

Lecture Notes for Mat-inf 4130, 2016

Tom Lyche

August 15, 2016

Contents

Preface	ix
0 A Short Review of Linear Algebra	1
0.1 Notation	1
0.2 Vector Spaces and Subspaces	4
0.2.1 Linear independence and bases	7
0.2.2 Subspaces	8
0.2.3 The vector spaces \mathbb{R}^n and \mathbb{C}^n	11
0.3 Linear Systems	11
0.3.1 Basic properties	12
0.3.2 The inverse matrix	14
0.4 Determinants	16
0.5 Eigenvalues, eigenvectors and eigenpairs	22
0.6 Algorithms and Numerical Stability	24
I Linear Systems	27
1 Diagonally dominant tridiagonal matrices; three examples	29
1.1 Cubic Spline Interpolation	29
1.1.1 Polynomial interpolation	30
1.1.2 Piecewise linear and cubic spline interpolation . .	30
1.1.3 Give me a moment	33
1.1.4 LU Factorization of a Tridiagonal System	36
1.1.5 Exercises for section 1.1	39
1.1.6 Computer exercises for section 1.1	41
1.2 A two point boundary value problem	43
1.2.1 Diagonal dominance	44
1.2.2 Exercises for section 1.2	45
1.3 An Eigenvalue Problem	47

1.3.1	The Buckling of a Beam	47
1.3.2	The eigenpairs of the 1D test matrix	48
1.4	Block Multiplication and Triangular Matrices	50
1.4.1	Block multiplication	50
1.4.2	Triangular matrices	53
1.4.3	Exercises for section 1.4	55
1.5	Review Questions	56
2	Gaussian elimination and LU Factorizations	57
2.1	Gaussian Elimination and LU-factorization	58
2.1.1	3 by 3 example	58
2.1.2	Gauss and LU	59
2.2	Banded Triangular Systems	62
2.2.1	Algorithms for triangular systems	62
2.2.2	Counting operations	66
2.3	The LU and LDU Factorizations	67
2.3.1	Existence and uniqueness	68
2.3.2	Block LU factorization	73
2.4	The PLU-Factorization	74
2.4.1	Pivoting	74
2.4.2	Permutation matrices	75
2.4.3	Pivot strategies	77
2.5	Review Questions	79
3	LDL* Factorization and Positive definite Matrices	81
3.1	The LDL* factorization	81
3.2	Positive Definite and Semidefinite Matrices	83
3.2.1	The Cholesky factorization.	85
3.2.2	Positive definite and positive semidefinite criteria .	87
3.3	Semi-Cholesky factorization of a banded matrix	90
3.4	The non-symmetric real case	94
3.5	Review Questions	95
4	Orthonormal and Unitary Transformations	97
4.1	Inner products, orthogonality and unitary matrices	98
4.1.1	Real and complex inner products	98
4.1.2	Orthogonality	100
4.1.3	Unitary and orthogonal matrices	103
4.2	The Householder Transformation	103
4.3	Householder Triangulation	107
4.3.1	The number of arithmetic operations	109
4.3.2	Solving linear systems using unitary transformations	110

4.4	The QR Decomposition and QR Factorization	110
4.4.1	Existence	110
4.4.2	QR and Gram-Schmidt	113
4.5	Givens Rotations	115
4.6	Review Questions	117
II	Some matrix theory and least squares	119
5	Eigenpairs and Similarity Transformations	121
5.1	Defective and nondefective matrices	121
5.1.1	Similarity transformations	123
5.2	Geometric multiplicity of eigenvalues and the Jordan Form .	125
5.2.1	Algebraic and geometric multiplicity of eigenvalues	126
5.2.2	The Jordan form	127
5.3	The Schur decomposition and normal matrices	131
5.3.1	The Schur decomposition	132
5.3.2	Unitary and orthogonal matrices	132
5.3.3	Normal matrices	134
5.3.4	The quasi triangular form	135
5.4	Hermitian Matrices	136
5.4.1	The Rayleigh Quotient	137
5.4.2	Minmax Theorems	137
5.4.3	The Hoffman-Wielandt Theorem	139
5.5	Left Eigenvectors	140
5.5.1	Biorthogonality	141
5.6	Review Questions	142
6	The Singular Value Decomposition	143
6.1	The SVD always exists	144
6.1.1	The matrices A^*A , AA^*	144
6.2	Further properties of SVD	147
6.2.1	The singular value factorization	148
6.2.2	SVD and the Four Fundamental Subspaces	150
6.2.3	A Geometric Interpretation	152
6.3	Determining the Rank of a Matrix Numerically	153
6.3.1	The Frobenius norm	153
6.3.2	Low rank approximation	155
6.4	Review Questions	156
7	Norms and perturbation theory for linear systems	157
7.1	Vector Norms	157

7.2	Matrix Norms	160
7.2.1	Consistent and subordinate matrix norms	161
7.2.2	Operator norms	162
7.2.3	The operator p -norms	164
7.2.4	Unitary invariant matrix norms	167
7.2.5	Absolute and monotone norms	168
7.3	The Condition Number with Respect to Inversion	168
7.4	Proof that the p -Norms are Norms	174
7.5	Review Questions	180
8	Least Squares	181
8.1	Examples	182
8.1.1	Curve Fitting	184
8.2	Geometric Least Squares theory	187
8.2.1	Sum of subspaces and orthogonal projections	187
8.3	Numerical Solution	189
8.3.1	Normal equations	189
8.3.2	QR factorization	190
8.3.3	Least squares and singular value decomposition	192
8.3.4	The generalized inverse	194
8.4	Perturbation Theory for Least Squares	197
8.4.1	Perturbing the right hand side	197
8.4.2	Perturbing the matrix	199
8.5	Perturbation Theory for Singular Values	200
8.5.1	The Minmax Theorem for Singular Values and the Hoffmann-Wielandt Theorem	200
8.6	Review Questions	203
III	Kronecker Products and Fourier Transforms	205
9	The Kronecker Product	207
9.0.1	The 2D Poisson problem	207
9.0.2	The test matrices	211
9.1	The Kronecker Product	211
9.2	Properties of the 2D Test Matrices	215
9.3	Review Questions	218
10	Fast Direct Solution of a Large Linear System	219
10.1	Algorithms for a Banded Positive Definite System	219
10.1.1	Cholesky factorization	220
10.1.2	Block LU factorization of a block tridiagonal matrix	220

10.1.3	Other methods	221
10.2	A Fast Poisson Solver based on Diagonalization	221
10.3	A Fast Poisson Solver based on the discrete sine and Fourier transforms	223
10.3.1	The discrete sine transform (DST)	224
10.3.2	The discrete Fourier transform (DFT)	224
10.3.3	The fast Fourier transform (FFT)	226
10.3.4	A poisson solver based on the FFT	229
10.4	Review Questions	232
IV	Iterative Methods for Large Linear Systems	233
11	The Classical Iterative Methods	235
11.1	Classical Iterative Methods; Component Form	236
11.1.1	The discrete Poisson system	238
11.2	Classical Iterative Methods; Matrix Form	241
11.2.1	Fixed-point form	241
11.2.2	The splitting matrices for the classical methods .	241
11.3	Convergence	243
11.3.1	Richardson's method.	244
11.3.2	Convergence of SOR	246
11.3.3	Convergence of the classical methods for the discrete Poisson matrix	248
11.3.4	Number of iterations	249
11.3.5	Stopping the iteration	252
11.4	Powers of a matrix	253
11.4.1	The spectral radius	253
11.4.2	Neumann series	255
11.5	The Optimal SOR Parameter ω	256
11.6	Review Questions	259
12	The Conjugate Gradient Method	261
12.1	Quadratic Minimization and Steepest Descent	262
12.2	The Conjugate Gradient Method	265
12.2.1	Derivation of the method	266
12.2.2	The conjugate gradient algorithm	268
12.2.3	Numerical example	269
12.2.4	Implementation issues	269
12.3	Convergence	271
12.3.1	The Main Theorem	271
12.3.2	The number of iterations for the model problems .	272

12.3.3	Krylov spaces and the best approximation property	273
12.4	Proof of the Convergence Estimates	277
12.4.1	Chebyshev polynomials	277
12.4.2	Convergence proof for steepest descent	280
12.4.3	Monotonicity of the error	282
12.5	Preconditioning	283
12.6	Preconditioning Example	286
12.6.1	A variable coefficient problem	286
12.6.2	Applying preconditioning	289
12.7	Review Questions	291
V	Eigenvalues and Eigenvectors	293
13	Numerical Eigenvalue Problems	295
13.1	Eigenpairs	295
13.2	Gerschgorin's Theorem	296
13.3	Perturbation of Eigenvalues	299
13.3.1	Nondefective matrices	301
13.4	Unitary Similarity Transformation of a Matrix into Upper Hessenberg Form	303
13.4.1	Assembling Householder transformations	304
13.5	Computing a Selected Eigenvalue of a Symmetric Matrix	305
13.5.1	The inertia theorem	307
13.5.2	Approximating λ_m	309
13.6	Review Questions	311
14	The QR Algorithm	313
14.1	The Power Method and its variants	313
14.1.1	The power method	314
14.1.2	The inverse power method	317
14.1.3	Rayleigh quotient iteration	318
14.2	The basic QR Algorithm	319
14.2.1	Relation to the power method	321
14.2.2	Invariance of the Hessenberg form	322
14.2.3	Deflation	323
14.3	The Shifted QR Algorithms	323
14.4	Review Questions	324
VI	Appendix	325
A	Computer Arithmetic	327

A.1	Absolute and Relative Errors	327
A.2	Floating Point Numbers	328
A.3	Rounding and Arithmetic Operations	331
A.3.1	Rounding	331
A.3.2	Arithmetic operations	332
B	Differentiation of Vector Functions	333
Bibliography		337
Index		346

Preface

These lecture notes contains the text for a course in matrix analysis and numerical linear algebra given at the beginning graduate level at the University of Oslo. The chapters correspond approximately to one week of lectures, but some contains more. In my own lectures I have not covered Sections 2.4, 3.3, 3.4, 5.5, 7.4, 8.4.2, 8.5, 11.5, 12.4, 12.5, 12.6.

Earlier versions of this manuscript were converted to LaTeX by Are Magnus Bruaset and Njål Foldnes. A special thanks goes to Christian Schulz, Georg Muntingh and Øyvind Ryan who helped me with the exercise sessions and have provided solutions to all problems in these notes.

Oslo, August 2016

Tom Lyche

Chapter 0

A Short Review of Linear Algebra

In this introductory chapter we give a compact introduction to linear algebra with emphasis on \mathbb{R}^n and \mathbb{C}^n . For a more elementary introduction, see for example the book [24]. We start by introducing the notation used.

0.1 Notation

The following sets and notations will be used in this book.

1. The sets of natural numbers, integers, rational numbers, real numbers, and complex numbers are denoted by $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}$, respectively.
2. We use the “colon equal” symbol $v := e$ to indicate that the symbol v is defined by the expression e .
3. \mathbb{R}^n is the set of n -tuples of real numbers which we will represent as column vectors. Thus $\mathbf{x} \in \mathbb{R}^n$ means

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix},$$

where $x_i \in \mathbb{R}$ for $i = 1, \dots, n$. Row vectors are normally identified using the transpose operation. Thus if $\mathbf{x} \in \mathbb{R}^n$ then \mathbf{x} is a column vector and \mathbf{x}^T is a row vector.

