniform grid of spacing h the piecewise-linear hat basis yields diagonal $2/h$ and off-diagonal $-1/h$ in K, and $2h/3, h/6$ in M. On each interval the slopes have magnitude $1/h$ and opposite signs, while the two shape functions are $1 - x/h, x/h$ in local coordinates. Integrating their products yields the entries after assembly. The continuous min–max principle for this Dirichlet problem, restricted to the trial space, bounds the exact eigenvalues from above by the conforming Ritz values. Session 7's dimension argument explains the direction of the bound. The laboratory separates trial-space error from algebraic solver residual.

21.10.3 When polynomials act like scalars

Let $p(t) \cdot v = p(T)v$. Polynomial identities give $(pq) \cdot v = p \cdot (q \cdot v)$, distributivity in both arguments, and $1 \cdot v = v$. Thus V is an $\mathbb{F}[t]$-module, a vector space equipped with these additional scalar actions. A subspace is a submodule exactly when it is T-invariant: stability under t is necessary, and stability under powers, sums and scalar multiples is sufficient. A map S between operator spaces respects this action precisely when $ST = T'S$, since checking on the generator t suffices. This recasts Module A's quotient diagram as a diagram of module maps.

Starting from v, the first k powers span $\mathcal{K}_k(T, v) = \operatorname{span}(v, Tv, \dots, T^{k-1}v)$. When a power first depends on its predecessors the space stops growing and equals the cyclic submodule generated by v. The first monic relation is the minimal polynomial of v; the operator minimal polynomial annihilates all vectors. For J in Session 1, e_2 generates both coordinates and $(J - 2I)^2 e_2 = 0$ while $(J - 2I)e_2 = e_1$. The module is $\mathbb{F}[t]/((t-2)^2)$, explaining why the eigenline and its quotient fit into an extension that does not split.

More generally factor the minimal polynomial as $m = \prod_i f_i^{a_i}$ over distinct monic irreducibles. Put $g_i = m/f_i^{a_i}$. Coprimality supplies b_i with $b_i g_i \equiv 1 \bmod f_i^{a_i}$ by Bézout's identity. The polynomials $b_i g_i$ are zero modulo every other factor. Their sum equals one modulo m, their pairwise products vanish modulo m, and each is idempotent modulo m. Evaluation at T therefore gives complementary projectors. Their images equal $\ker f_i(T)^{a_i}$: multiplication by $f_i^{a_i}$ kills each image, and on that kernel the corresponding projector acts as the identity. For a split polynomial these are the generalized eigenspaces. Powers of a single linear factor permit

Jordan chains within a primary component. Module F2 studies this decomposition via quotient rings and the Chinese remainder theorem.

21.11 Exercises

Attempt these before consulting the solutions. For each claim, identify whether it uses a field, an inner product, or a separation assumption.

1. **A line and a plane.** For $B = \left(\begin{smallmatrix} 2 & 0 & -3 \\ 1 & -1 & -1 \\ 0 & 0 & -1 \end{smallmatrix}\right)$ find both eigenspaces, their multiplicities, and the spectral projector for 2. Decide if it is orthogonal.
2. **What a quotient forgets.** Classify the invariant lines of $\left(\begin{smallmatrix} a & b \\ 0 & a \end{smallmatrix}\right)$ as b varies. Compute the quotient action by the first axis, and decide when the extension splits. Compare with a real quarter-turn after complexification.
3. **Reconstruct the spectral theorem.** Starting with one complex eigenvector of a normal matrix, prove its orthogonal complement reduces the matrix (is invariant under both the matrix and its adjoint). Which extra hypothesis forces the eigenvalues to be real?
4. **Separating a cluster.** For A_0 find the lowest-degree polynomial projection onto $\operatorname{span}(q_1, q_2)$. Compute $e^{-\tau A_0} e_1$. Which modes dominate at late time?
5. **Distance and invariance.** For $U = \operatorname{span}(q_1, q_2)$ and $W = \operatorname{span}(q_1, q_2 \cos t + q_3 \sin t)$, $0 < t < \pi/2$, compute both principal angles, the projector distance and the norm of the invariance residual of W for A_0.
6. **Min–max without coordinates.** If $B = A + vv^*$ with $A = A^*$, prove $\lambda_i(B) \geq \lambda_i(A)$ and $\lambda_i(B) \leq \lambda_{i+1}(A)$ for $i < n$ in increasing algebraic order. Identify the dimension argument.
7. **A misleading Ritz value.** For $\operatorname{diag}(-1, 0, 1)$ find a unit vector perpendicular to the zero eigenvector whose Rayleigh quotient is exactly zero. Calculate its residual and interpret the one-dimensional Ritz problem.
8. **A residual certificate.** A Hermitian problem has one target eigenvalue in $[2.9, 3.1]$ and all other eigenvalues outside $(1, 5)$. A unit Ritz vector has $\theta = 3.02$ and $\|r\| = 0.01$. Bound its eigenvalue error and its angle to the target space. What changes if there are two eigenvalues in the target window?
9. **A rectangular sensor.** Let A have columns $(1, 1, 0)^T$ and $(0, 0, 2)^T$. Give a thin SVD, the minimum-norm least-squares solution for $b = (1, -1, 4)^T$, and both optimal rank-one errors.
10. **Power and graph coordinates.** For eigenvalues $(9, 3, 1)$ in an orthonormal eigenbasis start with the line through $(1, 1, 1)^T$ and then with the plane whose graph over the first two coordinates is $(1, 1)$. Find the exact tangent after j ordinary power steps in both cases.
11. **Reconstruct the filter.** With unwanted $[1, 3]$, normalize a degree-three Chebyshev filter at 9. Bound its contraction and prove optimality for the continuous interval. Can

an unwanted eigenvalue at -6 reverse the gain comparison?

12. **A mesh with only two unknowns.** On $(0,1)$ take three equal elements with zero endpoints. Assemble K, M, solve for both eigenvalues, and verify mass orthogonality. Compare with the first two string eigenvalues.
13. **Primary components.** Let $T = \operatorname{diag}(J, 5)$ where J is the size-two Jordan block at 2. Give the minimal polynomial, a polynomial projector onto the 5-space, and an explanation of why ordinary two-point linear interpolation fails.
14. **Spectrum at a limit.** For $Te_j = e_j/j$ on ℓ^2, exhibit an element outside its range. Explain why testing every basis vector for membership in the range would miss the obstruction.

21.12 Worked solutions

1. The diagonal block structure gives $\chi_B = (t-2)(t+1)^2$. Solving $(B-2I)x = 0$ gives $x_3 = 0$ and $x_1 = 3x_2$, so $E_2 = \operatorname{span}((3,1,0)^T)$. For -1 the single constraint is $x_1 = x_3$, giving $E_{-1} = \operatorname{span}(e_2, (1,0,1)^T)$. Geometric multiplicities equal the algebraic ones. The coefficient along $(3,1,0)^T$ is $(x_1 - x_3)/3$ because removing that multiple leaves equal first and third coordinates. Hence $P_2 = (3,1,0)^T(1,0,-1)/3 = (B+I)/3$. Its kernel and image are not perpendicular, as the dot product of e_2 with $(3,1,0)^T$ is one.
2. Every invariant line is an eigenline. For $b = 0$ the matrix is scalar, so every line works and every complementary line splits off. For $b \neq 0$ the kernel after subtracting aI is the first axis, so no complementary invariant line exists. The quotient is scalar a for every b. The real quarter-turn has no invariant real line and two distinct complex eigenlines since its polynomial t^2+1 splits into simple factors over $\mathbb{C}$. Field extension cures this rotation obstruction, while the Jordan obstruction survives it.
3. Set $N = A - \lambda I$. Expansion shows $N^*N = NN^*$, so $Nv = 0$ implies $\|N^*v\| = \|Nv\| = 0$. Thus $A^*v = \bar{\lambda}v$. If $w \perp v$, moving A or A^* across the inner product proves both Aw and A^*w are perpendicular to v. The restriction is normal because both operators preserve the complement and still commute. Induction yields the orthonormal eigenbasis. Self-adjointness gives $\lambda = v^*Av/\|v\|^2 \in \mathbb{R}$; conversely real eigenvalues in the unitary diagonalization imply self-adjointness.
4. The polynomial $p(t) = 1-(t-1)(t-3)/48$ is one at $1, 3$ and zero at 9. A linear function taking the same value at the first two points is constant, so degree two is minimal. Since $e_1 = (q_1+q_2)/\sqrt{2}$, the solution is $(e^{-\tau}q_1 + e^{-3\tau}q_2)/\sqrt{2}$. As $\tau \to \infty$ both decay and the relative amplitude of the second decays as $e^{-2\tau}$.
5. The cross-Gram matrix is $\operatorname{diag}(1, \cos t)$, giving angles $0, t$ and distance $\sin t$. The first residual column vanishes. For the second, project onto the only excluded direction $-q_2 \sin t + q_3 \cos t$. Its coefficient is $(9-3)\sin t \cos t$. Hence the norm is $6 \sin t \cos t$. Near $\pi/2$ the trial space again becomes nearly invariant while being far from U.
6. The quotient of B exceeds or equals that of A pointwise because the difference is $|v^*x|^2/\|x\|^2$. Taking maxima and then minima proves the lower bound. Intersect the

span of the first $i+1$ eigenvectors of A with $v^{\perp}$. The intersection has dimension at least i. Choose an i-space there. Its B quotient equals its A quotient and is bounded above by $\lambda_{i+1}(A)$. Minimize over all i-spaces for B to conclude.

7. Take $u = (1,0,1)^T/\sqrt{2}$. The two energy contributions cancel. The single Ritz value is zero since $u^*Au = 0$. The residual is $(-1,0,1)^T/\sqrt{2}$ with norm one. The Ritz value equals a true eigenvalue even though its vector has angle $\pi/2$ to that eigenline. Residual orthogonality here says $u^*Au = 0$ and gives no smallness by itself.
8. Eigenvalue inclusion gives a spectral point in $[3.01, 3.03]$, necessarily the target. The nearest possible outside eigenvalue is at distance at least $\min(3.02-1, 5-3.02) = 1.98$. Hence $\sin t \le 0.01/1.98$ and $t \le \arcsin(0.01/1.98)$, about 0.29°. With two target eigenvalues the same angle bound concerns their joint span. Identifying a particular eigenline requires separation within that span as well.
9. The orthogonal columns have lengths $\sqrt{2}, 2$. In decreasing order take right vectors e_2, e_1 and left vectors $e_3, (1,1,0)^T/\sqrt{2}$. The corresponding singular values are $2, \sqrt{2}$. The data component $(1,-1,0)^T$ is perpendicular to the range. Removing it and undoing the gain two gives $x = (0,2)^T$, the unique minimizer because the map is injective. Keeping the gain-two term leaves one singular value $\sqrt{2}$, so both optimal errors equal $\sqrt{2}$.
10. Dividing the line coordinates by 9^j gives tangent $\sqrt{(3/9)^{2j} + (1/9)^{2j}}$. For the plane the lower block evolves with scalar 1 while the upper block has powers $9^j, 3^j$. The new graph row is $(9^{-j}, 3^{-j})$ and its norm is $\sqrt{9^{-2j} + 3^{-2j}}$. Here the two asymptotic factors happen to be $1/3$ for different reasons: the line compares 3 with 9, the plane compares 1 with 3.
11. $s(t) = t-2$ and $T_3(x) = 4x^3 - 3x$, giving $p(t) = T_3(t-2)/1351$. The maximum unwanted gain is at most $1/1351$ and the wanted gain equals one. A smaller interval maximum would force the difference from p to alternate signs at the four mapped extrema and vanish at 9, hence have four distinct zeros at degree at most three. At -6 the mapped argument is -8 and $|T_3(-8)| = 2024 > 1351$. That unwanted component could dominate the intended one.
12. Here $K = \left(\begin{smallmatrix} 6 & -3 \\ -3 & 6 \end{smallmatrix}\right)$ and $M = \left(\begin{smallmatrix} 4 & 1 \\ 1 & 4 \end{smallmatrix}\right)/18$. The symmetric vector $(1,1)^T$ gives eigenvalue $3/(5/18) = 10.8$. The antisymmetric vector $(1,-1)^T$ gives $9/(3/18) = 54$. Their mass product vanishes since they are orthogonal and M scales each vector. The exact values π^2 and $4\pi^2$ are smaller; the coarse trial space penalizes the more oscillatory mode more severely.
13. The minimal polynomial is $(t-2)^2(t-5)$ because of the nilpotent part in J and the independent scalar block. Take $p(t) = (t-2)^2/9$. This vanishes on the whole Jordan block and equals one at 5. The linear selector $(t-2)/3$ only vanishes on the ordinary 2-eigenspace: on the Jordan block it has residual action $N/3$. The derivative condition $p'(2) = 0$ removes that action.
14. The sequence $y_j = 1/j$ is square summable. Its forced preimage has entries 1 and is not in ℓ^2. Each basis vector has preimage je_j, whose norm grows without bound. Finite combinations are in the range and dense, explaining how a proper range can look surjective on every finite truncation.

21.13 Computational laboratory

Run the cells in chapter order. The symbolic cells establish identities for the particular examples. The floating-point cells test quantitative predictions and print the quantities whose sizes matter. All figures are generated here. The optional HTML control varies the filter degree while the static figure compares representative degrees in print.

```
import numpy as np
import sympy as sp
import matplotlib.pyplot as plt
from numpy.linalg import norm, eigh, qr, svd, solve

np.set_printoptions(precision=5, suppress=True)
A = np.array([[2., 1., 0.], [1., 2., 0.], [0., 0., 9.]])
lam, V = eigh(A)  # Increasing algebraic order
eye = np.eye(3)
```

21.13.1 A. Count modes before computing directions

A numerical eigensolver can return nearly parallel columns for a defective matrix. Exact nullspaces here distinguish a genuinely missing direction from a poorly scaled approximate basis. SymPy lists a basis for each eigenspace over the complex numbers.

```
examples = {
    "Hermitian": sp.Matrix(A.astype(int)),
    "oblique": sp.Matrix([[1, 2], [0, 3]]),
    "Jordan": sp.Matrix([[2, 1], [0, 2]]),
    "quarter turn": sp.Matrix([[0, -1], [1, 0]]),
}
for name, matrix in examples.items():
    data = matrix.eigenvects()
    for value, multiplicity, basis in data:
        print(name, value, "algebraic", multiplicity, "geometric",
              len(basis), "basis", [tuple(v) for v in basis])
    print("Complex eigenbasis:", sum(len(b) for _, _, b in data) == matrix.rows)
```

```
Hermitian 1 algebraic 1 geometric 1 basis [(-1, 1, 0)]
Hermitian 3 algebraic 1 geometric 1 basis [(1, 1, 0)]
Hermitian 9 algebraic 1 geometric 1 basis [(0, 0, 1)]
Complex eigenbasis: True
oblique 1 algebraic 1 geometric 1 basis [(1, 0)]
```

```
oblique 3 algebraic 1 geometric 1 basis [(1, 1)]
Complex eigenbasis: True
Jordan 2 algebraic 2 geometric 1 basis [(1, 0)]
Complex eigenbasis: False
quarter turn -I algebraic 1 geometric 1 basis [(-I, 1)]
quarter turn I algebraic 1 geometric 1 basis [(I, 1)]
Complex eigenbasis: True
```

21.13.2 B. Reassemble a Hermitian operator from orthogonal projectors

For A_0 each eigenvalue is simple. Dividing vv^* by v^*v makes the construction independent of the arbitrary eigenvector length. All the following checks are exact.

```
S = examples["Hermitian"]
projectors = {}
for val, _, basis in S.eigenvects():
    v = basis[0]
    projectors[val] = v * v.H / (v.H * v)[0]
for val, P in projectors.items():
    assert P*P == P and P.H == P
    for other, R in projectors.items():
        if other != val:
            assert P*R == sp.zeros(3)
assert sum(projectors.values(), sp.zeros(3)) == sp.eye(3)
assert sum((v*P for v, P in projectors.items()), sp.zeros(3)) == S
print("Projector onto the antisymmetric mode:")
print(projectors[1])
```

```
Projector onto the antisymmetric mode:
Matrix([[1/2, -1/2, 0], [-1/2, 1/2, 0], [0, 0, 0]])
```

21.13.3 C. Compare polynomial evaluation routes

Here $p(t) = t^2 - 4t + 2$ takes the same value on 1 and 3. The resulting matrix acts as a scalar throughout the horizontal plane. The interpolating polynomial for 9 singles out the complementary axis.

```
I = sp.eye(3)
direct = S**2 - 4*S + 2*I
modal = sum(((v**2 - 4*v + 2)*P for v, P in projectors.items()),
```

```
            sp.zeros(3))
assert direct == modal
assert (S-I)*(S-3*I)/48 == projectors[9]
print(direct)
```

```
Matrix([[-1, 0, 0], [0, -1, 0], [0, 0, 47]])
```

21.13.4 D. Rotate one direction in a plane

Use the two planes from Exercise 5. At the endpoint $t = \pi/2$ the wrong plane is exactly invariant. Figure 21.2 shows why interpreting zero residual as recovery of a specified target would be a mistake. Dividing the residual by the gap keeps all curves on a common scale.

```
def tilted_frame(t):
    return np.column_stack((V[:, 0], np.cos(t)*V[:, 1] + np.sin(t)*V[:, 2]))

target = V[:, :2]
angles = np.linspace(0, np.pi/2, 151)
distances, defects = [], []
for t in angles:
    Q = tilted_frame(t)
    P = Q @ Q.T
    cosines = svd(target.T @ Q, compute_uv=False)
    principal = np.arccos(np.clip(cosines, 0, 1))
    distance = norm(target @ target.T - P, 2)
    assert abs(distance - np.sin(principal[-1])) < 5e-8
    distances.append(distance)
    defects.append(norm((eye-P) @ A @ Q, 2))
fig, ax = plt.subplots(figsize=(6.4, 3.4))
ax.plot(np.degrees(angles), distances, label="Projector distance")
ax.plot(np.degrees(angles), np.array(defects)/6, label="Residual / gap")
ax.set(xlabel="Tilt (degrees)", ylabel="Scaled magnitude")
ax.legend()
fig.tight_layout()
plt.show()
```

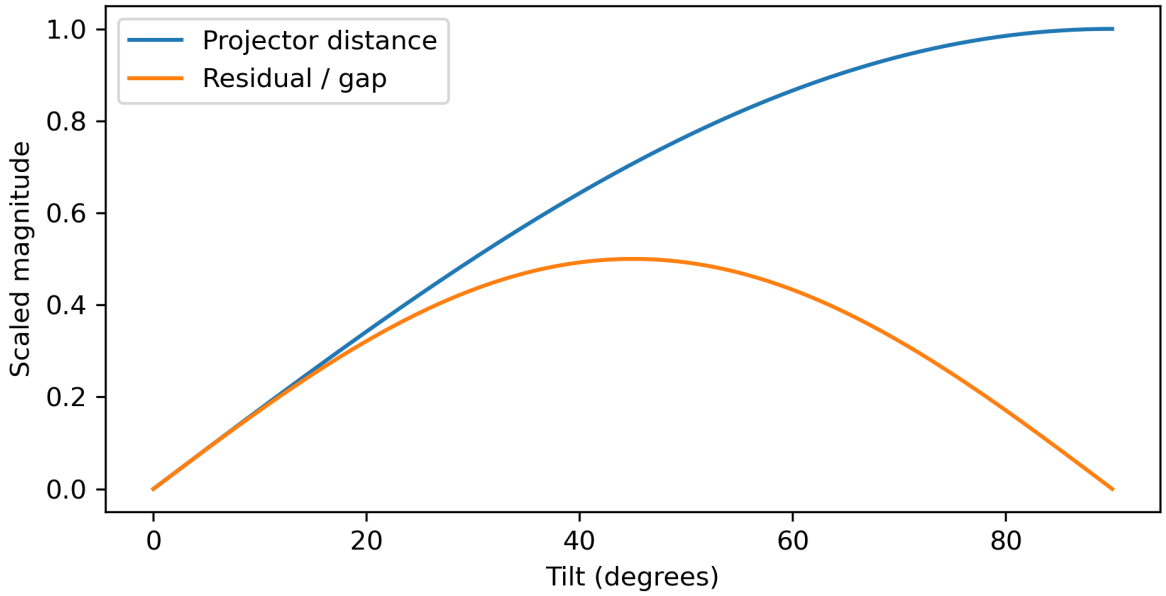


Figure 21.2: Tilting one principal direction. The projector distance reaches one at the second invariant plane.

21.13.5 E. Follow the energy around a circle

Restrict A_0 to the plane of modes $3, 9$. The minimum and maximum of this restriction are attained at the axes. Figure 21.3 also compares orders near the lower mode. Ratios to the indicated powers of t tend to the gap 6.

```
def energy_test(t):
    x = V[:, 1]*np.cos(t) + V[:, 2]*np.sin(t)
    rho = x @ A @ x / (x @ x)
    return rho, norm(A @ x - rho*x)

circle = np.linspace(0, 2*np.pi, 361)
energies = [energy_test(t)[0] for t in circle]
assert np.allclose([min(energies), max(energies)], [3, 9])
small = np.geomspace(1e-5, 0.2, 60)
energy, residual = np.array([energy_test(t) for t in small]).T
fig, axes = plt.subplots(1, 2, figsize=(8, 3.3))
axes[0].plot(np.degrees(circle), energies)
axes[0].set(xlabel="Angle (degrees)", ylabel="Rayleigh quotient")
axes[1].loglog(small, energy-3, label="Energy error")
axes[1].loglog(small, residual, label="Residual")
```

```
axes[1].set(xlabel="Angle (radians)", ylabel="Magnitude")
axes[1].legend()
fig.tight_layout()
plt.show()
```

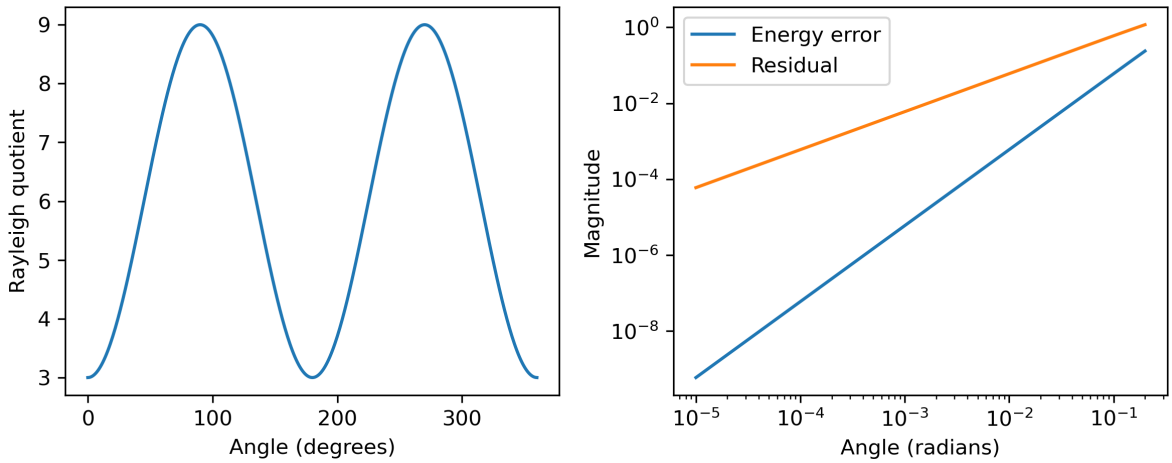


Figure 21.3: Rayleigh quotient along a unit circle and error orders near the mode of eigenvalue three.

21.13.6 F. Solve the reduced problem and test it in the full space

Compare the residual within the trial space with its full norm. Checking only the smaller residual would miss the coupling to the excluded direction.

```
Q = tilted_frame(np.pi/4)
H = Q.T @ A @ Q
ritz, Y = eigh(H)
U = Q @ Y
R = A @ U - U*ritz
print("Compression:\n", H, "\nRitz values:", ritz)
print("Full residual norms:", norm(R, axis=0))
print("Projected residual norm:", norm(Q.T @ R))
assert np.all(ritz >= lam[:2]-1e-12) and np.all(ritz <= lam[1:]+1e-12)
assert norm(Q.T @ R) < 1e-12
```

```
Compression:
 [[1. 0.]
```

```
 [0. 6.]]
Ritz values: [1. 6.]
Full residual norms: [0. 3.]
Projected residual norm: 1.1348741955820997e-15
```

21.13.7 G. Keep only the strongest channel of a rectangular map

The two singular-vector equations test input and output coordinates separately. The pseudoinverse discards the invisible antisymmetric input, while a rank-one approximation discards the weak output channel as well.

```
sensor = np.array([[1., 1., 0.], [0., 0., 1.]])
left, sig, right_t = svd(sensor, full_matrices=False)
assert np.allclose(sensor @ right_t.T, left*sig)
assert np.allclose(sensor.T @ left, right_t.T*sig)
pinv = (right_t.T / sig) @ left.T
assert np.allclose(pinv, np.linalg.pinv(sensor))
rank_one = sig[0]*np.outer(left[:, 0], right_t[0])
error = sensor-rank_one
assert np.allclose([norm(error, 2), norm(error, "fro")], [sig[1], sig[1]])
print("Minimum-norm input for output (2,3):", pinv @ [2., 3.])
print("Optimal rank-one error:", norm(error, 2))
```

```
Minimum-norm input for output (2,3): [1. 1. 3.]
Optimal rank-one error: 1.0
```

21.13.8 H. Compare vector and block convergence with predicted factors

Use a rotated symmetric problem with prescribed spectrum. The graph norm below measures small angles without taking the inverse cosine of a number rounded close to one. For the vector the reference rate is 6/9, for the two-dimensional space it is 3/6. Figure 21.4 includes upper bounds computed from the initial errors. A bound can overestimate the observed error if the starting overlap with the slow component is small.

```
rng = np.random.default_rng(2718)
basis, _ = qr(rng.normal(size=(6, 6)))
values = np.array([9., 6., 3., 2., 1.5, 1.])
B = (basis*values) @ basis.T
initial, _ = qr(rng.normal(size=(6, 2)))
```

```
def graph_tangent(X, k):
    upper = basis[:, :k].T @ X
    lower = basis[:, k:].T @ X
    graph = solve(upper.T, lower.T).T
    return norm(graph, 2)

x = initial[:, :1].copy()
X = initial.copy()
line_err, plane_err, value_err = [], [], []
for step in range(20):
    line_err.append(graph_tangent(x, 1))
    plane_err.append(graph_tangent(X, 2))
    value_err.append(abs(values[0] - (x.T @ B @ x).item()))
    x = B @ x
    x /= norm(x)
    X, _ = qr(B @ X)
steps = np.arange(len(line_err))
fig, ax = plt.subplots(figsize=(7, 3.9))
for errors, ratio, name, color in [
        (line_err, 6/9, "Line tangent", "tab:blue"),
        (plane_err, 3/6, "Plane tangent", "tab:orange")]:
    bound = errors[0]*ratio**steps
    assert np.all(np.array(errors) <= bound + 1e-12)
    ax.semilogy(steps, errors, color=color, label=name)
    ax.semilogy(steps, bound, "--", color=color, label=f"Bound ({ratio:.2f})")
ax.semilogy(steps, value_err, ":", color="black", label="Rayleigh error")
ax.set(xlabel="Iteration", ylabel="Error or bound")
ax.legend(ncol=2, fontsize=8)
fig.tight_layout()
plt.show()
```

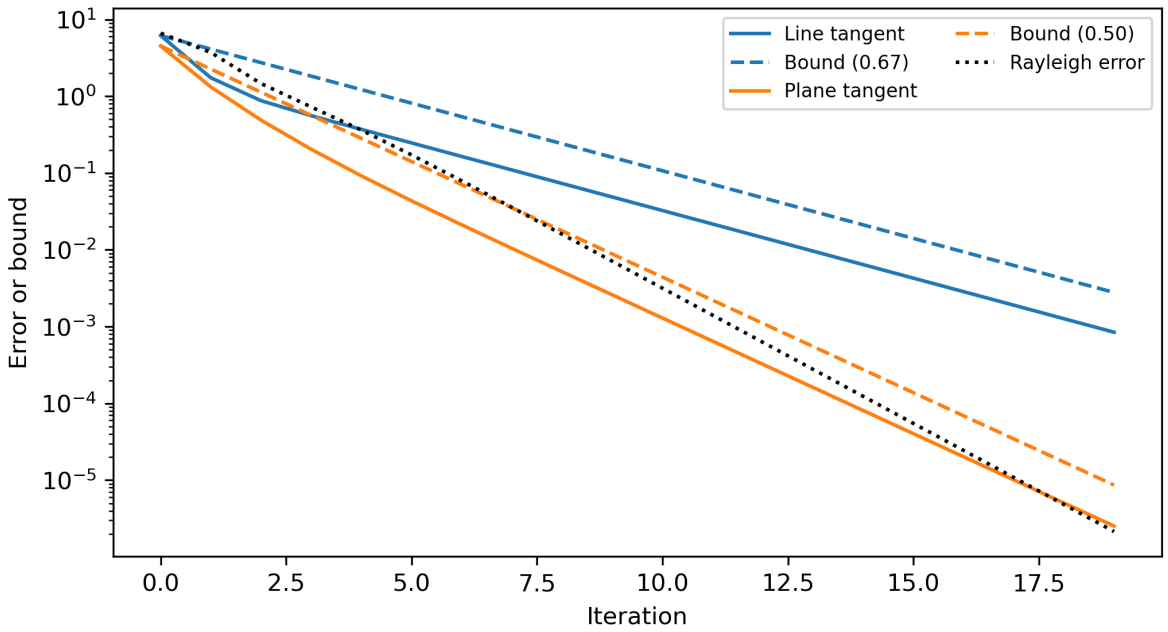


Figure 21.4: Power and block iteration on a controlled spectrum. Dashed lines give exact-arithmetic tangent bounds.

21.13.9 I. Change the degree and inspect the response

Return to A_0 with the external reference point 9. The following implementation handles constant and linear polynomials as well as higher degrees and accepts either a vector or a block. It uses unnormalized intermediate blocks, suitable for the modest degrees here. Figure 21.5 shows the absolute gain on a logarithmic axis; exact filter zeros are displayed at a positive plotting floor.

```
def cheb_value(x, degree):
    coeff = np.zeros(degree+1)
    coeff[-1] = 1
    return np.polynomial.chebyshev.chebval(x, coeff)

def affine(t, a, b):
    return (2*t-a-b)/(b-a)

def filter_block(matrix, block, degree, a, b, reference):
    def shifted_product(x):
        return (2*(matrix @ x) - (a+b)*x)/(b-a)
    if degree == 0:
```

```
        return block.copy()
    prev, curr = block, shifted_product(block)
    for _ in range(1, degree):
        prev, curr = curr, 2*shifted_product(curr) - prev
    return curr/cheb_value(affine(reference, a, b), degree)

grid = np.linspace(1, 9, 801)
fig, ax = plt.subplots(figsize=(6.4, 3.7))
for degree in (1, 2, 4):
    gain = cheb_value(affine(grid, 1, 3), degree)/cheb_value(7., degree)
    ax.semilogy(grid, np.maximum(np.abs(gain), 1e-7), label=f"Degree {degree}")
    output = filter_block(A, V @ np.ones(3), degree, 1, 3, 9)
    expected = cheb_value(lam-2, degree)/cheb_value(7., degree)
    assert np.allclose(V.T @ output, expected)
ax.axvspan(1, 3, alpha=.1, color="gray")
ax.set(xlabel="Spectral value", ylabel="Absolute gain", ylim=(1e-7, 2))
ax.legend()
fig.tight_layout()
plt.show()
```

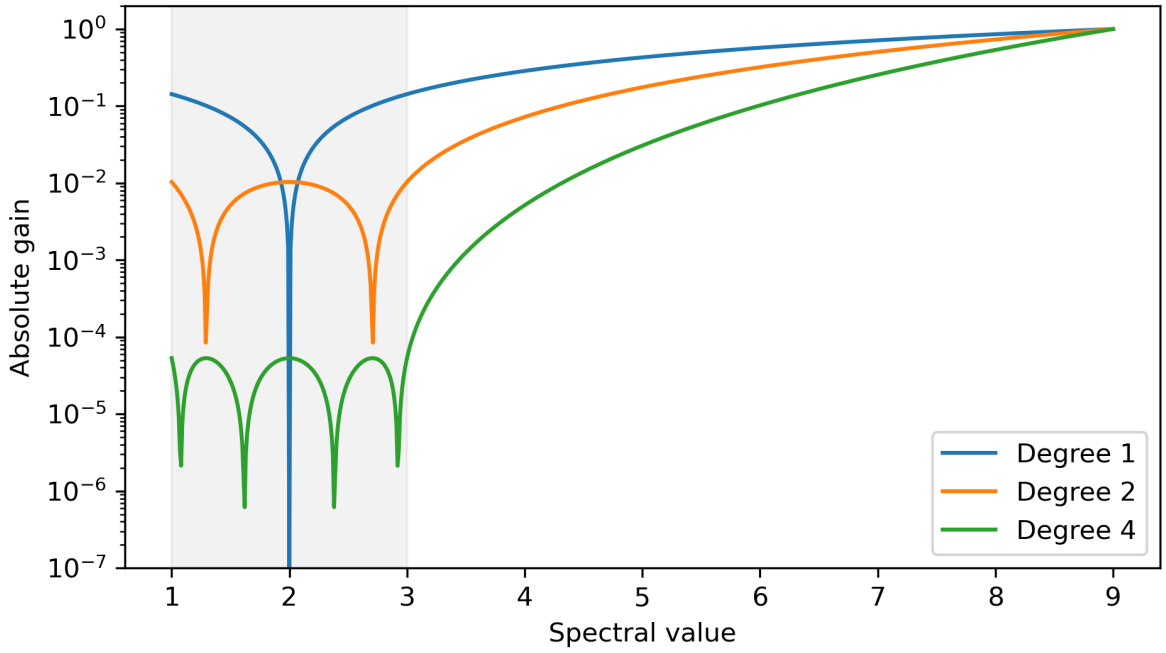


Figure 21.5: Normalized Chebyshev gain. The shaded interval contains unwanted eigenvalues and the reference at nine has unit gain.