4. Addition and scalar multiplication are denoted and defined by

$$\mathbf{x} + \mathbf{y} = \begin{bmatrix} x_1 + y_1 \\ \vdots \\ x_n + y_n \end{bmatrix}, \quad a\mathbf{x} = \begin{bmatrix} ax_1 \\ \vdots \\ ax_n \end{bmatrix}, \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \quad a \in \mathbb{R}.$$

5. $\mathbb{R}^{m \times n}$ is the set of matrices \mathbf{A} with real elements. The integers m and n are the number of rows and columns in the tableau

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}.$$

The element in the i th row and j th column of \mathbf{A} will be denoted by $a_{i,j}$, a_{ij} , $\mathbf{A}(i,j)$ or $(\mathbf{A})_{i,j}$. We use the notations

$$\mathbf{a}_{:j} = \begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{bmatrix}, \quad \mathbf{a}_{i:}^T = [a_{i1}, a_{i2}, \dots, a_{in}], \quad \mathbf{A} = [\mathbf{a}_{:1}, \mathbf{a}_{:2}, \dots, \mathbf{a}_{:n}] = \begin{bmatrix} \mathbf{a}_{1:}^T \\ \mathbf{a}_{2:}^T \\ \vdots \\ \mathbf{a}_{m:}^T \end{bmatrix}$$

for the columns $\mathbf{a}_{:j}$ and rows $\mathbf{a}_{i:}^T$ of \mathbf{A} . We often drop the colon and write \mathbf{a}_j and \mathbf{a}_i^T with the risk of some confusion. If $m = 1$ then \mathbf{A} is a row vector, if $n = 1$ then \mathbf{A} is a column vector, while if $m = n$ then \mathbf{A} is a square matrix. In this text we will denote matrices by boldface capital letters $\mathbf{A}, \mathbf{B}, \mathbf{C}, \dots$ and vectors most often by boldface lower case letters $\mathbf{x}, \mathbf{y}, \mathbf{z}, \dots$.

6. A **complex number** is a number written in the form $x = a + ib$, where a, b are real numbers and i , the **imaginary unit**, satisfies $i^2 = -1$. The set of all such numbers is denoted by \mathbb{C} . The numbers $a = \operatorname{Re} x$ and $b = \operatorname{Im} x$ are the **real and imaginary part** of x . The number $\bar{x} := a - ib$ is called the **complex conjugate** of $x = a + ib$, and $|x| := \sqrt{\bar{x}x} = \sqrt{a^2 + b^2}$ the **absolute value** or **modulus** of x . The **complex exponential function** can be defined by

$$e^x = e^{a+ib} := e^a(\cos b + i \sin b).$$

In particular,

$$e^{i\pi/2} = i, \quad e^{i\pi} = -1, \quad e^{2i\pi} = 1.$$

We have $e^{x+y} = e^x e^y$ for all $x, y \in \mathbb{C}$. The **polar form** of a complex number is

$$x = a + ib = re^{i\theta}, \quad r = |x| = \sqrt{a^2 + b^2}, \quad \cos \theta = \frac{a}{r}, \quad \sin \theta = \frac{b}{r}.$$

7. For matrices and vectors with complex elements we use the notation $\mathbf{A} \in \mathbb{C}^{m \times n}$ and $\mathbf{x} \in \mathbb{C}^n$. We define complex row vectors using either the transpose \mathbf{x}^T or the conjugate transpose operation $\mathbf{x}^* := \bar{\mathbf{x}}^T = [\bar{x}_1, \dots, \bar{x}_n]$.
8. For $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$ and $a \in \mathbb{C}$ the operations of vector addition and scalar multiplication is defined by component operations as in the real case (cf. 4.).
9. The arithmetic operations on rectangular matrices are
 - **matrix addition** $\mathbf{C} = \mathbf{A} + \mathbf{B}$ if $\mathbf{A}, \mathbf{B}, \mathbf{C}$ are matrices of the same size, i.e., with the same number of rows and columns, and $c_{ij} = a_{ij} + b_{ij}$ for all i, j .
 - **multiplication by a scalar** $\mathbf{C} = \alpha \mathbf{A}$, where $c_{ij} = \alpha a_{ij}$ for all i, j .
 - **matrix multiplication** $\mathbf{C} = \mathbf{AB}$, $\mathbf{C} = \mathbf{A} \cdot \mathbf{B}$ or $\mathbf{C} = \mathbf{A} * \mathbf{B}$, where $\mathbf{A} \in \mathbb{C}^{m \times p}$, $\mathbf{B} \in \mathbb{C}^{p \times n}$, $\mathbf{C} \in \mathbb{C}^{m \times n}$, and $c_{ij} = \sum_{k=1}^p a_{ik} b_{kj}$ for $i = 1, \dots, m$, $j = 1, \dots, n$.
 - **element-by-element matrix operations** $\mathbf{C} = \mathbf{A} \times \mathbf{B}$, $\mathbf{D} = \mathbf{A}/\mathbf{B}$, and $\mathbf{E} = \mathbf{A} \wedge r$ where all matrices are of the same size and $c_{ij} = a_{ij} b_{ij}$, $d_{ij} = a_{ij}/b_{ij}$ and $e_{ij} = a_{ij}^r$ for all i, j and suitable r . The element-by-element product $\mathbf{C} = \mathbf{A} \times \mathbf{B}$ is known as the **Schur product** and also the **Hadamard product**.
10. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ or $\mathbf{A} \in \mathbb{C}^{m \times n}$. The **transpose** \mathbf{A}^T and **conjugate transpose** \mathbf{A}^* are $n \times m$ matrices with elements $a_{ij}^T = a_{ji}$ and $a_{ij}^* = \bar{a}_{ji}$, respectively. If \mathbf{B} is an n, p matrix then $(\mathbf{AB})^T = \mathbf{B}^T \mathbf{A}^T$ and $(\mathbf{AB})^* = \mathbf{B}^* \mathbf{A}^*$.
11. The **unit vectors** in \mathbb{R}^n and \mathbb{C}^n are denoted by

$$\mathbf{e}_1 := \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 := \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{e}_3 := \begin{bmatrix} 0 \\ 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \quad \dots, \quad \mathbf{e}_n := \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix},$$

while $\mathbf{I}_n = \mathbf{I} := [\delta_{ij}]_{i,j=1}^n$, where

$$\delta_{ij} := \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise,} \end{cases} \quad (1)$$

- is the **identity matrix** of order n . Both the columns and the transpose of the rows of \mathbf{I} are the unit vectors $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_n$.
12. Some matrices with many zeros have names indicating their “shape”. Suppose $\mathbf{A} \in \mathbb{R}^{n \times n}$ or $\mathbf{A} \in \mathbb{C}^{n \times n}$. Then \mathbf{A} is
 - **diagonal** if $a_{ij} = 0$ for $i \neq j$.

- **upper triangular** or **right triangular** if $a_{ij} = 0$ for $i > j$.
- **lower triangular** or **left triangular** if $a_{ij} = 0$ for $i < j$.
- **upper Hessenberg** if $a_{ij} = 0$ for $i > j + 1$.
- **lower Hessenberg** if $a_{ij} = 0$ for $i < j + 1$.
- **tridiagonal** if $a_{ij} = 0$ for $|i - j| > 1$.
- **d -banded** if $a_{ij} = 0$ for $|i - j| > d$.

13. We use the following notations for diagonal- and tridiagonal $n \times n$ matrices

$$\text{diag}(d_i) = \text{diag}(d_1, \dots, d_n) := \begin{bmatrix} d_1 & 0 & \cdots & 0 \\ 0 & d_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d_n \end{bmatrix} = \begin{bmatrix} d_1 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & d_n \end{bmatrix},$$

$$B = \text{tridiag}(a_i, d_i, c_i) = \text{tridiag}(\mathbf{a}, \mathbf{d}, \mathbf{c}) := \begin{bmatrix} d_1 & c_1 & & & \\ a_1 & d_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{n-2} & d_{n-1} & c_{n-1} \\ & & & a_{n-1} & d_n \end{bmatrix}.$$

Here $b_{ii} = d_i$ for $i = 1, \dots, n$, $b_{i+1,i} = a_i$, $b_{i,i+1} = c_i$ for $i = 1, \dots, n-1$, and $b_{ij} = 0$ otherwise.

14. Suppose $\mathbf{A} \in \mathbb{C}^{m \times n}$ and $1 \leq i_1 < i_2 < \dots < i_r \leq m$, $1 \leq j_1 < j_2 < \dots < j_c \leq n$. The matrix $\mathbf{A}(\mathbf{i}, \mathbf{j}) \in \mathbb{C}^{r \times c}$ is the submatrix of \mathbf{A} consisting of rows $\mathbf{i} := [i_1, \dots, i_r]$ and columns $\mathbf{j} := [j_1, \dots, j_c]$

$$\mathbf{A}(\mathbf{i}, \mathbf{j}) := \mathbf{A} \left(\begin{array}{cccc} i_1 & i_2 & \cdots & i_r \\ j_1 & j_2 & \cdots & j_c \end{array} \right) = \begin{bmatrix} a_{i_1, j_1} & a_{i_1, j_2} & \cdots & a_{i_1, j_c} \\ a_{i_2, j_1} & a_{i_2, j_2} & \cdots & a_{i_2, j_c} \\ \vdots & \vdots & \ddots & \vdots \\ a_{i_r, j_1} & a_{i_r, j_2} & \cdots & a_{i_r, j_c} \end{bmatrix}.$$

For the special case of consecutive rows and columns we use the notation

$$\mathbf{A}(r_1 : r_2, c_1 : c_2) := \begin{bmatrix} a_{r_1, c_1} & a_{r_1, c_1+1} & \cdots & a_{r_1, c_2} \\ a_{r_1+1, c_1} & a_{r_1+1, c_1+1} & \cdots & a_{r_1+1, c_2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{r_2, c_1} & a_{r_2, c_1+1} & \cdots & a_{r_2, c_2} \end{bmatrix}.$$

0.2 Vector Spaces and Subspaces

Many mathematical systems have analogous properties to vectors in \mathbb{R}^2 or \mathbb{R}^3 .

Definition 0.1 (Real vector space)

A **real vector space** is a nonempty set \mathcal{V} , whose objects are called **vectors**, together with two operations $+ : \mathcal{V} \times \mathcal{V} \rightarrow \mathcal{V}$ and $\cdot : \mathbb{R} \times \mathcal{V} \rightarrow \mathcal{V}$, called **addition** and **scalar multiplication**, satisfying the following axioms for all vectors $\mathbf{u}, \mathbf{v}, \mathbf{w}$ in \mathcal{V} and scalars c, d in \mathbb{R} .

(V1) The sum $\mathbf{u} + \mathbf{v}$ is in \mathcal{V} ,

(V2) $\mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$,

(V3) $\mathbf{u} + (\mathbf{v} + \mathbf{w}) = (\mathbf{u} + \mathbf{v}) + \mathbf{w}$,

(V4) There is a **zero vector** $\mathbf{0}$ such that $\mathbf{u} + \mathbf{0} = \mathbf{u}$,

(V5) For each \mathbf{u} in \mathcal{V} there is a vector $-\mathbf{u}$ in \mathcal{V} such that $\mathbf{u} + (-\mathbf{u}) = \mathbf{0}$,

(S1) The scalar multiple $c \cdot \mathbf{u}$ is in \mathcal{V} ,

(S2) $c \cdot (\mathbf{u} + \mathbf{v}) = c \cdot \mathbf{u} + c \cdot \mathbf{v}$,

(S3) $(c + d) \cdot \mathbf{u} = c \cdot \mathbf{u} + d \cdot \mathbf{u}$,

(S4) $c \cdot (d \cdot \mathbf{u}) = (cd) \cdot \mathbf{u}$,

(S5) $1 \cdot \mathbf{u} = \mathbf{u}$.