21.13.10 J. Filter, orthonormalize, extract and diagnose

Use the six-dimensional problem from H and select both leading modes. Enclose the other four in $[1, 3]$, and normalize at 6, the weaker of the two target eigenvalues under this filter. For degree two the outside minimum gain is one and the unwanted maximum is $1/T_2(4) = 1/31$. Here "outside" refers to the target outside the unwanted interval. Inspect the bound alongside each measured tangent. The residual needs a separate computation with the original matrix.

```
d, a, b, ref = 2, 1., 3., 6.
gains = cheb_value(affine(values, a, b), d)/cheb_value(affine(ref, a, b), d)
gamma = max(abs(gains[2:]))/min(abs(gains[:2]))
X = initial.copy()
start = graph_tangent(X, 2)
print("Step  Ritz values          max residual   tangent      bound")
for step in range(1, 8):
    X, _ = qr(filter_block(B, X, d, a, b, ref))
    theta, rotation = eigh(X.T @ B @ X)
    X = X @ rotation
    max_residual = max(norm(B @ X - X*theta, axis=0))
    tangent = graph_tangent(X, 2)
    predicted = start*gamma**step
    assert tangent <= predicted + 1e-12
    print(f"{step:>3}  {theta}  {max_residual:12.3e} "
          f"{tangent:11.3e} {predicted:11.3e}")
```

```
Step  Ritz values          max residual   tangent      bound
  1  [5.93278 8.99579]     5.360e-01   1.282e-01   1.458e-01
  2  [5.99993 9.     ]     1.743e-02   4.070e-03   4.702e-03
  3  [6. 9.]     5.612e-04   1.309e-04   1.517e-04
  4  [6. 9.]     1.809e-05   4.222e-06   4.893e-06
  5  [6. 9.]     5.836e-07   1.362e-07   1.578e-07
  6  [6. 9.]     1.883e-08   4.393e-09   5.091e-09
  7  [6. 9.]     6.073e-10   1.417e-10   1.642e-10
```

Ritz values are printed in increasing algebraic order, so the leading pair approaches $(6, 9)$. Measuring against a known target makes this a controlled experiment. For a large unknown problem, residuals and estimated separation are accessible, while the true tangent usually is not. FP64 data can follow the geometric estimate for a finite range of steps without establishing a theorem about rounding-error propagation.

21.13.11 K. Discretize a differential eigenproblem and refine the mesh

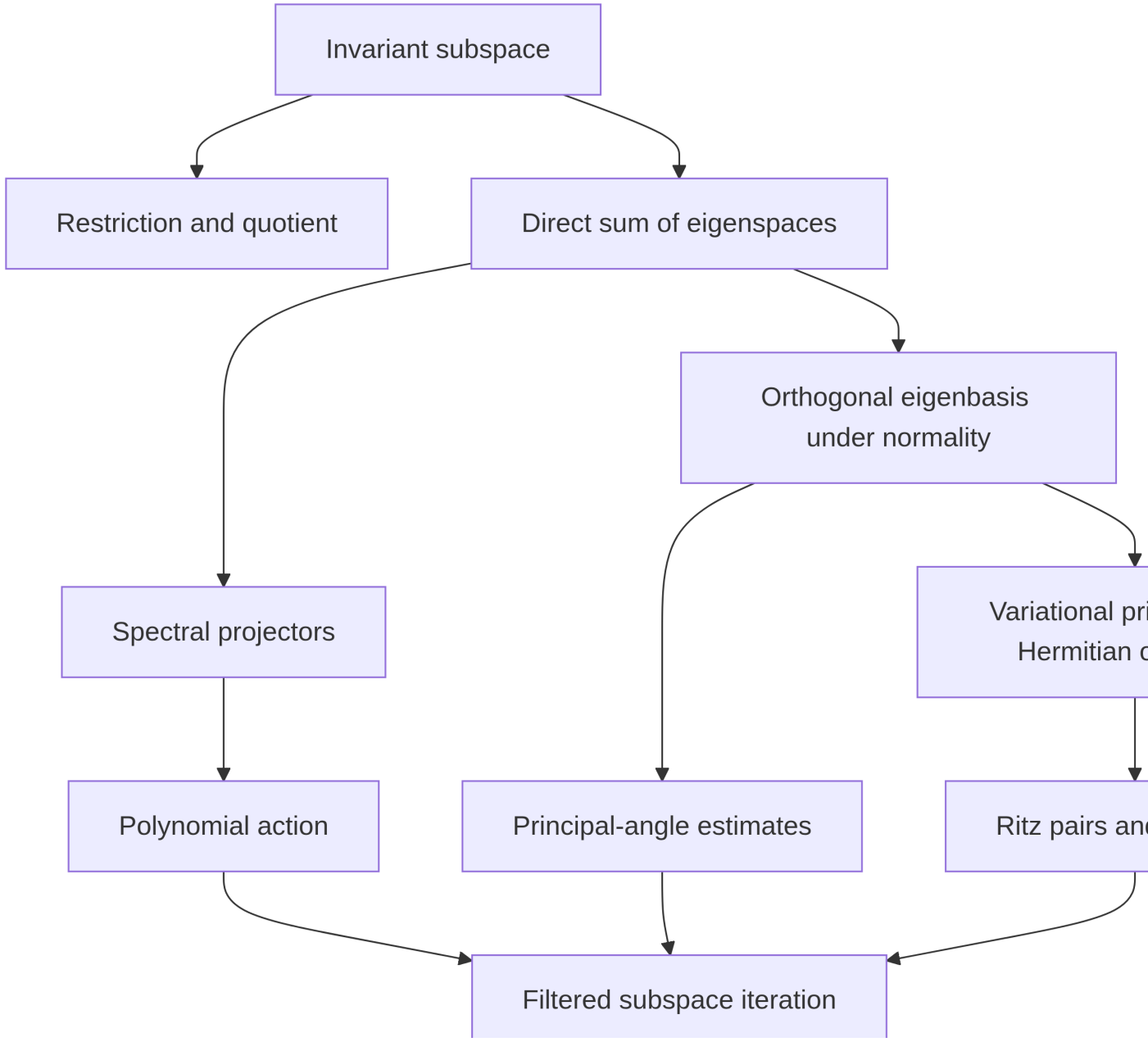
Factor $M = LL^T$. The substitution $z = L^T c$ changes $Kc = \lambda Mc$ into $L^{-1}KL^{-T}z = \lambda z$. Two triangular systems on opposite sides implement the change of coordinates; the code uses NumPy solves. The continuum error changes with refinement even when the algebraic residuals are tiny on every mesh.

```
for elements in (4, 8, 16, 32):
    n, h = elements-1, 1/elements
    off = np.diag(np.ones(n-1), 1)
    K = (2*np.eye(n) - off - off.T)/h
    M = (4*np.eye(n) + off + off.T)*(h/6)
    L = np.linalg.cholesky(M)
    left_solved = solve(L, K)
    standard = solve(L, left_solved.T).T
    vals, Z = eigh(standard)
    C = solve(L.T, Z[:, :2])
    residual = K @ C - (M @ C)*vals[:2]
    exact = (np.arange(1, 3)*np.pi)**2
    print(f"{elements:2} elements  continuum errors {vals[:2]-exact}"
          f"  residual {norm(residual):.2e}"
          f"  mass orthogonality {norm(C.T @ M @ C-np.eye(2)):.2e}")
    assert np.all(vals[:2] > exact)
    assert np.allclose(C.T @ M @ C, np.eye(2))
```

```
 4 elements  continuum errors [0.51704 8.52158]  residual 1.21e-14  mass orthogonality 4.20e
 8 elements  continuum errors [0.12748 2.06815]  residual 7.33e-14  mass orthogonality 1.10e
16 elements  continuum errors [0.03175 0.50991]  residual 3.59e-13  mass orthogonality 7.20e
32 elements  continuum errors [0.00793 0.127  ]  residual 7.53e-13  mass orthogonality 6.00e
```

These coefficient vectors have unit *mass* norm. Their ordinary Euclidean lengths depend on the mesh and basis. For the first mode, halving h approximately quarters the eigenvalue error once the mesh resolves it. One can check this rate directly here: the discrete sine vector has stiffness factor $(2-2\cos(j\pi h))/h$ and mass factor $h(2+\cos(j\pi h))/3$. Taylor expansion of their ratio at fixed j gives $(j\pi)^2 + (j\pi)^4 h^2/12 + O(h^4)$. The algebraic residual tests the assembled equations and cannot detect this discretization term.

21.14 Synthesis — Which structure does the computation use?



Read the arrows as dependencies with hypotheses. A single invariant subspace always gives a restriction and quotient. Reaching the direct-sum branch requires enough eigenvectors. On the right, orthogonality supplies norm identities, Hermitian structure orders energies, and separation makes an angular certificate possible.

Class of operator	Available structure	Example or diagnostic
General endomorphism	Invariance gives block triangular form.	Jordan block has an eigenline and no invariant complement to it.
Diagonalizable over $\mathbb{F}$	Direct sum, dual coordinates, possibly oblique projectors.	B from Session 2 has $P_1^* \neq P_1$.
Complex normal	Unitary eigenbasis, orthogonal spectral projectors.	A quarter-turn has unit-modulus complex eigenvalues.
Hermitian	Real spectrum and variational order.	A_0 has three orthogonal vibration modes.
Hermitian positive definite	Strictly positive energies, invertible positive square root.	A mass matrix defines a metric on coefficients.

Construction	Hypotheses	Quantity recovered
Algebraic projector	Diagonalizable, direct sum of eigenspaces	Component along the other eigenspaces; the kernel specifies the direction of projection.
Orthogonal projector	Orthonormal frame of a subspace	Closest component in the subspace, kernel its perpendicular complement.
Eigenvalue	Endomorphism over the chosen field	Scalar action on an invariant line; invariant under similarity.
Singular value	Any map between finite-dimensional inner-product spaces	Gain between right and left axes; invariant under independent unitary basis changes.
Principal angles	Two orthonormal frames	Singular values of their cross-Gram matrix give angle cosines.
Courant–Fischer	Hermitian, increasing algebraic ordering	Ordered eigenvalues via extrema over subspaces.
Ritz residual	Hermitian, orthonormal trial basis	A spectral inclusion; an angle bound after identifying an outside gap.

Construction	Hypotheses	Quantity recovered
Truncated SVD	Rank budget and spectral or Frobenius norm	Best approximation error given by the discarded singular values.
Graph contraction	Normal, full target overlap, nonzero wanted gains	Worst tangent reduced by at most the unwanted-to-wanted gain ratio.

Three diagnostic questions. If a program returns different vectors after changing basis, compare projectors first: a repeated eigenspace has many bases. If a Ritz value looks accurate, compute the full residual and identify the neighboring spectrum: energy errors can cancel. If filtering stagnates, inspect the actual wanted and unwanted gains and the initial overlap before assigning the behavior to finite precision.

21.14.1 Integrated problem — Certify a filtered mode without its coordinates

For A_0, start from $x = (q_2 + q_3)/\sqrt{2}$. Construct the exact projector onto the highest mode, apply one degree-two interval filter, compute the resulting one-dimensional Ritz pair, then use its residual to certify the angle without inserting the exact eigenvector into the certificate. Compare the result with two ordinary power steps.

Solution. The spectral projector is $(A_0 - I)(A_0 - 3I)/48$ and maps x to $q_3/\sqrt{2}$. The filter normalized at 9 has gains $1/97, 1/97, 1$. The new unit vector is $u = (q_2 + 97q_3)/\sqrt{9410}$. Its tangent to the target is $1/97$. Its Ritz value is $\theta = 9 - 6/9410$, and the two-mode residual formula gives $\|r\| = 6 \cdot 97/9410$. Knowing only that the unwanted spectrum is at most 3 gives $\delta = \theta - 3$ and $\sin t \le \|r\|/(\theta - 3) = 1/97$. Direct geometry gives $\sin t = 1/\sqrt{9410}$, verifying the certificate. Two ordinary power steps give tangent $1/9$. The exact polynomial projector, the interval filter and the residual answer three distinct questions: what the target component is, how to amplify it, and how to assess the extracted result.

21.14.2 Mastery assessment

Reconstruct without notes	A successful check
Restrict an operator and pass to its quotient.	Identify both diagonal blocks, exhibit a nonsplitting extension, and explain field dependence.

Reconstruct without notes	A successful check
Decide diagonalizability and recover projectors.	Count geometric multiplicities, compute the dual basis, and distinguish rank-one choices from whole eigenspaces.
Prove both spectral theorems.	Explain why the orthogonal complement reduces the operator and where reality of eigenvalues enters.
Derive polynomial functional calculus.	Construct a Lagrange projector and explain the derivative term on a Jordan block.
Compare subspaces.	Derive the cross-Gram singular values and the principal-plane projector formula.
Recover variational characterizations.	Derive quotient bounds and stationarity, then prove both min–max formulas by dimension counting.
Certify a Ritz approximation.	Derive orthogonality, interlacing, spectral inclusion and an outside-gap angle bound, including the generalized metric.
Construct an SVD.	Identify all four subspaces, solve minimum-norm least squares and prove both rank-truncation bounds.
Explain exact iteration.	Derive the power factor, the graph update, full-overlap condition and largest-angle bound.
Design a Chebyshev filter.	Map the interval, prove normalized minimax optimality, evaluate by recurrence and produce a failed-window example.
Change the mathematical setting.	Derive stiffness and mass forms, distinguish spectrum from point spectrum, and identify submodules and primary projectors.
Transfer to Module E.	Separate a predicted exact contraction, a measured residual, a gap estimate and the precision in which each was evaluated.

Module E starts where the last certificate becomes a practical question. A block product is computed with error, each Chebyshev recurrence step propagates earlier errors, and orthonormalization can lose accuracy. A residual evaluated in higher precision can reveal an accuracy floor hidden by a cheaper product. The exact contraction derived here supplies a reference curve against which mixed-precision arithmetic, communication costs and stopping criteria can be investigated.

22 Module E — Numerical Eigenvalue Methods: Conditioning, Stability, Polynomial Filtering, and Mixed Precision

22.1 From a spectral decomposition to a numerical claim

Suppose a program returns a unit vector x and a number θ. You can multiply by A, but you do not know any of its eigenvectors. What calculation would justify the statement "this is an accurate approximation to the second eigenvector"? Computing $\|Ax - \theta x\|$ is a first step. You still need to identify the eigenvalue, establish separation from competing directions, and know how accurately you evaluated the residual itself. Each of the ten sessions develops one part of that argument.

Starting point in Review II. Module A (§4.11) supplied invariant subspaces and direct sums. Module B (§5.7) explained how a functional pulls back through a map. Module C (§§6.7, 6.11) supplied orthogonal projections and the mass-matrix inner product. Module D (§§7.5–7.10) established the spectral theorem, principal angles, Rayleigh–Ritz, and polynomial filtering. Its §7.18 asks what accuracy and cost survive implementation. Here a spectral theorem becomes both a way to design an algorithm and a way to audit its output.

Running example and conventions. Work over a finite-dimensional real or complex inner-product space, using orthonormal coordinates unless specified. A star denotes conjugate transpose. Write $\|\cdot\|$ for the Euclidean or induced spectral norm, $\|\cdot\|_F$ for the Frobenius norm. The Rayleigh quotient is $\rho_A(x) = x^*Ax/(x^*x)$. Reuse the matrix from §7.5.5,

$$A_0 = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 9 \end{pmatrix}, \quad q_1 = (1,-1,0)^T/\sqrt{2}, \quad q_2 = (1,1,0)^T/\sqrt{2}, \quad q_3 = e_3.$$

The corresponding eigenvalues are $1, 3, 9$. These subscripts stay attached to these vectors. For general results we state the ordering separately, since largest magnitude, largest algebraic value and closest to a shift are three different targets.

Route through the chapter. Sessions 1–3 build certificates for computed answers. Sessions 4–6 derive vector, block and Krylov methods. Sessions 7–8 follow one Chebyshev filter from interval design to a bound on the angle of its computed output. Sessions 9–10 turn those

bounds into precision choices and a common-target solver comparison. Experiments A–J form one executable Python laboratory in reading order.

Write $\sigma(A)$ for the spectrum and $\rho(A) = \max_{\lambda\in\sigma(A)} |\lambda|$ for its radius. A hat denotes a computed quantity. A tilde denotes intentionally perturbed or quantized data. A residual written without a hat is the mathematical residual of the supplied pair, even when the pair itself was computed.

22.2 Session 1 — How accurately does a computer apply a matrix?

22.2.1 What one rounding step changes

In binary precision p a normalized positive number has form $(1.b_2 \dots b_p)_2 2^e$ with a bounded exponent. Adjacent numbers in $[1, 2)$ are 2^{1-p} apart. Rounding to nearest therefore gives unit roundoff $u = 2^{-p}$. In binary32, $p = 24$; in binary64, $p = 53$. The spacing at one is $2u$. The exponent controls range independently of the precision. For a correctly rounded real operation with exact result in the normal range and no overflow,

$$\mathrm{fl}(a \circ b) = (a \circ b)(1 + \delta), \quad |\delta| \le u.$$

Indeed half an ulp in the binade starting at 2^e is $2^e u$, while the result has magnitude at least 2^e. Zero is exact. Near underflow one needs an absolute-error term. A fused multiply-add rounds the combined expression once. Complex arithmetic implemented with real operations has different operation counts.

Hand experiment. With $p = 3$ and ties to even, $1 + 1/8$ rounds to 1. Thus $(1 + 1/8) - 1$ evaluates to zero while $(1 - 1) + 1/8$ evaluates exactly. The final subtraction in the first expression is itself exact. It exposes the loss in the preceding sum. This is the same difficulty that arises when subtracting two large nearly equal terms in an eigenpair residual.

22.2.2 From dot products to a matrix product

For $ku < 1$ write $\gamma_k = ku/(1 - ku)$. A product of at most k factors $1 + \delta_j$ differs from one by at most γ_k: its upper bound is $(1+u)^k \le 1/(1-ku)$ by binomial expansion and the geometric series, and its lower bound is at least $1 - ku$. Unroll a sequential real dot product. Each exact summand $a_j x_j$ is multiplied by the rounding factor from its product and factors from subsequent additions, at most n in total when the first product initializes the sum. Thus

$$|\,\mathrm{fl}(a^T x) - a^T x| \le \gamma_n \sum_j |a_j x_j|, \qquad |\,\mathrm{fl}(Ax) - Ax| \le \gamma_n |A|\,|x|.$$

Absolute values and the second inequality are entrywise. Equivalently, for this multiply there exists ΔA with $|\Delta A| \le \gamma_n |A|$ giving the computed result exactly as $(A + \Delta A)x$. The perturbation can change with the input and with each multiply. For sparse rows one can replace n by their maximum number of summed products under the same arithmetic model. Cancellation can make the *relative output error* large even when the componentwise backward error is small.

For a sequential sum of n numbers there are at most $n - 1$ additions on the path of any summand, giving $|\hat{s} - s| \le \gamma_{n-1} \sum |x_j|$. A balanced pairwise reduction has depth $\lceil \log_2 n \rceil$ and gives the corresponding depth bound. Applying the dot-product argument to entry (i, j) of a block gives

$$|\widehat{AX} - AX| \le \gamma_n |A|\,|X|.$$

Each output column admits its own nearby matrix. Replacing these by a single small perturbation shared across the block requires another argument. On a GPU, fused instructions, accumulation format and reduction order determine which operation count belongs in the bound.

Binary format	Significand bits including leading bit	Normal exponents	Unit roundoff
binary16	11	-14 to 15	2^{-11}
bfloat16	8	-126 to 127	2^{-8}
TF32 input format	11	-126 to 127	2^{-11} for nearest rounding
binary32	24	-126 to 127	2^{-24}
binary64	53	-1022 to 1023	2^{-53}

Below the smallest normal $2^{e_{\min}}$, gradual underflow uses a fixed spacing $2^{e_{\min}-p+1}$. For example, binary16 represents 2^{-24} exactly and rounds 2^{-25} to zero with ties to even. At the other end, 2^{16} overflows binary16 although it is a power of two. TF32 describes a matrix-product input mode with a wider accumulator on supported hardware. A storage cast alone does not implement that pipeline.

22.3 Session 2 — Is the problem sensitive or the algorithm unstable?

22.3.1 Problem sensitivity versus the error of a method

For differentiable f the local absolute condition number in specified norms is $\|Df(d)\|$. For $d, f(d) \neq 0$, the relative one is $\|Df(d)\|\|d\|/\|f(d)\|$. This follows by maximizing the first-order

change $Df(d)h$ over a ball of input perturbations. Specifying which data may change is part of defining the problem.

A forward error measures distance from the exact output. A backward error is the least size of a permitted perturbation for which the computed output is exact. Backward stability means this size is small on the scale of arithmetic error under stated hypotheses. For $Ax = b$ with nonsingular A, fixed A, and $r = b - A\hat{x}$, the exact perturbed right side is $b - r$. Hence

$$x - \hat{x} = A^{-1}r, \qquad \frac{\|x - \hat{x}\|}{\|x\|} \le \kappa_2(A)\frac{\|r\|}{\|b\|}, \quad \kappa_2(A) = \|A\|\|A^{-1}\| = \sigma_{\max}/\sigma_{\min}.$$

Here $\|b\| \le \|A\|\|x\|$ supplies the last factor. For $A = \mathrm{diag}(1, \varepsilon)$, $b = e_1$ and $\hat{x} = (1, 1)^T$, the relative residual is ε and the relative forward error is one. The SVD from Module D explains this amplification by the long axis of the inverse image ellipsoid. In iterative refinement a more accurate residual helps recover the correction. The eigenvalue analogue requires spectral separation, developed next.

22.3.2 Which entries may change?

With simultaneous relative perturbations $|E| \le \eta|A|$ and $|f| \le \eta|b|$, feasibility of $(A + E)\hat{x} = b + f$ requires $|r_i| \le \eta(|A||\hat{x}| + |b|)_i$. This is also sufficient for real data. In each row select the signs of $E_{ij}\hat{x}_j$ and $-f_i$ to agree with r_i, and scale every allowed term in that row by the same fraction until their sum is r_i. Thus the smallest feasible η is the maximum of the row ratios. A zero denominator contributes zero if its residual is zero and infinity otherwise. This model preserves structural zeros. A normwise perturbation of the same relative size can fill them.

A backward-stable algorithm gives $\hat{y} = f(d + \Delta d)$ for $\|\Delta d\|/\|d\| \le Cu$ on the stated data class. Applying the definition of the derivative bounds the relative forward error locally by $Cu\kappa_f(d) + o(u)$ for fixed d. Forward stability asks directly for accuracy at this conditioning-dependent scale. A mixed analysis permits a small final output perturbation as well as perturbed inputs. A meaningful analysis specifies how C depends on dimension, implementation and any growth factors.

Checkpoint. For $a = (1, -1)$, $x = (1, 1 + \varepsilon)^T$, explain why the dot-product bound divided by $|a^T x|$ grows like $2\gamma_2/|\varepsilon|$. Which two terms cancel?

22.4 Session 3 — What does an eigenpair residual certify?

22.4.1 A residual as the price of exactness

For a unit vector x, set $r = Ax - \theta x$. A matrix E makes (θ, x) exact for $A + E$ precisely when $Ex = -r$. Consequently $\|E\| \ge \|r\|$, with equality for $E = -rx^*$. This is an exact,

normwise, absolute backward error allowing arbitrary matrix perturbations. For a nonzero unnormalized vector divide the residual norm by $\|x\|$. Normalizing further by $\|A\|$ gives a relative perturbation in A when $\|A\| > 0$.

If $A = A^*$ and $\theta = \rho_A(x)$, then $x^*r = 0$. Now $E = -rx^* - xr^*$ is Hermitian and has the same optimal norm. On the orthonormal pair $x, r/\|r\|$ its matrix has zero diagonal and off-diagonals $-\|r\|$, and it vanishes on their orthogonal complement. The case $r = 0$ takes $E = 0$. This construction preserves self-adjointness. Additional requirements such as a fixed sparsity pattern change the backward-error problem.

22.4.2 Locating an eigenvalue and locating a direction

Let $A = A^* = U\Lambda U^*$, $x = \sum_i c_i u_i$, $\|x\| = 1$. Pythagoras gives the identity

$$\|r\|^2 = \sum_i |\lambda_i - \theta|^2 |c_i|^2.$$

Since $\sum_i |c_i|^2 = 1$, some eigenvalue lies within $\|r\|$ of θ. To identify a target set I, let $P_I = \sum_{i \in I} u_i u_i^*$ and $\delta = \min_{i \notin I} |\lambda_i - \theta| > 0$. Discarding the target terms gives

$$\sin \angle(x, \operatorname{ran} P_I) = \|(I - P_I)x\| \le \|r\|/\delta.$$

Knowing which eigenvalues are excluded is substantive information. An exact eigenvector of the wrong eigenvalue has zero residual. Eigenvectors inside a multiple eigenspace are interchangeable, so a particular basis there cannot be certified from the equation.

For a simple target λ_j, expansion of the Rayleigh quotient gives $\rho_A(x) - \lambda_j = \sum_{i \ne j} (\lambda_i - \lambda_j)|c_i|^2$. Its absolute value is bounded by $\max_i |\lambda_i - \lambda_j| \sin^2 \angle(x, u_j)$. Thus the value can settle to many digits while its direction still moves appreciably. For $x = \cos t\, q_2 + \sin t\, q_3$, the value is $3 + 6\sin^2 t$ and the residual norm is $6|\sin t \cos t|$.

22.4.3 Weyl and subspace separation

For two Hermitian matrices A and $A + E$, their eigenvalues in increasing order satisfy $|\lambda_i(A + E) - \lambda_i(A)| \le \|E\|$. **Proof.** For every unit vector the Rayleigh quotients differ by at most $\|E\|$. Taking the maximum over vectors in an i-dimensional subspace and then the minimum over such subspaces preserves both inequalities. Apply Module D's min–max theorem. This is an absolute bound. Relative sensitivity may be large near zero.

Here is a residual version of the subspace separation argument behind Davis–Kahan. Let $X^*X = I$, $H = X^*AX$, and $R = AX - XH$. Rotate X by a unitary matrix to diagonalize

$H = \operatorname{diag}(\theta_j)$ in its new basis. If U_2 spans the exact unwanted eigenvectors with diagonal matrix Λ_2, then $S = U_2^* X$ satisfies

$$\Lambda_2 S - SH = U_2^* R.$$

If every $|\lambda_i(\Lambda_2) - \theta_j| \geq \delta > 0$, division entry by entry and summation of squares gives $\|S\|_F \leq \|R\|_F/\delta$. The square roots of the eigenvalues of S^*S are the sines of the principal angles when the target and trial dimensions agree. This proves a Frobenius sin-theta bound and therefore also bounds the largest sine. Entrywise division alone does not prove the same constant for a spectral-norm right side. For a perturbed invariant subspace of $A + E$, $R = -(I - XX^*)EX$; the denominator must still separate the perturbed Ritz spectrum from the unwanted spectrum of A.

Small-gap example. For $\operatorname{diag}(1, 1+g)$ add a real symmetric off-diagonal ε. The diagonalizing rotation has $|\tan(2t)| = 2|\varepsilon|/g$. Taking $\varepsilon = g/2$ rotates the axes by $\pi/8$ even as $g \to 0$. Appending a third eigenvalue 9 leaves the first two coordinates as an exact invariant plane. A well-determined plane may contain poorly determined individual directions.

22.4.4 Why nonnormality changes the certificate

If $A = V\Lambda V^{-1}$ and μ is an eigenvalue of $A + E$, then $\min_i |\mu - \lambda_i| \leq \kappa_2(V)\|E\|$ (Bauer–Fike). For μ outside the spectrum, apply $(\mu I - A)^{-1}$ to Ex for a unit perturbed eigenvector x. The resolvent has norm at most $\kappa_2(V)/\min_i |\mu - \lambda_i|$, giving the claim. A normal matrix admits unitary V. For a simple eigenvalue, differentiating $Ax = \lambda x$ and testing with a left eigenvector y^* gives $\dot\lambda = y^*\dot{A}x/(y^*x)$. This is Module B's dual measurement in action. Nearly orthogonal left and right eigenvectors amplify perturbations.

For the Jordan block $J = \left(\begin{smallmatrix} 0 & 1 \\ 0 & 0 \end{smallmatrix}\right)$, adding ε in position $(2,1)$ produces eigenvalues $\pm\sqrt{\varepsilon}$. Even the order of the displacement changes at a defective eigenvalue. More generally the ε-pseudospectrum consists of the possible eigenvalues under $\|E\| \leq \varepsilon$. The rank-one construction identifies it exactly with those θ admitting a unit vector of residual at most ε.

Checkpoint. If a normalized Hermitian Ritz vector has residual 10^{-8} and every unwanted eigenvalue is at least 10^{-3} away from its Ritz value, what is certified? Some eigenvalue is within 10^{-8} and the sine to the identified target space is at most 10^{-5}.

22.5 Session 4 — Vector iterations as spectral transformations

22.5.1 Power iteration and its rate

Given a unit x_0, form $y = Ax_k$, $x_{k+1} = y/\|y\|$ whenever $y \neq 0$. Normalization controls the scale and preserves the line generated by $A^{k+1}x_0$. For Hermitian A, order $|\lambda_1| > |\lambda_2| \geq \cdots$ and

write $x_0 = \sum c_i u_i$ with $c_1 \neq 0$. Divide the squared norm of the unwanted components by that of the desired component:

$$\tan^2 \angle(x_k, u_1) = \sum_{i>1} \left|\frac{c_i}{c_1}\right|^2 \left|\frac{\lambda_i}{\lambda_1}\right|^{2k}.$$

Consequently the tangent decreases at least as fast as the bound $(|\lambda_2/\lambda_1|)^k \tan \angle(x_0, u_1)$. Equality holds if all active unwanted components have modulus $|\lambda_2|$. The Rayleigh quotient error is quadratic in the sine by Session 3. A negative dominant eigenvalue reverses sign each iteration. Monitor the line or its residual. A zero starting coefficient stays zero in exact arithmetic. Equal dominant moduli can prevent convergence to a line, as $\mathrm{diag}(1, -1)$ shows.

For A_0, start at $(q_2 + q_3)/\sqrt{2}$. After two multiplications the tangent to q_3 is $1/9$, and the Rayleigh quotient is $(3 + 9 \cdot 81)/82$. Compare the error $6/82$ with the angle $\arctan(1/9)$. In an optimization interpretation, ρ_A on the unit sphere has tangential gradient $2(Ax - \rho_A(x)x)$ in real coordinates. A stationary point is an eigenvector. Power iteration selects by modulus, so on an indefinite matrix it need not maximize the quotient.

22.5.2 Inverse iteration and Rayleigh quotient iteration

Choose a real shift σ outside the spectrum. Solve $(A - \sigma I)y = x_k$ and normalize y. In the eigenbasis the component multiplier is $1/(\lambda_i - \sigma)$. For a unique closest eigenvalue λ_t, nonzero starting overlap, and second-closest distance d_2, the tangent bound contracts per step by $|\lambda_t - \sigma|/d_2$. A fixed shift permits reuse of a factorization. For A_0, $\sigma = 2.8$ gives distances $1.8, 0.2, 6.2$. The target is q_2 and the worst-case factor is $1/9$. Near-singularity is the source of selectivity. It also calls for stable shifted solves and residual checks on the original eigenproblem. A solve residual $s = x_k - (A - \sigma I)\hat{y}$ changes the inverse application by $-(A - \sigma I)^{-1}s$.

Rayleigh quotient iteration (RQI) recomputes $\sigma = \rho_A(x_k)$ each time. Assume A Hermitian, λ_t simple, gap $g > 0$, and x_k close enough that $|\rho_A(x_k) - \lambda_t| < g/2$. Put $t = \angle(x_k, u_t)$ and $D = \max_i |\lambda_i - \lambda_t|$. When the shifted solve exists, the same component calculation gives

$$\tan t_{\mathrm{new}} \leq \frac{|\lambda_t - \rho_A(x_k)|}{\min_{i \neq t} |\lambda_i - \rho_A(x_k)|} \tan t \leq (2D/g) \sin^2 t \, \tan t.$$

This proves local cubic convergence of the direction. Each changing shift can require a new factorization. Check convergence before solving: at an eigenvector the shifted matrix is singular. At an interior eigenvalue cancellation can also give a singular shift away from an eigenvector. For $\mathrm{diag}(-1, 0, 1)$, the equal mixture of the outer directions has quotient zero. A practical safeguard detects a failed solve and selects a perturbed shift. The cubic estimate makes no global claim about which eigenpair a distant start selects.

Newton interpretation. For real symmetric A, linearize $(Ax - \lambda x, (x^T x - 1)/2) = 0$ at a unit vector and its Rayleigh quotient. The correction obeys $(A - \lambda I)\Delta x - x\Delta\lambda = -r$ and $x^T \Delta x = 0$. Rearrangement yields $(A - \lambda I)(x + \Delta x) = x\Delta\lambda$. Whenever the Newton system and shifted solve are nonsingular, the normalized Newton direction is the RQI direction up to sign. Rayleigh quotient extraction supplies the next shift.

Checkpoint. Explain why a smaller shift distance helps inverse iteration in exact arithmetic and simultaneously increases the potential amplification of a linear-solve residual.

22.6 Session 5 — Block iteration, principal angles, and extraction

22.6.1 Why a basis needs attention

To find a p-dimensional invariant space, start with $X_0^* X_0 = I_p$, form $Y = AX_k$, and take X_{k+1} as the thin Q in $Y = QR$. Provided Y has full column rank, R is invertible and the range is unchanged by this operation. Independent power iterations tend to align all columns with the strongest direction. A QR step restores a basis with which the weaker directions can still be represented. A unitary change of basis $X \mapsto XW$ leaves the orthogonal projector XX^* unchanged.

Write $A = [U_1, U_2]\,\mathrm{diag}(\Lambda_1, \Lambda_2)[U_1, U_2]^*$ for Hermitian A, with p target columns in U_1. Put $C = U_1^* X_0$, $D = U_2^* X_0$. Assume C invertible and every target eigenvalue nonzero. In eigencoordinates a basis of the iterated space can be written

$$\begin{pmatrix} \Lambda_1^k C \\ \Lambda_2^k D \end{pmatrix} (\Lambda_1^k C)^{-1} = \begin{pmatrix} I \\ Z_k \end{pmatrix}, \qquad Z_k = \Lambda_2^k D C^{-1} \Lambda_1^{-k}.$$

This is the graph of a map from the target to its orthogonal complement. A vector on it has tangent $\|Z_k v\|/\|v\|$ to the target. Maximization therefore gives $\tan\theta_{\max} = \|Z_k\|$. In particular

$$\tan\theta_{\max,k} \le \left(\frac{\max_{\text{out}} |\lambda|}{\min_{\text{in}} |\lambda|} \right)^k \|DC^{-1}\|.$$

Separation is needed across the boundary of the target. Multiplicities wholly inside it cause no difficulty. Taking a block larger than the requested number can move that boundary beyond a cluster; then interpret the bound for the enlarged space and extract the requested Ritz pairs.

Two orthogonalization algorithms. For an incoming column z, classical Gram–Schmidt computes all coefficients $Q^* z$ before subtracting $Q(Q^* z)$. Modified Gram–Schmidt updates $z \leftarrow z - q_i(q_i^* z)$ after each coefficient, so later measurements see the revised vector. Both formulas coincide with exact orthonormal Q. For $z_1 = (1, \epsilon)^T$, $z_2 = (1, 0)^T$, the independent component has size about ϵ. An absolute error of size u before normalization can therefore produce angular error of order u/ϵ. A second pass measures and removes components left by the first. Rank-revealing QR or an SVD can diagnose the point at which the residual direction is indistinguishable from perturbations in the input.

22.6.2 Rayleigh–Ritz and finite precision

With orthonormal X, compress to $H = X^*AX$. Solve $Hw_j = \theta_j w_j$ with orthonormal w_j, form $v_j = Xw_j$. Then $r_j = Av_j - \theta_j v_j$ satisfies $X^*r_j = 0$ by substitution. Conversely this orthogonality for $v = Xw$ is precisely the projected eigenproblem. Thus the defect annihilates every trial-space test direction (Module B), or equivalently has zero orthogonal projection there (Module C). For nonorthonormal full-rank Y the corresponding problem is $Y^*AYw = \theta Y^*Yw$. An ill-conditioned Gram matrix squares the basis condition number. Orthogonalize first.

For A_0, use $X = [q_2, (q_1 + q_3)/\sqrt{2}]$. The compression is $\mathrm{diag}(3, 5)$, yet 5 is not an eigenvalue of A_0. Its residual is $4(q_3 - q_1)/\sqrt{2}$, perpendicular to both columns. One exact Ritz pair can coexist with a poor second pair. Sorted Ritz values alone do not verify a block.

Householder QR uses unitary reflections $I - 2vv^*/(v^*v)$ to zero entries below a pivot without changing exact lengths. In floating point it is a standard stable choice for a dense block. Gram–Schmidt subtracts projections onto earlier columns. If this leaves a tiny vector, errors made in subtracting large components are magnified upon normalization. A second orthogonalization pass often repairs the defect. If the input block has lost numerical rank, reorthogonalization cannot recover information already lost. Monitor $\|I - X^*X\|$, the size and conditioning of the small triangular factor, and the direct residual. Computing small angles through $\arccos(\sigma_{\min}(U_1^*X))$ loses digits near one. Prefer $\|(I - U_1U_1^*)X\|$ or $\|U_2^*X\|$ for the largest sine.

Checkpoint. For trial columns $q_3, (q_2 + q_1)/\sqrt{2}$ and target $\mathrm{span}(q_3, q_2)$, calculate the tangent after one power step. It is $1/3$, even though the first column is already exact.

22.7 Session 6 — Krylov spaces and projected eigenproblems

22.7.1 Keeping the history

For $v \neq 0$, $\mathcal{K}_m(A, v) = \mathrm{span}(v, Av, \ldots, A^{m-1}v) = \{p(A)v : \deg p < m\}$. Power iteration keeps one polynomial. Krylov projection searches the span of all these candidates. If $A^m v$ lies in the earlier span then applying A to any spanning vector stays inside and the space is invariant. Algebraically, $p \mapsto p(A)v$ maps the polynomial ring onto the cyclic space generated by v. Its kernel is the ideal generated by the least-degree monic polynomial annihilating v. Polynomial division shows that its degree equals the eventual Krylov dimension. This is the first isomorphism theorem of Module A in the language of Module F2.

22.7.2 Arnoldi, Lanczos, and the residual shortcut

Normalize $v_1 = v/\|v\|$. Given $V_j = [v_1, \dots, v_j]$, project Av_j onto the basis, subtract that projection and normalize the remainder:

$$h_{ij} = v_i^* A v_j, \quad w = Av_j - \sum_{i=1}^{j} h_{ij} v_i, \quad h_{j+1,j} = \|w\|, \quad v_{j+1} = w/h_{j+1,j}.$$

A zero norm gives an invariant space in exact arithmetic. Stacking the identities gives

$$AV_m = V_m H_m + h_{m+1,m} v_{m+1} e_m^*, \qquad H_m = V_m^* A V_m.$$

H_m is upper Hessenberg because Av_j lies in the first $j+1$ basis vectors. For a unit eigenvector $H_m y = \theta y$, its lifted residual has norm $|h_{m+1,m}|\,|e_m^* y|$. Only the last coefficient and the last subdiagonal enter.

For Hermitian A, Hermitian symmetry of H_m forces all entries above its first superdiagonal to vanish as well. Writing its real diagonal as α_j and nonnegative subdiagonal as β_j gives the Lanczos recurrence

$$\beta_j v_{j+1} = Av_j - \alpha_j v_j - \beta_{j-1} v_{j-1}, \quad \beta_0 = 0.$$

In exact arithmetic earlier projections vanish by the Hessenberg identity and symmetry. In finite precision tiny components along converged Ritz directions can grow back, causing duplicate approximate eigenvalues. Full or selective reorthogonalization trades additional work and communication for control of this effect. The direct residual checks both the recurrence shortcut and the projected solve.

Two steps by hand. Start A_0 at e_1. Then $Ae_1 = 2e_1 + e_2$, so $\alpha_1 = 2$, $\beta_1 = 1$, $v_2 = e_2$. Next $Ae_2 = e_1 + 2e_2$, yielding $\alpha_2 = 2$, $\beta_2 = 0$. The projected problem has eigenvalues $1, 3$. The missing 9 cannot be recovered from this start because every polynomial in A preserves this coordinate plane.

Nonnormal diagnostic. A simple, diagonalizable example makes the dual measurement explicit. Let $N = \left(\begin{smallmatrix} 1 & K \\ 0 & 2 \end{smallmatrix}\right)$ for large $K > 0$. At eigenvalue one, right vector e_1 and left row $(1, -K)$ have pairing one but product of norms $\sqrt{1+K^2}$. A perturbation $\epsilon e_2 e_1^T$ changes the characteristic equation to $(\lambda - 1)(\lambda - 2) - K\epsilon = 0$. The eigenvalue near one is $(3 - \sqrt{1+4K\epsilon})/2 = 1 - K\epsilon + O((K\epsilon)^2)$ for $|K\epsilon| \ll 1$. Merely seeing a unit gap between eigenvalues would miss this amplification.

A sparse Arnoldi step needs one matrix application and projections against earlier vectors. For m steps, storing the full basis costs order nm and orthogonalization costs order nm^2. Short-recurrence Lanczos can generate coefficients with a few vectors, but recovering Ritz vectors later requires stored or regenerated basis information. Restarting bounds storage by retaining selected approximate spectral information before building a new space. Discarding too much

can erase progress near a cluster. Block iteration has fixed width and can exploit matrix–block products. A degree-d filtered block also belongs to a block Krylov space, but evaluates a prescribed polynomial without storing every power.

Checkpoint. What evidence distinguishes exact invariance from numerical breakdown? A tiny computed β_j is a clue. Directly test $\|AV_j - V_jH_j\|$ and $\|V_j^*V_j - I\|$ at the accuracy required by the application.

22.8 Session 7 — Designing and evaluating a Chebyshev filter

22.8.1 The design problem

Suppose the unwanted eigenvalues of a Hermitian operator lie in $[a, b]$ and all desired eigenvalues lie above b. Let $c = (a + b)/2$, $h = (b - a)/2 > 0$, and $B = (A - cI)/h$. The unwanted interval maps to $[-1, 1]$. A filter should be bounded there and large at every target. The same map works for a target below a using the parity of the polynomial. An interior band generally needs a different filter design: two exterior tails cannot both be mapped to this single unwanted interval by one affine map without also including the interior target.

Define $T_j(\cos t) = \cos(jt)$. The addition formula gives $T_0 = 1$, $T_1 = z$, $T_{j+1} = 2zT_j - T_{j-1}$. Thus $|T_j| \le 1$ on $[-1, 1]$, and $T_j(-z) = (-1)^j T_j(z)$. For $z > 1$ substitute $z = \cosh s$ into the same recurrence to obtain $T_j(z) = \cosh(js)$, $s = \operatorname{arcosh} z$. Growth is exponential for fixed exterior z. It slows as the target approaches the cutoff. Taking higher powers z^j also amplifies exterior components, but misses the extremal property below.

Normalized minimax theorem. Fix $z_* > 1$, $d \ge 1$. Among real polynomials p of degree at most d satisfying $p(z_*) = 1$, the least possible maximum of $|p|$ on $[-1, 1]$ is $1/T_d(z_*)$. The candidate is $p = T_d/T_d(z_*)$. To prove optimality, suppose a competitor has strictly smaller maximum. The difference $T_d - T_d(z_*)p$ alternates in sign at $\cos(j\pi/d)$ for $j = 0, \dots, d$. It has d distinct interior zeros and another zero at z_*. A nonzero polynomial of degree at most d cannot have $d + 1$ roots. This optimality concerns uniform suppression of an interval with a specified exterior normalization. It does not optimize wall time or rounding errors.

22.8.2 Subspace convergence and degree choice

The spectral theorem gives $T_d(B)u_i = T_d((\lambda_i - c)/h)u_i$. Apply the graph proof from Session 5 with each eigenvalue replaced by its filter value. For full initial target overlap,

$$\tan\theta_{\max,k} \le \rho_d^k \tan\theta_{\max,0}, \qquad \rho_d = \frac{\max_{\text{out}} |T_d((\lambda - c)/h)|}{\min_{\text{in}} |T_d((\lambda - c)/h)|}.$$

If the nearest target maps to $z_* > 1$ and all unwanted eigenvalues lie in the design interval, $\rho_d \le \operatorname{sech}(d \operatorname{arcosh} z_*)$. For a desired one-block contraction $\tau < 1$ choose $d \ge \operatorname{arcosh}(1/\tau)/\operatorname{arcosh} z_*$.

Round upward. This counts polynomial degree, hence block products, per application. It is a sufficient interval estimate. The finite set of actual unwanted eigenvalues can give a smaller ratio.

Running matrix. Seek q_3 of A_0. For $[1,3]$, $B = A_0 - 2I$ has eigenvalues $-1, 1, 7$. A degree-two filter has values $1, 1, 97$. Starting at $(q_2 + q_3)/\sqrt{2}$, two ordinary power steps leave tangent $1/9$. One degree-two filter also takes two products and leaves tangent $1/97$. Both calculations start with the same angle and cost the same number of applications. Differences in QR, memory access and coefficient operations enter an actual timing comparison.

22.8.3 Evaluating a polynomial without forming powers

Use $Y_0 = X$, $Y_1 = BX$, $Y_{j+1} = 2BY_j - Y_{j-1}$. Induction gives $Y_j = T_j(B)X$. Only successive blocks are needed. The product $BY = (AY - cY)/h$ can be implemented as an operator action. Forming B once can change cancellation, rounding and memory requirements. An exact-arithmetic scalar normalization common to the entire block leaves the resulting space unchanged.

To control range, choose an exterior reference $z_s > 1$, typically near the largest mapped target, and set $W_j = Y_j/T_j(z_s)$. Define $s_j = T_{j-1}(z_s)/T_j(z_s)$. Then

$$s_1 = 1/z_s, \quad s_{j+1} = 1/(2z_s - s_j), \quad W_0 = X, \quad W_1 = s_1 BX, \quad W_{j+1} = 2s_{j+1}BW_j - s_j s_{j+1} W_{j-1}.$$

This follows by dividing the recurrence by $T_{j+1}(z_s)$. It avoids computing the huge normalization factor explicitly. Common scaling controls overflow but leaves the ratio between the strongest and weakest wanted components intact. Large degree can make the filtered block nearly rank deficient before QR. Compare singular values of the small R factor, shorten the polynomial or orthogonalize between complete shorter filters. Inserting arbitrary independent normalizations into an unmodified three-term recurrence changes the polynomial.

22.9 Session 8 — What happens to an error made inside a filter?

22.9.1 What a local error becomes

For the unscaled recurrence define a local additive defect F_j in each computed step, incorporating whatever product and update errors occurred:

$$\widehat{Y}_0 = X, \quad \widehat{Y}_1 = BX + F_1, \quad \widehat{Y}_j = 2B\widehat{Y}_{j-1} - \widehat{Y}_{j-2} + F_j, \quad j \geq 2.$$

Let $U_0(z) = 1$, $U_1(z) = 2z$, $U_{k+1} = 2zU_k - U_{k-1}$ be the Chebyshev polynomials of the second kind (here U_k is a polynomial). Subtract the exact recurrence. Induction, using $U_0 = 1$ for

the newest defect, gives the exact error representation

$$\widehat{Y}_d - Y_d = \sum_{j=1}^{d} U_{d-j}(B) F_j.$$

This identity assumes a fixed exact B and exact initial block. Errors in storing or applying B can be included in F_j. It does not presume independent random signs. Projection onto any eigendirection reduces the propagation to scalar weights. For $z = \cos t$, $U_k(z) = \sin((k+1)t)/\sin t$, bounded by $k+1$ including the limiting endpoints (expand the sine quotient as a sum of unit complex exponentials). Consequently for the unwanted spectral projector P_2,

$$\|P_2(\widehat{Y}_d - Y_d)\| \le \sum_{j=1}^{d} (d-j+1)\|P_2 F_j\|.$$

For $z = \cosh s > 1$, the corresponding weight is $\sinh((k+1)s)/\sinh s$. An early error in a desired component can grow very large in absolute norm. Whether it rotates the range is a separate question. For instance, at $d = 3$ a defect in the first product contributes $(4B^2 - I)F_1$, while a defect in the last update contributes F_3 directly.

22.9.2 Initialization, storage and one local product

If the stored start is $X + G_0$ and the first multiply is $B(X + G_0) + F_1$, linearity contributes the additional term $T_d(B)G_0$ to the preceding sum. Alternatively, prescribe independent initial errors Z_0, Z_1 relative to X, BX. For $d \ge 2$, homogeneous propagation gives $U_{d-1}(B)Z_1 - U_{d-2}(B)Z_0$, followed by the sum starting at $j = 2$. Substitution of $Z_1 = BG_0 + F_1$ recovers the first convention. Counting both initial terms and a separate storage defect for the same conversion double-counts the error.

For a real stored B, let $D_j = \mathrm{fl}(B\widehat{Y}_{j-1}) - B\widehat{Y}_{j-1}$. Sequential products give $|D_j| \le \gamma_n |B||\widehat{Y}_{j-1}|$. If doubling is exact and a single rounded subtraction finishes the update, then

$$|F_j| \le 2|D_j| + u(2|B\widehat{Y}_{j-1} + D_j| + |\widehat{Y}_{j-2}|).$$

This follows by adding and subtracting the exact update of the computed operands. If B was stored as $B + E_s$, include $E_s\widehat{Y}_{j-1}$ in the product defect. The bound varies with block size in norm, input magnitudes and degree. A single fitted constant times unit roundoff can conceal all three effects.

22.9.3 Turning application error into angle error

In the target/complement coordinates write the input block as $\binom{C}{D}$, and write one computed filter output as $\binom{P_1 C + G_1}{P_2' D + G_2}$, where P_1, P_2' are diagonal target and unwanted filter values, and

G_i are the total filter errors in those coordinates. Put $\alpha = \min|\operatorname{diag} P_1|$, $\beta = \max|\operatorname{diag} P_2'|$. Right-multiplying by C^{-1} shows that the graph of the output has tangent bounded by

$$t_{\text{new}} \le \frac{\beta t + \|G_2 C^{-1}\|}{\alpha - \|G_1 C^{-1}\|}, \qquad t = \|DC^{-1}\|.$$

Indeed the smallest singular value of the top block after multiplication is at least the denominator, which must be positive. The unwanted numerator is bounded by the triangle inequality. This inequality explains why a global absolute error bound alone is a poor predictor of angle accuracy. Unwanted leakage adds a new direction and poor target overlap can amplify it.

If the resulting step bounds yield **uniform** constants $t_{k+1} \le qt_k + \eta$ with $0 \le q < 1$, summing a geometric progression gives $t_k \le q^k t_0 + \eta(1-q^k)/(1-q)$. The second term predicts a possible stagnation scale. It is an upper envelope, never a proof that every run stagnates there. Errors can cancel or decrease as the residual shrinks. An observed plateau can also be real. For a scalar graph update $z_{k+1} = qz_k + f$ with fixed positive leakage f and $0 < q < 1$, the exact limit is $f/(1-q)$. This is realized by target multiplier one, unwanted multiplier q and additive unwanted component f in graph coordinates. A verified residual and a gap estimate are needed to diagnose a real run.

22.9.4 Bounds that miss part of the spectrum

If an unwanted eigenvalue falls just outside the designed interval, its multiplier becomes $\cosh(d \operatorname{arcosh}|z|)$ in magnitude. Recompute the *actual* ratio ρ_d: convergence can slow and an unwanted direction can even dominate. Conversely an overly wide interval can bring the nearest target closer to 1, requiring higher degree. Obtain bounds from analysis of the operator, enclosing intervals such as Gershgorin bounds, or preliminary iterations with explicit margins. An interior Ritz estimate of an extreme eigenvalue alone is generally not an outer enclosure, as min–max already shows.

Checkpoint. With $[a, b] = [1, 5]$ and target 8, $z_* = 2.5$. Show $T_3(2.5) = 55$. If an unwanted 6 was missed, its mapped value is 1.5, giving $T_3(1.5) = 9$. The ratio estimate changes from $1/55$ to $9/55$ when this is the only missed eigenvalue.

22.10 Session 9 — Mixed precision and hardware-aware computation

22.10.1 An error budget by operation

Operation	Where accuracy enters	Measurement or design choice
Store and apply A	Quantization perturbs the operator before accumulation begins	Retain a reference action for verification
Evaluate the filter	Every recurrence defect is propagated by a different polynomial	Select degree, scaling, product and update precisions separately
QR	Small singular directions of the input may be obscured	Check orthogonality and inspect the triangular factor
Rayleigh–Ritz	Projection errors perturb a small Hermitian problem	Form and solve the compression accurately, check its symmetry
Stop and report	Cancellation in $AX - X\Theta$ may hide the true residual	Reapply the original operator in a specified verification precision

If storage gives $\widetilde{A} = A + E_s$ and arithmetic returns $\widetilde{A}X + F$, the product error relative to the original data is $E_sX + F$. Higher-precision accumulation can reduce F but does not remove E_sX. If X itself is quantized, expand $(A + E_s)(X + E_x)$ to see the additional terms $AE_x + E_sE_x$. In a distributed computation rounding communicated entries contributes to E_x, possibly only on boundary data. A uniform normwise model may overestimate that effect. Floating-point format, product precision, accumulation precision and communication format must be specified separately.

Verification bound. If the computed verification residual is $\hat{r} = r + e$ with $\|e\| \le \epsilon_v$, then $\|r\| \le \|\hat{r}\| + \epsilon_v$. A matvec bound from Session 1 plus bounds for the scale and subtraction can supply ϵ_v under a known arithmetic model. A double-precision direct residual is a practical independent diagnostic when the filter ran in single. Calling it a rigorous enclosure additionally requires controlling its own evaluation and norm rounding. Increasing the verification precision can reveal an existing error without changing the iterate.

A scalar identity behind residual-based methods. For a polynomial p and approximate eigenpair (x, θ), the factor theorem gives $p(z) - p(\theta) = (z - \theta)s_\theta(z)$. Therefore

$$p(A)x = p(\theta)x + s_\theta(A)r, \qquad r = Ax - \theta x.$$

For $p(z) = 2z^2 - 1$, $s_\theta(z) = 2(z + \theta)$. Near convergence the argument of the operator in the correction is small. If a product error is proportional to the norm of its input, the error in applying an operator to a small residual may shrink with convergence. An accurately evaluated residual and a controlled correction recurrence are essential. When the input residual is already corrupted at the scale of $u\|A\|$, the argument loses force. This identity provides a way into the seminar paper's residual-based recurrence. The ordinary recurrence in the lab is its comparison

case. For the actual algorithm and its numerical evidence see Kodali, Ramakrishnan and Motamarri, *Residual-based Chebyshev filtered subspace iteration*.

22.10.2 Cost of reaching an accuracy target

A dense matrix–block product with an $n \times p$ real block costs approximately $2n^2p$ scalar floating-point operations. A sparse product costs approximately $2\,\mathrm{nnz}(A)p$. A thin dense QR costs order np^2 and the compressed eigensolve costs order p^3. A degree-d filter takes d matrix–block products before extraction. These counts give no sustained bandwidth, occupancy or collective latency. For an illustrative machine model with peak rate F and bandwidth B_w, moving S bytes and doing W operations requires at least $\max(W/F, S/B_w)$ time for the modeled resources. Sparse indices, cache reuse, conversions, synchronization and contention affect actual time. A block may reuse matrix entries across columns. Distributed inner products require reductions, whereas stencil products often require neighbor exchanges.

For a fair comparison specify a target cluster, a common residual tolerance on the original operator, an orthogonality threshold, starting block and stopping rule. Time setup, transfers, filtering, QR, extraction and verification; include all repetitions needed to reach tolerance. Record failures to meet tolerance. A low-precision run with faster individual products can take longer to deliver an acceptable subspace, or never meet the criterion. Timing the dense teaching example says little about a sparse device implementation.

Checkpoint. A single-precision filter and double-precision QR produce $\|I - X^*X\| \approx 10^{-15}$ but a verified residual of 10^{-6}. These diagnostics can coexist: one measures internal geometry of the basis, the other tests its relation to A.

22.11 Session 10 — From a contraction factor to a verified eigenspace

22.11.1 Construct a problem with known answers

Take $n = 32$ and the orthonormal discrete cosine basis U. Its zeroth column is constant $1/\sqrt{n}$, and for row $i = 0, \dots, n-1$ and column $j = 1, \dots, n-1$ its entry is $\sqrt{2/n}\cos(\pi(i+1/2)j/n)$. Set $A = U\Lambda U^T$ with 29 equally spaced unwanted eigenvalues from 1 to 5 and targets $8, 9, 10$. The dense rotation is deliberate: rounding a diagonal example can preserve exact zeros and hide leakage between directions. The exact mathematical construction and its stored floating-point realization differ slightly.

For each target eigenvector take its equal mixture with a different unwanted eigenvector. Pair 8 with 5, 9 with 1, and 10 with the second-smallest eigenvalue. These three vectors are orthonormal. Each angle is initially $\pi/4$. For the pair determining the worst angle the power multiplier ratio is $5/8$ and the degree-three Chebyshev ratio is $1/55$. All other ratios are smaller.

The graph estimate is attained, so the predicted largest tangents are **exactly** $(5/8)^k$ and 55^{-k} in mathematical arithmetic. Convert a tangent t to a sine by $t/\sqrt{1+t^2}$ before comparing with the lab's sine diagnostic.

Each outer step applies the chosen operator, orthogonalizes, solves the projected eigenproblem, and checks the residuals in the original coordinates. The lab retains the QR basis for the next step; rotating it to the Ritz basis gives the same next range in exact arithmetic. It records the largest column residual, largest sine, maximum sorted Ritz-value error and orthogonality defect. Only the synthetic example supplies the exact target basis and exact target values. In applications, identification and separation must come from other information.

22.11.2 Predict before running

Experiments E and I use this paired start to isolate the boundary eigenvalue ratio. Experiment J uses a common random block so the Krylov trial space can reach the entire spectrum. Compare verified residual, orientation and work at the first accepted step. A residual check alone would accept an exact invariant plane at the wrong end of the spectrum. In this reference problem the midpoint between 5 and 8 supplies a cluster identification check.

22.11.3 Transfer to a variational PDE problem

In §6.11, $Kx = \lambda Mx$ came from testing $-u'' = \lambda u$ with finite-element basis functions. Assume $K = K^*$, $M = M^* > 0$ and $M = LL^*$. Set $y = L^*x$ and $C = L^{-1}KL^{-*}$. Then $Cy = \lambda y$ is Hermitian, $\|y\|^2 = x^*Mx$, and

$$Cy - \theta y = L^{-1}(Kx - \theta Mx).$$

For an M-normalized approximate eigenvector, $\|Kx - \theta Mx\|_{M^{-1}}$ is precisely the Euclidean residual norm of the transformed problem. Solve triangular systems to apply the transformations. An M-orthonormal trial basis X gives the small matrix X^*KX and the Ritz residual satisfies $X^*(KX - MXH) = 0$. These are the Galerkin test equations again. A verified algebraic residual certifies the *discrete* eigenproblem. Errors due to the mesh and the choice of trial functions require an additional approximation argument.

Checkpoint. Why can a change of finite-element basis alter the Euclidean norm of a residual without changing the mathematical approximation? The residual represents a functional; its coordinates and the mass Gram matrix transform together. The dual norm accounts for both changes.

22.12 Exercises and worked solutions

Try each problem before reading its solution. Problems 4, 6, 8 and 11 form a short proof-reconstruction route into the seminar.

22.12.1 1. Which data are allowed to move?

For $A = \operatorname{diag}(1, 10^{-6})$, $b = e_1$ and $\hat{x} = (1, 1)^T$, find the minimum absolute normwise perturbations when (i) only b may move, (ii) only A may move. Compare with the relative forward error.

Solution. The residual is $(0, -10^{-6})^T$. The new right side is forced to be $A\hat{x}$, so its distance is 10^{-6}. For a matrix change $E\hat{x} = r$, the lower bound is $\|r\|/\|\hat{x}\| = 10^{-6}/\sqrt{2}$. It is attained by $E = r\hat{x}^*/\|\hat{x}\|^2$. The exact $x = e_1$ gives relative forward error 1. Backward error has meaning only after specifying the allowed data and norm.

22.12.2 2. A Rayleigh quotient can conceal an inaccurate vector

For A_0 take $x = (q_2 + 0.01q_3)/\sqrt{1.0001}$. Compute the quotient error relative to 3 in absolute units, the residual norm and the sine to q_2.

Solution. With $t = 0.01$, $\rho = (3+9t^2)/(1+t^2)$, so the absolute error is $6t^2/(1+t^2) \approx 0.000600$. The two residual coordinates in the eigenbasis are $(3-\rho)/\sqrt{1+t^2}$ and $(9-\rho)t/\sqrt{1+t^2}$. Squaring and adding gives $6t/(1+t^2) \approx 0.059994$ for the norm. The sine is $t/\sqrt{1+t^2} \approx 0.0099995$. To call the quotient error relative, one would divide it by 3.

22.12.3 3. A backward error with symmetry

For $A = \operatorname{diag}(1, 3)$, $x = (1, 1)^T/\sqrt{2}$, find an optimal Hermitian perturbation for its Rayleigh quotient.

Solution. $\theta = 2$, $r = (-1, 1)^T/\sqrt{2}$, $\|r\| = 1$. Expanding $-rx^* - xr^*$ gives $\operatorname{diag}(1, -1)$. The corrected matrix is $2I$, so $(2, x)$ is exact. Any admissible E has $\|E\| \geq \|Ex\| = 1$, proving optimality. The perturbation merges two simple eigenvalues; the backward-error condition imposes no constraint on multiplicities.

22.12.4 4. Fixed versus adaptive inverse iteration

In a two-dimensional Hermitian eigenspace mixture take $x = (u_1 + tu_2)/\sqrt{1+t^2}$, real $t \neq 0$, and distinct eigenvalues. Derive exactly the new signed coefficient ratio under RQI.

Solution. $\rho = (\lambda_1 + t^2\lambda_2)/(1+t^2)$. The new ratio equals $t(\lambda_1 - \rho)/(\lambda_2 - \rho) = -t^3$. Thus the tangent cubes even with the sign change. A fixed shift produces $t(\lambda_1 - \sigma)/(\lambda_2 - \sigma)$ instead. Starting at $t = 1$ gives a two-cycle in the line for RQI, illustrating the locality of the convergence statement.

22.12.5 5. A block that cannot recover its target

For A_0 target $\operatorname{span}(q_2, q_3)$ and start with $[q_1, q_3]$. Predict the residuals and the sine of the largest target angle after any number of power steps.

Solution. Multiplication and QR preserve the starting invariant plane. Both Ritz residuals are zero, whereas q_1 is orthogonal to the target plane and the largest sine is one. In target coordinates the overlap matrix has rank one. The invertibility hypothesis in the graph argument fails. This is why stopping on residuals requires an eigenvalue selection or cluster identification rule as well.

22.12.6 6. From an approximate Arnoldi relation to a trustworthy residual

Suppose $AV = VH + hve_m^* + F$, $\|v\| = 1$, and $Hy = \theta y + s$ with $\|y\| = 1$. Bound the lifted residual and explain the role of $\|Vy\|$.

Solution. Substitution gives $(A-\theta I)Vy = hve_m^*y + Vs + Fy$. By the reverse triangle inequality its norm differs from $|h|\,|e_m^*y|$ by at most $\|V\|\|s\| + \|F\|$. To report the residual for a unit vector divide by $\|Vy\|$. If $\|V^*V - I\| \leq \epsilon < 1$ then $\sqrt{1-\epsilon} \leq \|Vy\| \leq \sqrt{1+\epsilon}$. A shortcut reporting zero can coexist with a nonzero direct residual when the recurrence has a defect.

22.12.7 7. Designing a degree

For an unwanted interval $[1, 5]$, a nearest target at 8, and a desired one-step angle contraction 10^{-3}, find the smallest degree certified by the interval bound.

Solution. $z_* = (8-3)/2 = 2.5$. Successive values $T_j(2.5)$ are $1, 2.5, 11.5, 55, 263.5, 1262.5$ by the recurrence. Since the requirement is $T_d(2.5) \geq 1000$, the first sufficient degree is five. Degrees lower than five might meet the goal on a particular starting block if the most weakly amplified target component carries no error, but the interval certificate is uniform.

22.12.8 8. Endpoint accumulation

Consider the scalar recurrence with $B = 1$, exact initial value, and the same additive defect f at every computed step starting with step one. Find the exact error after d steps. Compare it with a single defect only at step one.

Solution. $U_k(1) = k + 1$ by induction. Summation gives $f \sum_{j=1}^{d} (d - j + 1) = fd(d + 1)/2$. A single first defect gives df. The polynomial endpoint satisfies $|T_d(1)| = 1$, yet perturbations in recurrence evaluation can accumulate quadratically with degree. Using a uniform bound $\|F_j\| \le |f|$ attains equality in this example.

22.12.9 9. Verification with a mass matrix

Let $M = \mathrm{diag}(4, 1)$, $K = \mathrm{diag}(4, 3)$, $x = (1/2, 1)^T/\sqrt{2}$. Determine the generalized Rayleigh quotient and the dual residual norm.

Solution. $x^*Mx = 1$, $x^*Kx = 2$. Thus $r = Kx - 2Mx = (-2, 1)^T/\sqrt{2}$. The Euclidean norm is $\sqrt{5/2}$, whereas $r^*M^{-1}r = (1 + 1)/2 = 1$. Under $L = \mathrm{diag}(2, 1)$ the transformed matrix is $\mathrm{diag}(1, 3)$, $y = (1, 1)^T/\sqrt{2}$, and its Euclidean residual norm is exactly one.

22.12.10 10. Distinguishing a measured floor from a theorem

Suppose $t_{k+1} \le 0.1t_k + 10^{-7}$. Can one prove that t_k can never fall below 10^{-7}? For $t_0 = 1$ bound t_8.

Solution. The zero sequence after step zero obeys the inequality, so it supplies no lower bound. Repeated substitution gives $t_8 \le 10^{-8} + 10^{-7}(1 - 10^{-8})/0.9 \approx 1.21\,10^{-7}$. A measured history that levels off there is consistent with this model. A smaller measured value is equally consistent.

22.12.11 11. A filter written in terms of its residual

With $p(z) = 4z^3 - 3z$, derive a correction formula involving $r = Ax - \theta x$. Which products would you evaluate accurately if the requested tolerance is much smaller than the filter's working unit roundoff?

Solution. Polynomial division gives $s_\theta(z) = 4(z^2 + \theta z + \theta^2) - 3$, hence $p(A)x = p(\theta)x + [4(A^2 + \theta A + \theta^2 I) - 3I]r$. Compute the initial residual using the original operator in verification precision and accumulate the correction accurately. Products on the correction may tolerate lower precision since their absolute errors can scale with $\|r\|$. This conclusion requires bounds on the correction operator, conversion errors and dynamic range. A small $\|r\|$ does not authorize treating its *relative* error as small automatically.

22.12.12 12. Design an equal-work comparison

For the case study, compare one degree-three application with three ordinary power steps from the same start. Which conclusion follows before any timing experiment?

Solution. Both use three matrix–block products. The respective largest tangents are $1/55 \approx 0.01818$ and $(5/8)^3 = 125/512 \approx 0.24414$. The filter gives the smaller angle for this spectrum, block and work metric. Three ordinary outer steps may perform more QR and extraction operations. Timing must specify their frequency, data movement, precision, matrix layout and verification tolerance. Neither theoretical ratio determines the faster implementation on a given machine.

22.12.13 13. Reconstruct the four proof steps behind a filtered solver

Without using the final estimates, (a) deduce inclusion from a spectral expansion, (b) compare the ordered spectra of A and $A + E$ using test spaces, (c) turn a block in eigen-coordinates into a graph after filtering, and (d) isolate a single recurrence defect inserted at step j. Which step fails if a computed output is substituted directly for the exact polynomial?

Solution. (a) For a unit vector, the squared residual is a weighted average of $|\lambda_i - \theta|^2$ with weights $|u_i^* x|^2$. Its minimum term is at most its average. (b) Every unit-vector quotient changes by an amount between $-\|E\|$ and $\|E\|$. Taking the maximum on each i-plane preserves the bounds; taking the minimum over planes proves both sides for the ith smallest eigenvalue. (c) With wanted and unwanted blocks C, D, the filtered basis has blocks $P_1 C, P_2 D$. Multiplication on the right by $(P_1 C)^{-1}$ gives graph $P_2 D C^{-1} P_1^{-1}$, so submultiplicativity bounds its norm by $(\beta/\alpha)\|DC^{-1}\|$. Invertibility of both wanted factors is essential. (d) Starting from zero error, one defect contributes F_j at j, $2BF_j$ at $j+1$, then $(4B^2 - I)F_j$. Induction produces $U_{d-j}(B)F_j$. Linearity superposes defects. In (c) computed blocks contain additive errors and the exact factorization no longer holds. A lower bound on the smallest singular value of the perturbed wanted block is now necessary.

22.13 Computational laboratory — From one rounded sum to a verified cluster

Run the cells in order. Each experiment poses a question with an analytical reference. The examples use real symmetric matrices so transpose suffices. Matrix products use the installed NumPy backend. All reference residuals and all QR and Ritz operations use binary64 unless explicitly indicated. Reported residuals are absolute, for unit vectors.

```
import numpy as np
import scipy.linalg as la
import matplotlib.pyplot as plt
from numpy.polynomial.chebyshev import chebval

def orth(block):
    if not np.isfinite(block).all():
        raise FloatingPointError("nonfinite basis input")
    q, r = la.qr(block, mode="economic")
    singular = la.svdvals(r)
    if singular[-1] <= max(block.shape) * np.finfo(block.dtype).eps * singular[0]:
        raise ValueError("block has lost numerical rank")
    return q

def norm2(x):
    return la.norm(x, 2)

def symmetric(a):
    return (a + a.T) * 0.5

def response(z, degree):
    return chebval(z, [0] * degree + [1])

def ritz(a, basis, count):
    compressed = symmetric(basis.T @ (a @ basis))
    values, weights = la.eigh(compressed)
    vectors = basis @ weights[:, -count:]
    values = values[-count:]
    residuals = la.norm(a @ vectors - vectors * values, axis=0)
    return values, vectors, residuals

def plot_history(ax, data, label):
    ax.semilogy(np.arange(len(data)), np.maximum(data, 1e-18), label=label)
    ax.set_xlabel("Iteration")
```

22.13.1 A. Which digits did cancellation expose?

The sum $1 + 2^{-25}$ lies closer to one than to its binary32 successor. The small singular value in the second example is an independent source of lost forward accuracy. We measure both using data whose mathematical answers are known.

```
one = np.float32(1)
tiny = np.float32(2.0**-25)
print("Rounded increment, exact increment:", (one + tiny) - one, tiny)
small = np.diag([1., 1e-8])
b = np.array([1., 0.])
xhat = np.ones(2)
print("Relative residual, relative forward error, condition:",
      norm2(b - small @ xhat) / norm2(b), norm2(xhat - b), np.linalg.cond(small))
```

```
Rounded increment, exact increment: 0.0 2.9802322e-08
Relative residual, relative forward error, condition: 1e-08 1.0 100000000.0
```

The first result is zero. In the second experiment the residual 10^{-8} and condition number 10^{8} attain the bound in Session 2. The displayed candidate was chosen deliberately. This tests the sharpness of an estimate and says nothing about the stability of a particular linear solver.