The scalar multiplication symbol \cdot is often omitted, writing $c\mathbf{v}$ instead of $c \cdot \mathbf{v}$. We define $\mathbf{u} - \mathbf{v} := \mathbf{u} + (-\mathbf{v})$. We call \mathcal{V} a **complex vector space** if the scalars consist of all complex numbers \mathbb{C} . In this book a vector space is either real or complex.

From the axioms it follows that

1. The zero vector is unique.
2. For each $\mathbf{u} \in \mathcal{V}$ the **negative** $-\mathbf{u}$ of \mathbf{u} is unique.
3. $0\mathbf{u} = \mathbf{0}$, $c\mathbf{0} = \mathbf{0}$, and $-\mathbf{u} = (-1)\mathbf{u}$.

Here are some examples

1. The space \mathbb{R}^n , where $n \in \mathbb{N}$, is a real vector space.
2. Similarly, \mathbb{C}^n is a complex vector space.
3. Let \mathcal{D} be a subset of \mathbb{R} and $d \in \mathbb{N}$. The set \mathcal{V} of all functions $\mathbf{f}, \mathbf{g} : \mathcal{D} \rightarrow \mathbb{R}^d$ is a real vector space with

$$(\mathbf{f} + \mathbf{g})(t) := \mathbf{f}(t) + \mathbf{g}(t), \quad (c\mathbf{f})(t) := c\mathbf{f}(t), \quad t \in \mathcal{D}, \quad c \in \mathbb{R}.$$

Two functions \mathbf{f}, \mathbf{g} in \mathcal{V} are equal if $\mathbf{f}(t) = \mathbf{g}(t)$ for all $t \in \mathcal{D}$. The zero element is the **zero function** given by $\mathbf{f}(t) = \mathbf{0}$ for all $t \in \mathcal{D}$ and the negative of \mathbf{f} is given by $-\mathbf{f} = (-1)\mathbf{f}$. In the following we will use boldface letters for functions only if $d > 1$.

4. For $n \geq 0$ the space Π_n of polynomials of degree at most n consists of all polynomials $p : \mathbb{R} \rightarrow \mathbb{R}$, $p : \mathbb{R} \rightarrow \mathbb{C}$, or $p : \mathbb{C} \rightarrow \mathbb{C}$ of the form

$$p(t) = a_0 + a_1 t + a_2 t^2 + \cdots + a_n t^n, \quad (2)$$

where the coefficients a_0, \dots, a_n are real or complex numbers. p is called the **zero polynomial** if all coefficients are zero. All other polynomials are said to be **nontrivial**. The **degree** of a nontrivial polynomial p given by (2) is the smallest integer $0 \leq k \leq n$ such that $p(t) = a_0 + \cdots + a_k t^k$ with $a_k \neq 0$. The degree of the zero polynomial is not defined. Π_n is a vector space if we define addition and scalar multiplication as for functions.

Definition 0.2 (Linear combination)

For $n \geq 1$ let $\mathcal{X} := \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ be a set of vectors in a vector space \mathcal{V} and let c_1, \dots, c_n be scalars.

1. The sum $c_1 \mathbf{x}_1 + \cdots + c_n \mathbf{x}_n$ is called a **linear combination** of $\mathbf{x}_1, \dots, \mathbf{x}_n$.
2. The linear combination is **nontrivial** if $c_j \mathbf{x}_j \neq \mathbf{0}$ for at least one j .
3. The set of all linear combinations of elements in \mathcal{X} is denoted $\text{span}(\mathcal{X})$.
4. A vector space is **finite dimensional** if it has a finite spanning set; i.e., there exists $n \in \mathbb{N}$ and $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ in \mathcal{V} such that $\mathcal{V} = \text{span}(\{\mathbf{x}_1, \dots, \mathbf{x}_n\})$.

Example 0.3 (Linear combinations)

1. Any $\mathbf{x} = [x_1, \dots, x_m]^T$ in \mathbb{C}^m can be written as a linear combination of the unit vectors as $\mathbf{x} = x_1 \mathbf{e}_1 + x_2 \mathbf{e}_2 + \cdots + x_m \mathbf{e}_m$. Thus, $\mathbb{C}^m = \text{span}(\{\mathbf{e}_1, \dots, \mathbf{e}_m\})$ and \mathbb{C}^m is finite dimensional. Similarly \mathbb{R}^m is finite dimensional.
2. Let $\Pi = \cup_n \Pi_n$ be the space of all polynomials. Π is a vector space that is not finite dimensional. For suppose Π is finite dimensional. Then $\Pi = \text{span}(\{p_1, \dots, p_m\})$ for some polynomials p_1, \dots, p_m . Let d be an integer such that the degree of p_j is less than d for $j = 1, \dots, m$. A polynomial of degree d cannot be written as a linear combination of p_1, \dots, p_m , a contradiction.

0.2.1 Linear independence and bases

Definition 0.4 (Linear independence)

A set $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ of nonzero vectors in a vector space is **linearly dependent** if $\mathbf{0}$ can be written as a nontrivial linear combination of $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$. Otherwise \mathcal{X} is **linearly independent**.

A set of vectors $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ is linearly independent if and only if

$$c_1\mathbf{x}_1 + \cdots + c_n\mathbf{x}_n = \mathbf{0} \quad \Rightarrow \quad c_1 = \cdots = c_n = 0. \quad (3)$$

Suppose $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ is linearly independent. Then

1. If $\mathbf{x} \in \text{span}(\mathcal{X})$ then the scalars c_1, \dots, c_n in the representation $\mathbf{x} = c_1\mathbf{x}_1 + \cdots + c_n\mathbf{x}_n$ are unique.
2. Any nontrivial linear combination of $\mathbf{x}_1, \dots, \mathbf{x}_n$ is nonzero,

Lemma 0.5 (Linear independence and span)

Suppose $\mathbf{v}_1, \dots, \mathbf{v}_n$ span a vector space \mathcal{V} and that $\mathbf{w}_1, \dots, \mathbf{w}_k$ are linearly independent vectors in \mathcal{V} . Then $k \leq n$.

Proof. Suppose $k > n$. Write \mathbf{w}_1 as a linear combination of elements from the set $\mathcal{X}_0 := \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$, say $\mathbf{w}_1 = c_1\mathbf{v}_1 + \cdots + c_n\mathbf{v}_n$. Since $\mathbf{w}_1 \neq \mathbf{0}$ not all the c 's are equal to zero. Pick a nonzero c , say c_{i_1} . Then \mathbf{v}_{i_1} can be expressed as a linear combination of \mathbf{w}_1 and the remaining \mathbf{v} 's. So the set $\mathcal{X}_1 := \{\mathbf{w}_1, \mathbf{v}_1, \dots, \mathbf{v}_{i_1-1}, \mathbf{v}_{i_1+1}, \dots, \mathbf{v}_n\}$ must also be a spanning set for \mathcal{V} . We repeat this for \mathbf{w}_2 and \mathcal{X}_1 . In the linear combination $\mathbf{w}_2 = d_{i_1}\mathbf{w}_1 + \sum_{j \neq i_1} d_j\mathbf{v}_j$, we must have $d_{i_2} \neq 0$ for some i_2 with $i_2 \neq i_1$. For otherwise $\mathbf{w}_2 = d_1\mathbf{w}_1$ contradicting the linear independence of the \mathbf{w} 's. So the set \mathcal{X}_2 consisting of the \mathbf{v} 's with \mathbf{v}_{i_1} replaced by \mathbf{w}_1 and \mathbf{v}_{i_2} replaced by \mathbf{w}_2 is again a spanning set for \mathcal{V} . Repeating this process $n - 2$ more times we obtain a spanning set \mathcal{X}_n where $\mathbf{v}_1, \dots, \mathbf{v}_n$ have been replaced by $\mathbf{w}_1, \dots, \mathbf{w}_n$. Since $k > n$ we can then write \mathbf{w}_k as a linear combination of $\mathbf{w}_1, \dots, \mathbf{w}_n$ contradicting the linear independence of the \mathbf{w} 's. We conclude that $k \leq n$. \square

Definition 0.6 (basis)

A finite set of vectors $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ in a vector space \mathcal{V} is a **basis** for \mathcal{V} if

1. $\text{span}\{\mathbf{v}_1, \dots, \mathbf{v}_n\} = \mathcal{V}$.
2. $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ is linearly independent.

Theorem 0.7 (Basis subset of a spanning set)

Suppose \mathcal{V} is a vector space and that $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ is a spanning set for \mathcal{V} . Then we can find a subset $\{\mathbf{v}_{i_1}, \dots, \mathbf{v}_{i_k}\}$ that forms a basis for \mathcal{V} .

Proof. If $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ is linearly dependent we can express one of the \mathbf{v} 's as a nontrivial linear combination of the remaining \mathbf{v} 's and drop that \mathbf{v} from the spanning set. Continue this process until the remaining \mathbf{v} 's are linearly independent. They still span the vector space and therefore form a basis. \square

Corollary 0.8 (Existence of a basis)

A vector space is finite dimensional if and only if it has a basis.

Proof. Let $\mathcal{V} = \text{span}\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ be a finite dimensional vector space. By Theorem 0.7, \mathcal{V} has a basis. Conversely, if $\mathcal{V} = \text{span}\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ and $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ is a basis then it is by definition a finite spanning set. \square

Theorem 0.9 (Dimension of a vector space)

Every basis for a vector space \mathcal{V} has the same number of elements. This number is called the **dimension** of the vector space and denoted $\dim \mathcal{V}$.

Proof. Suppose $\mathcal{X} = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ and $\mathcal{Y} = \{\mathbf{w}_1, \dots, \mathbf{w}_k\}$ are two bases for \mathcal{V} . By Lemma 0.5 we have $k \leq n$. Using the same Lemma with \mathcal{X} and \mathcal{Y} switched we obtain $n \leq k$. We conclude that $n = k$. \square

The set of unit vectors $\{\mathbf{e}_1, \dots, \mathbf{e}_n\}$ form a basis for both \mathbb{R}^n and \mathbb{C}^n .

Theorem 0.10 (Enlarging vectors to a basis)

Every linearly independent set of vectors $\{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ in a finite dimensional vector space \mathcal{V} can be enlarged to a basis for \mathcal{V} .

Proof. If $\{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ does not span \mathcal{V} we can enlarge the set by one vector \mathbf{v}_{k+1} which cannot be expressed as a linear combination of $\{\mathbf{v}_1, \dots, \mathbf{v}_k\}$. The enlarged set is also linearly independent. Continue this process. Since the space is finite dimensional it must stop after a finite number of steps. \square

0.2.2 Subspaces

Definition 0.11 (Subspace)

A nonempty subset \mathcal{S} of a real or complex vector space \mathcal{V} is called a **subspace** of \mathcal{V} if

(V1) The sum $\mathbf{u} + \mathbf{v}$ is in \mathcal{S} for any $\mathbf{u}, \mathbf{v} \in \mathcal{S}$.

(S1) The scalar multiple $c\mathbf{u}$ is in \mathcal{S} for any scalar c and any $\mathbf{u} \in \mathcal{S}$.

Using the operations in \mathcal{V} , any subspace \mathcal{S} of \mathcal{V} is a vector space, i. e., all 10 axioms $V1 - V5$ and $S1 - S5$ are satisfied for \mathcal{S} . In particular, \mathcal{S} must contain the zero element in \mathcal{V} . This follows since the operations of vector addition and scalar multiplication are inherited from \mathcal{V} .