22.13.2 B. Can we move a Hermitian eigenvalue farther than the perturbation?

Rotate a prescribed spectrum and apply a positive rank-one perturbation. The triangle inequality behind Weyl permits the full perturbation norm. Equality would require favorable alignment with an extremal direction.

```
rng = np.random.default_rng(20261118)
n, p = 32, 3
i, j = np.indices((n, n))
U = np.sqrt(2/n) * np.cos(np.pi * (i + .5) * j / n)
U[:, 0] = 1 / np.sqrt(n)
spectrum = np.r_[np.linspace(1, 5, n-p), [8., 9., 10.]]
A = symmetric((U * spectrum) @ U.T)
target, outside = U[:, -p:], U[:, :-p]
assert norm2(U.T @ U - np.eye(n)) < 1e-12
w = rng.normal(size=n)
w /= norm2(w)
E = 1e-4 * np.outer(w, w)
newvals, newvec = la.eigh(A + E)
change = np.max(abs(newvals - la.eigvalsh(A)))
print("Max shift, perturbation norm:", change, norm2(E))
print("Max residual on original and perturbed matrices:",
      la.norm(A @ newvec - newvec * newvals, axis=0).max(),
      la.norm((A+E) @ newvec - newvec * newvals, axis=0).max())
assert change <= norm2(E) + 1e-12
```

```
Max shift, perturbation norm: 1.999035056154952e-05 0.00010000000000000003
Max residual on original and perturbed matrices: 4.4711375986704965e-05 7.625702338003613e-15
```

The two residuals test different equations. On the original operator the exact residual of a perturbed eigenvector equals $-Ev$. On the perturbed operator only eigensolve and verification errors remain. The comparison with the eigenvalues of stored A isolates the applied perturbation from the initial cosine-basis construction.

22.13.3 C. Which changes more, an eigenvector or its cluster?

Apply the same off-diagonal perturbation to gaps decreasing by eight orders of magnitude. The exact angle of the lowest eigenvector to the first coordinate is half an arctangent. The two-dimensional target plane is invariant throughout.

```
eps = 1e-5
print("gap       measured sine   exact sine     residual/gap*   plane sine")
for gap in [1., 1e-2, 1e-4, 1e-6, 1e-8]:
    base = np.diag([1., 1.+gap, 5.])
    changed = base.copy()
    changed[0, 1] = changed[1, 0] = eps
    vals, vecs = la.eigh(changed)
    v = vecs[:, 0]
    delta = 1 + gap - vals[0]
    posterior = norm2(base @ v - vals[0]*v) / delta
    print(f"{gap:.0e}    {abs(v[1]):.4e}     "
          f"{np.sin(.5*np.arctan2(2*eps,gap)):.4e}    "
          f"{posterior:.4e}      {norm2(vecs[2, :2]):.1e}")
```

```
gap       measured sine   exact sine     residual/gap*   plane sine
1e+00    1.0000e-05     1.0000e-05    1.0000e-05      0.0e+00
1e-02    1.0000e-03     1.0000e-03    1.0000e-03      0.0e+00
1e-04    9.8538e-02     9.8538e-02    9.9020e-02      0.0e+00
1e-06    6.8923e-01     6.8923e-01    9.5125e-01      0.0e+00
1e-08    7.0693e-01     7.0693e-01    9.9950e-01      0.0e+00
```

Here "gap*" is the distance from the perturbed eigenvalue to the original unwanted spectrum. The angle approaches $\pi/4$ as the internal gap closes. The zero plane error results from the location of the perturbation. A perturbation coupling the plane to the third coordinate would instead test its external gap.

22.13.4 D. What changes when the shift follows the vector?

Target the eigenvalue 9 in the diagonal coordinates of A_0. A two-component start makes the predicted power and fixed-shift tangent factors 1/3 and 1/11 respectively (shift 8.5). RQI cubes the tangent until the accuracy of the quotient or the shifted solve intervenes.

```
diag = np.diag([1., 3., 9.])
def vector_run(method):
    x = np.array([0., .5, 1.])
    x /= norm2(x)
    fixed = la.lu_factor(diag - 8.5*np.eye(3))
    rows = []
    for k in range(40):
        value = x @ diag @ x
        residual = norm2(diag @ x - value*x)
        rows.append([abs(value-9), residual, norm2(x[:2])])
        if residual < 1e-13:
            break
        if method == "power":
            y = diag @ x
        elif method == "fixed":
            y = la.lu_solve(fixed, x)
        else:
            # At a rounded eigenvalue use an adjacent representable shift.
            shift = np.nextafter(value, np.inf) if value in diag.diagonal() else value
            y = np.linalg.solve(diag - shift*np.eye(3), x)
        if not np.isfinite(y).all() or norm2(y) == 0:
            raise FloatingPointError("invalid vector update")
        x = y / norm2(y)
    return np.array(rows)

fig, axes = plt.subplots(1, 3, figsize=(9, 3), constrained_layout=True)
for method in ["power", "fixed", "RQI"]:
    rows = vector_run(method)
    for col, ax in enumerate(axes):
        plot_history(ax, rows[:, col], method)
for ax, title in zip(axes, ["Value error", "Residual norm", "Sine of angle"]):
    ax.set_title(title)
axes[0].legend()
plt.show()
```

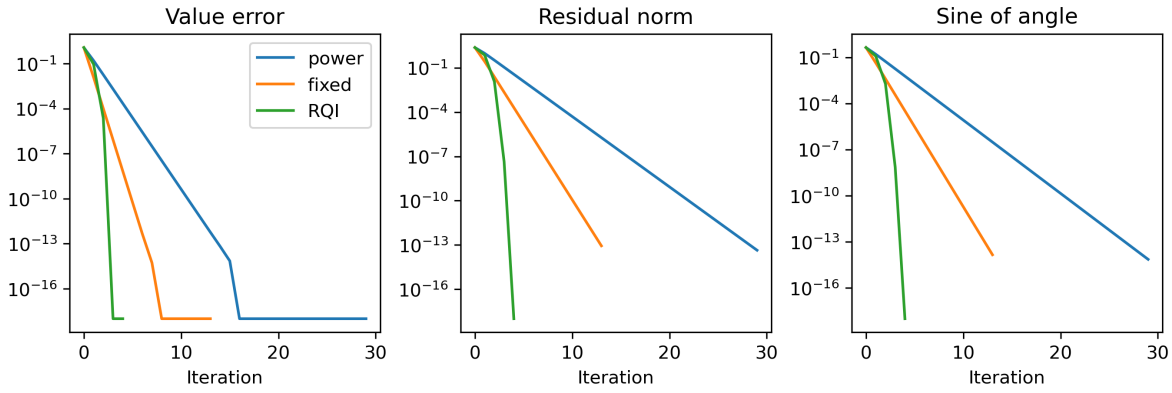


Figure 22.1: Error in the Ritz value, original residual and angle to the target. Values below the plotting threshold are drawn at the threshold.

A rounded value can appear exact while the residual still decreases. Read the residual and angle panels together. RQI has fewer steps here, but each new shift normally costs another factorization. A diagonal example cannot measure that cost on a sparse matrix.

22.13.5 E. Does block iteration attain the graph bound?

Use the paired start from Session 10. The largest principal angle should be controlled by the $5, 8$ pair even if QR rotates the individual columns. Compare measured sines with the prediction for tangents converted to sines.

```
paired = (target + outside[:, [-1, 0, 1]]) / np.sqrt(2)
def metrics(basis):
    values, vectors, residuals = ritz(A, basis, p)
    return [max(residuals), norm2(outside.T @ vectors),
            norm2(vectors.T @ vectors - np.eye(p)),
            np.max(abs(values - spectrum[-p:])), values.min()]

Q = paired.copy()
print("step    measured sine   predicted sine   orthogonality")
for step in range(13):
    measured = metrics(Q)
    t = (5/8)**step
    if step % 3 == 0:
        print(f"{step:3}     {measured[1]:.6e}    {t/np.hypot(1,t):.6e}    {measured[2]:.1e}"
    Q = orth(A @ Q)
```

```
step    measured sine   predicted sine   orthogonality
  0     7.071068e-01    7.071068e-01    1.6e-15
  3     2.371746e-01    2.371746e-01    4.4e-16
  6     5.949905e-02    5.949905e-02    8.9e-16
  9     1.455037e-02    1.455037e-02    3.0e-16
 12     3.552691e-03    3.552691e-03    6.8e-16
```

Agreement tests a worst-case construction where the estimate is attained. For a generic start the most persistent graph entry can have a small initial coefficient. A slower or faster transient than this example would then be consistent with the same upper bound.

22.13.6 F. When can the final Krylov coefficient be trusted?

Build Arnoldi using two projection passes. Switch to the three-term recurrence for a Hermitian operator and compare orthogonality with and without a second projection against the entire basis. The rank-one tail is retained explicitly, so the code handles an invariant space without dividing by a zero subdiagonal.

```
def krylov(a, start, m, lanczos=False, repair=False):
    nloc = len(start)
    V = np.zeros((nloc, m+1))
    H = np.zeros((m+1, m))
    V[:, 0] = start / norm2(start)
    for j in range(m):
        basis = V[:, :j+1]
        w = a @ V[:, j]
        if lanczos:
            if j:
                H[j-1, j] = H[j, j-1]
                w -= H[j-1, j] * V[:, j-1]
            H[j, j] = V[:, j] @ w
            w -= H[j, j] * V[:, j]
            if repair:
                # Keep the correction coefficients in the recorded relation.
                correction = basis.T @ w
                w -= basis @ correction
                H[:j+1, j] += correction
        else:
            for _ in range(2):
                h = basis.T @ w
                w -= basis @ h
```

```
                H[:j+1, j] += h
        beta = norm2(w)
        H[j+1, j] = beta
        if beta <= 1e-14 or j+1 == m:
            return basis, H[:j+1, :j+1], w
        V[:, j+1] = w/beta

vstart = rng.normal(size=n)
general = A + .25*np.triu(np.ones_like(A), 1)
print("method            orth defect    relation defect   Ritz check error")
for name, mat, short, repair in [
        ("Arnoldi", general, False, False),
        ("Lanczos", A, True, False),
        ("Lanczos+repair", A, True, True)]:
    V, H, tail = krylov(mat, vstart, 28, short, repair)
    defect = mat @ V - V @ H
    defect[:, -1] -= tail
    vals, Y = la.eig(H)
    lifted = V @ Y
    direct = la.norm(mat @ lifted - lifted*vals, axis=0)
    predicted = norm2(tail)*abs(Y[-1])
    print(f"{name:16}  {norm2(V.T@V-np.eye(V.shape[1])):.2e}       "
          f"{norm2(defect):.2e}         {max(abs(direct-predicted)):.2e}")
```

```
method            orth defect    relation defect   Ritz check error
Arnoldi           4.44e-16       2.02e-15          9.36e-15
Lanczos           1.00e+00       2.64e-15          2.80e-14
Lanczos+repair    5.29e-16       1.98e-15          2.75e-14
```

A small recurrence defect can coexist with a large orthogonality defect. The last column tests the **unnormalized** lifted residual, as in Exercise 6. Trusting Ritz values as eigenvalues of an orthogonal compression requires the geometric test as well. The repaired variant stores tiny off-tridiagonal correction coefficients. A run showing loss of orthogonality demonstrates a mechanism allowing ghost values, without proving a duplicate occurs in every spectrum.

22.13.7 G. What if the interval misses an unwanted mode?

Choose an upper target at 8 and vary the assumed enclosing interval. Degree enlarges the exterior response on both sides. Inspect the entire unwanted spectrum before accepting a filter design.

```
fig, axes = plt.subplots(1, 2, figsize=(8, 3), constrained_layout=True)
grid = np.linspace(1, 10, 501)
for ax, (lo, hi) in zip(axes, [(1, 5), (3.5, 5)]):
    def mapped(x):
        return (2*x-lo-hi)/(hi-lo)
    for d in [1, 3, 5]:
        scale = abs(response(mapped(8.), d))
        curve = abs(response(mapped(grid), d)) / scale
        ax.semilogy(grid, np.maximum(curve, 1e-5), label=f"d={d}")
        factor = max(abs(response(mapped(spectrum[:-p]), d))) / scale
        print((lo, hi), d, "actual ratio", round(factor, 5))
    ax.axvspan(1, 5, alpha=.12)
    ax.axhline(1, color="gray", lw=.7)
    ax.set(title=f"Design [{lo}, {hi}]", xlabel="Eigenvalue", ylabel="Normalized magnitude")
axes[0].legend()
plt.show()
```

```
(1, 5) 1 actual ratio 0.4
(1, 5) 3 actual ratio 0.01818
(1, 5) 5 actual ratio 0.00079
(3.5, 5) 1 actual ratio 0.86667
(3.5, 5) 3 actual ratio 0.64429
(3.5, 5) 5 actual ratio 0.48062
```

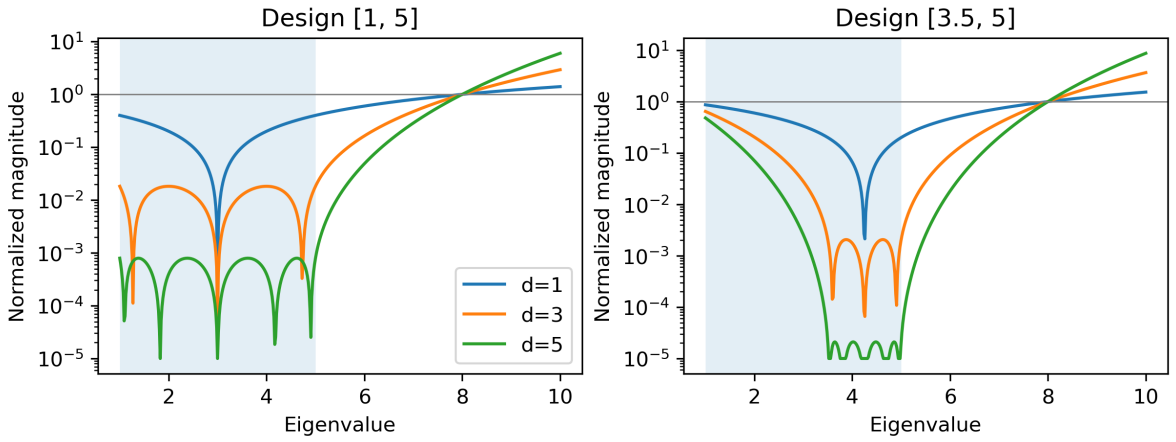


Figure 22.2: Absolute filter response normalized at the weakest target. Shading marks the actual unwanted spectrum's interval.

The erroneous interval amplifies low unwanted values. In this example the weakest target

still wins, so the consequence is slower convergence. For a genuine reversal, change the left endpoint to 4.5 while keeping the right endpoint 5. The distances from the midpoint 4.75 are 3.75 for the unwanted 1 and 3.25 for the wanted 8, so the unwanted multiplier exceeds the weakest wanted multiplier at every positive degree.

22.13.8 H. Where does a local filter defect go?

Measure the actual defects of binary32 recurrence steps by replaying their operands in binary64. Propagate each defect in the known cosine eigenbasis. Compare the resulting bound with the error in the unwanted projection and the full relative output error. The reference is spectral evaluation in binary64. Degree 12 keeps the unscaled values in range.

```
B = symmetric((A - 3*np.eye(n))/2)
z = (spectrum - 3)/2
def second(z, k):
    previous, current = np.zeros_like(z), np.ones_like(z)
    for _ in range(k):
        previous, current = current, 2*z*current-previous
    return current

stored = B.astype(np.float32)
x0 = paired.astype(np.float32)
Y = [x0, stored @ x0]
local = [Y[1].astype(float) - B @ x0.astype(float)]
print("degree  relative error   unwanted error   projected envelope")
for d in range(1, 13):
    if d >= 2:
        y = 2*(stored @ Y[-1])-Y[-2]
        local.append(y.astype(float) - (2*(B @ Y[-1].astype(float))-Y[-2]))
        Y.append(y)
    weights = response(z, d)
    reference = U @ (weights[:, None]*(U.T @ paired))
    err = Y[d].astype(float) - reference
    initial = outside.T @ (x0.astype(float)-paired)
    envelope = norm2(weights[:-p, None]*initial)
    for j, defect in enumerate(local, start=1):
        envelope += norm2(second(z[:-p], d-j)[:, None]*(outside.T @ defect))
    measured = norm2(outside.T @ err)
    print(f"{d:4}    {norm2(err)/norm2(reference):.3e}       {measured:.3e}       {envelope:
    # The ideal cosine basis and the stored B differ by binary64 construction error.
    assert measured <= envelope + 5e-13*norm2(reference)
```

```
degree  relative error   unwanted error   projected envelope
   1    1.238e-07       3.055e-07       3.276e-07
   2    1.312e-07       2.061e-06       2.384e-06
   3    1.193e-07       1.144e-05       1.366e-05
   4    9.896e-08       5.785e-05       7.546e-05
   5    1.288e-07       6.286e-04       7.140e-04
   6    1.181e-07       3.772e-03       4.198e-03
   7    1.170e-07       2.145e-02       2.576e-02
   8    1.506e-07       1.720e-01       1.908e-01
   9    1.693e-07       1.136e+00       1.352e+00
  10    1.973e-07       1.010e+01       1.131e+01
  11    1.991e-07       5.213e+01       6.343e+01
  12    1.876e-07       2.455e+02       3.016e+02
```

The first line already includes conversion of the starting block. At higher degree, the stage operands have grown and even late local defects can be large in absolute units. The sum of projected norms discards cancellation. Its comparison is a diagnostic up to reference rounding, whereas the exact-arithmetic identity in Session 8 is a theorem. Relative full-block error and absolute unwanted leakage answer different questions.

22.13.9 I. Which precision limits the final residual?

Hold QR, Rayleigh–Ritz and residual verification fixed at binary64. Change only the filter. The third configuration rounds the mapped matrix and each stage output to binary16 and promotes them for binary64 products and updates. It simulates storage quantization and deliberately avoids any claim about native half-precision accumulation. Degree three fits the exponent range here.

```
def filter_block(mapped_matrix, X, degree, mode):
    if mode == "FP64":
        dtype = np.float64
    elif mode == "FP32":
        dtype = np.float32
    elif mode == "FP16 storage":
        dtype = np.float16
    else:
        raise ValueError(mode)
    def store_stage(value):
        converted = value.astype(dtype)
        if not np.isfinite(converted).all():
            raise FloatingPointError("filter exceeded storage range")
```

```
        return converted.astype(float) if dtype == np.float16 else converted
    matrix, prev = store_stage(mapped_matrix), store_stage(X)
    if degree == 0:
        return prev.astype(float)
    curr = store_stage(matrix @ prev)
    for _ in range(2, degree+1):
        prev, curr = curr, store_stage(2*(matrix @ curr)-prev)
    return curr.astype(float)

# The reference spectral norm is ten in the mathematical construction.
def accepted(row):
    return row[0] <= 1e-10 and row[2] <= 1e-12 and row[4] > 6.5

histories = {}
for mode in ["FP64", "FP32", "FP16 storage"]:
    Q = paired.copy()
    history = [metrics(Q)]
    for _ in range(14):
        Q = orth(filter_block(B, Q, 3, mode))
        history.append(metrics(Q))
    histories[mode] = np.array(history)
    first = next((k for k, row in enumerate(history) if accepted(row)), None)
    print(mode, "first accepted:", first, "last residual, sine, orthogonality:",
          np.array(history[-1][:3]))
```

```
FP64 first accepted: 7 last residual, sine, orthogonality: [4.15430599e-15 9.39035302e-15 9.4
FP32 first accepted: None last residual, sine, orthogonality: [8.01954321e-07 1.25260796e-07
FP16 storage first accepted: None last residual, sine, orthogonality: [1.25977060e-03 2.76888
```

```
fig, axes = plt.subplots(1, 3, figsize=(9, 3), constrained_layout=True)
for label, history in histories.items():
    for j, ax in enumerate(axes):
        plot_history(ax, history[:, j], label)
t = 55.**(-np.arange(15))
axes[1].semilogy(t/np.hypot(1,t), "k--", label="exact")
for ax, title in zip(axes, ["Max residual", "Largest principal sine", "Orthogonality defect"]
    ax.set_title(title)
axes[0].legend(fontsize=8)
plt.show()
```

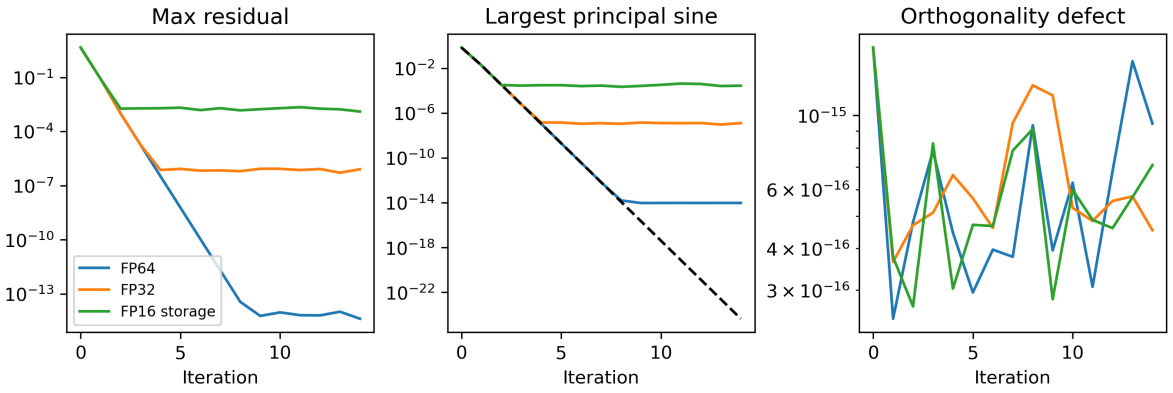


Figure 22.3: A common binary64 verifier applied to three filter configurations. The dashed line is the exact paired-start sine.

In Figure 22.3 the vertical separation between the residual curves can be large even when the orthogonality curves overlap. QR can restore orthogonality within the range it receives. It cannot undo rotation of that range by a quantized operator. Inspect several late points before calling a plateau. A repeatable nonzero residual with higher-precision verification is evidence about this configuration, not a universal lower bound for a format.

22.13.10 J. Which method reaches the same verified cluster with the least operator work?

Use the **same random three-column start** for all methods and the same cluster threshold and stopping predicate from I. Block Krylov extends the orthogonal trial space with the latest block action. The final extension may contain fewer than three independent vectors, so an SVD selects the surviving directions. At every extraction we charge one operator action per basis column for projection and three fresh actions for direct verification. This intentionally transparent implementation recomputes the compression. A production implementation can cache earlier products.

```
start = orth(np.random.default_rng(41).normal(size=(n,p)))
print("method       steps   iteration AVs   assessment AVs   residual   sine      orth      s
for method in ["ordinary", "Chebyshev", "block Krylov"]:
    Q, front = start.copy(), start.copy()
    iteration_work = assessment_work = 0
    for step in range(101):
        row = metrics(Q)
        assessment_work += Q.shape[1] + p
        if accepted(row) or step == 100:
```

```
            break
        if method == "block Krylov":
            W = A @ front
            iteration_work += front.shape[1]
            for _ in range(2):
                W -= Q @ (Q.T @ W)
            uvec, sv, _ = la.svd(W, full_matrices=False)
            keep = sv > 100*np.finfo(float).eps*10
            if not keep.any():
                break
            front = uvec[:, keep]
            Q = np.column_stack((Q, front))
        elif method == "ordinary":
            Q = orth(A @ Q)
            iteration_work += p
        else:
            Q = orth(filter_block(B, Q, 3, "FP64"))
            iteration_work += 3*p
    print(f"{method:12} {step:4}       {iteration_work:4}           {assessment_work:4}
          f"{row[0]:.1e}  {row[1]:.1e}  {row[2]:.1e}  "
          f"{'met' if accepted(row) else 'unmet'}")
```

```
method       steps   iteration AVs   assessment AVs   residual   sine      orth      status
ordinary       51         153             312        7.6e-11  2.5e-11  4.8e-16  met
Chebyshev       7          63              48        3.8e-12  8.8e-13  6.8e-16  met
block Krylov   10          30             230        1.2e-14  9.4e-15  1.4e-15  met
```

"AVs" counts single-vector equivalents. One degree-three application to a width-three block costs nine. The scaled dense B is formed once outside the iteration. Cost columns exclude one-time setup, basis operations, small dense solves and the reference-only angle calculation. The Krylov method can reach the full space on this small example, at which point a projected eigensolve is essentially a dense solve. A sparse production study would limit basis size, count restarts, and measure communication. The table can establish accuracy and work for these particular implementations. It cannot establish a hardware speedup.

22.14 Synthesis — What makes the computation reliable?

Module A's direct sum provides the coordinates $\binom{C}{D}$ in which algorithmic convergence becomes a graph contraction. Module B explains both why left eigenvectors measure perturbations and why Ritz tests annihilate the trial-space residual. Module C makes these tests computable

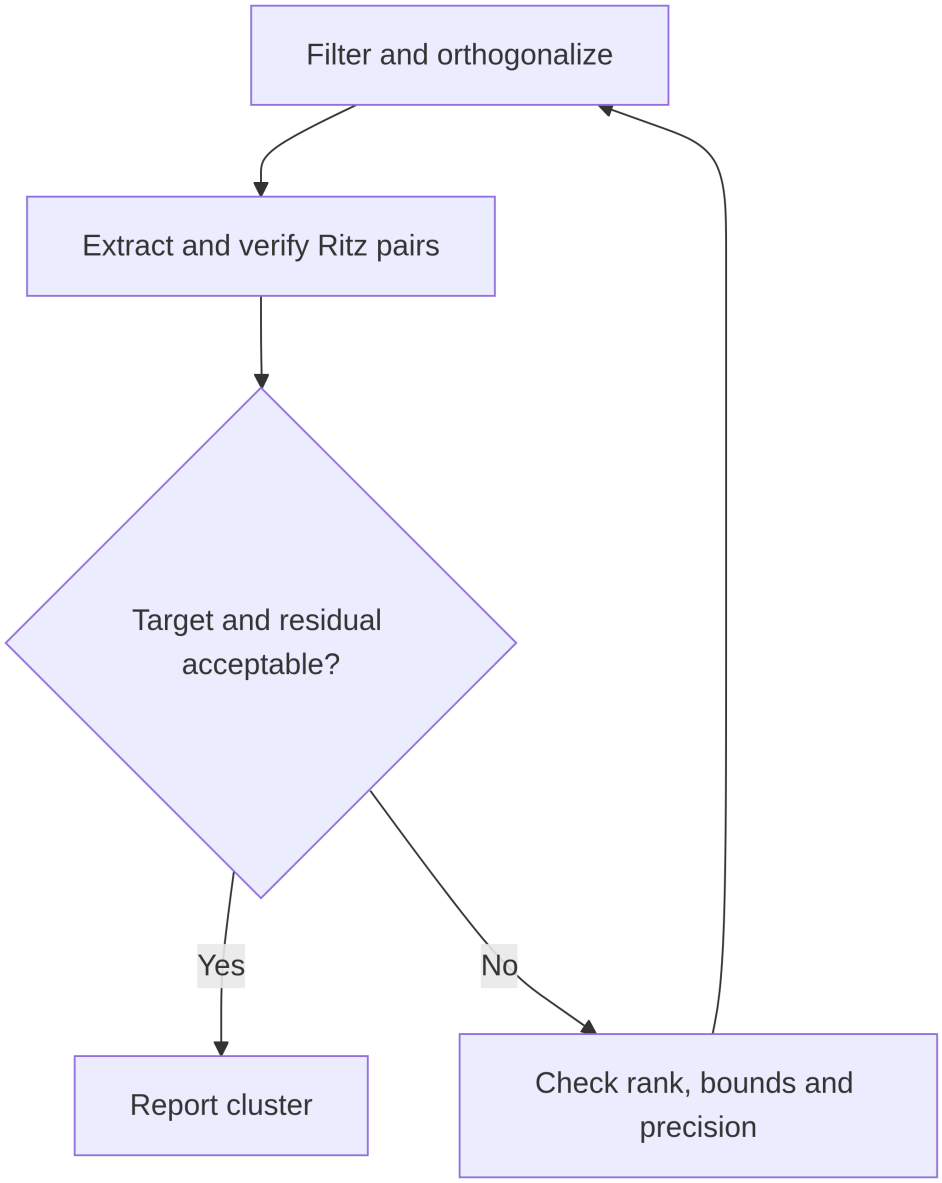
through inner products and makes "close" precise through orthogonal projection. Module D diagonalizes a polynomial of a Hermitian operator. Module E uses all four results twice: first to choose the algorithm, then to decide what its output warrants. The hypotheses in the middle column below are as important as the formulas.

Quantity or result	Hypotheses	Interpretation
$\gamma_n \lvert A\rvert\lvert x\rvert$	Sequential real matvec, $nu < 1$, normal-range model	Componentwise absolute error envelope
$\min \lVert E\rVert = \lVert r\rVert$	Unit x, unrestricted matrix perturbation	Price of making the approximate pair exact
$\operatorname{dist}(\theta, \sigma(A)) \le \lVert r\rVert$	Hermitian or normal A, unit vector	Existence of a nearby eigenvalue
$\lVert (I-P)x\rVert \le \lVert r\rVert/\delta$	Hermitian A, unwanted spectrum separated from θ	Target-space angle certificate
$\tan t_k \le \rho^k \tan t_0$	Self-adjoint filter, full overlap, $\rho = \beta/\alpha < 1$	Block convergence across a cluster boundary
$X^*(AX - XH) = 0$	$X^*X = I$, $H = X^*AX$	Galerkin orthogonality
$\lVert r_{\text{Ritz}}\rVert = h_{m+1,m}\lvert e_m^* y\rvert$	Exact Arnoldi relation, normalized projected eigenpair	Cheap residual estimate
$\sum_{j=1}^{d} U_{d-j}(B)F_j$	Unscaled recurrence, fixed B, exact initialization	Total error from local filter defects
$\lVert L^{-1}(Kx - \theta Mx)\rVert$	$M = LL^* > 0$, $x^*Mx = 1$	Residual in the correct variational geometry

22.14.1 Choosing an algorithm from the spectral information available

Method	What information and overlap does it need?	Main work and likely limitation
Power	Unique largest modulus with a nonzero initial coefficient	One product per step. Nearly equal moduli make the ratio close to one.
Fixed inverse	Unique closest eigenvalue to a shift outside the spectrum	Reusable factorization and one solve per step. Fill-in can dominate sparse storage.
RQI	Start near a simple Hermitian eigenpair, with solvable shifted systems	Local cubic angle convergence. A new factorization may be needed each step.

Method	What information and overlap does it need?	Main work and likely limitation
Block power	Full-rank overlap with a dominant cluster and a gap in magnitude across its boundary	One block product and QR. A weakly represented wanted direction can disappear.
Arnoldi	Krylov start sees the target. A small residual and identification validate extraction	Increasing basis, projection and reductions. Nonnormality weakens residual-based forward guarantees.
Lanczos	Hermitian action, starting overlap and a suitably informative Krylov space	Three-term generation, possible reorthogonalization and restart. Repeated Ritz values need independent checks.
Chebyshev block	Outer spectral bounds, full target overlap and $\beta/\alpha < 1$	Several block products per QR. A missed spectral tail or amplified recurrence defects can dominate.



22.14.2 Reading the measured evidence

In one run of Experiment J, ordinary iteration reached tolerance after 51 steps, Chebyshev after 7, and block Krylov after 10 extensions. The iteration-only counts were 153, 63 and 30 vector products. Including the explicit recompression and direct checks changed the totals to 465, 111 and 260. The ordering of the last two methods therefore depends on what work is charged. Caching the basis images would especially change the Krylov count. The acceptance threshold,

target and directly verified accuracy must travel with any reported speed comparison.

Experiment F illustrates a different accounting mistake. The unrepaired Lanczos relation had defect about 2.3×10^{-15} while the basis orthogonality defect reached order one. The recurrence residual formula was still numerically accurate for the *unnormalized* lifted vectors. It could not certify independent directions. In I the FP32 filter showed the opposite combination of diagnostics: orthogonality near 4×10^{-16} and an original residual near 3.8×10^{-7}. These examples locate two distinct ways a plausible-looking convergence report can fail.

> **Verification note.** A tolerance must name a quantity and a scale. For example, for this matrix an absolute residual of 10^{-10} is a relative matrix backward error of about 10^{-11} for a unit Ritz vector. To request an angle, also specify which cluster and how its separation is known. To request a wall time, specify the operator, storage layout, precision stages, machine and verification rule.

22.14.3 Integrated problem — Write a numerical acceptance report

You seek the three upper modes in the experiment. The other eigenvalues lie in $[1, 5]$, your trial block is orthonormal, every Ritz value is at least 7.9, and each of its three residuals is at most 10^{-7}. Can you certify a largest principal sine below 10^{-7}? Would this certify each direction to the same tolerance? Predict how many ideal degree-three steps suffice for tangent 10^{-10} from the paired start. Then use Experiments I and J to decide whether FP32 filtering and binary64 verification alone achieved the laboratory acceptance criterion.

Worked solution. The excluded spectrum is at distance at least 2.9 from every Ritz value. In the Sylvester argument the Frobenius residual is at most $\sqrt{3}\,10^{-7}$. Hence the largest principal sine is at most $\sqrt{3}\,10^{-7}/2.9 < 6.0 \times 10^{-8}$. This controls the plane. A vector certificate also requires identifying its particular eigenvalue and separating it from the *other two wanted* eigenvalues. From $55^{-k} \le 10^{-10}$, take $k = \lceil 10\log(10)/\log(55)\rceil = 6$. This is a prediction for the angle; acceptance in J also tests the residual. The FP32 run in I retained good orthogonality but missed the residual criterion. Increasing verification precision diagnoses the computed subspace. Changing the filter or improving the iterate is required to change its true residual.

22.14.4 Mastery checklist

- ☐ Given a computed eigenpair, can you construct the optimal perturbation and specify exactly which structure it preserves?
- ☐ Given a cluster, can you explain why an orthonormal basis may change substantially even when the projector barely moves?
- ☐ Starting from a spectral decomposition, can you derive the tangent contraction for power, shifted inverse and filtered block iteration with their respective hypotheses?

- ☐ Can you derive an Arnoldi residual using only the last subdiagonal, then identify the two numerical defects that compromise it?
- ☐ Given spectral bounds and a requested contraction, can you map the interval, choose a degree and diagnose a missed unwanted eigenvalue?
- ☐ Can you propagate a single early recurrence defect, then explain why an absolute filter error need not equal an angle error?
- ☐ Can you design a mixed-precision comparison that separates storage error, arithmetic error, verification error and time to tolerance?
- ☐ Can you derive the summation and block-product bounds and identify the underflow assumption?
- ☐ Can you distinguish problem conditioning, algorithmic stability and forward accuracy using one concrete counterexample?
- ☐ Can you prove Weyl from min–max and explain why left eigenvectors enter a nonnormal sensitivity estimate?
- ☐ Can you justify reorthogonalization, distinguish numerical rank loss from exact breakdown, and explain the role of the external cluster gap?
- ☐ Can you describe input, accumulation, communication and verification precision well enough for someone else to reproduce a run?

22.14.5 Where to go next

For HASC. Read the seminar recurrence with three quantities on paper: the object being multiplied at each step, the norm in which its product error is bounded, and the factor that transfers this error into the unwanted space. Compare an error proportional to the norm of a full iterate with one proportional to a residual correction. Track which steps require access to the higher-precision operator. In an experiment, maintain a separate column for measured time and another for the verified final residual. This lets you judge a performance claim at a stated mathematical accuracy.

For functional analysis. Orthogonal projections, Rayleigh quotients and bounded polynomials of a bounded self-adjoint operator extend to Hilbert space. A compact self-adjoint operator has an eigen-expansion with possible accumulation at zero, so a uniform positive gap cannot be assumed along its whole spectrum. Differential operators can be unbounded and require domains and boundary conditions. A useful next step is to prove spectral convergence of finite-element approximations using compact solution operators, distinguishing discretization error from iteration error.

For structural algebra. The polynomial action $p \mapsto p(A)$ and the cyclic quotient by an annihilating ideal explain exact Krylov termination independently of the coordinate basis. An

inner product, an error norm and a rounding model add information that an algebraic quotient does not encode. Under a badly conditioned nonunitary change of coordinates the abstract operator is isomorphic while measured sensitivity can change substantially.

Foundation established in principle. After reconstructing the proofs and reproducing the experiment, you have the tools to justify selection, contraction, extraction and verification for finite-dimensional Hermitian problems. **Further study** includes sharper clustered-subspace bounds, implicit restarting and locking, interior filter design, communication-avoiding QR, sparse generalized eigensolvers with inexact solves, and precision-dependent convergence proofs for a specific residual-based recurrence. Each needs assumptions beyond the elementary graph and residual estimates proved here.

23 Module F1 — Functional Analysis: Infinite-Dimensional Geometry and Operator Theory

A matrix can be injective and still amplify errors enormously when inverted. Now imagine a sequence of unit input directions whose outputs tend to zero. Each direction remains distinguishable in exact arithmetic, yet no single stability constant controls inversion. In an infinite-dimensional space a *single operator* can exhibit this behavior. Understanding how it happens will take us from linear algebra to inverse problems, and ultimately to the numerical spectrum of a differential equation.

Starting point. Review II, §4.16 connects kernels, quotients and uniqueness. Section 5.11.3 interprets a load as a functional on test functions. Module C supplies the geometry of representing vectors and adjoints. Sections 7.11.5–6 arrive at the stiffness–mass eigenproblem. Here we recover those constructions with precise hypotheses on the underlying function spaces. F1 and F2 are parallel extensions of the core modules. The numerical connections to Module E can be read along the way.

Conventions. Scalars are real or complex. As in Module C, $\langle x, y\rangle$ is linear in x. Functionals in X^* are linear. Write T' for the continuous dual map (Module B's pullback) and T^* for the Hilbert adjoint. Sessions 6–8 use real spaces for the PDE examples.

23.1 Session 1 — What is a limit of vectors?

23.1.1 Distance adds structure to a vector space

A norm assigns x a length $\|x\| \geq 0$, zero exactly at 0, with $\|cx\| = |c|\|x\|$ and $\|x+y\| \leq \|x\|+\|y\|$. Translation now preserves distance $d(x, y) = \|x - y\|$. The open ball $B(x, r)$ consists of points at distance less than r. A set is open if each of its points belongs to a ball contained in the set. These open sets form the **norm topology**. The reverse triangle inequality $|\|x\|-\|y\|| \leq \|x-y\|$ follows by bounding each norm in terms of the other.

A sequence converges to x if eventually every term lies in every prescribed ball about x. It is **Cauchy** if for every $\varepsilon > 0$, all pairs of sufficiently late terms have distance less than ε. Convergence implies Cauchy by inserting the limit into the triangle inequality. Completeness

is the converse for every Cauchy sequence in the space. A complete normed space is a **Banach space**.

The closure $\overline{M}$ consists of limits in the ambient space of sequences from M. Equivalently every ball about such a point meets M (choose radii $1/n$). A closed set contains all its limit points. A dense set has closure equal to the whole space. A map S is continuous at x if for each output tolerance $\varepsilon > 0$ some input radius $\delta > 0$ ensures $d(Sz, Sx) < \varepsilon$ whenever $d(z, x) < \delta$. Equivalently, $x_n \to x$ implies $Sx_n \to Sx$; failure of the radius condition gives a counterexample sequence by choosing $\delta = 1/n$.

23.1.2 Why the choice of norm used to be harmless

Two norms are equivalent if $c\|x\|_a \le \|x\|_b \le C\|x\|_a$ for positive constants independent of x. Both identity maps are continuous and the two notions of Cauchy sequence agree.

Finite-dimensional norm equivalence. Choose a basis and write $x = \sum x_j b_j$. For any norm N, Cauchy–Schwarz in the coordinates gives $N(x) \le (\sum N(b_j)^2)^{1/2} |x|_2$. The reverse triangle inequality now makes N continuous in Euclidean coordinates. On the compact Euclidean unit sphere it attains a strictly positive minimum m. Scaling gives $m|x|_2 \le N(x)$. Comparing each of two norms with $|\cdot|_2$ proves the theorem. Coordinatewise completeness of $\mathbb{R}$ or $\mathbb{C}$ then gives completeness in every such norm. Every finite-dimensional subspace of any normed space is closed: a convergent sequence in it is Cauchy there, has an internal limit, and limits are unique. The compact sphere was the decisive ingredient.

23.1.3 A dictionary of spaces

Space	Elements and norm	Meaning of small error
ℓ^p, $1 \le p < \infty$	Sequences with $\|x\|_p = (\sum_{j \ge 1} \|x_j\|^p)^{1/p} < \infty$	Small aggregate error over all coordinates
ℓ^∞	Bounded sequences, $\|x\|_\infty = \sup_j \|x_j\|$	Every coordinate accurate to the same tolerance
$C[a, b]$	Continuous functions, $\|u\|_\infty = \max \|u\|$	Uniform control, including endpoints
$L^p(a, b)$, $1 \le p < \infty$	Measurable functions modulo equality almost everywhere, $\|u\|_p = (\int_a^b \|u\|^p)^{1/p}$	Integrated error, allowing high narrow peaks
$L^\infty(a, b)$	Essentially bounded measurable functions, essential supremum norm	Uniform bound outside a set of measure zero

Here length extends to a measure of sets. A null set can be covered by intervals with arbitrarily small total length. "Almost everywhere" permits exceptions on a null set. Measurability ensures that level sets can be measured and the Lebesgue integral defined; it agrees with the usual integral for continuous functions. Changing a function at a single point leaves its integral unchanged. Passing to equivalence classes makes the integral expression a genuine norm. For $1 < p < \infty$ the triangle inequality for these norms is Minkowski's inequality. The table lists Banach spaces. Completeness of general L^p uses integration theory (the Riesz–Fischer completeness theorem); the following sequence-space argument gives its essential mechanism without measure theory.

Proof for ℓ^p, $p < \infty$. If $x^{(k)}$ is Cauchy, each coordinate is Cauchy since $|x_j^{(k)} - x_j^{(l)}| \le \|x^{(k)} - x^{(l)}\|_p$. Call its limit x_j. For k, l sufficiently large, the sum of the first m difference powers is at most ε^p. First send $l \to \infty$ in that finite sum, then $m \to \infty$. Thus $\|x - x^{(k)}\|_p \le \varepsilon$, and $x \in \ell^p$ by addition. For the sup norm take the coordinate limit in the bound for each coordinate and then the supremum. The same uniform Cauchy argument works in $C[a, b]$ because uniform limits preserve continuity: approximate the limit uniformly to $\varepsilon/3$ by one continuous function and use its continuity for the middle term.

Three convergence tests. On $[0, 1]$, x^n converges pointwise to zero except at 1, where it remains one. Its L^2 norm is $(2n + 1)^{-1/2}$, whereas its uniform norm stays one. Pointwise asks about each fixed point, uniform uses a single index for every point, and norm convergence depends on the specified norm. For pointwise convergence everywhere to zero without uniform convergence, take the triangular hat supported on $[0, 2/n]$ of height one at $1/n$. Conversely, growing the height of that hat to $\sqrt{n}$ keeps its squared integral $2/3$ while it still tends to zero pointwise. In L^2 even the phrase "value at x" requires a choice of representative.

23.1.4 Completing an unfinished space

Let c_{00} consist of finite-support sequences with the ℓ^2 norm. Truncate $x = (1, 1/2, 1/3, \dots)$ at index N. The truncations are Cauchy by convergence of $\sum j^{-2}$, but their only possible coordinatewise limit has infinite support. Every ℓ^2 sequence is approximated by its truncations, so the completion of c_{00} is ℓ^2.

In general, take Cauchy sequences modulo $(x_n) \sim (y_n)$ when $\|x_n - y_n\| \to 0$. Coordinatewise operations and $\|[(x_n)]\| = \lim \|x_n\|$ are well defined by the triangle inequality. Constant sequences embed X isometrically and densely: the constants x_k approach $[(x_n)]$ as $k \to \infty$. For a Cauchy sequence of classes z_k, choose $y_k \in X$ within $1/k$ of z_k. Then (y_k) is Cauchy and its class is the limit of z_k. This proves completeness.

Universal property. For any bounded linear $T : X \to Y$ with Y Banach there is a unique bounded linear extension to the completion, given by $[(x_n)] \mapsto \lim Tx_n$. Boundedness makes the images Cauchy and equivalent representatives have equal limits. Density forces uniqueness and preserves the norm. This determines the completion up to a unique isometry fixing

X. The coordinates used in an example are optional; the extension property identifies the construction.

Checkpoint. Find the precise step in the norm-equivalence proof that fails on ℓ^2. The unit vectors e_j have pairwise distance $\sqrt{2}$, so the unit sphere contains a sequence with no convergent subsequence.

23.2 Session 2 — Which measurements survive approximation?

23.2.1 A global bound from continuity at one point

For a linear $T : X \to Y$, the following are equivalent: continuity, continuity at zero, and existence of $C < \infty$ with $\|Tx\| \le C\|x\|$ for all x. Indeed, continuity at zero supplies $\delta > 0$ such that $\|z\| < \delta$ gives $\|Tz\| < 1$. Apply this to $z = \delta x/(2\|x\|)$ for $x \neq 0$. Conversely a bound gives $\|Tx - Ty\| \le C\|x - y\|$, hence continuity everywhere. The least possible bound is

$$\|T\| = \sup_{\|x\| \le 1} \|Tx\|.$$

The supremum need not be attained. The diagonal map $e_j \mapsto (1 - 1/j)e_j$ on ℓ^2 has norm one; every individual unit vector has strictly smaller image norm.

Write $\mathcal{L}(X, Y)$ for the bounded linear maps. Norm submultiplicativity $\|ST\| \le \|S\|\|T\|$ follows by applying the two bounds successively. If Y is Banach, $\mathcal{L}(X, Y)$ is Banach even when X is incomplete. Indeed a norm-Cauchy sequence T_n has pointwise limits $Tx = \lim T_n x$, linear by the limit laws. For large n, m, $\|(T_n - T_m)x\| \le \varepsilon\|x\|$. Passing m to infinity shows both boundedness of T and norm convergence. In particular $X^* = \mathcal{L}(X, \mathbb{F})$ is always Banach.

23.2.2 Two notions of duality

The algebraic dual contains every linear functional. A continuous functional permits reliable passage from approximate inputs to measured outputs. On $C[0, 1]$ with the sup norm, $\delta_t(u) = u(t)$ has norm one, as seen from the constant function. With the L^2 norm on the same continuous functions, shrinking triangular hats centered at t give arbitrarily small inputs with output one. Evaluation is now discontinuous. Its formula cannot even be defined on all L^2 classes by choosing an arbitrary representative. On c_{00} the sum $s(x) = \sum_j x_j$ is algebraically legitimate, but a vector with its first N entries $1/\sqrt{N}$ has unit norm and $s(x) = \sqrt{N}$.

For bounded $T : X \to Y$, $T'g = g \circ T$ pulls continuous measurements back from Y to X. It satisfies $(ST)' = T'S'$ and $\|T'\| \le \|T\|$. Its definition uses only composition. An adjoint acting on *vectors* requires something further in Session 4.

Hahn–Banach. Every bounded linear functional f on a subspace M of a real or complex normed space extends to the whole space with exactly the same norm. Neither closedness of M nor completeness of X is required.

Proof outline with the extension step. Over $\mathbb{R}$, set $p(x) = \|f\|\|x\|$. To assign the value c at one new vector z, domination by p requires

$$\sup_{m\in M}\{f(m) - p(m - z)\} \le c \le \inf_{n\in M}\{p(n + z) - f(n)\}.$$

Every lower candidate is below every upper candidate because $f(m + n) \le p(m + n) \le p(m - z) + p(n + z)$. Choose a c in the interval. Scaling separately for positive and negative coefficients proves $f(m)+tc \le p(m+tz)$; applying this to the negative vector gives the absolute bound. Order all dominated extensions by inclusion. A chain has an upper bound by taking its union. Zorn's lemma (a choice principle) yields a maximal extension; the one-vector step forces its domain to be X. In the complex case extend $g = \operatorname{Re} f$ over the underlying real space and set $F(x) = g(x) - ig(ix)$ using the extended g. This is complex linear and extends f. Rotate x by a scalar of modulus one so F is real and nonnegative there; the real bound gives $|F(x)| \le \|f\|\|x\|$.

For $x \neq 0$ prescribe $f(cx) = c\|x\|$ on its line. Extension gives $\|f\| = 1$ and $f(x) = \|x\|$. Consequently $\|x\| = \sup_{\|f\|\le 1} |f(x)|$. Applying this at each Tx proves $\|T'\| = \|T\|$. The canonical evaluation map $J : X \to X^{**}$, $Jx(f) = f(x)$, is therefore isometric. Surjectivity is a separate property called **reflexivity**. Hilbert spaces have it by Riesz representation. The norming functional and the extension can depend on choices; the map J does not.

23.2.3 Quotients and annihilators revisited

For a linear subspace M, set $\|x + M\| = \inf_{m\in M} \|x - m\|$. Changing representatives leaves the infimum unchanged. Scaling and adding near-minimizers proves homogeneity and the triangle inequality. The null classes correspond precisely to $x \in \overline{M}$, so one gets a norm precisely when M is closed. If X is Banach and M closed the quotient is Banach: from a Cauchy sequence of classes select a subsequence with successive quotient distances below 2^{-k}. Lift each difference to a representative of norm below 2^{1-k}. The lifted increments sum in X by completeness, giving a limit class for the subsequence and hence the whole sequence.

Every bounded T vanishing on M factors uniquely through the quotient by $[x] \mapsto Tx$. Indeed $\|Tx\| \le \|T\|\|x - m\|$ for every m. Taking the infimum gives a factor of the same norm (the quotient map is contractive). Applying this to scalar-valued maps identifies $(X/M)^*$ isometrically with the annihilator $M^\circ = \{f \in X^* : f|_M = 0\}$ for closed M. For $x \notin M$ the nonzero class $[x]$ has a norming functional; pulling it back separates x from M. For a dense proper subspace no continuous functional can separate it from any point. This explains why the analytic counterpart of Module B's annihilator remembers a closure.

Checkpoint. If two bounded maps agree on a dense set, where can they differ? Approximate any input by a sequence from that set and pass to the limit. Why does the argument fail for the summation functional on c_{00} when one tries to extend it to ℓ^2?

23.3 Session 3 — Geometry that survives an infinite sum

An inner product is positive definite, conjugate symmetric and linear in its first argument. Minimizing $\|x-ty\|^2$ over scalar t gives $|\langle x,y\rangle| \le \|x\|\|y\|$, with $\|x\|^2 = \langle x,x\rangle$. A **Hilbert space** is complete for this particular norm. Examples are ℓ^2 with $\sum x_j\overline{y_j}$ and L^2 with $\int u\bar{v}$. The parallelogram identity follows by expanding the four inner products. A norm violating that identity cannot arise from any inner product. For instance $\|(1,0)\pm(0,1)\|_1 = 2$, which gives 8 on one side and 4 on the other.

23.3.1 Existence of a nearest vector

Projection theorem. For a closed linear subspace M of a Hilbert space H, each $x \in H$ has a unique nearest $p \in M$. It is characterized by $x-p \in M^{\perp}$, where $M^{\perp} = \{z : \langle z, m\rangle = 0 \ \forall m \in M\}$.

Proof. Put $d = \inf_{m\in M}\|x-m\|$. Choose $m_n \in M$ whose distances tend to d. Applying the parallelogram identity to $x-m_n, x-m_k$ gives

$$\|m_n - m_k\|^2 = 2\|x-m_n\|^2 + 2\|x-m_k\|^2 - 4\|x-(m_n+m_k)/2\|^2.$$

The midpoint lies in M, so the right side is at most the first two terms minus $4d^2$, tending to zero. Completeness gives a limit, closedness puts it in M, and norm continuity gives distance d. For any nonzero $m \in M$, subtracting $\langle x-p, m\rangle m/\|m\|^2$ from the residual would shorten it unless that coefficient vanishes. Conversely if $x-p \perp M$, Pythagoras gives $\|x-q\|^2 = \|x-p\|^2 + \|p-q\|^2$ for all $q \in M$. This proves uniqueness as well as optimality.

Thus $H = M \oplus M^{\perp}$. The decomposition is algebraically unique and its coordinate maps are continuous: $P_M x = p$ is linear by uniqueness and contractive by Pythagoras. For arbitrary M, $M^{\perp} = (\overline{M})^{\perp}$ by continuity, and applying the decomposition to $\overline{M}$ gives $(M^{\perp})^{\perp} = \overline{M}$. For $M = c_{00} \subset \ell^2$, $M^{\perp} = 0$; an infinite-support vector has infimum distance zero but no nearest vector in M. An algebraic complement obtained by extending a Hamel basis supplies no such best approximation.

23.3.2 Orthonormal coordinates and their limits

Let (u_j) be orthonormal, meaning $\langle u_i, u_j\rangle = \delta_{ij}$. For $P_n x = \sum_{j=1}^{n} \langle x, u_j\rangle u_j$ the residual is orthogonal to each u_i, $i \le n$. Hence

$$\|x - P_n x\|^2 = \|x\|^2 - \sum_{j=1}^{n} |\langle x, u_j\rangle|^2.$$

Nonnegativity and passage to the limit prove Bessel's inequality. Moreover $\|P_m x - P_n x\|^2$ is the corresponding tail sum for $m > n$. Completeness supplies a limit p in the closed span, and continuity makes $x - p$ orthogonal to every u_j. Thus p is the projection onto the closed span. If that span is H, the system is a **Hilbert basis**, $x = \sum_j \langle x, u_j\rangle u_j$ in norm, and Parseval's identity equates the full sum of squares to $\|x\|^2$. The standard vectors form such a basis of ℓ^2. A Hamel basis allows each vector a finite expansion; the vector $(1/j)$ immediately distinguishes the two concepts. We use countable systems here, as appropriate for the separable spaces in these examples.

23.3.3 The representing vector and weak convergence

Riesz representation. For each $f \in H^*$ there is exactly one $y \in H$ with $f(x) = \langle x, y\rangle$ for all x, and $\|f\| = \|y\|$.

Proof. Take $y = 0$ for the zero functional. Otherwise $N = \ker f$ is closed. Choose w with $f(w) = 1$ and set $z = w - P_N w$. Then $f(z) = 1$ and $z \perp N$. For each x, $x - f(x)z$ lies in N, so $\langle x, z\rangle = f(x)\|z\|^2$. Set $y = z/\|z\|^2$. If two vectors represent f, their difference is orthogonal to itself. Cauchy–Schwarz gives the upper norm bound, and evaluation at $y/\|y\|$ attains it. The Riesz map $R_H : y \mapsto \langle \cdot\,, y\rangle$ is a conjugate-linear isometric bijection, dependent on the inner product.

A sequence x_n **converges weakly** to x if $f(x_n) \to f(x)$ for every fixed $f \in X^*$. Norm convergence implies weak convergence by boundedness. In ℓ^2, $e_n \rightharpoonup 0$ because $\langle e_n, y\rangle = \bar{y}_n \to 0$ for every $y \in \ell^2$, yet $\|e_n\| = 1$. The norm formula in Session 2 takes a supremum over measurements. For weak convergence each measurement is fixed *before* taking the limit; the measurement attaining the norm can change with n. Weak limits can preserve all fixed linear observations while losing energy.

Checkpoint. If $x_n \rightharpoonup x$ in a Hilbert space and $\|x_n\| \to \|x\|$, recover norm convergence by expanding $\|x_n - x\|^2$. Identify the one weak observation the proof uses.

23.4 Session 4 — Solvability, adjoints and the missing closure

For bounded $T : H \to G$, the functional $x \mapsto \langle Tx, y\rangle$ has norm at most $\|T\|\|y\|$. Riesz supplies a unique vector T^*y satisfying $\langle Tx, y\rangle = \langle x, T^*y\rangle$. Uniqueness and the inner-product convention give linearity in y and $\|T^*\| \le \|T\|$. Conjugate symmetry yields $T^{**} = T$ here for the iterated Hilbert adjoint, hence equality of norms. In the notation of Session 2, $T^* = R_H^{-1} T' R_G$. This identity explains how a pullback of measurements becomes a map between vectors.

Operator on H	Definition	Geometric consequence
Self-adjoint	$T = T^*$	$\langle Tx, x\rangle$ is real
Positive	Self-adjoint with $\langle Tx, x\rangle \ge 0$	Measures a nonnegative quadratic energy
Normal	$TT^* = T^*T$	$\|Tx\| = \|T^*x\|$
Orthogonal projection	$P^2 = P = P^*$	Identity on its closed range and zero on its orthogonal complement

The last row follows by decomposing $x = Px + (I - P)x$ and testing the inner product of the two terms. Conversely the projection from Session 3 has these identities by the orthogonal decomposition. Mere idempotence permits oblique projections.

23.4.1 Compatibility and stability ask different questions

The identity defining the adjoint immediately yields

$$y \in (\operatorname{ran} T)^\perp \iff \langle x, T^*y\rangle = 0 \ \forall x \iff y \in \ker T^*.$$

Taking orthogonal complements proves $\overline{\operatorname{ran} T} = (\ker T^*)^\perp$. Thus compatibility with all adjoint null vectors guarantees arbitrarily accurate residuals. Exact solvability requires membership in the range itself. Injectivity is still exactly $\ker T = 0$ by linearity; it says nothing about onto-ness in infinite dimensions. The right shift $S(x_1, x_2, ...) = (0, x_1, x_2, ...)$ is injective and misses e_1. Its adjoint, the left shift, is onto and kills e_1.

A model inverse problem. Let $De_j = e_j/j$ on ℓ^2. It is positive, self-adjoint, injective and bounded. Its range consists precisely of f with $\sum j^2 |f_j|^2 < \infty$, since the only candidate solution is $x_j = jf_j$. Every finite sequence belongs to the range. The datum $f = (1/j)$ is in its closure but has no solution. Moreover $\|De_j\| = 1/j$ while $\|e_j\| = 1$: arbitrarily small data errors can conceal unit input errors. Finite truncation gives condition number N. Removing high indices stabilizes inversion at the price of a bias.

Closed range theorem (Hilbert spaces). For bounded $T : H \to G$, the following are equivalent:

1. $\operatorname{ran} T$ is closed.
2. $\|Tx\| \geq c\|x\|$ for some $c > 0$ on $H_0 = (\ker T)^\perp$.
3. $\operatorname{ran} T^*$ is closed.

Here is the proof with its Banach-space dependency explicit. Restriction $T_0 : H_0 \to \operatorname{ran} T$ is a bounded bijection by projection onto H_0. If the range is closed both spaces are Banach, so the **bounded inverse theorem** gives a bounded inverse. This general theorem follows from the open mapping theorem via the Baire category theorem; we use it here without proof. Conversely the lower bound makes the representatives of any convergent sequence of images Cauchy in H_0, proving closedness. Under the lower bound, for $z \in H_0$ define $g(Tx) = \langle x, z\rangle$. It is well defined and bounded by $\|z\|\|Tx\|/c$ using the orthogonal representative. Extend it to G by Hahn–Banach and represent it by y. Then $T^*y = z$, giving $\operatorname{ran} T^* = H_0$. Interchanging T and T^* gives the reverse implication.

Least squares. Minimizing $\|Tx - f\|$ asks whether $P_{\overline{\operatorname{ran} T}} f$ is in the range. A minimizer exists exactly then and satisfies $T^*(Tx - f) = 0$ by orthogonality. Its companions differ by $\ker T$. For closed range the unique representative in H_0 is $T^\dagger f = T_0^{-1} P_{\operatorname{ran} T} f$, a bounded pseudoinverse. For D above, the formal inverse on the range is unbounded. In Module A the quotient by the kernel always gives an algebraic isomorphism onto the image. The closed range criterion tells us when its inverse also respects errors in the ambient data norm.

Checkpoint. For $f = (1/j)$, construct approximants with residual tending to zero and compute their solution norms. Which component of the finite-dimensional least-squares argument fails?

23.5 Session 5 — From eigenvalues to spectra

23.5.1 Compactness as uniform finite-dimensional approximation

A bounded linear operator is **compact** if images of bounded sequences always have norm-convergent subsequences. Finite-rank maps are compact by finite-dimensional compactness of closed bounded sets. An operator-norm limit K of finite-rank maps into a Banach space is compact: approximate K on the unit ball uniformly to ε by one finite-rank map, whose bounded image has a finite ε-net. Thus K's image has a finite 2ε-net for every ε. Successively extracting subsequences from balls of shrinking radii gives a Cauchy subsequence; completeness gives its limit. Conversely for a compact operator into a Hilbert space, project onto the span of a finite ε-net of its unit-ball image to get a finite-rank approximation within ε.

For $De_j = e_j/j$, truncation after N coordinates has uniform error $1/(N+1)$. For the identity the same projections P_N converge **strongly** ($\|P_N x - x\| \to 0$ for each x), but $\|I - P_N\| = 1$ for every N, by testing e_{N+1}. The identity on ℓ^2 is noncompact by the separated sequence (e_j). This is the difference between resolving a fixed vector and resolving a whole unit ball.

For a continuous kernel k on $[0,1]^2$, $Ku(s) = \int k(s,t)u(t)\,dt$ is compact on L^2. Uniform continuity allows approximation by kernels constant on finitely many rectangles of a grid. Their outputs are piecewise constant on one fixed finite partition. Cauchy–Schwarz in t followed by integration in s gives $\|K - K_h\| \le \|k - k_h\|_{L^2([0,1]^2)} \to 0$. Integral smoothing can therefore have the same inverse-instability mechanism as a rapidly decaying diagonal.

23.5.2 An inverse-based definition of spectrum

For a bounded operator T on a complex Banach space the resolvent set consists of λ for which $T - \lambda I$ has a bounded everywhere-defined inverse. Its complement is $\sigma(T)$. A bijective bounded map between Banach spaces has bounded inverse; checking injectivity alone no longer suffices. For $|\lambda| > \|T\|$ the geometric series for $(I - T/\lambda)^{-1}$ converges in operator norm, so the spectrum lies in the closed norm disk. We use the disjoint classification below.

Type	Behavior of $T - \lambda I$	Example
Point	Nonzero kernel	$1/j$ for D
Continuous	Injective, dense range, fails to be onto	0 for D
Residual	Injective, range fails to be dense	0 for the right shift

An interval of spectrum without eigenfunctions. On $L^2(0,1)$ take $Mu(t) = tu(t)$. Multiplication by a real bounded function is self-adjoint. If $Mu = \lambda u$, the support of u lies up to null sets in a singleton or the empty set. Thus $u = 0$ as an L^2 class. For $\lambda \notin [0,1]$, division by $t - \lambda$ is bounded by $1/\operatorname{dist}(\lambda, [0,1])$. For $\lambda \in [0,1]$ take the normalized indicator of $(\lambda - \varepsilon, \lambda + \varepsilon) \cap (0,1)$. Its norm is one but $\|(M - \lambda I)u_\varepsilon\| \le \varepsilon$. A bounded inverse is impossible. The range is dense by the adjoint kernel identity, so the entire interval is continuous spectrum, endpoints included. This operator is noncompact: normalized indicators of disjoint subintervals of $(1/2, 1)$ have mutually orthogonal images of norm at least $1/2$.

23.5.3 What additional structure restores eigenvectors?

Compact self-adjoint spectral theorem. For compact self-adjoint K on a Hilbert space, its nonzero spectral values are real eigenvalues of finite multiplicity. Only zero can be their accumulation point. The space is the orthogonal sum of $\ker K$ and the closed span of eigenvectors for the nonzero eigenvalues. Choosing orthonormal vectors in these spaces yields a Hilbert basis. Zero lies in the spectrum in infinite dimension (otherwise $I = K^{-1}K$ would be compact); it may or may not be an eigenvalue.

Proof mechanism. For self-adjoint K, $\|K\| = \sup_{\|x\|=1} |\langle Kx, x\rangle|$. To see the difficult inequality, write $q(x) = \langle Kx, x\rangle$ and bound $4\operatorname{Re}\langle Kx, y\rangle = q(x+y) - q(x-y)$ using the parallelogram

identity; rotate a unit y to make the pairing real. For $K \neq 0$, choose unit vectors with quadratic forms approaching $\lambda = \|K\|$ or $-\|K\|$ along a subsequence. Then

$$\|Kx_n - \lambda x_n\|^2 \leq 2\lambda^2 - 2\lambda\langle Kx_n, x_n\rangle \to 0.$$

Compactness makes a subsequence of Kx_n converge. Since $\lambda \neq 0$, the corresponding x_n converge to a unit eigenvector. Its orthogonal complement is invariant by self-adjointness. Repeat there. An infinite orthonormal sequence of extracted vectors must have eigenvalues tending to zero, since otherwise their images have a separated subsequence. This also rules out infinite multiplicity away from zero. The remaining restriction vanishes: its norm is bounded by each successive extremal modulus. In these diagonal coordinates division by the diagonal entries minus λ is bounded whenever λ is separated from their closure, proving the spectral assertion.

Beyond compactness (theorem statement). Every bounded self-adjoint T has a projection-valued spectral measure E on the real line. To each measurable spectral band B it assigns an orthogonal projection, with $E(\mathbb{R}) = I$, $E(B)E(C) = E(B \cap C)$, and countable additivity on disjoint bands in the strong sense. Then

$$T = \int_{\sigma(T)} t \, dE(t), \qquad g(T) = \int_{\sigma(T)} g(t) \, dE(t)$$

for bounded Borel functions g on the spectrum. For a step function the second integral is simply a weighted sum of band projections. For M, $E(B)u = \mathbf{1}_B u$ and $g(M)u(t) = g(t)u(t)$. A singleton projection detects an eigenspace; for M all singleton projections vanish while the projection for the whole interval is the identity. The general existence proof requires spectral integration or the representation theory of commutative operator algebras and is deferred. For polynomials this calculus is the usual repeated operator multiplication and addition from Module D. Choosing a polynomial large on a desired band and small elsewhere approximately selects that band.

Checkpoint. Does a diagonal matrix obtained by sampling the function t accurately represent the eigenvectors of M? Explain how a collection of sampled eigenvalues can approximate a spectral interval even though the limiting operator has no eigenvectors.

23.6 Session 6 — Solving an equation by testing it

Let V be a real Hilbert space. A bilinear form a assigns each pair (u, v) a scalar. Think of u as the state and v as the probe used to measure its response. Suppose for all u, v,

$$|a(u, v)| \leq C\|u\|\|v\|, \qquad a(v, v) \geq \alpha\|v\|^2, \quad \alpha > 0.$$

These are **boundedness** and **coercivity**. The second requires a single lower constant across the unit sphere. Strict positivity on each nonzero vector is weaker: $\langle De_j, e_j\rangle = 1/j$ for the

strictly positive diagonal of Session 4. The form defines $A : V \to V^*$ by $(Au)(v) = a(u, v)$. A given load $F \in V^*$ prescribes the response to *every* probe. Seeking u with $a(u, v) = F(v)$ for all v is exactly solving $Au = F$.

Lax–Milgram. Under these hypotheses each load has a unique solution, with $\|u\| \le \|F\|/\alpha$. Symmetry is optional.

Proof. Riesz converts $a(u, \cdot)$ to a vector Bu such that $a(u, v) = \langle Bu, v\rangle$. Then B is bounded and linear, and $\|Bu\| \ge \alpha\|u\|$ by coercivity and Cauchy–Schwarz. Hence B is injective and has closed range by the Cauchy argument of Session 4. If $y \perp \operatorname{ran} B$, set $u = y$ to obtain $0 = \langle By, y\rangle \ge \alpha\|y\|^2$. Thus the range is dense as well, hence all of V. Represent F by a vector g and solve $Bu = g$. The lower bound gives the estimate. In a complex space take a linear in the trial variable and conjugate-linear in the test variable, require $\operatorname{Re} a(v, v) \ge \alpha\|v\|^2$, and take F in the continuous **antidual**. Represent $F(v) = \langle g, v\rangle$; the same proof applies.

23.6.1 From infinitely many probes to a linear system

Choose a finite-dimensional conforming space $V_h \subset V$ and require $u_h \in V_h$ to satisfy $a(u_h, v_h) = F(v_h)$ for all $v_h \in V_h$. Restricted to this Hilbert space the same constants work, so existence, uniqueness and stability survive discretization. Subtraction gives **Galerkin orthogonality** $a(u - u_h, v_h) = 0$. Setting $e = u - u_h$ and taking any $w_h \in V_h$ yields

$$\alpha\|e\|^2 \le a(e, e) = a(e, u - w_h) \le C\|e\|\|u - w_h\|.$$

If $e = 0$ the estimate is immediate; otherwise divide and take the infimum. **Céa's lemma** gives

$$\|u - u_h\| \le (C/\alpha) \inf_{w_h \in V_h} \|u - w_h\|.$$

The theorem shifts the analytical work to constructing good approximating spaces. For a symmetric form a is an inner product and $\|v\|_a^2 = a(v, v)$ is equivalent to $\|v\|^2$. Now Pythagoras gives *equality* with the best approximation error in energy norm. Equivalently u_h minimizes $\frac{1}{2}a(w, w) - F(w)$ on V_h, since its excess at w equals $\frac{1}{2}\|w - u_h\|_a^2$. For nonsymmetric coercive forms Céa still applies; the form itself supplies no orthogonal distance.

In a basis $\phi_1, \dots, \phi_n$, $K_{ij} = a(\phi_j, \phi_i)$, $b_i = F(\phi_i)$ and $Kc = b$. With a symmetric coercive form, $c^\top Kc = a(\sum c_j\phi_j, \sum c_j\phi_j) > 0$ for nonzero c. This proves positive definiteness but does not guarantee small Euclidean matrix condition number. The relation between coefficient norm and function norm depends on the basis and mesh.