Example 0.12 (Examples of subspaces)

1. $\{\mathbf{0}\}$, where $\mathbf{0}$ is the zero vector is a subspace, the **trivial subspace**. The dimension of the trivial subspace is defined to be zero. All other subspaces are **nontrivial**.
2. \mathcal{V} is a subspace of itself.
3. $\text{span}(\mathcal{X})$ is a subspace of \mathcal{V} for any $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\} \subseteq \mathcal{V}$. Indeed, it is easy to see that **(V1)** and **(S1)** hold.
4. The **sum** of two subspaces \mathcal{R} and \mathcal{S} of a vector space \mathcal{V} is defined by

$$\mathcal{R} + \mathcal{S} := \{\mathbf{r} + \mathbf{s} : \mathbf{r} \in \mathcal{R} \text{ and } \mathbf{s} \in \mathcal{S}\}. \quad (4)$$

Clearly **(V1)** and **(S1)** hold and it is a subspace of \mathcal{V} .

5. The **intersection** of two subspaces \mathcal{R} and \mathcal{S} of a vector space \mathcal{V} is defined by

$$\mathcal{R} \cap \mathcal{S} := \{\mathbf{x} : \mathbf{x} \in \mathcal{R} \text{ and } \mathbf{x} \in \mathcal{S}\}. \quad (5)$$

It is a subspace of \mathcal{V} .

6. The **union** of two subspaces \mathcal{R} and \mathcal{S} of a vector space \mathcal{V} is defined by

$$\mathcal{R} \cup \mathcal{S} := \{\mathbf{x} : \mathbf{x} \in \mathcal{R} \text{ or } \mathbf{x} \in \mathcal{S}\}. \quad (6)$$

In general it is not a subspace of \mathcal{V} .

7. A sum of two subspaces \mathcal{R} and \mathcal{S} of a vector space \mathcal{V} is called a **direct sum** and denoted $\mathcal{R} \oplus \mathcal{S}$ if $\mathcal{R} \cap \mathcal{S} = \{\mathbf{0}\}$. The subspaces \mathcal{R} and \mathcal{S} are called **complementary** in the subspace $\mathcal{R} \oplus \mathcal{S}$.

Theorem 0.13 (Dimension formula for sums of subspaces)

Let \mathcal{R} and \mathcal{S} be two finite dimensional subspaces of a vector space \mathcal{V} . Then

$$\dim(\mathcal{R} + \mathcal{S}) = \dim(\mathcal{R}) + \dim(\mathcal{S}) - \dim(\mathcal{R} \cap \mathcal{S}). \quad (7)$$

In particular, for a direct sum

$$\dim(\mathcal{R} \oplus \mathcal{S}) = \dim(\mathcal{R}) + \dim(\mathcal{S}). \quad (8)$$

Proof. Let $\{\mathbf{u}_1, \dots, \mathbf{u}_p\}$ be a basis for $\mathcal{R} \cap \mathcal{S}$, where $\{\mathbf{u}_1, \dots, \mathbf{u}_p\} = \emptyset$, the empty set, in the case $\mathcal{R} \cap \mathcal{S} = \{\mathbf{0}\}$. We use Theorem 0.10 to extend $\{\mathbf{u}_1, \dots, \mathbf{u}_p\}$ to a basis $\{\mathbf{u}_1, \dots, \mathbf{u}_p, \mathbf{r}_1, \dots, \mathbf{r}_q\}$ for \mathcal{R} and a basis $\{\mathbf{u}_1, \dots, \mathbf{u}_p, \mathbf{s}_1, \dots, \mathbf{s}_t\}$ for \mathcal{S} . Every $\mathbf{x} \in \mathcal{R} + \mathcal{S}$ can be written as a linear combination of $\{\mathbf{u}_1, \dots, \mathbf{u}_p, \mathbf{r}_1, \dots, \mathbf{r}_q, \mathbf{s}_1, \dots, \mathbf{s}_t\}$ so these vectors span $\mathcal{R} + \mathcal{S}$. We show that they are linearly independent and hence a basis. Suppose $\mathbf{u} + \mathbf{r} + \mathbf{s} = \mathbf{0}$, where $\mathbf{u} := \sum_{j=1}^p \alpha_j \mathbf{u}_j$, $\mathbf{r} := \sum_{j=1}^q \rho_j \mathbf{r}_j$, and $\mathbf{s} := \sum_{j=1}^t \sigma_j \mathbf{s}_j$. Now $\mathbf{r} = -(\mathbf{u} + \mathbf{s})$ belongs to both \mathcal{R} and to \mathcal{S} and hence $\mathbf{r} \in \mathcal{R} \cap \mathcal{S}$. Therefore \mathbf{r} can be written as a linear combination of $\{\mathbf{u}_1, \dots, \mathbf{u}_p\}$ say $\mathbf{r} := \sum_{j=1}^p \beta_j \mathbf{u}_j$. But then $\mathbf{0} = \sum_{j=1}^p \beta_j \mathbf{u}_j - \sum_{j=1}^q \rho_j \mathbf{r}_j$ and since $\{\mathbf{u}_1, \dots, \mathbf{u}_p, \mathbf{r}_1, \dots, \mathbf{r}_q\}$ is linearly independent we must have $\beta_1 = \dots = \beta_p = \rho_1 = \dots = \rho_q = 0$ and hence $\mathbf{r} = \mathbf{0}$. We then have $\mathbf{u} + \mathbf{s} = \mathbf{0}$ and by linear independence of $\{\mathbf{u}_1, \dots, \mathbf{u}_p, \mathbf{s}_1, \dots, \mathbf{s}_t\}$ we obtain $\alpha_1 = \dots = \alpha_p = \sigma_1 = \dots = \sigma_t = 0$. We have shown that the vectors $\{\mathbf{u}_1, \dots, \mathbf{u}_p, \mathbf{r}_1, \dots, \mathbf{r}_q, \mathbf{s}_1, \dots, \mathbf{s}_t\}$ constitute a basis for $\mathcal{R} + \mathcal{S}$. But then

$$\dim(\mathcal{R} + \mathcal{S}) = p + q + t = (p + q) + (p + t) - p = \dim(\mathcal{R}) + \dim(\mathcal{S}) - \dim(\mathcal{R} \cap \mathcal{S})$$

and (7) follows. (7) implies (8) since $\dim\{\mathbf{0}\} = 0$. \square

It is convenient to introduce a matrix transforming a basis in a subspace into a basis for the space itself.

Lemma 0.14 (Change of basis matrix)

Suppose \mathcal{S} is a subspace of a finite dimensional vector space \mathcal{V} and let $\{\mathbf{s}_1, \dots, \mathbf{s}_n\}$ be a basis for \mathcal{S} and $\{\mathbf{v}_1, \dots, \mathbf{v}_m\}$ a basis for \mathcal{V} . Then each \mathbf{s}_j can be expressed as a linear combination of $\mathbf{v}_1, \dots, \mathbf{v}_m$, say

$$\mathbf{s}_j = \sum_{i=1}^m a_{ij} \mathbf{v}_i \text{ for } j = 1, \dots, n. \quad (9)$$

If $\mathbf{x} \in \mathcal{S}$ then $\mathbf{x} = \sum_{j=1}^n c_j \mathbf{s}_j = \sum_{i=1}^m b_i \mathbf{v}_i$ for some coefficients $\mathbf{b} := [b_1, \dots, b_m]^T$, $\mathbf{c} := [c_1, \dots, c_n]^T$. Moreover $\mathbf{b} = \mathbf{A}\mathbf{c}$, where $\mathbf{A} = [a_{ij}] \in \mathbb{C}^{m \times n}$ is given by (9). The matrix \mathbf{A} has linearly independent columns.

Proof. (9) holds for some a_{ij} since $\mathbf{s}_j \in \mathcal{V}$ and $\{\mathbf{v}_1, \dots, \mathbf{v}_m\}$ spans \mathcal{V} . Since $\{\mathbf{s}_1, \dots, \mathbf{s}_n\}$ is a basis for \mathcal{S} and $\{\mathbf{v}_1, \dots, \mathbf{v}_m\}$ a basis for \mathcal{V} , every $\mathbf{x} \in \mathcal{S}$ can be written $\mathbf{x} = \sum_{j=1}^n c_j \mathbf{s}_j = \sum_{i=1}^m b_i \mathbf{v}_i$ for some scalars (c_j) and (b_i) . But then

$$\sum_{i=1}^m b_i \mathbf{v}_i = \mathbf{x} = \sum_{j=1}^n c_j \mathbf{s}_j \stackrel{(9)}{=} \sum_{j=1}^n c_j \left(\sum_{i=1}^m a_{ij} \mathbf{v}_i \right) = \sum_{i=1}^m \left(\sum_{j=1}^n a_{ij} c_j \right) \mathbf{v}_i.$$

Since $\{\mathbf{v}_1, \dots, \mathbf{v}_m\}$ is linearly independent it follows that $b_i = \sum_{j=1}^n a_{ij} c_j$ for $i = 1, \dots, m$ or $\mathbf{b} = \mathbf{A}\mathbf{c}$. Finally, to show that \mathbf{A} has linearly independent columns

suppose $\mathbf{b} := \mathbf{A}\mathbf{c} = \mathbf{0}$ for some $\mathbf{c} = [c_1, \dots, c_n]^T$. Define $\mathbf{x} := \sum_{j=1}^n c_j \mathbf{s}_j$. Then $\mathbf{x} = \sum_{i=1}^m b_i \mathbf{v}_i$ and since $\mathbf{b} = \mathbf{0}$ we have $\mathbf{x} = \mathbf{0}$. But since $\{\mathbf{s}_1, \dots, \mathbf{s}_n\}$ is linearly independent it follows that $\mathbf{c} = \mathbf{0}$. \square

The matrix \mathbf{A} in Lemma 0.14 is called a **change of basis matrix**.

0.2.3 The vector spaces \mathbb{R}^n and \mathbb{C}^n

When $\mathcal{V} = \mathbb{R}^m$ we can think of n vectors in \mathbb{R}^m , say $\mathbf{x}_1, \dots, \mathbf{x}_n$, as a set $\mathcal{X} := \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ or as the columns of a matrix $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$. A linear combination can then be written as a matrix times vector $\mathbf{X}\mathbf{c}$, where $\mathbf{c} = [c_1, \dots, c_n]^T$ is the vector of scalars. Thus

$$\text{span}(\mathcal{X}) = \text{span}(\mathbf{X}) = \{\mathbf{X}\mathbf{c} : \mathbf{c} \in \mathbb{R}^n\}.$$

Of course the same holds for \mathbb{C}^m .

In \mathbb{R}^m and \mathbb{C}^m each of the following statements is equivalent to linear independence of \mathcal{X} .

- (i) $\mathbf{X}\mathbf{c} = \mathbf{0} \Rightarrow \mathbf{c} = \mathbf{0}$,
- (ii) \mathbf{X} has linearly independent columns,

Definition 0.15 (Column space and null space)

Associated with a matrix $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$ are the following subspaces

1. The subspace $\text{span}(\mathbf{X})$ is called the **column space** of \mathbf{X} . It is the smallest subspace containing $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$.
2. $\text{span}(\mathbf{X}^T)$ is called the **row space** of \mathbf{X} . It is generated by the rows of \mathbf{X} written as column vectors.
3. The subspace $\ker(\mathbf{X}) := \{\mathbf{y} \in \mathbb{R}^n : \mathbf{X}\mathbf{y} = \mathbf{0}\}$ is called the **null space** or **kernel space** of \mathbf{X} .

Note that the subspace $\ker(\mathbf{X})$ is nontrivial if and only if \mathcal{X} is linearly dependent.

0.3 Linear Systems

Consider a linear system

$$\begin{array}{ccccccccc} a_{11}x_1 & + & a_{12}x_2 & + & \cdots & + & a_{1n}x_n & = & b_1 \\ a_{21}x_1 & + & a_{22}x_2 & + & \cdots & + & a_{2n}x_n & = & b_2 \\ \vdots & & \vdots & & & & \vdots & & \vdots \\ a_{m1}x_1 & + & a_{m2}x_2 & + & \cdots & + & a_{mn}x_n & = & b_m \end{array}$$

of m equations in n unknowns. Here for all i, j , the coefficients a_{ij} , the unknowns x_j , and the components of the right hand sides b_i , are real or complex numbers. The system can be written as a vector equation

$$x_1 \mathbf{a}_1 + x_2 \mathbf{a}_2 + \cdots + x_n \mathbf{a}_n = \mathbf{b},$$

where $\mathbf{a}_j = [a_{1j}, \dots, a_{mj}]^T \in \mathbb{C}^m$ for $j = 1, \dots, n$ and $\mathbf{b} = [b_1, \dots, b_m]^T \in \mathbb{C}^m$. It can also be written as a matrix equation

$$\mathbf{A}\mathbf{x} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} = \mathbf{b}.$$

The system is **homogeneous** if $\mathbf{b} = \mathbf{0}$ and it is said to be **underdetermined**, **square**, or **overdetermined** if $m < n$, $m = n$, or $m > n$, respectively.

0.3.1 Basic properties

A linear system has a unique solution, infinitely many solutions, or no solution. To discuss this we first consider the real case, and a homogeneous underdetermined system.

Lemma 0.16 (Underdetermined system)

Suppose $\mathbf{A} \in \mathbb{R}^{m \times n}$ with $m < n$. Then there is a nonzero $\mathbf{x} \in \mathbb{R}^n$ such that $\mathbf{A}\mathbf{x} = \mathbf{0}$.

Proof. Suppose $\mathbf{A} \in \mathbb{R}^{m \times n}$ with $m < n$. The n columns of \mathbf{A} span a subspace of \mathbb{R}^m . Since \mathbb{R}^m has dimension m the dimension of this subspace is at most m . By Lemma 0.5 the columns of \mathbf{A} must be linearly dependent. It follows that there is a nonzero $\mathbf{x} \in \mathbb{R}^n$ such that $\mathbf{A}\mathbf{x} = \mathbf{0}$. \square

A square matrix is either **nonsingular** or **singular**.

Definition 0.17 (Real nonsingular or singular matrix)

*A square matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ is said to be **nonsingular** if the only real solution of the homogeneous system $\mathbf{A}\mathbf{x} = \mathbf{0}$ is $\mathbf{x} = \mathbf{0}$. The matrix is **singular** if there is a nonzero $\mathbf{x} \in \mathbb{R}^n$ such that $\mathbf{A}\mathbf{x} = \mathbf{0}$.*

Theorem 0.18 (Linear systems; existence and uniqueness)

Suppose $\mathbf{A} \in \mathbb{R}^{n \times n}$. The linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ has a unique solution $\mathbf{x} \in \mathbb{R}^n$ for any $\mathbf{b} \in \mathbb{R}^n$ if and only if the matrix \mathbf{A} is nonsingular.

Proof. Suppose \mathbf{A} is nonsingular. We define $\mathbf{B} = [\mathbf{A} \ \mathbf{b}] \in \mathbb{R}^{n \times (n+1)}$ by adding a column to \mathbf{A} . By Lemma 0.16 there is a nonzero $\mathbf{z} \in \mathbb{R}^{n+1}$ such that $\mathbf{B}\mathbf{z} = \mathbf{0}$. If we write $\mathbf{z} = \begin{bmatrix} \tilde{\mathbf{z}} \\ z_{n+1} \end{bmatrix}$ where $\tilde{\mathbf{z}} = [z_1, \dots, z_n]^T \in \mathbb{R}^n$ and $z_{n+1} \in \mathbb{R}$, then

$$\mathbf{B}\mathbf{z} = [\mathbf{A} \ \mathbf{b}] \begin{bmatrix} \tilde{\mathbf{z}} \\ z_{n+1} \end{bmatrix} = \mathbf{A}\tilde{\mathbf{z}} + z_{n+1}\mathbf{b} = \mathbf{0}.$$

We cannot have $z_{n+1} = 0$ for then $\mathbf{A}\tilde{\mathbf{z}} = \mathbf{0}$ for a nonzero $\tilde{\mathbf{z}}$, contradicting the nonsingularity of \mathbf{A} . Define $\mathbf{x} := -\tilde{\mathbf{z}}/z_{n+1}$. Then

$$\mathbf{A}\mathbf{x} = -\mathbf{A}\left(\frac{\tilde{\mathbf{z}}}{z_{n+1}}\right) = -\frac{1}{z_{n+1}}\mathbf{A}\tilde{\mathbf{z}} = -\frac{1}{z_{n+1}}(-z_{n+1}\mathbf{b}) = \mathbf{b},$$

so \mathbf{x} is a solution.

Suppose $\mathbf{Ax} = \mathbf{b}$ and $\mathbf{Ay} = \mathbf{b}$ for $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. Then $\mathbf{A}(\mathbf{x} - \mathbf{y}) = \mathbf{0}$ and since \mathbf{A} is nonsingular we conclude that $\mathbf{x} - \mathbf{y} = \mathbf{0}$ or $\mathbf{x} = \mathbf{y}$. Thus the solution is unique.

Conversely, if $\mathbf{Ax} = \mathbf{b}$ has a unique solution for any $\mathbf{b} \in \mathbb{R}^n$ then $\mathbf{Ax} = \mathbf{0}$ has a unique solution which must be $\mathbf{x} = \mathbf{0}$. Thus \mathbf{A} is nonsingular. \square

For the complex case we have

Lemma 0.19 (Complex underdetermined system)

Suppose $\mathbf{A} \in \mathbb{C}^{m \times n}$ with $m < n$. Then there is a nonzero $\mathbf{x} \in \mathbb{C}^n$ such that $\mathbf{Ax} = \mathbf{0}$.

Definition 0.20 (Complex nonsingular matrix)

A square matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ is said to be **nonsingular** if the only complex solution of the homogeneous system $\mathbf{Ax} = \mathbf{0}$ is $\mathbf{x} = \mathbf{0}$. The matrix is **singular** if it is not nonsingular.

Theorem 0.21 (Complex linear system; existence and uniqueness)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$. The linear system $\mathbf{Ax} = \mathbf{b}$ has a unique solution $\mathbf{x} \in \mathbb{C}^n$ for any $\mathbf{b} \in \mathbb{C}^n$ if and only if the matrix \mathbf{A} is nonsingular.



James Joseph Sylvester, 1814-1897. The word matrix to denote a rectangular array of numbers, was first used by Sylvester in 1850.

0.3.2 The inverse matrix

Suppose $A \in \mathbb{C}^{n \times n}$ is a square matrix. A matrix $B \in \mathbb{C}^{n \times n}$ is called a **right inverse** of A if $AB = I$. A matrix $C \in \mathbb{C}^{n \times n}$ is said to be a **left inverse** of A if $CA = I$. We say that A is **invertible** if it has both a left- and a right inverse. If A has a right inverse B and a left inverse C then

$$C = CI = C(AB) = (CA)B = IB = B$$

and this common inverse is called the **inverse** of A and denoted by A^{-1} . Thus the inverse satisfies $A^{-1}A = AA^{-1} = I$.

We want to characterize the class of invertible matrices and start with a lemma.

Theorem 0.22 (Product of nonsingular matrices)

If $A, B, C \in \mathbb{C}^{n \times n}$ with $AB = C$ then C is nonsingular if and only if both A and B are nonsingular. In particular, if $AB = I$ or $BA = I$ then A is nonsingular and $A^{-1} = B$.

Proof. Suppose both A and B are nonsingular and let $Cx = \mathbf{0}$. Then $ABx = \mathbf{0}$ and since A is nonsingular we see that $Bx = \mathbf{0}$. Since B is nonsingular we have $x = \mathbf{0}$. We conclude that C is nonsingular.

For the converse suppose first that B is singular and let $\mathbf{x} \in \mathbb{C}^n$ be a nonzero vector so that $B\mathbf{x} = \mathbf{0}$. But then $C\mathbf{x} = (AB)\mathbf{x} = A(B\mathbf{x}) = A\mathbf{0} = \mathbf{0}$ so C is singular. Finally suppose B is nonsingular, but A is singular. Let $\tilde{\mathbf{x}}$ be a nonzero vector such that $A\tilde{\mathbf{x}} = \mathbf{0}$. By Theorem 0.21 there is a vector \mathbf{x} such that $B\mathbf{x} = \tilde{\mathbf{x}}$ and \mathbf{x} is nonzero since $\tilde{\mathbf{x}}$ is nonzero. But then $C\mathbf{x} = (AB)\mathbf{x} = A(B\mathbf{x}) = A\tilde{\mathbf{x}} = \mathbf{0}$ for a nonzero vector \mathbf{x} and C is singular. \square

Theorem 0.23 (When is a square matrix invertible?)

A square matrix is invertible if and only if it is nonsingular.

Proof. Suppose first \mathbf{A} is a nonsingular matrix. By Theorem 0.21 each of the linear systems $\mathbf{Ab}_i = \mathbf{e}_i$ has a unique solution \mathbf{b}_i for $i = 1, \dots, n$. Let $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_n]$. Then $\mathbf{AB} = [\mathbf{Ab}_1, \dots, \mathbf{Ab}_n] = [\mathbf{e}_1, \dots, \mathbf{e}_n] = \mathbf{I}$ so that \mathbf{A} has a right inverse \mathbf{B} . By Theorem 0.22 \mathbf{B} is nonsingular since \mathbf{I} is nonsingular and $\mathbf{AB} = \mathbf{I}$. Since \mathbf{B} is nonsingular we can use what we have shown for \mathbf{A} to conclude that \mathbf{B} has a right inverse \mathbf{C} , i.e. $\mathbf{BC} = \mathbf{I}$. But then $\mathbf{AB} = \mathbf{BC} = \mathbf{I}$ so \mathbf{B} has both a right inverse and a left inverse which must be equal so $\mathbf{A} = \mathbf{C}$. Since $\mathbf{BC} = \mathbf{I}$ we have $\mathbf{BA} = \mathbf{I}$, so \mathbf{B} is also a left inverse of \mathbf{A} and \mathbf{A} is invertible.

Conversely, if \mathbf{A} is invertible then it has a right inverse \mathbf{B} . Since $\mathbf{AB} = \mathbf{I}$ and \mathbf{I} is nonsingular, we again use Theorem 0.22 to conclude that \mathbf{A} is nonsingular. \square

To verify that some matrix \mathbf{B} is an inverse of another matrix \mathbf{A} it is enough to show that \mathbf{B} is either a left inverse or a right inverse of \mathbf{A} . This calculation also proves that \mathbf{A} is nonsingular. We use this observation to give simple proofs of the following results.

Corollary 0.24 (Basic properties of the inverse matrix)

Suppose $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n \times n}$ are nonsingular and c is a nonzero constant.