Checkpoint. In the existence proof separately identify the argument that makes the range closed and the argument that makes it dense. Why would a lower bound on $\|Bv\|$ alone be insufficient? Test the right shift.

23.7 Session 7 — Which space should a differential equation live in?

A piecewise linear finite-element function can have corners. Its ordinary second derivative vanishes inside every element, so asking for $-u_h'' = f$ pointwise would discard the changes of slope at the nodes. Integration against smooth probes records exactly those changes.

23.7.1 One derivative in the weak sense

A **test function** $\varphi \in C_c^\infty(0,1)$ is smooth and zero near both endpoints. For $u, g \in L^2(0,1)$ we call g the weak derivative u' when

$$\int u\varphi' = -\int g\varphi \quad \text{for all such } \varphi.$$

This agrees with integration by parts for smooth u. Uniqueness follows from density of smooth compactly supported functions in L^2. For $u(x) = |x - 1/2|$, split the integral at $1/2$. The two interior boundary terms cancel by continuity and g equals the ordinary slope on each side. A jump in u leaves a multiple of $\varphi(1/2)$, which cannot equal integration against an L^2 function: smooth shrinking bumps of height one there have L^2 norms tending to zero.

Define $H^1 = \{u \in L^2 : u' \in L^2\}$ with norm squared $\|u\|_2^2 + \|u'\|_2^2$. It is Hilbert: a Cauchy sequence has limits u, g in L^2 for its two components. Passing to the limit in the test identity shows $g = u'$. Likewise H^2 requires the weak derivative of u' to belong to L^2. Define H_0^1 as the closure of C_c^∞ in H^1.

We use two elementary one-dimensional Sobolev facts. Each H^1 class has an absolutely continuous representative satisfying $u(x) = u(0) + \int_0^x u'(t)\,dt$. Also H_0^1 consists exactly of representatives with both endpoint values zero. For the first fact, subtract the integral of the weak derivative: the remainder has zero weak derivative, so is a constant almost everywhere. For the latter assertion the forward inclusion follows by continuity of endpoint evaluation in H^1 (bound $|u(0)|$ by $\|u\|_2 + \|u'\|_2$ using the integral formula and averaging). The converse is the one-dimensional density theorem, proved by cutoff and smooth approximation. These density and representative results are the Sobolev prerequisites used here.

For $u(0) = 0$, Cauchy–Schwarz gives $|u(x)| \le \sqrt{x}\|u'\|_2$. Integrating gives the **Poincaré inequality** $\|u\|_2 \le \|u'\|_2/\sqrt{2}$ (a sufficient constant). Hence $V = H_0^1$ is Hilbert in the equivalent energy norm $\|u\|_V = \|u'\|_2$. Bounded sets in V obey $|u(x) - u(y)| \le C|x - y|^{1/2}$ and are uniformly bounded. The Arzelà–Ascoli theorem gives a uniformly convergent subsequence, hence an L^2-convergent one. Its mechanism is diagonal subsequence extraction on a countable dense grid followed by the common modulus of continuity to control between grid points. Thus the embedding $V \hookrightarrow L^2$ is compact.

23.7.2 One boundary-value problem, two operators

For $-u'' = f$ on $(0,1)$ with $u(0) = u(1) = 0$, multiply by v, integrate and remove the boundary term using $v(0) = v(1) = 0$. The resulting problem is

$$u \in V, \qquad \int_0^1 u'v' = \int_0^1 fv \qquad (v \in V).$$

The left side is bounded and coercive with both constants one in the energy norm. For $f \in L^2$, $|F(v)| \le \|f\|_2\|v\|_2 \le \|f\|_2\|v\|_V/\sqrt{2}$. Lax–Milgram applies. The notation $H^{-1} = (H_0^1)^*$ means all continuous loads for this energy space, including point loads $F(v) = v(1/2)$ by the pointwise estimate above. Thus the variational operator $A : V \to H^{-1}$ sending u to the functional $v \mapsto \int u'v'$ is a bounded isometric isomorphism in the chosen norms.

Alternatively define $Lu = -u''$ **on** L^2 with domain $\mathcal{D}(L) = H^2(0,1) \cap H_0^1(0,1)$. This domain specifies both regularity and boundary conditions. On unit vectors $s_k = \sqrt{2}\sin(k\pi x)$, $\|Ls_k\|_2 = k^2\pi^2$, proving unboundedness in L^2. For $f \in L^2$, the weak equation implies $u'' = -f$ weakly by testing smooth functions, hence $u \in \mathcal{D}(L)$. Conversely integration by parts gives the weak equation for each such solution. Thus L and A agree via the embedding of L^2 loads into H^{-1} on this domain. The broader class of weak loads can yield solutions outside H^2.

23.7.3 Assembly on an interval

On nodes $x_i = ih$, $h = 1/N$, let ϕ_i equal one at the ith interior node, zero at the others and linear between nodes. Adjacent slopes are $\pm 1/h$. On a single element the two local shape functions $1 - t/h, t/h$ in local coordinates give

$$K_e = \frac{1}{h}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}, \qquad M_e = \frac{h}{6}\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}.$$

The first integrates products of slopes, the second integrates products of values. Adding contributions at shared nodes and dropping the fixed endpoint unknowns gives $K = h^{-1}\,\mathrm{tridiag}(-1, 2, -1)$ and $M = (h/6)\,\mathrm{tridiag}(1, 4, 1)$. Both are SPD since a nonzero coefficient vector gives a nonzero function and the corresponding quadratic forms are its squared energy and L^2 norms.

For $f = 1$ every interior load is the area of a hat, $b_i = h$. With $N = 2$ there is one interior unknown and $4c = 1/2$, so $c = 1/8$. The exact solution $u = x(1-x)/2$ also has midpoint value $1/8$, yet the energy error is positive: one trial function has two constant slopes and the exact slope varies continuously. Nodal comparisons alone would miss this error.

Checkpoint. Remove both endpoint conditions. Which part of the stiffness argument breaks down? The constant function now has zero energy. Solvability of the pure Neumann problem requires the load to annihilate constants; fixing the mean restores uniqueness.

23.8 Session 8 — From a compact inverse to computed eigenpairs

Let Sf solve the weak Dirichlet problem for $f \in L^2$, considered as a map from L^2 back to L^2. Its factorization is a bounded load embedding $L^2 \to H^{-1}$, the bounded solution map $A^{-1} : H^{-1} \to V$, and the compact embedding $V \hookrightarrow L^2$. Thus $S = L^{-1}$ is compact. Symmetry gives

$$\langle Sf, g\rangle = a(Sg, Sf) = a(Sf, Sg) = \langle f, Sg\rangle.$$

Also $\langle Sf, f\rangle = a(Sf, Sf) \geq 0$. It is injective since $Sf = 0$ implies f annihilates the dense space $V \subset L^2$. Its nonzero eigenvalues μ correspond to eigenvalues $\lambda = 1/\mu$ of L, since $Su = \mu u$ implies $u \in \mathcal{D}(L)$ and $Lu = u/\mu$.

Testing $a(u, v) = \lambda\langle u, v\rangle$ with $v = u$ gives $\lambda > 0$. The differential equation then gives $u(x) = b\sin(\sqrt{\lambda}x) + c\cos(\sqrt{\lambda}x)$ (the weak equation gives H^2 and, here, classical regularity by integration). The endpoints force $c = 0$ and $\sqrt{\lambda} = k\pi$. Consequently

$$\lambda_k = (k\pi)^2, \qquad s_k(x) = \sqrt{2}\sin(k\pi x), \quad k \geq 1.$$

Solving the ODE identifies the possible eigenfunctions. The compact self-adjoint spectral theorem for S, together with $\ker S = 0$, supplies the additional conclusion that these functions span L^2 after closure. The inverse suppresses the kth mode by $(k\pi)^{-2}$. Zero is in the continuous spectrum of S while the eigenvalues of L escape to infinity.

23.8.1 The variational characterization and its discretization

For $0 \neq v \in V$ set $\rho(v) = \|v'\|_2^2/\|v\|_2^2$. The scaled functions $s_j/\sqrt{\lambda_j}$ are an orthonormal basis in the energy inner product: orthogonality follows from the eigen-equation, and energy orthogonality to all s_j implies L^2 orthogonality to all s_j, hence vanishing. Applying Parseval twice gives

$$\rho(v) = \frac{\sum_j \lambda_j |\langle v, s_j\rangle|^2}{\sum_j |\langle v, s_j\rangle|^2}.$$

On the first k modes this weighted average is at most λ_k. Every k-dimensional trial space has a nonzero vector orthogonal in L^2 to the first $k-1$ modes by rank–nullity, and that vector's quotient is at least λ_k. Together these give

$$\lambda_k = \min_{\substack{W \subset V \\ \dim W = k}} \max_{0 \neq w \in W} \rho(w).$$

Restricting the competition to $W \subset V_h$ gives the Ritz value $\lambda_{k,h} \geq \lambda_k$ for $\dim V_h \geq k$. Enlarging a nested space decreases each ordered value. For fixed k, if the spaces approximate every element of V in energy norm as $h \to 0$, these upper bounds converge: approximate each of the first k modes by trial vectors. Their $k \times k$ mass and stiffness Gram matrices converge entrywise to I and $\mathrm{diag}(\lambda_1, \dots, \lambda_k)$. On the unit coefficient sphere this convergence is uniform,

so their maximal quotient tends to λ_k. These k vectors are eventually independent and furnish admissible competitors. For refining linear elements the density hypothesis follows by approximating first by smooth functions and then interpolating: the elementwise slope is the average derivative, converging uniformly to that derivative for smooth inputs.

The discrete quotient is $c^\top Kc/(c^\top Mc)$; stationarity yields $Kc = \lambda_h Mc$. The mass matrix is the L^2 inner product in nodal coordinates. For $M = RR^\top$ (Cholesky), $R^{-1}KR^{-\top}$ is symmetric in ordinary coordinates and $y = R^\top c$. In large problems apply factors or solves without forming dense inverses. Alternatively work directly with the M inner product. An eigenvector residual $Kc - \theta Mc$ measures failure of the **discrete** equation. A zero discrete residual can coexist with nonzero continuum eigenvalue error.

Seminar connection. In block subspace iteration each column carries a mixture of modes. For a symmetric transformed operator B with eigenvectors q_j, $p(B)q_j = p(\lambda_j)q_j$. If a target cluster is separated and the starting block has full overlap with it, repeated filtering attenuates unwanted components according to $\max_{\text{out}} |p(\lambda)| / \min_{\text{in}} |p(\lambda)|$, provided this ratio is below one. Orthonormalization maintains a basis and Rayleigh–Ritz extracts pairs in its span. Chebyshev polynomials offer a concrete design: map the unwanted interval to $[-1, 1]$, where $|T_m(t)| \le 1$; outside it their magnitudes grow as $\cosh(m \operatorname{arcosh} |t|)$. This gives a filter for a separated exterior cluster. The desired spectral interval, mesh-dependent spectral bounds, conditioning of the basis, and rounding errors in the recurrence all matter. F1 explains the operator and approximation target; Module E and the HASC seminar analyze how accurately a finite iteration captures the corresponding subspace.

Checkpoint. Which operation changes the mathematical space: mesh refinement or increasing the filter degree on a fixed mesh? Which error can persist after an eigensolver has converged to machine precision?

23.9 Exercises

Try each problem before consulting its solution. For proofs, mark where boundedness, closedness or completeness first enters.

1. **Two norms on one vector space.** On c_{00} decide which identity map between the 1- and 2-norms is continuous. Construct vectors converging to zero in only one of the norms. Explain why each fixed-dimensional coordinate section gives a different impression.
2. **A dual measurement.** On c_{00} define $f_a(x) = \sum x_j \bar{a}_j$ for an arbitrary sequence a. In the 2-norm, characterize exactly when f_a extends continuously to ℓ^2 and compute the norm.
3. **Dual of a quotient.** With $X = C[0, 1]$ in the sup norm and $M = \{u : u(0) = 0\}$, identify X/M isometrically. Determine M°. What changes if X is instead the continuous functions equipped with the L^2 norm?

4. **Reconstruct the projection proof.** Find the nearest vector to $x = (1, 0, ...)$ in $M = \{y \in \ell^2 : \sum y_j/j = 0\}$. Explain first why a minimizer exists without computing it. Give an example of an incomplete inner-product space with a relatively closed subspace for which a nearest vector fails to exist.
5. **Weak limits and compactness.** Verify that $e_n \rightharpoonup 0$ and $\|(I-P_n)e_{n+1}\| = 1$. Prove that a compact operator takes any bounded weakly convergent sequence to a norm-convergent sequence.
6. **Compatible data and exploding solutions.** For $De_j = e_j/j$, find the least residual norm for $f_j = 1/j$. Is it attained? For $f_j = j^{-s}$, distinguish membership in the data space and exact solvability.
7. **Locate a spectral value.** For the right shift S, classify 0 for S and S^*. Prove directly that 1 is in the continuous spectrum of S using normalized finite blocks.
8. **A compact integral inverse.** Let $G(x,t) = \min(x,t) - xt$. Show that integration against G gives the Dirichlet solution operator. Explain how to see compactness and self-adjointness without solving an eigenproblem.
9. **Coercivity without symmetry.** On real $\mathbb{R}^2$ take $a(u,v) = v^\top Bu$ with $B = \left(\begin{smallmatrix} 1 & -2 \\ 2 & 1 \end{smallmatrix}\right)$. Find C, α, solve with $F(v) = v_2$, and compute the Galerkin approximation in $\operatorname{span}(e_1)$. Check which orthogonality actually holds.
10. **A point load.** Find $u \in H_0^1$ such that $\int u'v' = v(1/2)$. Compute its energy and decide if $u \in H^2$. What happens when a finite-element mesh includes the midpoint?
11. **From interpolation to actual error.** For the constant-load Dirichlet problem, show on a uniform mesh that the piecewise linear interpolant is the Galerkin solution. Compute its exact energy and L^2 errors.
12. **Discrete eigenvalues and their limits.** Insert $c_i = \sin(k\pi i/N)$ into the assembled generalized eigenproblem. Derive an explicit eigenvalue, the fixed-mode convergence rate, and the limiting relative error for the highest-frequency mode. Reconcile these with min–max.

23.9.1 Worked solutions

1. Expanding $(\sum |x_j|)^2$ shows $\|x\|_2 \le \|x\|_1$. Let the first n entries of $x^{(n)}$ equal $1/n$ and the rest zero. Then $\|x^{(n)}\|_1 = 1$ and $\|x^{(n)}\|_2 = n^{-1/2}$. Thus only the identity from the 1-norm to the 2-norm is continuous. On the first n coordinates Cauchy–Schwarz gives the reverse estimate with factor $\sqrt{n}$; our vector attains it. There is no common constant as the dimension grows.

2. If $a \in \ell^2$, Cauchy–Schwarz bounds $|f_a(x)|$ by $\|a\|_2\|x\|_2$. The extension follows by density or the absolutely convergent series. Conversely insert the normalized first N entries of a as the test vector whenever the truncated vector is nonzero. The output is $(\sum_{j\le N} |a_j|^2)^{1/2}$. Any continuous extension must bound every one of these quantities by its norm. This proves necessity and norm equality, including the zero case. The formula constructs Riesz's vector in coordinates.

3. Evaluation at zero is bounded in the sup norm, hence its kernel is closed. Every representative in $u + M$ has norm at least $|u(0)|$; the constant function $u(0)$ attains it. The quotient is isometric to the scalars by evaluation, so every functional annihilating M is a scalar multiple of evaluation. With the L^2 norm let $h_n(x) = \max(1 - nx, 0)$. Then $u - u(0)h_n$ vanishes at zero and approaches u in norm. Thus M is dense, every quotient class has seminorm zero, and the continuous annihilator is trivial. The algebraic quotient is still one-dimensional.

4. The defining functional has Riesz vector $a_j = 1/j$ and is continuous, so $M = a^\perp$ is closed in the complete space ℓ^2. Projection gives $p = x - \langle x, a\rangle a/\|a\|^2 = e_1 - a/\sum j^{-2}$. The subtraction removes the unique component normal to M. Now replace the ambient space by c_{00} and take the kernel of the same bounded functional restricted there. It is relatively closed. If e_1 had a nearest vector there, the residual r would be perpendicular to the kernel by the one-direction minimization argument. For each $j > 1$, $e_j - e_1/j$ is in this kernel. Thus $r_j = r_1/j$ for all j. Finite support forces $r = 0$, but e_1 is outside the kernel. The proof of the theorem constructed a Cauchy sequence; only completeness guaranteed a limit inside the ambient space.

5. Coordinate observations via Riesz give $\langle e_n, y\rangle \to 0$. The coordinate projector misses e_{n+1} entirely. If $x_n \rightharpoonup x$ is bounded and K is compact, each subsequence of Kx_n has a norm-convergent subsubsequence. Every such limit equals Kx, because bounded maps preserve weak convergence ($f(Kx_n) = (K'f)(x_n)$) and continuous functionals separate points. If norm convergence to Kx failed, one could choose a subsequence at distance at least ε, contradicting this conclusion.

6. The first N entries of f are attained by $x^{(N)} = (1, \dots, 1, 0, \dots)$. The residual norm is $(\sum_{j>N} j^{-2})^{1/2} \to 0$ but $\|x^{(N)}\| = \sqrt{N}$. Residual zero would force all entries equal to one. Thus the infimum is zero and no minimizer exists. For real s, a datum belongs to ℓ^2 exactly when $2s > 1$, and $x_j = j^{1-s}$ belongs to ℓ^2 exactly when $2s - 2 > 1$. The endpoints fail by divergence of the harmonic series.

7. S has zero kernel and closed range $e_1^\perp$, so zero is residual. $S^* e_1 = 0$, hence zero is point spectrum for S^*. At one, neither S nor its adjoint has a nonzero fixed vector: the first coordinate and recurrence force zero for S, and a constant sequence in ℓ^2 vanishes for S^*. Thus $S - I$ is injective with dense range. Let v_n be $1/\sqrt{n}$ on the first n coordinates and zero thereafter. Only the two boundary coordinates survive in $(S - I)v_n$, giving norm $\sqrt{2/n}$. A bounded inverse is impossible. The bounded inverse theorem rules out surjectivity, giving continuous spectrum.

8. For fixed t, G vanishes at both endpoints; its x-derivative is $1 - t$ to the left of t and $-t$ to the right. Integrating this slope against v' gives $(1 - t)v(t) + tv(t) = v(t)$. Multiply by $f(t)$ and integrate. Switching the order of integration gives $a(\int G(\cdot, t)f(t)\,dt, v) = \int fv$; the switch is justified by bounded slopes and integrability on a finite square. The endpoint values and bounded integrated slope place the candidate in V. Uniqueness identifies the inverse. The kernel is symmetric and continuous, so symmetry of integration yields self-adjointness and the rectangle approximation in Session 5 gives compactness.

9. $B^\top B = 5I$, hence $C = \sqrt{5}$; $u^\top Bu = \|u\|^2$ gives $\alpha = 1$. Solving $Bu = e_2$ gives $u = (2,1)^\top/5$. Testing ce_1 against e_1 gives $c = 0$. Therefore $a(u - u_h, e_1) = e_1^\top Bu = 0$, yet $\langle u - u_h, e_1\rangle = 2/5$. The Euclidean best approximation is $(2/5)e_1$, with error $1/5$. Galerkin's error is $1/\sqrt{5}$, attaining the Céa factor $\sqrt{5}$ in this example.

10. Take $u(x) = G(x, 1/2) = \frac{1}{2}\min(x, 1-x)$. Its slopes are $\pm 1/2$; splitting $\int u'v'$ yields $v(1/2)$ as in Exercise 8. Thus $\|u\|_V^2 = 1/4 = u(1/2)$. The slope has a jump, so its weak derivative cannot be an L^2 function by the bump test of Session 7. If the mesh contains $1/2$, u already belongs to the conforming piecewise linear space. The Galerkin solution is exact by uniqueness. A singular load and an exactly representable solution can occur together.

11. The derivative of the nodal interpolant $I_h u$ on each element is the average of u'. Therefore $\int_e (u - I_h u)' v_h' = 0$ since v_h' is constant there. Summing proves Galerkin orthogonality and uniqueness identifies $I_h u = u_h$. In a local coordinate $t \in [0, h]$ the interpolation error is $t(h-t)/2$ and its derivative is $h/2 - t$. The squared local errors integrate to $h^5/120$ and $h^3/12$ respectively. Summing $1/h$ elements gives $\|u - u_h\|_2 = h^2/\sqrt{120}$ and $\|u - u_h\|_V = h/\sqrt{12}$. This calculation measures error between the nodes.

12. Write $\theta = k\pi/N$ for $1 \le k < N$. The adjacent sine samples sum to $2\cos\theta\, c_i$. Thus the eigenvalues are

$$\lambda_{k,h} = \frac{6(1 - \cos\theta)}{h^2(2 + \cos\theta)}.$$

For fixed k, expanding the cosine gives $\lambda_{k,h} = (k\pi)^2[1 + (k\pi h)^2/12 + O((k\pi h)^4)]$. For $k = N-1$, $\theta \to \pi$ and $\lambda_{k,h}/(k\pi)^2 \to 12/\pi^2$, giving relative error approaching $12/\pi^2 - 1$. Min–max gives upper bounds in both cases, while the convergence theorem fixes the mode index *before* refining. The oscillation wavelength of the last mode always spans only about two elements.

23.10 Computational laboratory — What does refinement actually measure?

The three cells run in order in a Quarto Python notebook or Jupyter with NumPy, SciPy, and Matplotlib. The first experiment compares two ways of measuring the *same* truncation error. The second and third solve the source and eigenvalue problems using the assembled matrices. Predict each slope before running the cell.

23.10.1 Experiment 1 — A tail, a unit ball and amplified noise

For an exact test vector use $x_j = \sqrt{1/j - 1/(j+1)}$, whose squared tail beyond N is $1/(N+1)$ by telescoping. The unit-ball error for truncating I is witnessed by e_{N+1}; for D it is $1/(N+1)$. A data perturbation δe_N becomes $N\delta e_N$ under inversion of the retained block.

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.linalg import eigh
from scipy.sparse import diags
from scipy.sparse.linalg import spsolve

sizes = 2 ** np.arange(2, 10)
fig, axes = plt.subplots(1, 2, figsize=(9, 3.6))
axes[0].loglog(sizes, 1 / np.sqrt(sizes + 1), 'o-', label='fixed vector tail')
axes[0].loglog(sizes, np.ones_like(sizes), 's-', label='identity, unit ball')
axes[0].loglog(sizes, 1 / (sizes + 1), '^-', label='diagonal D, unit ball')
axes[0].set(xlabel='retained coordinates N', ylabel='norm error')
axes[0].legend(fontsize=8)
gains = []
for n in sizes:
    sigma = 1 / np.arange(1, n + 1)
    noise = np.zeros(n)
    noise[-1] = 1e-6
    gains.append(np.linalg.norm(noise / sigma) / np.linalg.norm(noise))
axes[1].loglog(sizes, gains, 'o-')
axes[1].set(xlabel='retained coordinates N', ylabel='solution error / data error')
fig.tight_layout()
plt.show()
```

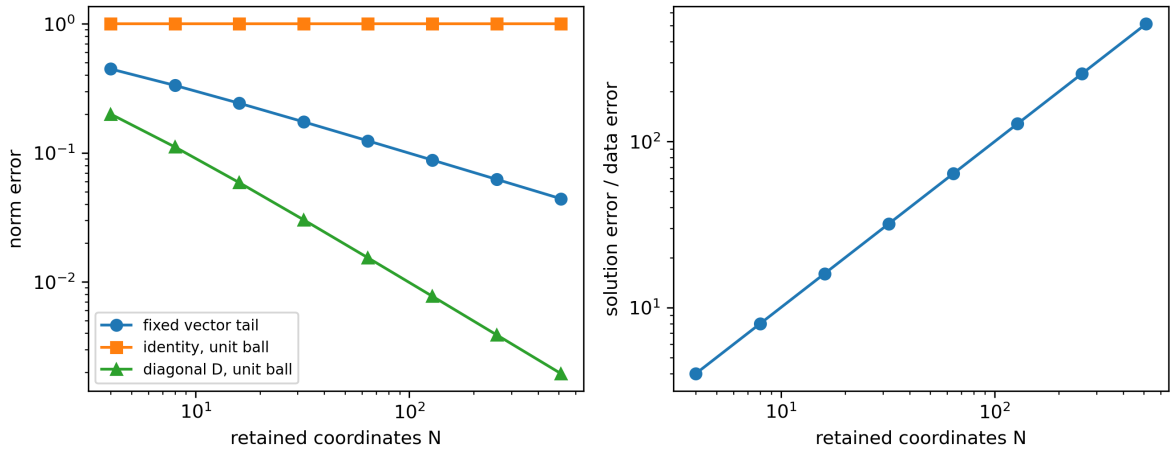


Figure 23.1: Truncation error for a fixed vector and on the unit ball, alongside inverse amplification.

The left panel uses exact infinite-tail formulas. Summing a fixed reference vector of length 512

would instead force its measured tail to zero at 512. The right panel performs actual finite-dimensional inversion in singular coordinates. Its growing slope is explained by the analytical range calculation in Session 4.

23.10.2 Experiment 2 — A Galerkin solution with zero nodal error

Assemble K and M after enforcing homogeneous boundary conditions. Evaluate errors inside each interval with three-point Gauss quadrature, exact for polynomials through degree five. The squared value error is quartic in this example.

```
def assemble(n):
    h = 1 / n
    main = np.ones(n - 1)
    side = np.ones(n - 2)
    stiffness = diags([-side, 2 * main, -side], [-1, 0, 1], format='csc') / h
    mass = diags([side, 4 * main, side], [-1, 0, 1], format='csc') * (h / 6)
    return stiffness, mass

meshes = np.array([8, 16, 32, 64, 128])
energy_errors, value_errors = [], []
points, weights = np.polynomial.legendre.leggauss(3)
t = (points + 1) / 2
for n in meshes:
    h = 1 / n
    K, M = assemble(n)
    coeff = np.r_[0., spsolve(K, np.full(n - 1, h)), 0.]
    left = np.arange(n)[:, None] * h
    x = left + h * t
    uh = coeff[:-1, None] * (1 - t) + coeff[1:, None] * t
    slopes = np.diff(coeff)[:, None] / h
    def integrated_norm(error):
        return np.sqrt(h / 2 * np.sum(error**2 * weights))
    value_errors.append(integrated_norm(x * (1 - x) / 2 - uh))
    energy_errors.append(integrated_norm(0.5 - x - slopes))

hvals = 1 / meshes
assert np.allclose(value_errors, hvals**2 / np.sqrt(120))
assert np.allclose(energy_errors, hvals / np.sqrt(12))
plt.figure(figsize=(6, 3.8))
plt.loglog(hvals, energy_errors, 'o-', label='energy norm')
plt.loglog(hvals, value_errors, 's-', label='L² norm')
plt.xlabel('mesh width h')
```

```
plt.ylabel('integrated error')
plt.legend()
plt.tight_layout()
plt.show()
```

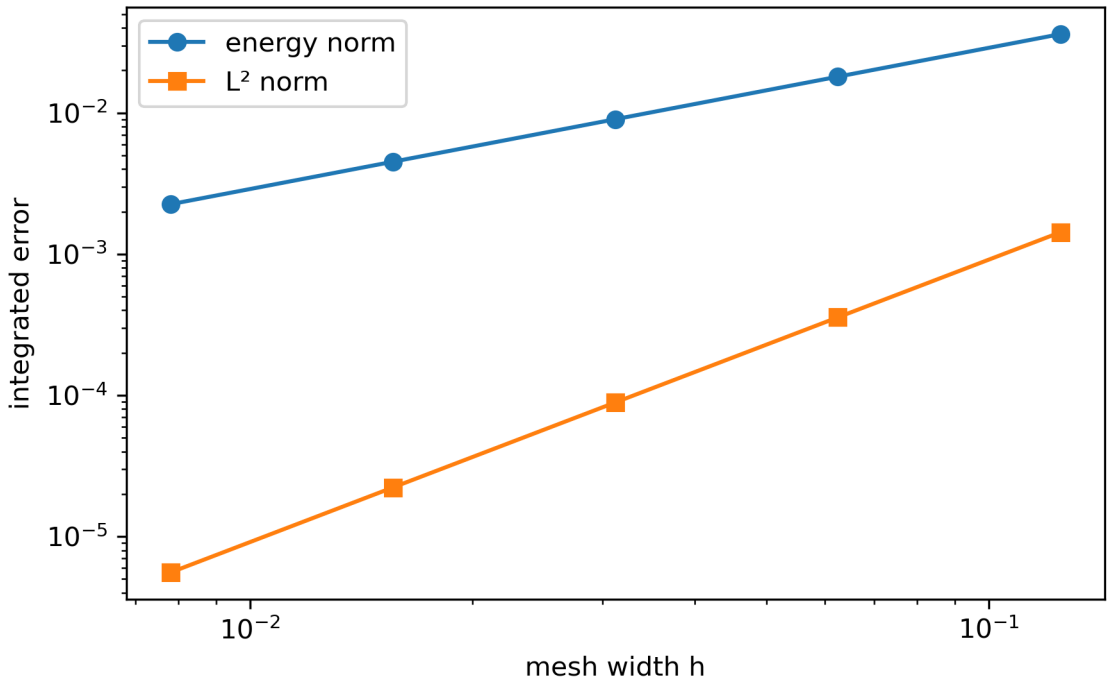


Figure 23.2: Energy and integral errors for the linear finite-element solution with constant load.

Halving h halves the energy error and quarters the L^2 error. Both conclusions here refer to this smooth quadratic solution and the chosen conforming spaces. Exercise 10 gives a different outcome for a point load.

23.10.3 Experiment 3 — Eigenvalue convergence and solver residual

Use the consistent mass matrix. The matrices are deliberately small so a dense symmetric generalized eigensolver can compute both a low and a high mode. Compare each with the corresponding continuum eigenvalue, and compare the low-mode error with the residual of the discrete equation.