1. \mathbf{A}^{-1} is nonsingular and $(\mathbf{A}^{-1})^{-1} = \mathbf{A}$.
2. $\mathbf{C} = \mathbf{AB}$ is nonsingular and $\mathbf{C}^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$.
3. \mathbf{A}^T is nonsingular and $(\mathbf{A}^T)^{-1} = (\mathbf{A}^{-1})^T =: \mathbf{A}^{-T}$.
4. \mathbf{A}^* is nonsingular and $(\mathbf{A}^*)^{-1} = (\mathbf{A}^{-1})^* =: \mathbf{A}^{-*}$.
5. $c\mathbf{A}$ is nonsingular and $(c\mathbf{A})^{-1} = \frac{1}{c}\mathbf{A}^{-1}$.

Proof.

1. Since $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$ the matrix \mathbf{A} is a right inverse of \mathbf{A}^{-1} . Thus \mathbf{A}^{-1} is nonsingular and $(\mathbf{A}^{-1})^{-1} = \mathbf{A}$.
2. We note that $(\mathbf{B}^{-1}\mathbf{A}^{-1})(\mathbf{AB}) = \mathbf{B}^{-1}(\mathbf{A}^{-1}\mathbf{A})\mathbf{B} = \mathbf{B}^{-1}\mathbf{B} = \mathbf{I}$. Thus \mathbf{AB} is invertible with the indicated inverse since it has a left inverse.
3. Now $\mathbf{I} = \mathbf{I}^T = (\mathbf{A}^{-1}\mathbf{A})^T = \mathbf{A}^T(\mathbf{A}^{-1})^T$ showing that $(\mathbf{A}^{-1})^T$ is a right inverse of \mathbf{A}^T . The proof of part 4 is similar.
4. The matrix $\frac{1}{c}\mathbf{A}^{-1}$ is a one sided inverse of $c\mathbf{A}$.

\square

Exercise 0.25 (The inverse of a general 2×2 matrix)

Show that

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \alpha \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}, \quad \alpha = \frac{1}{ad - bc},$$

for any a, b, c, d such that $ad - bc \neq 0$.

Exercise 0.26 (The inverse of a special 2×2 matrix)

Find the inverse of

$$A = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}.$$

Exercise 0.27 (Sherman-Morrison formula)

Suppose $A \in \mathbb{C}^{n \times n}$, and $B, C \in \mathbb{R}^{n \times m}$ for some $n, m \in \mathbb{N}$. If $(I + C^T A^{-1} B)^{-1}$ exists then

$$(A + BC^T)^{-1} = A^{-1} - A^{-1}B(I + C^T A^{-1}B)^{-1}C^TA^{-1}.$$

0.4 Determinants

The first systematic treatment of determinants was given by Cauchy in 1812. He adopted the word “determinant”. The first use of determinants was made by Leibniz in 1693 in a letter to De L'Hôpital. By the beginning of the 20th century the theory of determinants filled four volumes of almost 2000 pages (Muir, 1906–1923. Historic references can be found in this work). The main use of determinants in this text will be to study the characteristic polynomial of a matrix and to show that a matrix is nonsingular.

For any $A \in \mathbb{C}^{n \times n}$ the determinant of A is defined the number

$$\det(A) = \sum_{\sigma \in S_n} \text{sign}(\sigma) a_{\sigma(1),1} a_{\sigma(2),2} \cdots a_{\sigma(n),n}. \quad (10)$$

This sum ranges of all $n!$ permutations of $\{1, 2, \dots, n\}$. Moreover, $\text{sign}(\sigma)$ equals the number of times a bigger integer precedes a smaller one in σ . We also denote the determinant by (Cayley, 1841)

$$\left| \begin{array}{cccc} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{array} \right|.$$

From the definition we have

$$\left| \begin{array}{cc} a_{11} & a_{12} \\ a_{21} & a_{22} \end{array} \right| = a_{11}a_{22} - a_{21}a_{12}.$$

The first term on the right corresponds to the identity permutation ϵ given by $\epsilon(i) = i$, $i = 1, 2$. The second term comes from the permutation $\sigma = \{2, 1\}$. For $n = 3$ there are six permutations of $\{1, 2, 3\}$. Then

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11}a_{22}a_{33} - a_{11}a_{32}a_{23} - a_{21}a_{12}a_{33} + a_{21}a_{32}a_{13} + a_{31}a_{12}a_{23} - a_{31}a_{22}a_{13}.$$

This follows since $\text{sign}(\{1, 2, 3\}) = \text{sign}(\{2, 3, 1\}) = \text{sign}(\{3, 1, 2\}) = 1$, and noting that interchanging two numbers in a permutation reverses its sign we find $\text{sign}(\{2, 1, 3\}) = \text{sign}(\{3, 2, 1\}) = \text{sign}(\{1, 3, 2\}) = -1$.

To compute the value of a determinant from the definition can be a trying experience. It is often better to use elementary operations on rows or columns to reduce it to a simpler form. For example, if \mathbf{A} is triangular then $\det(\mathbf{A}) = a_{11}a_{22} \cdots a_{nn}$, the product of the diagonal elements. In particular, for the identity matrix $\det(\mathbf{I}) = 1$. The elementary operations using either rows or columns are

1. Interchanging two rows(columns): $\det(\mathbf{B}) = -\det(\mathbf{A})$,
2. Multiply a row(column) by a scalar: α , $\det(\mathbf{B}) = \alpha \det(\mathbf{A})$,
3. Add a constant multiple of one row(column) to another row(column): $\det(\mathbf{B}) = \det(\mathbf{A})$.

where \mathbf{B} is the result of performing the indicated operation on \mathbf{A} .

If only a few elements in a row or column are nonzero then a **cofactor expansion** can be used. These expansions take the form

$$\det(\mathbf{A}) = \sum_{j=1}^n (-1)^{i+j} a_{ij} \det(\mathbf{A}_{ij}) \text{ for } i = 1, \dots, n, \text{ row} \quad (11)$$

$$\det(\mathbf{A}) = \sum_{i=1}^n (-1)^{i+j} a_{ij} \det(\mathbf{A}_{ij}) \text{ for } j = 1, \dots, n, \text{ column}. \quad (12)$$

Here $\mathbf{A}_{i,j}$ denotes the submatrix of \mathbf{A} obtained by deleting the i th row and j th column of \mathbf{A} . For $\mathbf{A} \in \mathbb{C}^{n \times n}$ and $1 \leq i, j \leq n$ the determinant $\det(\mathbf{A}_{ij})$ is called the **cofactor** of a_{ij} .

Example 0.28 (Determinant equation for a straight line)

The equation for a straight line through two points (x_1, y_1) and (x_2, y_2) in the plane can be written as the equation

$$\det(\mathbf{A}) := \begin{vmatrix} 1 & x & y \\ 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \end{vmatrix} = 0$$

involving a determinant of order 3. We can compute this determinant using row operations of type 3. Subtracting row 2 from row 3 and then row 1 from row 2, and then using a cofactor expansion on the first column we obtain

$$\begin{aligned} \begin{vmatrix} 1 & x & y \\ 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \end{vmatrix} &= \begin{vmatrix} 1 & x & y \\ 0 & x_1 - x & y_1 - y \\ 0 & x_2 - x_1 & y_2 - y_1 \end{vmatrix} \\ &= \begin{vmatrix} x_1 - x & y_1 - y \\ x_2 - x_1 & y_2 - y_1 \end{vmatrix} = (x_1 - x)(y_2 - y_1) - (y_1 - y)(x_2 - x_1). \end{aligned}$$

Rearranging the equation $\det(\mathbf{A}) = 0$ we obtain

$$y - y_1 = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1)$$

which is the slope form of the equation of a straight line.

We will freely use, without proofs, the following properties of determinants. If \mathbf{A}, \mathbf{B} are square matrices of order n with real or complex elements, then

1. $\det(\mathbf{AB}) = \det(\mathbf{A}) \det(\mathbf{B})$.
2. $\det(\mathbf{A}^T) = \det(\mathbf{A})$, and $\det(\mathbf{A}^*) = \overline{\det(\mathbf{A})}$, (complex conjugate).
3. $\det(a\mathbf{A}) = a^n \det(\mathbf{A})$, for $a \in \mathbb{C}$.
4. \mathbf{A} is singular if and only if $\det(\mathbf{A}) = 0$.
5. If $\mathbf{A} = \begin{bmatrix} \mathbf{C} & \mathbf{D} \\ \mathbf{0} & \mathbf{E} \end{bmatrix}$ for some square matrices \mathbf{C}, \mathbf{E} then $\det(\mathbf{A}) = \det(\mathbf{C}) \det(\mathbf{E})$.
6. **Cramer's rule** Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is nonsingular and $\mathbf{b} \in \mathbb{C}^n$. Let $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$ be the unique solution of $\mathbf{Ax} = \mathbf{b}$. Then

$$x_j = \frac{\det(\mathbf{A}_j(\mathbf{b}))}{\det(\mathbf{A})}, \quad j = 1, 2, \dots, n,$$

where $\mathbf{A}_j(\mathbf{b})$ denote the matrix obtained from \mathbf{A} by replacing the j th column of \mathbf{A} by \mathbf{b} .

7. **Adjoint formula for the inverse.** If $\mathbf{A} \in \mathbb{C}^{n \times n}$ is nonsingular then

$$\mathbf{A}^{-1} = \frac{1}{\det(\mathbf{A})} \text{adj}(\mathbf{A}),$$

where the matrix $\text{adj}(\mathbf{A}) \in \mathbb{C}^{n \times n}$ with elements $\text{adj}(\mathbf{A})_{i,j} = (-1)^{i+j} \det(\mathbf{A}_{j,i})$ is called the **adjoint** of \mathbf{A} . Moreover, $\mathbf{A}_{j,i}$ denotes the submatrix of \mathbf{A} obtained by deleting the j th row and i th column of \mathbf{A} .

8. **Cauchy-Binet formula:** Let $\mathbf{A} \in \mathbb{C}^{m \times p}$, $\mathbf{B} \in \mathbb{C}^{p \times n}$ and $\mathbf{C} = \mathbf{AB}$. Suppose $1 \leq r \leq \min\{m, n, p\}$ and let $\mathbf{i} = \{i_1, \dots, i_r\}$ and $\mathbf{j} = \{j_1, \dots, j_r\}$ be integers with $1 \leq i_1 < i_2 < \dots < i_r \leq m$ and $1 \leq j_1 < j_2 < \dots < j_r \leq n$. Then

$$\begin{bmatrix} c_{i_1, j_1} & \cdots & c_{i_1, j_r} \\ \vdots & & \vdots \\ c_{i_r, j_1} & \cdots & c_{i_r, j_r} \end{bmatrix} = \sum_{\mathbf{k}} \begin{bmatrix} a_{i_1, k_1} & \cdots & a_{i_1, k_r} \\ \vdots & & \vdots \\ a_{i_r, k_1} & \cdots & a_{i_r, k_r} \end{bmatrix} \begin{bmatrix} b_{k_1, j_1} & \cdots & b_{k_1, j_r} \\ \vdots & & \vdots \\ b_{k_r, j_1} & \cdots & b_{k_r, j_r} \end{bmatrix},$$

where we sum over all $\mathbf{k} = \{k_1, \dots, k_r\}$ with $1 \leq k_1 < k_2 < \dots < k_r \leq p$. More compactly,

$$\det(\mathbf{C}(\mathbf{i}, \mathbf{j})) = \sum_{\mathbf{k}} \det(\mathbf{A}(\mathbf{i}, \mathbf{k})) \det(\mathbf{B}(\mathbf{k}, \mathbf{j})), \quad (13)$$

Note the resemblance to the formula for matrix multiplication.



Arthur Cayley, 1821-1895 (left), Gabriel Cramer 1704-1752 (center), Alexandre-Thophile Vandermonde, 1735-1796 (right). The notation $| |$ for determinants is due to Cayley 1841.

Exercise 0.29 (Cramer's rule; special case)

Solve the following system by Cramers rule:

$$\begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 6 \end{bmatrix}$$

Exercise 0.30 (Adjoint matrix; special case)

Show that if

$$\mathbf{A} = \begin{bmatrix} 2 & -6 & 3 \\ 3 & -2 & -6 \\ 6 & 3 & 2 \end{bmatrix},$$

then

$$\text{adj}(\mathbf{A}) = \begin{bmatrix} 14 & 21 & 42 \\ -42 & -14 & 21 \\ 21 & -42 & 14 \end{bmatrix}.$$

Moreover,

$$\text{adj}(\mathbf{A})\mathbf{A} = \begin{bmatrix} 343 & 0 & 0 \\ 0 & 343 & 0 \\ 0 & 0 & 343 \end{bmatrix} = \det(\mathbf{A})\mathbf{I}.$$

Exercise 0.31 (Determinant equation for a plane)

Show that

$$\begin{vmatrix} x & y & z & 1 \\ x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \end{vmatrix} = 0.$$

is the equation for a plane through three points (x_1, y_1, z_1) , (x_2, y_2, z_2) and (x_3, y_3, z_3) in space.

Exercise 0.32 (Signed area of a triangle)

Let $P_i = (x_i, y_i)$, $i = 1, 2, 3$, be three points in the plane defining a triangle T . Show that the area of T is¹

$$A(T) = \frac{1}{2} \begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{vmatrix}.$$

The area is positive if we traverse the vertices in counterclockwise order.

Exercise 0.33 (Vandermonde matrix)

Show that

$$\begin{vmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{vmatrix} = \prod_{i>j} (x_i - x_j),$$

where $\prod_{i>j} (x_i - x_j) = \prod_{i=2}^n (x_i - x_1)(x_i - x_2) \cdots (x_i - x_{i-1})$. This determinant is called the Vandermonde determinant.²

¹Hint: $A(T) = A(ABP_3P_1) + A(P_3BCP_2) - A(P_1ACP_2)$, c.f. Figure 1

²Hint: subtract x_n^k times column k from column $k+1$ for $k = n-1, n-2, \dots, 1$.

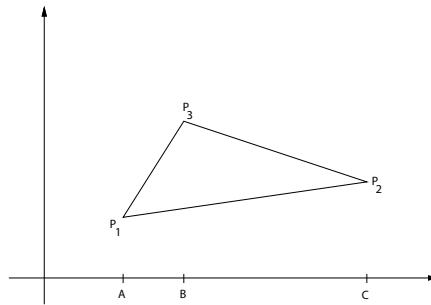


Figure 1: The triangle T defined by the three points P_1 , P_2 and P_3 .

Exercise 0.34 (Cauchy determinant (1842))

Let $\alpha = [\alpha_1, \dots, \alpha_n]^T$, $\beta = [\beta_1, \dots, \beta_n]^T$ be in \mathbb{R}^n .

- a) Consider the matrix $A \in \mathbb{R}^{n \times n}$ with elements $a_{i,j} = 1/(\alpha_i + \beta_j)$, $i, j = 1, 2, \dots, n$. Show that

$$\det(A) = Pg(\alpha)g(\beta)$$

where $P = \prod_{i=1}^n \prod_{j=1}^n a_{ij}$, and for $\gamma = [\gamma_1, \dots, \gamma_n]^T$

$$g(\gamma) = \prod_{i=2}^n (\gamma_i - \gamma_1)(\gamma_i - \gamma_2) \cdots (\gamma_i - \gamma_{i-1})$$

Hint: Multiply the i th row of A by $\prod_{j=1}^n (\alpha_i + \beta_j)$ for $i = 1, 2, \dots, n$. Call the resulting matrix C . Each element of C is a product of $n-1$ factors $\alpha_r + \beta_s$. Hence $\det(C)$ is a sum of terms where each term contain precisely $n(n-1)$ factors $\alpha_r + \beta_s$. Thus $\det(C) = q(\alpha, \beta)$ where q is a polynomial of degree at most $n(n-1)$ in α_i and β_j . Since $\det(A)$ and therefore $\det(C)$ vanishes if $\alpha_i = \beta_j$ for some $i \neq j$ or $\beta_r = \beta_s$ for some $r \neq s$, we have that $q(\alpha, \beta)$ must be divisible by each factor in $g(\alpha)$ and $g(\beta)$. Since $g(\alpha)$ and $g(\beta)$ is a polynomial of degree $n(n-1)$, we have

$$q(\alpha, \beta) = kg(\alpha)g(\beta)$$

for some constant k independent of α and β . Show that $k = 1$ by choosing $\beta_i + \alpha_i = 0$, $i = 1, 2, \dots, n$.

- b) Notice that the cofactor of any element in the above matrix A is the determinant of a matrix of similar form. Use the cofactor and determinant of A to represent the elements of $A^{-1} = (b_{j,k})$. Answer:

$$b_{j,k} = (\alpha_k + \beta_j)A_k(-\beta_j)B_j(-\alpha_k),$$

where

$$A_k(x) = \prod_{s \neq k} \left(\frac{\alpha_s - x}{\alpha_s - \alpha_k} \right), \quad B_k(x) = \prod_{s \neq k} \left(\frac{\beta_s - x}{\beta_s - \beta_k} \right).$$

Exercise 0.35 (Inverse of the Hilbert matrix)

Let $\mathbf{H}_n = (h_{i,j})$ be the $n \times n$ matrix with elements $h_{i,j} = 1/(i+j-1)$. Use Exercise 0.34 to show that the elements $t_{i,j}^n$ in $\mathbf{T}_n = \mathbf{H}_n^{-1}$ are given by

$$t_{i,j}^n = \frac{f(i)f(j)}{i+j-1},$$

where

$$f(i+1) = \left(\frac{i^2 - n^2}{i^2} \right) f(i), \quad i = 1, 2, \dots, \quad f(1) = -n.$$

0.5 Eigenvalues, eigenvectors and eigenpairs

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is a square matrix, $\lambda \in \mathbb{C}$ and $\mathbf{x} \in \mathbb{C}^n$. We say that (λ, \mathbf{x}) is an **eigenpair** for \mathbf{A} if $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ and \mathbf{x} is nonzero. The scalar λ is called an **eigenvalue** and \mathbf{x} is said to be an **eigenvector**.³ The set of eigenvalues is called the **spectrum** of \mathbf{A} and is denoted by $\sigma(\mathbf{A})$. For example, $\sigma(\mathbf{I}) = \{1, \dots, 1\} = \{1\}$.

Eigenvalues are the roots of the characteristic polynomial.

Lemma 0.36 (Characteristic equation)

For any $\mathbf{A} \in \mathbb{C}^{n \times n}$ we have $\lambda \in \sigma(\mathbf{A}) \iff \det(\mathbf{A} - \lambda\mathbf{I}) = 0$.

Proof. Suppose (λ, \mathbf{x}) is an eigenpair for \mathbf{A} . The equation $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ can be written $(\mathbf{A} - \lambda\mathbf{I})\mathbf{x} = \mathbf{0}$. Since \mathbf{x} is nonzero the matrix $\mathbf{A} - \lambda\mathbf{I}$ must be singular with a zero determinant. Conversely, if $\det(\mathbf{A} - \lambda\mathbf{I}) = 0$ then $\mathbf{A} - \lambda\mathbf{I}$ is singular and $(\mathbf{A} - \lambda\mathbf{I})\mathbf{x} = \mathbf{0}$ for some nonzero $\mathbf{x} \in \mathbb{C}^n$. Thus $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ and (λ, \mathbf{x}) is an eigenpair for \mathbf{A} . \square

The expression $\det(\mathbf{A} - \lambda\mathbf{I})$ is a polynomial of exact degree n in λ . For $n = 3$ we have

$$\det(\mathbf{A} - \lambda\mathbf{I}) = \begin{vmatrix} a_{11} - \lambda & a_{12} & a_{13} \\ a_{21} & a_{22} - \lambda & a_{23} \\ a_{31} & a_{32} & a_{33} - \lambda \end{vmatrix}.$$

³The word “eigen” is derived from German and means “own”

Expanding this determinant by the first column we find

$$\begin{aligned}\det(\mathbf{A} - \lambda\mathbf{I}) &= (a_{11} - \lambda) \begin{vmatrix} a_{22} - \lambda & a_{23} \\ a_{32} & a_{33} - \lambda \end{vmatrix} - a_{21} \begin{vmatrix} a_{12} & a_{13} \\ a_{32} & a_{33} - \lambda \end{vmatrix} \\ &\quad + a_{31} \begin{vmatrix} a_{12} & a_{13} \\ a_{22} - \lambda & a_{23} \end{vmatrix} = (a_{11} - \lambda)(a_{22} - \lambda)(a_{33} - \lambda) + r(\lambda)\end{aligned}$$

for some polynomial r of degree at most one. In general

$$\det(\mathbf{A} - \lambda\mathbf{I}) = (a_{11} - \lambda)(a_{22} - \lambda) \cdots (a_{nn} - \lambda) + r(\lambda), \quad (14)$$

where each term in $r(\lambda)$ has at most $n - 2$ factors containing λ . It follows that r is a polynomial of degree at most $n - 2$, $\det(\mathbf{A} - \lambda\mathbf{I})$ is a polynomial of exact degree n in λ and the eigenvalues are the roots of this polynomial.

We observe that $\det(\mathbf{A} - \lambda\mathbf{I}) = (-1)^n \det(\lambda\mathbf{I} - \mathbf{A})$ so $\det(\mathbf{A} - \lambda\mathbf{I}) = 0$ if and only if $\det(\lambda\mathbf{I} - \mathbf{A}) = 0$.

Definition 0.37 (Characteristic polynomial of a matrix)

The function $\pi_{\mathbf{A}}: \mathbb{C} \rightarrow \mathbb{C}$ given by $\pi_{\mathbf{A}}(\lambda) = \det(\mathbf{A} - \lambda\mathbf{I})$ is called the **characteristic polynomial of \mathbf{A}** . The equation $\det(\mathbf{A} - \lambda\mathbf{I}) = 0$ is called the **characteristic equation of \mathbf{A}** .

By the fundamental theorem of algebra an $n \times n$ matrix has, counting multiplicities, precisely n eigenvalues $\lambda_1, \dots, \lambda_n$ some of which might be complex even if \mathbf{A} is real. The complex eigenpairs of a real matrix occur in complex conjugate pairs. Indeed, taking the complex conjugate on both sides of the equation $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ with \mathbf{A} real gives $\mathbf{A}\bar{\mathbf{x}} = \bar{\lambda}\bar{\mathbf{x}}$.