```
low_errors, high_errors = [], []
print('elements   first Ritz value   relative residual')
for n in meshes:
    K, M = assemble(n)
    lam, vectors = eigh(K.toarray(), M.toarray())
    c = vectors[:, 0]
    residual = K @ c - lam[0] * (M @ c)
    scale = np.linalg.norm(K @ c) + lam[0] * np.linalg.norm(M @ c)
    print(f'{n:8d}   {lam[0]:16.10f}   {np.linalg.norm(residual)/scale:.2e}')
    low_errors.append(lam[0] / np.pi**2 - 1)
    high_errors.append(lam[-1] / ((n - 1) * np.pi)**2 - 1)
    k = np.arange(1, n)
    theta = k * np.pi / n
    exact_discrete = 12 * n**2 * np.sin(theta / 2)**2 / (2 + np.cos(theta))
    assert np.allclose(lam, exact_discrete)

plt.figure(figsize=(6, 3.8))
plt.loglog(hvals, low_errors, 'o-', label='first mode')
plt.loglog(hvals, high_errors, 's-', label='last mode')
plt.axhline(12 / np.pi**2 - 1, ls=':', color='black', label='last-mode limit')
plt.xlabel('mesh width h')
plt.ylabel('relative eigenvalue error')
plt.legend()
plt.tight_layout()
plt.show()
```

```
elements   first Ritz value   relative residual
       8       9.9970806562   6.34e-15
      16       9.9013536784   3.47e-14
      32       9.8775341175   1.63e-13
      64       9.8715863533   5.85e-13
     128       9.8700998593   3.72e-12
```

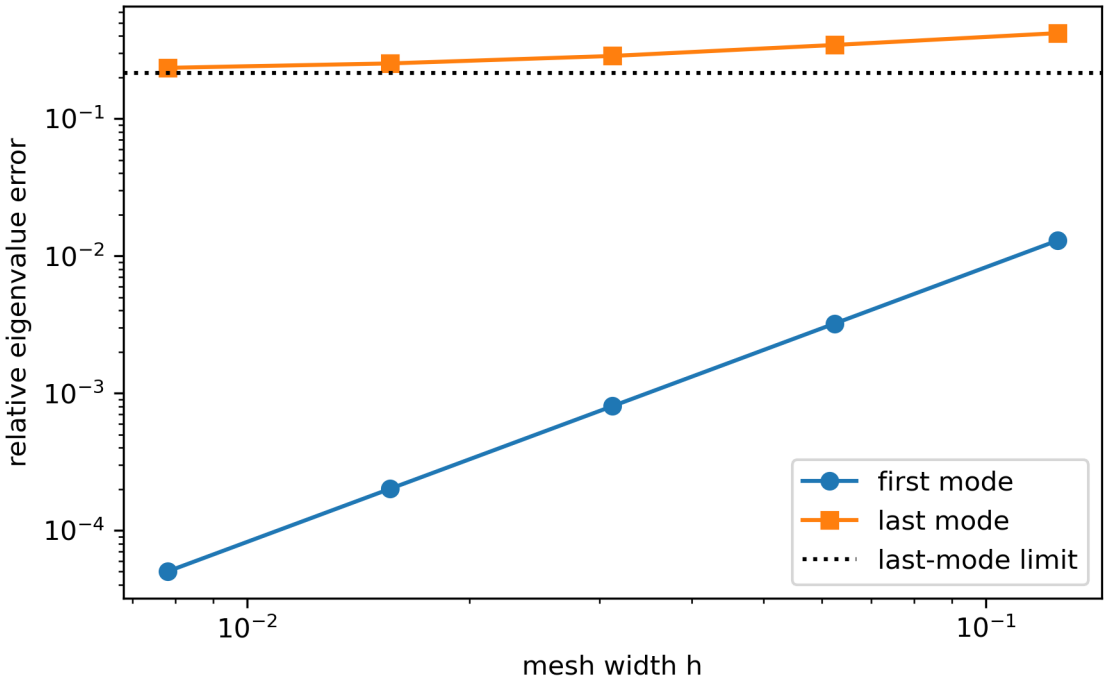


Figure 23.3: Relative continuum eigenvalue errors for a fixed mode and the highest available mode.

For $N = 8$ expect $\lambda_{1,h} \approx 9.9971$, above $\pi^2 \approx 9.8696$. The residual can already be tiny. Refinement steadily improves this fixed mode, whereas tracking the last mode on successive meshes changes the target. A finite spectrum alone cannot establish compactness, range closure or completeness of the sine functions; the proofs explain which limits the calculations illustrate.

23.11 Synthesis — What did the topology change?

Module A asks whether a linear equation has a solution and how the kernel accounts for its nonuniqueness. The quotient $X/\ker T$ still parametrizes the image algebraically. The diagonal example introduces a new question: does a small change in data produce a small change in the solution class? For a bounded Hilbert-space operator, closed range answers precisely that question. The right shift separates uniqueness from existence, while the decaying diagonal separates arbitrarily small residuals from actual solvability.

Module B views functionals as measurements. Restricting morphisms to bounded linear maps gives a category in which composition preserves continuity. Three familiar constructions now

have analytic content. Completion extends bounded maps to limits, the normed quotient factors maps that kill a closed subspace, and the continuous dual reverses arrows by composition. Hahn–Banach explains why continuous measurements still detect every point. They cannot detect the difference between a dense subspace and its closure by annihilation. In particular Module B's finite-dimensional identity $\operatorname{ran} T' = (\ker T)^\circ$ can fail: for the real diagonal D under the Riesz identification the right side is the entire dual, while the left is a proper dense range.

Module C adds an inner product. With completeness it turns minimizing sequences into projections, makes compatible infinite coordinate lists into vectors, and gives every continuous measurement a representing vector. These results lead directly to least squares and coercive variational equations. An adjoint depends on the two Hilbert structures. Changing the test-space norm also changes what counts as a bounded load. A point force is a particularly concrete witness: it is continuous on the one-dimensional energy space and cannot be represented by an L^2 load.

Module D resolves operators into modes. In F1 compact self-adjoint operators retain an orthonormal eigenvector expansion, while a general self-adjoint operator may require projections onto spectral bands. For the Dirichlet problem the decisive compact operator is the *solution* operator. It sends bounded data to a family of functions with enough common regularity for a convergent subsequence. This explains both the sine expansion and the decay of the inverse eigenvalues.

For numerical PDEs the trial space, test space, norm and boundary conditions are parts of the mathematical model. Galerkin approximation transports their structure into matrices. For numerical linear algebra there are then two approximation questions: how accurately the trial space captures a continuum invariant subspace, and how accurately a numerical iteration captures the corresponding matrix invariant subspace. For F2 a further question remains: which constructions are characterized by the maps they admit? Completion and quotient already provide examples. Riesz representation shows what an additional geometric structure can supply beyond the underlying algebraic category.

23.11.1 Theorem map

Result	Hypotheses	Conclusion	Why it matters
Equivalence of norms	Finite-dimensional space	Two-sided norm bounds	Choice of error norm preserves convergence there
Completion	Normed domain, Banach target	Bounded maps extend uniquely to the completion	Defines operations on limits of approximations

Result	Hypotheses	Conclusion	Why it matters
Boundedness criterion	Linear map of normed spaces	Continuity iff a global norm bound exists	Turns continuity into an error estimate
Hahn–Banach	Bounded functional on any subspace	Norm-preserving extension	Norm recovery and separation in quotients
Projection	Hilbert space, closed subspace	Unique orthogonal best approximation	Existence of a best vector even without a finite basis
Riesz representation	Hilbert space, continuous functional	Unique representing vector of equal norm	Converts dual equations into vector equations
Parseval	Total orthonormal system	Coefficient energy equals squared norm	Justifies infinite expansions and truncation errors
Closed range	Bounded Hilbert-space operator	Closed range iff bounded below off kernel	Stability of least squares and the pseudoinverse
Compact spectral theorem	Compact, self-adjoint	Mode expansion; possible accumulation only at zero	Explains smoothing and unstable inversion
Spectral measure theorem	Bounded, self-adjoint	Resolution by band projections	Functional calculus even without eigenvectors
Lax–Milgram	Hilbert space, bounded coercive form, continuous load	Unique solution, inverse bound $1/\alpha$	Well-posed weak boundary-value problems
Céa	Same assumptions, conforming Galerkin space	Error within C/α of best	Reduces the error analysis to approximation
Variational min–max	Positive Dirichlet operator with compact inverse here	Ordered Rayleigh quotient characterization	Conforming Ritz upper bounds and fixed-mode convergence

23.11.2 Mastery assessment

Work from a blank page. For each item aim to give a definition, a proof or calculation, and one example in which a hypothesis matters.

- ☐ Distinguish normed, Banach and Hilbert spaces. Exhibit an incomplete inner-product space and a Banach norm that cannot arise from an inner product.
- ☐ Construct the completion of c_{00} in the 2-norm. Explain why its identity into a complete target extends uniquely when bounded.
- ☐ Derive boundedness of a linear map from continuity at zero. Test whether a proposed measurement extends to the completion.
- ☐ Identify the continuous dual of a quotient and use Hahn–Banach to separate a point from a closed subspace.
- ☐ Rebuild the projection proof from the midpoint estimate, then derive Riesz from the kernel of a functional.
- ☐ Use ℓ^2 to distinguish weak convergence, norm convergence, strong operator convergence and operator-norm convergence.
- ☐ Derive the adjoint range identities and give data orthogonal to the adjoint kernel for which there is no exact solution.
- ☐ Recognize compactness from finite-rank approximation. Explain the differing spectral behavior of a decaying diagonal and multiplication by t.
- ☐ Derive the weak Dirichlet equation, identify its domain and loads, and explain the two different meanings of its operator bound.
- ☐ Reconstruct Lax–Milgram and Céa. Identify the extra symmetry that turns quasi-optimality into exact energy best approximation.
- ☐ Assemble a stiffness–mass pencil, justify both definiteness claims, and distinguish a discrete eigenpair residual from continuum error.
- ☐ Starting with compactness of the solution operator, derive the sine basis, min–max upper bounds, and a reason a fixed low mode improves under mesh refinement.

Final reconstruction problem. Explain to a fellow student how $\int u'v' = F(v)$ leads both to a stable inverse and to an inverse with eigenvalues tending to zero. Specify each domain, codomain and norm. In energy and dual norms $A : V \to V^*$ is an isometric isomorphism. Viewing the solution instead as $S : L^2 \to L^2$ introduces a compact embedding and yields eigenvalues $(k\pi)^{-2}$. Inverting this last map on its range differentiates twice and amplifies high frequencies. The choice of spaces is exactly what reconciles the two conclusions.

24 Module F2 — Structural Algebra: From Linear Algebra to Universal Constructions and Category Theory

Suppose a solver returns two different potentials for the same electric field. Their difference might be a constant. Which object should hold the answer? A quotient records precisely the ambiguity. Now suppose an eigensolver returns two different bases for the same eigenspace. Here the natural answer is a subspace, and, given a complementary spectral subspace, a projector onto it. Why does one problem call for a quotient and the other for a projector? The distinction is visible in the maps each answer must support.

Module A traced the information lost by a linear map. Module B followed measurements backward through it. Module C used an inner product to turn measurements into vectors. Module D studied the subspaces preserved by an operator. We will vary the structure that maps must preserve and see which of these arguments survive. This is a parallel extension of Review II. The main spectral payoff comes in Session 6, where a single polynomial congruence constructs a projector even for a matrix with a defective eigenvalue.

Conventions. $\mathbb{F}$ denotes a field. $V^* = V^\vee$ is the algebraic dual, $T^\vee$ the dual map, U^0 the annihilator. For complex inner products we follow Module C and take $\langle x, y\rangle = y^H x$ in standard coordinates. T^* denotes the inner-product adjoint. Composition gf means apply f first. Infinite-dimensional algebraic basis arguments use the usual basis-extension principle.

24.1 Session 1 — Which maps preserve the problem?

24.1.1 From changing coordinates to changing categories

Take the point set $\mathbb{R}^2$. A translation preserves Euclidean distances. It preserves the vector-space origin only when the translation is zero. An invertible shear fixes the origin and respects addition and scaling, yet changes lengths. Before calling two descriptions equivalent, we must decide which operations, relations, or measurements matter.

Setting	Objects	Morphisms	What an inverse must preserve
Set	Sets	Functions	A bijection suffices
Grp	Groups	Homomorphisms	Multiplication
$\mathbf{Vect}_{\mathbb{F}}$	Spaces over one field	$\mathbb{F}$-linear maps	Addition and scalar multiplication
Unital rings	Rings with identity	Unital ring maps	Addition, multiplication, and identity
R-modules	Left modules over a fixed ring	R-linear maps	Addition and the action of every scalar
Normed spaces	Normed spaces over $\mathbb{R}$ or $\mathbb{C}$	Bounded linear maps	Both directions must be bounded
Top	Topological spaces	Continuous maps	Both directions must be continuous

A **group** has an associative multiplication, an identity and inverses. A **ring** has an abelian group of additions and an associative multiplication distributing over addition. We take rings with identity. A field additionally permits division by every nonzero scalar and has commutative multiplication. A **left module** is an abelian group with a scalar action of a ring satisfying the familiar vector-space distributivity, associativity and unit axioms. Over $\mathbb{Z}$, integer scaling just means repeated addition and its negative. Over a matrix ring, the order of scalar multiplication matters. A bounded linear map satisfies $\|Tx\| \le C\|x\|$ for a fixed C and all x. Composition preserves this condition because the bounds multiply.

Two tests of equivalence. The sets underlying $\mathbb{Q}$ and $\mathbb{Q}^2$ are both countably infinite, so there is a bijection. A linear map from $\mathbb{Q}$ has image spanned by the image of 1, hence cannot cover $\mathbb{Q}^2$. For an example on literally the same set, equip $\mathbb{R}$ once with the discrete topology and once with the usual topology. Every singleton is open in the former. A homeomorphism takes open sets to open sets, so no homeomorphism exists between the two. The choice of morphisms changes the answer even before we write any coordinates.

24.1.2 What a diagram asserts

A **category** specifies objects, morphisms $\mathrm{Hom}(X, Y)$ with specified source and target, an identity 1_X at each object, and a composite $gf : X \to Z$ whenever $f : X \to Y$, $g : Y \to Z$. The laws are $h(gf) = (hg)f$ and $1_Y f = f = f 1_X$. An isomorphism has a two-sided inverse in the category. The inverse is unique since two inverses g, h satisfy $g = g(fh) = (gf)h = h$.

The diagram $V \xrightarrow{T} W$ records the type of a map. A **commutative** diagram additionally equates compositions along paths with common endpoints. For example, let $T(x, y) = x - y$ and $D = \{(a, a)\}$. The following square is a claim that can be tested on an arbitrary input:

$$\begin{array}{ccc} \mathbb{R}^2 & \xrightarrow{T} & \mathbb{R} \\ {\scriptstyle q}\downarrow & & \downarrow{\scriptstyle 1} \\ \mathbb{R}^2/D & \xrightarrow{\bar{T}} & \mathbb{R}. \end{array}$$

Here $q(x,y) = [(x,y)]$. Changing a representative to $(x+a, y+a)$ leaves $x-y$ unchanged. Thus the lower arrow exists and both paths give $x-y$. One elementwise calculation constructs a map; the equation $\bar{T}q = T$ then describes how to use it with every input map $X \to \mathbb{R}^2$ at once.

Reversing all arrows and reversing the order of composition gives the **opposite category** $\mathcal{C}^{\mathrm{op}}$. Its objects are the same and $\mathrm{Hom}_{\mathcal{C}^{\mathrm{op}}}(X,Y) = \mathrm{Hom}_{\mathcal{C}}(Y,X)$. Associativity and the unit laws still hold. This permits us to reverse a whole statement about arrows. It gives the product/coproduct pairing in the next session. Constructing actual dual *spaces* and *maps* requires extra mathematics.

24.2 Session 2 — Three problems that determine their own objects

24.2.1 Identifying inputs while preserving all admissible measurements

Let $U \le V$. The relation $v \sim v'$ if $v - v' \in U$ partitions V into affine translates of U. Addition and scaling on classes are well defined because differences stay in U. Let $q(v) = v + U$.

Quotient factorization theorem. For every W, composition with q gives a bijection

$$\mathrm{Hom}(V/U, W) \longrightarrow \{f : V \to W \text{ linear} \mid f|_U = 0\}.$$

Proof. For $h : V/U \to W$, hq kills U. Conversely define $\bar{f}([v]) = f(v)$. Equivalent representatives differ by $u \in U$, so linearity of f gives equal values. The quotient operations make $\bar{f}$ linear, and $\bar{f}q = f$. Any solution h must take $[v] = q(v)$ to $f(v)$, which proves uniqueness. The constructions undo one another. □

It is the quantifier *for every target and every admissible map* that makes this property universal. There is even a computational prescription. For $U = \mathrm{span}(1,1)$ take quotient coordinate $d = x - y$. Every matrix killing U has columns $b, -b$, hence sends (x,y) to bd. Once d is known one can change the output space or b without recovering a representative.

Why universality determines the answer. If $q' : V \to Q'$ also kills U and satisfies the same property, factor q' through q to obtain a, and q through q' to obtain b. Then $baq = q = 1q$. Uniqueness gives $ba = 1$; exchanging the roles gives $ab = 1$. Exactly one isomorphism satisfies $aq = q'$. The compatibility clause matters: the underlying spaces may admit many other isomorphisms.

For a group G, the same reasoning uses cosets of a **normal** subgroup N (one satisfying $gNg^{-1} = N$). Normality makes coset multiplication well defined. Every homomorphism killing N factors uniquely by $[g] \mapsto f(g)$. For a ring, the required additive subgroup is a two-sided **ideal**, closed under multiplication from either side, so the multiplication of classes is well defined. These hypotheses tell us precisely when the proposed identification respects the operations.

24.2.2 Assembling outputs and prescribing inputs

Suppose two instruments read $f : X \to V$ and $g : X \to W$. Their joint reading is the unique map $\langle f, g\rangle : X \to V \times W$, $x \mapsto (f(x), g(x))$, whose projections give f, g. Existence follows from the formula; uniqueness follows because the coordinates determine a pair. This argument works for arbitrary sets or for vector spaces with componentwise operations. In any category this mapping problem defines a **product** (P, π_1, π_2) whenever a solution exists.

Turn the problem around. Given linear rules $a : V \to X$, $b : W \to X$, can we extend both rules to a common domain? In the external direct sum $V \oplus W$ insert each input using $i_1(v) = (v, 0)$ and $i_2(w) = (0, w)$. Define $[a, b](v, w) = a(v) + b(w)$. This is linear and restricts correctly. Every pair is $i_1(v) + i_2(w)$, forcing the formula. The object and its insertions satisfy the **coproduct** property. Reversing the product's arrows gives exactly this uniqueness problem. The preceding two-factor uniqueness proof for quotients adapts verbatim by requiring the comparison maps to commute with both structure arrows.

For a matrix example take $f = [1\ 2]$, $g = [0\ 3]$ on $\mathbb{R}^2$. Pairing them stacks their rows. Given column maps $a = (1, 2)^T$, $b = (3, 0)^T$ out of $\mathbb{R}$, extending them places their columns side by side. Composition computes $af + bg$. This is the origin of block matrix multiplication.

The common finite object is a **biproduct**. Its identities

$$\pi_j i_k = \delta_{jk} 1 \quad (0 \text{ for } j \neq k), \qquad i_1\pi_1 + i_2\pi_2 = 1$$

recover the two universal maps as $i_1 f + i_2 g$ and $a\pi_1 + b\pi_2$. The formulas apply to any finite number of factors. In sets the coproduct is a *tagged disjoint union*: define the outgoing function separately on each tag. Two singleton sets have a one-element product and a two-element coproduct. For infinitely many vector spaces the product permits arbitrary families, whereas the coproduct consists of finitely supported families. Finite support is what makes the extension $\sum_j a_j(v_j)$ meaningful algebraically.

24.2.3 Adding subspaces in an ambient space

Given $U, W \leq V$, the external sum maps onto the internal sum by $s(u, w) = u + w$. Its kernel consists of $(z, -z)$ for $z \in U \cap W$. Thus the addition map identifies $U \oplus W$ with $U + W$ precisely when the intersection vanishes. For three subspaces one needs $U_i \cap \sum_{j \neq i} U_j = 0$ for

each i: a nonzero component of a zero sum witnesses failure. The two coordinate axes and the diagonal in $\mathbb{R}^2$ test why pairwise intersections alone do not suffice. In the partially ordered category of subspaces of *a fixed* V (one arrow for inclusion), $U+W$ is itself the coproduct: any subspace containing both contains all sums. Changing the category has changed the universal problem.

24.3 Session 3 — Kernels, obstructions and splitting

For $T : V \to W$, the inclusion $k : K = \ker T \hookrightarrow V$ is universal among maps $h : X \to V$ with $Th = 0$. Indeed $h(x) \in K$, so corestrict h to K; injectivity of k forces uniqueness. Conversely the projection $c : W \to W/\operatorname{im} T$ is universal among maps $g : W \to Y$ with $gT = 0$. The premise exactly says g kills the image, so the quotient theorem supplies the unique map. This is the **cokernel**. Both constructions require a notion of zero morphism; a zero object, simultaneously initial and terminal, supplies one by $X \to 0 \to Y$. In vector spaces the zero-dimensional space serves.

Object	Where it lives	Question concerning $Tv = w$
$\ker T$	Subspace of the domain	How do solutions differ?
$\operatorname{im} T$	Subspace of the target	Which right-hand sides occur?
$\operatorname{coker} T$	Quotient of the target	Which class $[w]$ obstructs solvability?

A **monomorphism** m is left cancellable ($ma = mb \Rightarrow a = b$). An **epimorphism** e is right cancellable ($ae = be \Rightarrow a = b$). In vector spaces injectivity gives left cancellation pointwise. If $0 \neq v \in \ker m$, the two maps from $\mathbb{F}$ given by $s \mapsto sv$ and zero disprove it. Surjectivity gives right cancellation by testing on preimages. If the image is proper, the nonzero cokernel projection and zero agree after composition with e. The same cancellation characterizations hold in modules, using maps out of R instead of $\mathbb{F}$. They need care in other categories: the inclusion $\mathbb{Z} \to \mathbb{Q}$ is epic among commutative unital rings, since any two outgoing ring maps agreeing on integers agree on m/n by preservation of inverses.

A pair $A \xrightarrow{f} B \xrightarrow{g} C$ is **exact at** B if $\operatorname{im} f = \ker g$. Composition zero only proves one inclusion. In a short exact sequence $0 \to A \xrightarrow{i} B \xrightarrow{p} C \to 0$, i is injective, p surjective, and p identifies $B/i(A)$ with C: $[b] \mapsto p(b)$ is well defined, injective by exactness, and onto. Exactness describes the ambiguity in lifting c to b. A **section** $s : C \to B$ with $ps = 1$ makes one lift linear in c.

Splitting theorem. Such a section exists exactly when there is a compatible decomposition $B \simeq A \oplus C$. Given s, send $(a, c) \mapsto i(a) + s(c)$. For b put $c = p(b)$; $b - s(c)$ is in $\ker p = i(A)$ and determines a unique a. This gives a linear inverse. Conversely the second coordinate

injection in a compatible decomposition is a section. A retraction $r : B \to A$ with $ri = 1$ also suffices: $p|_{\ker r}$ is bijective since $b - ir(b)$ has the same image as b and $\ker r \cap i(A) = 0$. Its inverse is a section.

Over a field choose a basis of C, lift its members separately along p, then extend linearly. This proves every such sequence splits. Two sections differ by ih with $h : C \to A$, since $p(s_1 - s_2) = 0$. The choice can be substantial. Over $\mathbb{Z}$ consider

$$0 \to \mathbb{Z} \xrightarrow{\times 2} \mathbb{Z} \xrightarrow{\text{mod } 2} \mathbb{Z}/2 \to 0.$$

The even integers are exactly the kernel. A section would send $[1]$ to an odd integer n satisfying $2n = s(2[1]) = 0$, an impossibility. Exactness alone cannot produce independent coordinates compatible with the scalar action.

Why the first isomorphism theorem recurs. Define the coimage as the cokernel of k, and the image as the kernel of c. In vector spaces these are $V/\ker T$ and $\operatorname{im} T$. The unique induced map $[v] \mapsto Tv$ is onto by definition and has zero kernel since $Tv = 0$ gives $[v] = 0$. Thus T factors canonically as quotient, isomorphism, inclusion. An **abelian category** abstracts this situation: morphisms can be added as abelian groups, composition distributes, finite biproducts and kernels and cokernels exist, and the canonical coimage-to-image map is always invertible. Modules over a ring satisfy these conditions even when sequences fail to split. Exactness and splitting express different properties.

24.4 Session 4 — Pulling measurements back

An output measurement $\lambda : W \to \mathbb{F}$ gives an input measurement without solving an inverse problem: apply T and then λ. In general, given $a : V' \to V$, $b : W \to W'$, define

$$\operatorname{Hom}(V, W) \to \operatorname{Hom}(V', W'), \qquad h \mapsto bha.$$

Postcomposition moves with the output arrow (covariance); precomposition moves against the input arrow (contravariance). Both operations preserve addition and commute with one another by associativity. Setting $W = \mathbb{F}$ and fixing the output gives $T^\vee \lambda = \lambda T$. Evaluating on v gives $(ST)^\vee \mu(v) = \mu(STv) = (T^\vee S^\vee \mu)(v)$, and $1^\vee = 1$. In dual bases $(T^\vee \epsilon^j)(e_i) = \epsilon^j(Te_i) = A_{ji}$, so the dual matrix is A^T over any field, including $\mathbb{C}$.

24.4.1 Evaluation and its limitations

A vector determines a way to read every measurement: $\iota_V(v)(\lambda) = \lambda(v)$. This is linear in λ and in v. If $v \neq 0$, extend v to a basis and take its coordinate functional; thus ι_V is injective. In

finite dimension the dual basis gives $\dim V^{\vee\vee} = \dim V$, hence bijectivity. For any $T : V \to W$ we have a commutative square:

$$\begin{array}{ccc} V & \xrightarrow{T} & W \\ {\scriptstyle \iota_V}\downarrow & & \downarrow{\scriptstyle \iota_W} \\ V^{\vee\vee} & \xrightarrow{T^{\vee\vee}} & W^{\vee\vee}. \end{array}$$

Indeed both routes evaluated at $\mu \in W^\vee$ give $\mu(Tv)$. This proof needs no finiteness assumption. Finiteness is used to turn the vertical arrows into isomorphisms.

For a concrete failure of surjectivity take $V = \mathbb{F}^{(\mathbb{N})}$, finite sequences extended by zeros. Its dual consists of **all** sequences, acting by finite dot products. On the subspace of eventually constant sequences define $L(a)$ to be their eventual value; extend L linearly to the entire dual by basis extension. If $L = \iota_V(v)$, testing on each coordinate sequence forces every $v_i = 0$. But $L(1, 1, ...) = 1$. This double-dual element is invisible to every candidate vector.

What would identify V with $V^\vee$? A basis supplies the bilinear pairing $\sum x_i y_i$ in those coordinates; changing the basis generally changes the pairing. Even invariance under all changes of basis is too much to demand over $\mathbb{R}$: invariance of a bilinear form B under $2I$ gives $B(v, w) = 4B(v, w)$, hence $B = 0$. An inner product supplies extra data. In finite dimension, or for the **continuous** dual of a Hilbert space by Riesz representation, $R_V(y) = \langle \cdot\, , y\rangle$ identifies the two spaces, conjugate-linearly over $\mathbb{C}$. The identity $T^\vee R_W = R_V T^*$ follows by evaluating on x. In Hilbert spaces take T bounded and restrict to continuous functionals. This is how A^H enters in orthonormal coordinates. Reversing arrows in a category, pulling back scalar measurements, and forming a Hilbert adjoint are three distinct operations linked by these identities.

24.4.2 Exactness seen by measurements

Dualizing a short exact sequence gives

$$0 \to C^\vee \xrightarrow{p^\vee} B^\vee \xrightarrow{i^\vee} A^\vee \to 0.$$

Surjectivity of p makes $p^\vee$ injective. A measurement on B vanishing on $i(A)$ factors uniquely through p, giving exactness in the middle and $(B/i(A))^\vee \simeq i(A)^0$. Every measurement on A extends to B by making it zero on a chosen complement. This proves surjectivity at the right, including in algebraic infinite dimension under basis extension. The set of extensions is canonical, even though an individual extension can require a choice.

For modules $\mathrm{Hom}_R(-, M)$ always gives injectivity at the left and exactness in the middle by the same factorization. Extension can fail. Applying $\mathrm{Hom}_{\mathbb{Z}}(-, \mathbb{Z})$ to the multiplication-by-two sequence gives restriction represented by multiplication by two on $\mathbb{Z}$, which misses 1. Splitting of the original sequence would guarantee extension for every M.

24.5 Session 5 — When constructions respect changes of data

Consider computing a kernel before and after changing a linear system. The relevant input is a map, and a change consists of a commuting square $bT = T'a$. For $v \in \ker T$, $T'av = bTv = 0$. Thus a restricts to a map of kernels. Restrictions of identities are identities, and restrictions compose. This is more informative than the bare association $T \mapsto \ker T$: it specifies exactly which changes can be transported.

A **functor** $F : \mathcal{C} \to \mathcal{D}$ assigns an object FX to each object and a morphism $Ff : FX \to FY$ to each $f : X \to Y$, preserving identities and composition. A contravariant functor is an ordinary functor from $\mathcal{C}^{\mathrm{op}}$. The kernel example has the *arrow category* as source: its objects are linear maps, its arrows the squares just used, and squares compose by composing their two sides. Forgetting scalars gives a functor to abelian groups; forgetting the addition too gives one to sets. Both keep the underlying function. Fixing X gives the covariant functor $\mathrm{Hom}(X, -)$ and fixing Y gives the contravariant $\mathrm{Hom}(-, Y)$. The dual functor is the latter at $Y = \mathbb{F}$, keeping the vector-space structure on the outputs; applying it twice restores the direction.

For two functors $F, G : \mathcal{C} \to \mathcal{D}$, a **natural transformation** is a family $\eta_X : FX \to GX$ satisfying $Gf\,\eta_X = \eta_Y Ff$ for every $f : X \to Y$. Think of two ways to transport an answer: translate it into the other description first, or first change the input. Naturality says these give identical results. If each η_X is invertible, its inverses are natural as well (multiply the identity on both sides by the inverses). This is a natural isomorphism. The evaluation square proves that ι is a natural transformation everywhere and a natural isomorphism on finite-dimensional spaces.

In contrast, an ordered basis B gives a coordinate isomorphism $c_B : V \to \mathbb{F}^n$. Replacing B by BS changes c_B to $S^{-1}c_B$. If one tried to keep coordinate outputs fixed under *all* automorphisms S of V, the condition $c_B S = c_B$ would force $S = 1$. Transporting the basis together with the space makes coordinates meaningful again. To test a claim of canonicity, specify the input data, the permitted maps, and the equation they must satisfy.

24.5.1 Why a mapping problem can have an object as its answer

Fix V, U. The rule $H(W) = \{f : V \to W \mid f(U) = 0\}$ transports a solution forward by postcomposition. Session 2 identified it with $\mathrm{Hom}(V/U, W)$. This identification is natural in W: factoring bf gives $b\bar{f}$, since they agree after composition with q. A functor of this form is called **represented** by its source object. Its universal element is the solution corresponding to the identity, here q.

There is a short reason the identity contains so much information. Given a covariant set-valued functor H and natural maps $\Phi_X : \mathrm{Hom}(Q, X) \to H(X)$, put $u = \Phi_Q(1_Q)$. Naturality along $g : Q \to X$ forces $\Phi_X(g) = H(g)(u)$. Conversely every $u \in H(Q)$ gives such a family because $H(bg) = H(b)H(g)$. These operations are inverse. This is the covariant form of the **Yoneda lemma** for a category with sets of morphisms. It gives a practical proof strategy: construct

one universal element, then show each solution is its unique transport. The tensor product will use exactly this strategy with a bilinear universal element.

24.6 Session 6 — A linear operator is a polynomial action

24.6.1 Why enlarge the scalars?

Suppose an algorithm is allowed to add vectors, scale them, and repeatedly apply one fixed operator T. From a starting vector v it can build exactly $p(T)v$ as p ranges over $\mathbb{F}[t]$. We can record all three permitted operations in one scalar action. For a ring R, a left action on an abelian group satisfies

$$(r+s)v = rv + sv, \quad r(v+w) = rv + rw, \quad (rs)v = r(sv), \quad 1v = v.$$

These rules include ordinary vector spaces, abelian groups with integer scalars, R acting on itself by multiplication, and $M_n(\mathbb{F})$ acting on column vectors. A module can have torsion: a nonzero scalar can kill a nonzero element, as $2[1] = 0$ in $\mathbb{Z}/2$. Such a module cannot have a basis over $\mathbb{Z}$, since a free abelian group has no nonzero torsion elements.

Set $p(t)v = p(T)v$, where $p(T) = \sum_j a_j T^j$ and $T^0 = I$. Polynomial evaluation preserves sums, products and identity: for products expand and use $T^j T^k = T^{j+k}$. Each $p(T)$ is linear. These facts prove every module axiom. Conversely, an $\mathbb{F}[t]$-module restricts to a vector space using constant polynomials, and the action of t recovers T. A linear map L between two such spaces is a module map precisely when $LT = SL$: necessity follows by taking $p = t$, sufficiency by induction on powers and linearity. An invertible intertwiner is exactly a similarity of operators.

Invariant-subspace theorem. An $\mathbb{F}$-subspace U is a submodule iff $T(U) \subseteq U$. Closure under the action implies closure under t; conversely invariance under T gives invariance under its powers and linear combinations. In particular T induces $\bar{T}[v] = [Tv]$ on V/U: changing the representative adds $Tu \in U$. A section of the quotient is a *module* section precisely when it also intertwines $\bar{T}$ and T. Its image is then an invariant complement.

24.6.2 Cyclic modules, Krylov spaces and annihilators

For $v \neq 0$ consider the module map $\phi_v : \mathbb{F}[t] \to V$, $p \mapsto p(T)v$. Its image is the cyclic submodule generated by v. Its kernel is an ideal. Every nonzero ideal of $\mathbb{F}[t]$ has a monic generator: pick a nonzero polynomial of least degree, scale it to be monic and divide any ideal member by it; the remainder lies in the ideal and must vanish. Polynomial division uses the field assumption. A commutative integral domain in which all ideals have a single generator is called a **principal ideal domain** (PID).

In finite dimension some powers $v, Tv, \dots, T^n v$ are dependent. Thus the kernel has a unique monic generator m_v, the vector minimal polynomial. The first isomorphism theorem gives

$$\mathbb{F}[t]v \simeq \mathbb{F}[t]/(m_v).$$

If $\deg m_v = d$, residues $1, \dots, t^{d-1}$ are a basis by polynomial division and minimality. Hence $\dim \mathcal{K}_j(T, v) = \min(j, d)$ for $j \geq 1$. The first dependent Krylov column yields the coefficients of m_v, and the space stops growing because the dependence expresses $T^d v$ in terms of earlier powers. In this basis the matrix is a companion matrix: it shifts all but the last basis vector and the last column is the recurrence.

The operator annihilator is the intersection of the vector annihilators. Testing on a basis e_i suffices, so its monic generator m_T is $\operatorname{lcm}_i m_{e_i}$. Consequently $p(T) = 0$ iff $m_T \mid p$. Two filters with equal residues modulo m_T act identically on *all* vectors. On one starting vector it suffices to reduce modulo m_v, which can have much smaller degree.

24.6.3 A two-dimensional extension that refuses to split

Let $T = \left(\begin{smallmatrix} 2 & 1 \\ 0 & 2 \end{smallmatrix}\right)$ on $\mathbb{R}^2$. Then $(T - 2I)e_2 = e_1$ and $(T - 2I)e_1 = 0$. The vector e_2 generates the whole space with $m_{e_2} = (t-2)^2$; e_1 generates a line killed by $t - 2$. The quotient by that line is also killed by $t - 2$. An invariant complementary line would be another eigenline, yet $\ker(T - 2I) = \mathbb{R}e_1$. Thus the extension splits linearly and fails to split as a polynomial module. Two copies of the same one-dimensional factor can assemble into a Jordan block. The coupling is the single off-diagonal entry.

24.6.4 Coprime annihilators give genuine projectors

Suppose $m_T = ab$ with a, b monic and coprime. The Euclidean algorithm gives $ua + vb = 1$. Put $P_a = v(T)b(T)$, $P_b = u(T)a(T)$. Then $P_a + P_b = I$ and $P_a P_b = P_b P_a = 0$ because ab annihilates T. Multiplication by P_a proves $P_a^2 = P_a$. Its image lies in $\ker a(T)$ and it is the identity there because P_b vanishes there. Similarly $\operatorname{im} P_b = \ker b(T)$. Thus

$$V = \ker a(T) \oplus \ker b(T).$$

The projectors commute with T and are independent of the Bézout coefficients: P_a is prescribed as identity on one summand and zero on the other. Equivalently, the residue of vb is 1 modulo a and 0 modulo b. This proves the two-factor **Chinese remainder theorem** here: the residue map $\mathbb{F}[t]/(ab) \to \mathbb{F}[t]/(a) \times \mathbb{F}[t]/(b)$ has zero kernel by coprimality, and (r, s) lifts to $rvb + sua$.

Iterating for $m_T = \prod_i p_i^{d_i}$ gives the **primary decomposition** into $\ker p_i(T)^{d_i}$ for distinct monic irreducibles p_i. A linear factor gives a generalized eigenspace; $\ker(T - \lambda I)$ itself is the eigenspace. If all factors are distinct and linear, each primary space is an eigenspace, hence

an eigenbasis exists. Conversely for a diagonal matrix a polynomial vanishes iff it vanishes at every distinct diagonal entry, so the minimal polynomial is the product of their distinct linear factors. This proves the diagonalizability criterion over the stated field. For simple factors the projectors use ordinary Lagrange interpolation, $P_i = \prod_{j \neq i}(T - \lambda_j I)/(\lambda_i - \lambda_j)$. A real plane rotation by 90° has $m_T = t^2 + 1$, square-free but irreducible over $\mathbb{R}$: the field matters.

Three-dimensional computation. Take the matrix from the eigenspace example

$$A = \begin{pmatrix} 2 & 0 & -3 \\ 1 & -1 & -1 \\ 0 & 0 & -1 \end{pmatrix}.$$

Its 2-eigenspace is $\operatorname{span}(3, 1, 0)$ and its -1-eigenspace is $\{(x, y, x)\}$. They span $\mathbb{R}^3$, so $m_A = (t - 2)(t + 1)$. For any $x = (x_1, x_2, x_3)$, the projector $(A + I)/3$ returns $(x_1 - x_3)(1, 1/3, 0)$. Subtracting this from x leaves first and third coordinates equal. It projects *along the other eigenspace*, which explains its oblique geometry. A single vector in either eigenspace has a degree-one annihilator; a vector with both components nonzero needs both factors.

24.6.5 Why Jordan blocks arise, and what the minimal polynomial omits

Structure theorem used without proof. A finitely generated torsion module over a PID is a finite direct sum of cyclic modules $R/(d_j)$, with nonunit invariant factors $d_1 \mid \cdots \mid d_r$, unique up to multiplication by units. Finite-dimensional V is generated by any vector-space basis and is torsion under $\mathbb{F}[t]$ by the annihilator argument above. A proof of the general theorem uses reduction of a presentation matrix to Smith normal form; no claim below requires assuming that every invariant subspace splits.

The cyclic and coprime-factor results already proved split these summands into prime powers. When the polynomial splits over $\mathbb{F}$, each piece is $\mathbb{F}[t]/((t - \lambda)^k)$. Write $u = t - \lambda$ and order the basis as $u^{k-1}, \ldots, u, 1$. Multiplication by t acts as λI plus a backward shift killing the first vector. Its matrix is exactly one Jordan block. This derives Jordan form from the stated theorem. The largest block at λ gives the exponent in m_T; smaller blocks can be present. More precisely a length-s block contributes $\min(j, s)$ to $\dim \ker(T - \lambda I)^j$, and blocks at other eigenvalues contribute zero. Successive differences count blocks of length at least j. These counts recover information absent from m_T.

Four-dimensional computation. Let $A = J_2(2) \oplus [2] \oplus [-1]$. Then $\chi_A = (t - 2)^3(t + 1)$ and $m_A = (t - 2)^2(t + 1)$. The 2-eigenspace is spanned by e_1, e_3 and its primary space also contains e_2. To project onto the -1 line, take $p(t) = (t - 2)^2/9$. It is zero modulo $(t - 2)^2$ and equals 1 at -1. Hence $p(A) = \operatorname{diag}(0, 0, 0, 1)$. Merely imposing $p(2) = 0$ would leave a term proportional to $p'(2)$ on the Jordan block: $p(2I + N) = p(2)I + p'(2)N$ when $N^2 = 0$. Repeated factors require interpolation with multiplicity. This explains a concrete difference between filtering a normal matrix and filtering a defective one.

24.7 Session 7 — Turning bilinear rules into linear maps

The product $B(x,y) = xy$ on $\mathbb{R}\times\mathbb{R}$ is linear in each argument separately. Joint linearity would imply $B(1,1) = B(1,0) + B(0,1) = 0$, whereas $B(1,1) = 1$. In fact *every* jointly linear bilinear map is zero by that argument. The direct sum therefore solves the wrong extension problem. We need a space generated by symbols for pairs, in which a linear map can reproduce exactly the separate linearity rules.

Take the free vector space L on the set $V\times W$: elements are finite formal sums of basis symbols $[v,w]$. Quotient by the span of the relations

$$[av+bv',w] - a[v,w] - b[v',w], \qquad [v,aw+bw'] - a[v,w] - b[v,w'].$$

Write the resulting space $V\otimes W$ and the class of $[v,w]$ as $v\otimes w$. The defining map $\tau(v,w) = v\otimes w$ is bilinear by construction. Given bilinear $B : V\times W \to Z$, extend $[v,w]\mapsto B(v,w)$ linearly to L. It kills the displayed relations, so quotient factorization gives a unique $\tilde{B}$ on $V\otimes W$. Conversely any linear h gives a bilinear $h\tau$. Since the pure tensors span, these operations are inverse:

$$\mathrm{Bil}(V,W;Z) \simeq \mathrm{Hom}(V\otimes W, Z).$$

The isomorphism is linear and natural in Z by postcomposition. The proof also constructs a tensor product over a commutative ring using free modules and the corresponding relations with ring scalars.

Coordinates and rank. Given bases e_i, f_j, bilinearity expands $v\otimes w = \sum_{ij} v_i w_j e_i\otimes f_j$. These elementary tensors span. For independence apply the functional induced by $(v,w)\mapsto e^k(v)f^\ell(w)$: it returns precisely coefficient (k,ℓ). Thus dimensions multiply in finite dimension. In $\mathbb{R}^2\otimes\mathbb{R}^2$, the coefficients of a pure tensor form the rank-at-most-one matrix vw^T. The tensor $e_1\otimes e_1 + e_2\otimes e_2$ has coefficient matrix I_2 and cannot be expressed as one pure tensor. Arbitrary finite sums are essential.

Maps and Kronecker matrices. For $A : V\to V'$, $B : W\to W'$, the bilinear rule $(v,w)\mapsto Av\otimes Bw$ gives $A\otimes B$. Its (i,j) input basis column has coefficients $A_{ki}B_{\ell j}$. With the second index varying fastest the matrix is the block array $(A_{ki}B)_{ki}$, the Kronecker product. On each pure tensor $(C\otimes D)(A\otimes B) = CA\otimes DB$; spanning proves the identity everywhere. The abstract map is independent of ordering, its array depends on ordering.

Why outer products build operators. The canonical map $\Phi : V^\vee\otimes W\to \mathrm{Hom}(V,W)$ sends $\lambda\otimes w$ to $x\mapsto \lambda(x)w$. If V is finite dimensional, an inverse sends T to $\sum_i e^i\otimes T(e_i)$. Applying the two composites to x and to $\lambda\otimes w$ respectively gives $T(x)$ and $\sum_i \lambda(e_i)e^i\otimes w = \lambda\otimes w$. Thus the inverse is independent of the basis used to compute it. In arbitrary dimension the image is exactly the finite-rank maps: every finite sum has finite-dimensional image, and reading coordinates in a basis for a finite-dimensional range supplies the reverse inclusion. In particular the infinite-dimensional identity lies outside the image. Likewise $V^\vee\otimes W^\vee\to (V\otimes W)^\vee$ pairs elementary tensors by multiplication of evaluations; it is an isomorphism for

finite-dimensional factors by the basis calculation. The universal description of bilinear forms as $(V \otimes W)^\vee$ holds in every dimension.

A discretization you can assemble without a full matrix. On a rectangular grid with separable boundary conditions take one-dimensional second-difference matrices L_x, L_y. Write an array U with the y index down columns, and stack columns into $\operatorname{vec} U$. Acting on each x or y fiber gives

$$(I_x \otimes L_y + L_x \otimes I_y) \operatorname{vec} U = \operatorname{vec}(L_y U + U L_x^T).$$

Check this first on $U = wv^T$, for which $\operatorname{vec} U = v \otimes w$, then use linearity. The tensor product explains why independent coordinate actions extend to general grid data. Applying the two small matrices to arrays is the computational benefit. A general mesh or variable coefficient need not have this two-term separable representation.

24.8 Session 8 — When zero composition leaves a hole

Orient the triangle edges $a : 1 \to 2$, $b : 2 \to 3$, $c : 3 \to 1$. Regard edge flows as elements of $C_1 = \mathbb{R}^3$ and vertex loads as $C_0 = \mathbb{R}^3$. Send an edge to head minus tail:

$$D_1 = \begin{pmatrix} -1 & 0 & 1 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{pmatrix}.$$

Then $D_1(x, y, z) = 0$ exactly when $x = y = z$. A circulating flow has zero net load everywhere. The image is the plane of zero-sum vertex loads: each column sums to zero, and the first two columns are independent. The graph alone has no faces, so it contains a cycle that is the boundary of no available face.

Now fill the triangle. Take $C_2 = \mathbb{R}$ and send its oriented face to $a + b + c$, so $D_2 = (1, 1, 1)^T$. Then $D_1 D_2 = 0$. The circulation is still present, but now every circulation bounds a face multiple. The distinction cannot be seen from D_1 alone.

A **chain complex** consists of maps $d_k : C_k \to C_{k-1}$ with $d_k d_{k+1} = 0$. Every boundary $d_{k+1}x$ therefore belongs to $Z_k = \ker d_k$, the cycles. Put $B_k = \operatorname{im} d_{k+1}$. Two cycles define the same homology class when their difference bounds:

$$H_k = Z_k / B_k, \qquad \dim H_k = \dim C_k - \operatorname{rank} d_k - \operatorname{rank} d_{k+1}$$

in finite-dimensional vector spaces. The dimension formula follows by rank-nullity and quotient dimension. Exactness at C_k is equivalent to $H_k = 0$. Taking $d_0 = 0$, the triangle graph has dimensions $(\dim H_0, \dim H_1) = (1, 1)$. Filling it gives $(1, 0, 0)$ in degrees zero through two, since D_2 is injective. The surviving H_0 measures total load; more generally each connected component contributes one independent total since differences of vertices along paths are edge boundaries.

24.8.1 From chains to differential operators

Transposing the incidence matrix sends vertex potentials to oriented edge differences. A discrete potential u gives $g = D_1^T u$. The sum around the face is $D_2^T g = 0$ since it telescopes. This is the direction of a **cochain** complex: the degree increases. With inner products, the adjoint of the gradient gives a divergence up to sign, and mass matrices encode the chosen inner products. The incidence identity itself only uses orientations.

For smooth scalar and vector fields on an open subset of $\mathbb{R}^3$, the gradient, curl and divergence are successive arrows with zero composites by equality of mixed partials. Their common generalization on differential forms is the exterior derivative d satisfying $d^2 = 0$, the **de Rham complex**. Exactness asks whether every closed field has a potential of the appropriate degree. On $\mathbb{R}^2 \setminus \{0\}$, $F = (-y, x)/(x^2 + y^2)$ has zero scalar curl by differentiation. Its circulation on $(\cos t, \sin t)$ is $\int_0^{2\pi} 1\, dt = 2\pi$, whereas a global gradient integrates to zero on a closed path. Smoothness alone cannot fill the missing point. On a star-shaped domain, for a smooth curl-free field and star center 0, set $u(x) = \int_0^1 F(tx) \cdot x\, dt$. Differentiating in x_i and using $\partial_i F_j = \partial_j F_i$ makes the integrand $\frac{d}{dt}(tF_i(tx))$, hence $\partial_i u = F_i(x)$. This proves exactness at this position under a concrete geometric hypothesis.

A compatible discretization chooses spaces and discrete derivatives forming a complex. If interpolation or projection maps Π_k satisfy $d_h \Pi_k = \Pi_{k+1} d$, differentiating interpolated data agrees with interpolating its derivative. For instance the difference of scalar values at an edge's endpoints equals the edge integral of its gradient. This is a commuting square with a numerical meaning. Stability estimates and approximation rates require additional analysis; the commuting equation alone establishes an algebraic identity.

24.9 Session 9 — Four problems seen through several constructions

24.9.1 A. Recovering a potential from measurements

Take $T = D_1^T$ from the triangle, so $Tu = (u_2 - u_1, u_3 - u_2, u_1 - u_3)$. Let $K = \mathbb{R}(1, 1, 1)$ and $S = \{g : \sum_i g_i = 0\}$. A lift of $(a, b, -a - b)$ is $(0, a, a + b)$; every other lift adds a constant. Thus the two short exact sequences are

$$0 \to K \to \mathbb{R}^3 \xrightarrow{T} S \to 0, \qquad 0 \to S \to \mathbb{R}^3 \xrightarrow{\sum} \mathbb{R} \to 0.$$

In the first sequence T is corestricted to S. The induced quotient isomorphism $[u] \mapsto Tu$ is canonical. In the second, the sum gives explicit coordinates on the cokernel. On dualizing, $0 \to S^\vee \to (\mathbb{R}^3)^\vee \to K^\vee \to 0$ describes the potential measurements insensitive to constants as exactly the pulled-back edge measurements. The second dual sequence says that two ambient edge measurements restricting equally to S differ by a multiple of the sum functional. Choosing $u_1 = 0$ selects a section; choosing mean-zero potentials selects another, singled out by

the Euclidean geometry and its symmetry under relabeling. Both recover the same quotient class.

24.9.2 B. What a Krylov process can recover

Return to $A = J_2(2) \oplus [2] \oplus [-1]$ and take $v = e_2 + e_4$. Its two nonzero primary components have annihilators $(t-2)^2$ and $t+1$. A polynomial kills their sum iff it kills each component, so $m_v = m_A$ and the Krylov dimension stabilizes at three. Yet e_3 can never appear: every polynomial image lies in $\operatorname{span}(e_1, e_2, e_4)$. Knowing the full minimal polynomial does not make a single starting vector cyclic for the entire space. A block with a second start e_3 supplies the missing direction. The CRT filter $p(A)$ from Session 6 extracts e_4 from v in one polynomial application.

For a Hermitian operator with orthonormal eigenbasis, $q(A)v = \sum_i q(\lambda_i) c_i u_i$. If the wanted and unwanted spectral sets have $\min_{\text{wanted}} |q| = \alpha > 0$ and $\max_{\text{unwanted}} |q| = \beta$, then the ratio of unwanted to wanted component norms after filtering is at most β/α times its initial value, provided the wanted component is nonzero. This follows by squaring and summing coefficients in each orthogonal subspace. A Chebyshev filter exploits small values on an unwanted interval and large values on the desired part. Exact CRT interpolation presupposes detailed spectral data; interval filtering can use bounds. Both use the polynomial action, while their information requirements and numerical behavior differ.

24.9.3 C. A mesh with one remaining circulation

In a square, orient edges $12, 23, 34, 41$ cyclically and add diagonal 13. Attach only face 123. The boundary vectors of the two triangular loops in edge coordinates are $z = (1, 1, 0, 0, -1)$ and $w = (0, 0, 1, 1, 1)$. The incidence matrix and face column are

$$D = \begin{pmatrix} -1 & 0 & 0 & 1 & -1 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 1 \\ 0 & 0 & 1 & -1 & 0 \end{pmatrix}, \qquad E = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ -1 \end{pmatrix},$$

where the columns of D follow the stated edge order. Direct multiplication gives $Dz = Dw = 0$ when loops are columns. The first three edge columns are independent (a path on four vertices), and zero column sums bound the rank by three. Thus z, w form a basis of the two-dimensional kernel. Quotienting by $\operatorname{im} E$ leaves one class $[w]$; the outer boundary $z + w$ represents that same class. A zero face-sum constraint on edge differences enforces local compatibility on the filled triangle. The remaining loop integral tests a second, global compatibility condition for a vertex potential.

24.9.4 D. From a quotient to an operator equation

A real weak problem seeks $u \in V$ with $a(u,v) = \ell(v)$ for all $v \in V$. A bounded bilinear form gives a bounded operator $\mathcal{A} : V \to V'$ by $(\mathcal{A}u)(v) = a(u,v)$, where the prime denotes continuous dual. For the pure Neumann energy $a(u,v) = \int_\Omega \nabla u \cdot \nabla v$ on $H^1(\Omega)$ of a bounded connected Lipschitz domain, constants are killed in both slots. Necessarily $\ell(1) = 0$. Then the form and load descend to the quotient by constants, and its elements describe potentials up to an additive constant. Existence and bounded dependence use a Poincaré inequality and coercivity on that quotient (analytical results used here without proof). In a Galerkin subspace V_h the residual $\ell - \mathcal{A}u_h$ vanishes on all test vectors in V_h: this is membership in an annihilator.

For a generalized eigenproblem $Kx = \lambda Mx$ with $K = K^H$ and M Hermitian positive definite, M identifies coefficient vectors with dual loads in matrix coordinates. $T = M^{-1}K$ is self-adjoint for $\langle x, y\rangle_M = y^H Mx$, since $MT = T^H M = K$. Polynomials in T preserve its eigenspaces, and orthogonal projectors use the M geometry. In particular for a full-column-rank basis Q of a subspace, $Q(Q^H MQ)^{-1}Q^H M$ gives the M-orthogonal projector: its output lies in the range of Q and its error is annihilated by $Q^H M$. Applying T computationally can be realized by solving with M.

24.9.5 Bridge to F1 — Where topology changes the answer

For a normed space X the quotient seminorm is $\|[x]\| = \inf_{u\in U} \|x - u\|$. It vanishes precisely when $x \in \overline{U}$, so a normed quotient needs U closed; a Banach space modulo a closed subspace is Banach. If a bounded f kills U, continuity makes it kill $\overline{U}$, and $\|\bar{f}[x]\| \le \|f\| \inf_{u\in U} \|x - u\|$ proves bounded quotient factorization. Thus the cokernel of a bounded T in Banach spaces is the quotient by the **closure** of its range.

For example $D : \ell^2 \to \ell^2$, $(Dx)_n = x_n/n$, is bounded and injective. Its range contains all finitely supported sequences, hence is dense, but misses $(1/n)$ because its preimage would be the constant sequence 1. Its Banach-space cokernel is zero although its algebraic cokernel is nonzero. Its inverse on the range is unbounded since $\|De_n\| = 1/n$. Kernels of bounded maps are always closed by continuity. Algebraic exactness alone gives no closed-range estimate.

A topological direct sum requires bounded coordinate projections. For a bounded idempotent P, $X = \operatorname{im} P \oplus \ker P$ with closed summands since the image is $\ker(I - P)$. Conversely a decomposition into two closed subspaces of a Banach space has bounded projections: the addition map from the Banach product onto X is a bounded bijection, so its inverse is bounded by the bounded inverse theorem. This argument assumes a complement exists. Hilbert spaces provide a bounded orthogonal projection onto each closed subspace; general Banach spaces need not. Finally, algebraic functionals need not be bounded: on the finite sequences with ℓ^2 norm the map $x \mapsto \sum n x_n$ sends the unit vectors to n. An algebraic dual construction, a continuous dual construction and a choice of geometric representative answer different questions.

24.10 Exercises

Try these in order with the solutions covered. For each factorization identify the arrow whose existence is claimed and the property that forces uniqueness.

1. **Quotients in stages.** For $U \subseteq W \subseteq V$, construct a canonical isomorphism $(V/U)/(W/U) \simeq V/W$ using universal properties. Derive the first isomorphism theorem as a factorization.
2. **A product that changes with its category.** Compute the product and coproduct of two singleton sets. For vector spaces derive the product and coproduct formulas from the biproduct identities alone. Explain where addition is used.
3. **Gluing measurements.** Given $\alpha : U \to X$, $\beta : W \to X$ on subspaces of V, determine exactly when there is a linear $h : U + W \to X$ extending both.
4. **A cokernel compatibility test.** For $T(x, y, z) = (x + y, 2x + 2y, z)$, compute kernel, image and cokernel. Give a compatible section on the image and all solutions for $(3, 6, 4)$.
5. **Splitting depends on the scalars.** Classify sections of $(x, y) \mapsto x$ on $\mathbb{R}^2$. Decide which sections are module maps when the domain operator is $(x, y) \mapsto (x, x+y)$ and the target operator is the identity. Compare reduction modulo n on $\mathbb{Z}$, $n > 1$.
6. **Dualizing the gradient.** With $Tu = (u_2 - u_1, u_3 - u_2, u_1 - u_3)$, pull back the measurement (a, b, c) and find all measurements pulling back to zero. Verify double-dual naturality on this example.
7. **A functor that detects obstructions.** A square $bT = T'a$ induces a map of cokernels. Construct it and verify functoriality and naturality of the cokernel projections.
8. **Krylov recurrence.** For $A = J_2(2) \oplus [-1]$ find m_v for $v = e_1$, e_2, $e_1 + e_3$, $e_2 + e_3$. Predict the first index of Krylov stagnation in each case.
9. **Interpolation with a repeated node.** Construct the projector onto the generalized 2-eigenspace for the operator in Exercise 8. Verify its action on the Jordan chain and explain why a degree-one filter cannot work.
10. **Reading block sizes.** A nilpotent N on a six-dimensional space has nullities $3, 5, 6$ for its first three powers. Determine the blocks, the minimal polynomial, and a module presentation as a sum of cyclic modules.
11. **A bilinear rule and its matrix.** Factor $B(v, w) = v_1 w_2 - v_2 w_1$ through $\mathbb{R}^2 \otimes \mathbb{R}^2$. Determine its kernel. Decide whether $e_1 \otimes e_2 - e_2 \otimes e_1$ is pure.
12. **The effect of a second face.** In the square example, attach face 134 with column w. Compute all homology dimensions. What happens if the second face instead duplicates the first with the same boundary?
13. **Moving homology classes.** Suppose maps $f_k : C_k \to C'_k$ commute with boundaries, $d'_k f_k = f_{k-1} d_k$. Construct their action on H_k, prove it is well defined, and check what inclusion of the triangle boundary into the filled triangle does in degrees zero and one.
14. **An analytic counterexample.** Show that any two bounded maps out of ℓ^2 into a normed space agreeing after composition with $D(x)_n = x_n/n$ agree everywhere. Why does the proof fail for arbitrary algebraic maps?

24.11 Worked solutions

1. The map $v \mapsto [v]_W$ kills U, so it factors uniquely through V/U. Its factor sends $[v]_U$ to $[v]_W$, kills exactly W/U, and is onto. Factoring once more gives $[[v]_U] \mapsto [v]_W$. An inverse sends $[v]_W$ to $[[v]_U]$ since changing v by $w \in W$ changes the middle class by an element of W/U. Both composites fix every represented class. For an arbitrary T, corestrict to $\operatorname{im} T$ and factor through $V/\ker T$. The factor is onto by corestriction and injective because $Tv = Tw$ means $v - w \in \ker T$. Uniqueness after composition with the quotient singles out this isomorphism.

2. The product is a singleton, since every pair of incoming maps is the sole pair of constant functions. The coproduct has two tagged points; arbitrary outgoing functions specify one value at each point independently. For a biproduct, incoming f, g induce $h = i_1 f + i_2 g$. Projecting returns f, g. Conversely any h' with these projections satisfies $h' = (i_1\pi_1 + i_2\pi_2)h' = h$. Outgoing a, b extend to $k = a\pi_1 + b\pi_2$, and $k' = k'(i_1\pi_1 + i_2\pi_2) = k$ for any other extension. Both constructions used addition of parallel maps and distributivity of composition.

3. Necessity follows by restricting h to $U \cap W$. For sufficiency define $H(u, w) = \alpha(u) + \beta(w)$ on the external sum. It kills every $(z, -z)$ for $z \in U \cap W$ precisely when the restrictions agree. These pairs form the entire kernel of addition, so H factors uniquely through $U + W$. This gluing property is called a pushout of the two inclusions from $U \cap W$: it allows independent rules once they agree on their shared inputs.

4. Vanishing requires $x = -y, z = 0$, giving kernel $\operatorname{span}(1, -1, 0)$. Outputs fill the plane $(a, 2a, b)$ because $(a, 0, b)$ is a lift. The functional $(r, s, t) \mapsto s - 2r$ is onto and kills exactly that plane, hence provides a coordinate on the cokernel. The lift just given is linear and is a section on the plane. All solutions are $(3, 0, 4) + h(1, -1, 0)$.

5. A section is $s_c(x) = (x, cx)$ with arbitrary $c \in \mathbb{R}$. Intertwining would require $(x, (1+c)x) = (x, cx)$ for every x, impossible at $x = 1$. In the integer example a section must send $[1]$ to some $k \equiv 1 \bmod n$ with $nk = 0$ in $\mathbb{Z}$. This forces $k = 0$, a contradiction. The two failures arise from preserving an additional operation: the operator action in the first example and integer addition in the second.

6. Expanding $a(u_2 - u_1) + b(u_3 - u_2) + c(u_1 - u_3)$ gives coefficients $(c - a, a - b, b - c)$. All are zero exactly when $a = b = c$. For $\mu = (a, b, c)$ and u, $T^{\vee\vee}\iota(u)$ evaluated at μ is $\iota(u)(T^\vee\mu) = (c-a)u_1 + (a-b)u_2 + (b-c)u_3$. The expansion shows this equals $\mu(Tu) = \iota(Tu)(\mu)$ for every measurement.

7. Define $\hat{b}([w]) = [bw]$. A difference Tx is sent to $bTx = T'ax$, which vanishes in the new quotient. This proves independence of representative. The identity square fixes all classes, and two successive squares send $[w]$ to $[b'bw]$, the action of their composite. Write q_T for the quotient; $\hat{b}q_T = q_{T'}b$ by evaluation at w. These are the naturality squares from the functor taking an arrow to its codomain to the cokernel functor.

8. On e_1, $t - 2$ suffices. On e_2, $t - 2$ produces e_1 and applying it again kills e_1, so $(t - 2)^2$ is minimal. A nonzero e_3 component contributes $t + 1$; the two coprime factors must both divide

any annihilator of a sum from the distinct primary subspaces. Thus the last two answers are $(t-2)(t+1)$ and $(t-2)^2(t+1)$. The first j with $\mathcal{K}_{j+1} = \mathcal{K}_j$ is respectively $1, 2, 2, 3$ by polynomial division.

9. Take $q(t) = 1 - (t-2)^2/9$. Then $q(-1) = 0$ and $q \equiv 1 \bmod (t-2)^2$. Acting on e_1, e_2 the squared term vanishes, while on e_3 it is $9I$. Thus $q(A) = \operatorname{diag}(1, 1, 0)$, whose square equals itself. A linear polynomial r satisfying $r(2) = 1, r(-1) = 0$ has slope $1/3$; it sends e_2 to $e_2 + e_1/3$ and fails to fix the entire primary space.

10. Differences from the initial nullity zero are $3, 2, 1$. These count blocks of length at least one, two, and three. Since all six dimensions have appeared by the third power there are none longer. Subtracting consecutive counts gives one block of each length $1, 2, 3$. The largest length gives $m_N = t^3$ and the module is $\mathbb{F}[t]/(t) \oplus \mathbb{F}[t]/(t^2) \oplus \mathbb{F}[t]/(t^3)$.

11. In pair order $11, 12, 21, 22$ the induced row is $(0, 1, -1, 0)$ by evaluation on elementary tensors. Its kernel consists of coefficient arrays with $c_{12} = c_{21}$, spanned by the two diagonal tensors and $e_1 \otimes e_2 + e_2 \otimes e_1$. The proposed difference has coefficient matrix with determinant 1 and hence rank two. A single pure tensor has rank at most one.

12. The independent columns z, w span $\ker D$, so $H_1 = 0$, and the face map is injective so $H_2 = 0$. Vertices and edges have not changed, giving $\dim H_0 = 1$. With duplicate columns the span has dimension one and $[w]$ survives in H_1. The difference of the two face generators spans the kernel in degree two. The dimensions are then $(1, 1, 1)$; the duplicate attachment is interpreted as a second two-cell, not a second copy of a simplex in an abstract simplicial complex.

13. For a cycle x, $d'_k f_k x = f_{k-1} d_k x = 0$, so $f_k x$ is a cycle. Replacing x by $x + d_{k+1} y$ changes its image by $d'_{k+1} f_{k+1} y$, a boundary. Hence $[x] \mapsto [f_k x]$ is well defined and linear. Identities and composition pass to classes. Under inclusion the three vertices and edges keep their names, so total load gives the identity on $H_0 \simeq \mathbb{R}$; the old generator of H_1 becomes a face boundary, giving the zero map.

14. If $gD = hD$, $g - h$ vanishes on the dense range. For any y, approximate it by its finite truncations, all in the range, and use continuity to get $(g - h)y = 0$. Thus D is epic among bounded maps even though it misses $(1/n)$. Algebraically the map to $\ell^2/\operatorname{im} D$ is nonzero and vanishes after D. It detects the missing class because algebraic factorization requires no continuity.

24.12 Computational laboratory — Exact identities in MATLAB

Use a single MATLAB script with Symbolic Math Toolbox. Each block continues the script. Enter the matrices as exact integers before division. The assertions test structural equations independently of the particular basis returned by a null-space routine. In a floating-point experiment a rank decision would also require a scale and tolerance.

24.12.1 1. Compute quotient coordinates and factor a map

```
T = sym([1 1 0; 2 2 0; 0 0 1]);
K = null(T);
[~, pivot] = rref(T);
B = T(:,pivot);                 % basis for the range
Q = sym([1 1 0; 0 0 1]);       % coordinates on the domain modulo K
C = sym([-2 1 0]);             % quotient coordinate on the codomain
S = sym([1 0; 0 0; 0 1]);      % lift a pair of quotient coordinates
assert(isequal(B*Q,T))
assert(isequal(Q*S,sym(eye(2))))
assert(isequal(Q*K,sym(zeros(2,1))))
assert(isequal(C*T,sym(zeros(1,3))))
assert(rank(B)==2 && rank(C)==1)
disp(K), disp(B)
```

The columns of B span the solvable loads. Rank plus $CB = 0$ proves $\operatorname{im} B = \ker C$. The matrices B, Q represent two different maps even though the factorization passes through the same coordinate space $\mathbb{F}^2$. Alter the lift S by adding a column of K to either column. Predict which assertion detects this change. None should: adding a kernel vector leaves quotient coordinates unchanged.

24.12.2 2. Separate primary spaces and measure Krylov growth

```
A = sym([2 1 0 0; 0 2 0 0; 0 0 2 0; 0 0 0 -1]);
I = sym(eye(4));
P = (A-2*I)^2/9;
assert(isequal(P*P,P) && isequal(P*A,A*P))
assert(isequal(P,diag(sym([0 0 0 1]))))
assert(isequal((A-2*I)^2*(A+I),sym(zeros(4))))
v = sym([0;1;0;1]);
Y = [v,A*v,A^2*v,A^3*v];
dims = arrayfun(@(j) double(rank(Y(:,1:j))),1:4);
assert(isequal(dims,[1 2 3 3]))
assert(isequal(Y*sym([4;0;-3;1]),sym(zeros(4,1))))
disp(P*v), disp(dims)
```

The last relation reads $A^3v - 3A^2v + 4v = 0$, the expansion of $(t-2)^2(t+1)$. The first three columns are independent, proving minimality on this vector. Repeating with e_3 gives

immediate stagnation. Using e_3 as an additional block start instead raises the final reachable dimension to four.

24.12.3 3. Identify tensor coordinates

```
L = sym([2 1; 0 3]);
M = sym([1 -1; 2 0]);
v = sym([1;2]); w = sym([3;-1]);
U = w*v.';
assert(isequal(kron(v,w),U(:)))
assert(isequal(kron(L,M)*kron(v,w),kron(L*v,M*w)))
Vgrid = M*U+U*L.';
Ksum = kron(sym(eye(2)),M)+kron(L,sym(eye(2)));
assert(isequal(Ksum*U(:),Vgrid(:)))
```

MATLAB stacks columns, so the second factor varies fastest. The dot transpose here implements a bilinear coordinate identity. Reversing the factor order without changing the vectorization convention would apply the operators along the wrong axes.

24.12.4 4. Count classes before and after attaching faces

```
D = sym([-1 0 0 1 -1; 1 -1 0 0 0; 0 1 -1 0 1; 0 0 1 -1 0]);
z = sym([1;1;0;0;-1]); w = sym([0;0;1;1;1]);
faces = {z,[z w],[z z]};
expected = [1 1 0; 1 0 0; 1 1 1];
for j = 1:numel(faces)
    E = faces{j};
    assert(isequal(D*E,sym(zeros(4,size(E,2)))))
    b = [size(D,1)-rank(D), size(D,2)-rank(D)-rank(E), ...
         size(E,2)-rank(E)];
    assert(isequal(double(b),expected(j,:)))
    disp(b)
end
```

Observe that $DE = 0$ succeeds in all three complexes. Only the ranks reveal whether a surviving cycle bounds. To test whether the particular cycle w survives, compare $\operatorname{rank}[E\ w]$ with $\operatorname{rank} E$. The class is zero exactly when the ranks are equal.

24.13 Final synthesis — Choosing the object that answers the question

Return to the potential problem. If measurements must ignore constants, the quotient is forced by how maps *leave* the potential space. If the question is which loads can be solved, the appropriate subspace is the image, and its inclusion describes which maps *land in* the solvable loads. If we need a test for failure, we project to the cokernel. Once a representative is wanted for plotting or numerical solution, a section enters. The input admits many sections; an inner product or a prescribed reference vertex can supply the missing choice. This one problem explains why universal constructions can be canonical even when the arrays produced by a solver depend on conventions.

The spectral problem adds another datum, T. Respecting that datum restricts the allowed maps to intertwiners. A vector-space splitting can thereby cease to be a splitting, as the Jordan chain shows. Conversely a relatively prime factorization gives more than a dimension count: it constructs actual intertwiners $p(T)$ selecting invariant components. A Krylov basis, an exact spectral projector and a polynomial filter use the same polynomial action. Their objectives are respectively to represent a cyclic space, to split a module, and to change the relative sizes of spectral components.

Structural concept	Concrete manifestation	Equation to reconstruct
Universal factorization	Descending a map to a quotient	$\bar{f}q = f$
Product	Joint output of several measurements	$\pi_j h = f_j$
Coproduct	Extending from independent inputs	$h i_j = f_j$
Biproduct	A finite direct sum supporting both operations	$\sum i_j \pi_j = I$
Kernel	Null space with its inclusion	$Th = 0 \Rightarrow h = k\bar{h}$ uniquely
Cokernel	Codomain modulo the range	$gT = 0 \Rightarrow g = \bar{g}c$ uniquely
Exactness	Every compatible datum at a position is produced upstream	$\operatorname{im} f = \ker g$
Splitting	Coherent choice of a lift	$ps = I$
Contravariant functor	Pullback of measurements	$(ST)^\vee = T^\vee S^\vee$
Natural transformation	Evaluation into the double dual	$T^{\vee\vee}\iota_V = \iota_W T$
Polynomial module	Operator acting as a scalar generator	$tv = Tv$
Tensor product	Linear representation of a bilinear rule	$\tilde{B}(v \otimes w) = B(v, w)$

Structural concept	Concrete manifestation	Equation to reconstruct
Chain complex	Successive boundary maps	$d_k d_{k+1} = 0$
Homology	Closed data modulo boundaries	$H_k = \ker d_k / \operatorname{im} d_{k+1}$

Logical dependencies. Composition and equality of paths suffice to formulate products and coproducts. Zero arrows allow one to pose kernel and cokernel problems. The vector-space theory of exactness uses images and kernels; homology also needs their quotient. Functoriality asks how any of these constructions responds to changing its input, so it runs across the branches. Polynomial modules require a scalar action and polynomial arithmetic; they can be studied without homology. Tensor products begin with bilinear maps; their construction here uses a free space and quotient. Analytic applications add bounds, closedness and completeness at the point where passage to limits is required.

For further study, abstract algebra asks which features of $\mathbb{F}[t]$ made classification possible. Category theory asks which mapping problems admit representing objects. Homological algebra studies how failure of exactness propagates through constructions such as Hom. Functional analysis asks whether a factor, inverse or projection is bounded. Numerical mathematics asks how accurately and economically these maps can be applied when data are approximate. Each direction can be entered through an explicit example already worked above.

24.14 Mastery assessment

Use a blank page for the proofs. For computations, write down the space containing the answer before choosing coordinates.

- ☐ Give the objects, arrows and composition of two categories on familiar data. Exhibit a bijection that fails to give a structural isomorphism and explain why.
- ☐ Prove quotient factorization from representatives, then prove uniqueness of the representing pair using only arrows.
- ☐ Recover pairing and extension maps from biproduct identities. Explain the finite-support condition for an infinite coproduct.
- ☐ Prove the kernel and cokernel universal properties in opposite directions. Use each on a concrete linear system.
- ☐ Given a zero composite, decide what else must be checked for exactness. Construct a section when possible and distinguish its choice from the quotient isomorphism.
- ☐ Produce two nonsplitting examples, one over the integers and one involving an invariant subspace. Identify the operation a candidate section fails to preserve.
- ☐ Derive the dual matrix in dual bases, the double-dual naturality square, and the role of conjugation in the Riesz map.

- ☐ State a functor's action on morphisms and check its laws. Prove the map induced on cokernels by a commuting square is well defined.
- ☐ Reconstruct the universal-element argument by evaluating a natural family on an identity map.
- ☐ Derive the polynomial module axioms and characterize intertwiners, invariant subspaces and quotient operators.
- ☐ Starting from a Krylov dependence, find a vector annihilator and prove minimality. Explain why an operator minimal polynomial alone cannot count all Jordan blocks.
- ☐ Construct complementary polynomial projectors by Bézout's identity. Explain the extra interpolation condition at a defective eigenvalue.
- ☐ Construct a tensor product from bilinearity relations, prove the basis theorem and derive a Kronecker matrix in a declared ordering.
- ☐ Compute homology from two successive boundary matrices and determine whether a specific cycle represents zero. Interpret the dual compatibility condition on potentials.
- ☐ For a bounded operator with dense proper range, compare algebraic solvability, topological cokernel and bounded inversion.
- ☐ In a new numerical example, identify which parts use only uniqueness and factorization, which need an inner product, and which require an estimate.