Theorem 0.38 (Sums and products of eigenvalues; trace)

For any $\mathbf{A} \in \mathbb{C}^{n \times n}$

$$\text{trace}(\mathbf{A}) = \lambda_1 + \lambda_2 + \cdots + \lambda_n, \quad \det(\mathbf{A}) = \lambda_1 \lambda_2 \cdots \lambda_n, \quad (15)$$

where the **trace** of $\mathbf{A} \in \mathbb{C}^{n \times n}$ is the sum of its diagonal elements

$$\text{trace}(\mathbf{A}) := a_{11} + a_{22} + \cdots + a_{nn}. \quad (16)$$

Proof. We compare two different expansions of $\pi_{\mathbf{A}}$. On the one hand from (14) we find

$$\pi_{\mathbf{A}}(\lambda) = (-1)^n \lambda^n + c_{n-1} \lambda^{n-1} + \cdots + c_0,$$

where $c_{n-1} = (-1)^{n-1} \text{trace}(\mathbf{A})$ and $c_0 = \pi_{\mathbf{A}}(0) = \det(\mathbf{A})$. On the other hand

$$\pi_{\mathbf{A}}(\lambda) = (\lambda_1 - \lambda) \cdots (\lambda_n - \lambda) = (-1)^n \lambda^n + d_{n-1} \lambda^{n-1} + \cdots + d_0,$$

where $d_{n-1} = (-1)^{n-1}(\lambda_1 + \cdots + \lambda_n)$ and $d_0 = \lambda_1 \cdots \lambda_n$. Since $c_j = d_j$ for all j we obtain (15). \square

For a 2×2 matrix the characteristic equation takes the convenient form

$$\lambda^2 - \text{trace}(\mathbf{A})\lambda + \det(\mathbf{A}) = 0. \quad (17)$$

Thus, if $\mathbf{A} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ then $\text{trace}(\mathbf{A}) = 4$, $\det(\mathbf{A}) = 3$ so that $\pi_{\mathbf{A}}(\lambda) = \lambda^2 - 4\lambda + 3$. Since \mathbf{A} is singular $\iff \mathbf{Ax} = \mathbf{0}$, some $\mathbf{x} \neq \mathbf{0} \iff \mathbf{Ax} = \mathbf{0x}$, some $\mathbf{x} \neq \mathbf{0} \iff$ zero is an eigenvalue of \mathbf{A} , we obtain

Theorem 0.39 (Zero eigenvalue)

The matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ is singular if and only if zero is an eigenvalue.

Since the determinant of a triangular matrix is equal to the product of the diagonal elements the eigenvalues of a triangular matrix are found on the diagonal. In general it is not easy to find all eigenvalues of a matrix. However, sometimes the dimension of the problem can be reduced. Since the determinant of a block triangular matrix is equal to the product of the determinants of the diagonal blocks we obtain

Theorem 0.40 (Eigenvalues of a block triangular matrix)

If $\mathbf{A} = \begin{bmatrix} \mathbf{B} & \mathbf{D} \\ \mathbf{0} & \mathbf{C} \end{bmatrix}$ is block triangular then $\pi_{\mathbf{A}} = \pi_{\mathbf{B}} \cdot \pi_{\mathbf{C}}$.

0.6 Algorithms and Numerical Stability

In this text we consider mathematical problems (i.e., linear algebra problems) and many detailed numerical algorithms to solve them. Complexity is discussed briefly in Section 2.2.2. As for programming issues we often vectorize the algorithms leading to shorter and more efficient programs. Stability is important both for the mathematical problems and for the numerical algorithms. Stability can be studied in terms of perturbation theory leading to condition numbers, see Chapters 7, 8, 13. We will often use phrases like “the algorithm is numerically stable” or “the algorithm is not numerically stable” without saying precisely what we mean by this. Loosely speaking, an algorithm is numerically stable if the solution, computed in floating point arithmetic, is the exact solution of a slightly perturbed problem. To determine upper bounds for these perturbations is the topic of **backward error analysis**. We give a rather limited introduction to floating point arithmetic and backward error analysis in Appendix A, but in the text we will not discuss this. This does not mean that numerical stability is not an important issue. In fact, numerical stability is crucial for a good algorithm. For thorough treatments of numerical stability issues we refer to the books [13] and [28, 29].

A list of freely available software for solving linear algebra problems can be found at

<http://www.netlib.org/utk/people/JackDongarra/la-sw.html>

Part I

Linear Systems

Chapter 1

Diagonally dominant tridiagonal matrices; three examples

In this introductory chapter we consider three problems originating from:

- cubic spline interpolation,
- a two point boundary value problem,
- an eigenvalue problem for a two point boundary value problem.

Each of these problems leads to a linear algebra problem with a matrix which is diagonally dominant and tridiagonal. Taking advantage of structure we can show existence, uniqueness and characterization of a solution, and derive efficient algorithms to compute a numerical solution.

For a particular tridiagonal test matrix we determine all its eigenvectors and eigenvalues. We will need these later when studying more complex problems.

We end the chapter with an introduction to block multiplication, a powerful tool in matrix analysis and numerical linear algebra. Block multiplication is applied to derive some basic facts about triangular matrices.

1.1 Cubic Spline Interpolation

We consider the following interpolation problem.

Given an interval $[a, b]$, $n + 1 \geq 2$ equidistant sites

$$x_i = a + \frac{i - 1}{n}(b - a), \quad i = 1, 2, \dots, n + 1 \quad (1.1)$$

and y values $\mathbf{y} := [y_1, \dots, y_{n+1}]^T \in \mathbb{R}^{n+1}$. We seek a function $g : [a, b] \rightarrow \mathbb{R}$ such that

$$g(x_i) = y_i, \text{ for } i = 1, \dots, n + 1. \quad (1.2)$$

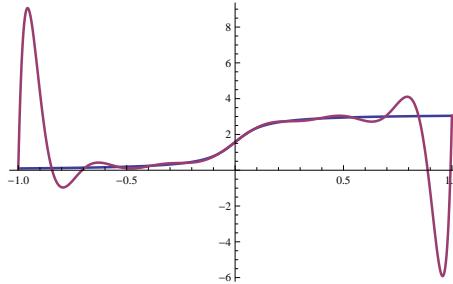


Figure 1.1: The polynomial of degree 13 interpolating $f(x) = \arctan(10x) + \pi/2$ on $[-1, 1]$. See text.

For simplicity we only consider equidistant sites. More generally they could be any $a \leq x_1 < x_2 < \dots < x_{n+1} \leq b$.

1.1.1 Polynomial interpolation

Since there are $n+1$ interpolation conditions in (1.2) a natural choice for a function g is a polynomial of degree n . As shown in most books on numerical methods such a g is uniquely defined and there are good algorithms for computing it. Evidently, when $n = 1$, g is the straight line

$$g(x) = y_1 + \frac{y_2 - y_1}{x_2 - x_1}(x - x_1), \quad (1.3)$$

known as the **linear interpolation polynomial**.

Polynomial interpolation is an important technique which often gives good results, but the interpolant g can have undesirable oscillations when n is large. As an example, consider the function given by

$$f(x) = \arctan(10x) + \pi/2, \quad x \in [-1, 1].$$

The function f and the polynomial g of degree at most 13 satisfying (1.2) with $[a, b] = [-1, 1]$ and $y_i = f(x_i)$, $i = 1, \dots, 14$ is shown in Figure 1.1. The interpolant has large oscillations near the end of the range. This is an example of the **Runge phenomenon** and is due to the fact that the sites are uniformly spaced. Using larger n will only make the oscillations bigger.

1.1.2 Piecewise linear and cubic spline interpolation

To avoid oscillations like the one in Figure 1.1 piecewise linear interpolation can be used. An example is shown in Figure 1.2. The interpolant g approximates the

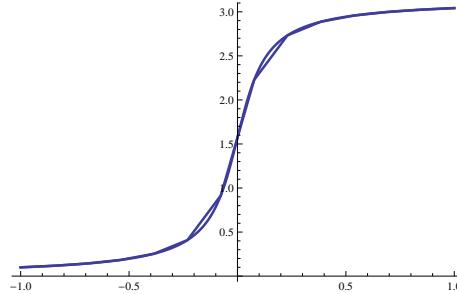


Figure 1.2: The piecewise linear polynomial interpolating $f(x) = \arctan(10x) + \pi/2$ at $n = 14$ uniform points on $[-1, 1]$.

original function quite well, and for some applications, like plotting, the linear interpolant using many points is what is used. Note that g is a piecewise polynomial of the form

$$g(x) := \begin{cases} p_1(x), & \text{if } x_1 \leq x < x_2, \\ p_2(x), & \text{if } x_2 \leq x < x_3, \\ \vdots & \\ p_{n-1}(x), & \text{if } x_{n-1} \leq x < x_n, \\ p_n(x), & \text{if } x_n \leq x \leq x_{n+1}, \end{cases} \quad (1.4)$$

where each p_i is a polynomial of degree ≤ 1 . In particular, p_1 is given in (1.3) and the other polynomials p_i are given by similar expressions.

The piecewise linear interpolant is continuous, but the first derivative will usually have jumps at the interior sites. We can obtain a smoother approximation by letting g be a piecewise polynomial of higher degree. With degree 3 (cubic) we obtain continuous derivatives of order ≤ 2 (C^2). We consider here the following functions giving examples of **C^2 cubic spline interpolants**.

Definition 1.1 (The D_2 -spline problem)

Given $n \in \mathbb{N}$, an interval $[a, b]$, $\mathbf{y} \in \mathbb{R}^{n+1}$, knots (sites) x_1, \dots, x_{n+1} given by (1.1) and numbers μ_1, μ_{n+1} . The problem is to find a function $g : [a, b] \rightarrow \mathbb{R}$ such that

- g is of the form (1.4) with each p_i a cubic polynomial, (**a piecewise cubic polynomial**),
- $g \in C^2[a, b]$, i.e., derivatives of order ≤ 2 are continuous on \mathbb{R} , (**smooth**),
- $g(x_i) = y_i$, $i = 1, 2, \dots, n + 1$, (**interpolation conditions**),
- $g''(a) = \mu_1$, $g''(b) = \mu_{n+1}$, (**D_2 boundary conditions**).

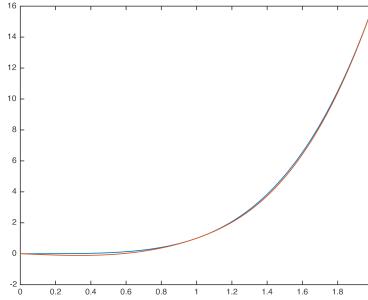


Figure 1.3: A cubic spline with one knot interpolating $f(x) = x^4$ on $[0, 2]$.

We call g a **D_2 -spline**. It is called an **N -spline** or **natural spline** if $\mu_1 = \mu_{n+1} = 0$.

Example 1.2 (A D_2 -spline)

Suppose we choose $n = 2$ and sample data from the function $f : [0, 2] \rightarrow \mathbb{R}$ given by $f(x) = x^4$. Thus we consider the D_2 -spline problem with $[a, b] = [0, 2]$, $\mathbf{y} := [0, 1, 16]^T$ and $\mu_1 = g''(0) = 0$, $\mu_3 = g''(2) = 48$. The knots are $x_1 = 0$, $x_2 = 1$ and $x_3 = 2$. The function g given by

$$g(x) := \begin{cases} p_1(x) = -\frac{1}{2}x + \frac{3}{2}x^3, & \text{if } 0 \leq x < 1, \\ p_2(x) = 1 + 4(x-1) + \frac{9}{2}(x-1)^2 + \frac{13}{2}(x-1)^3, & \text{if } 1 \leq x \leq 2, \end{cases} \quad (1.5)$$

is a D_2 -spline solving this problem. Indeed, p_1 and p_2 are cubic polynomials. For smoothness we find $p_1(1) = p_2(1) = 1$, $p'_1(1) = p'_2(1) = 4$, $p''_1(1) = p''_2(1) = 9$ which implies that $g \in C^2[0, 2]$. Finally we check that the interpolation and boundary conditions hold. Indeed, $g(0) = p_1(0) = 0$, $g(1) = p_2(1) = 1$, $g(2) = p_2(2) = 16$, $g''(0) = p''_1(0) = 0$ and $g''(2) = p''_2(2) = 48$. Note that $p'''_1(x) = 9 \neq 39 = p'''_2(x)$ showing that the third derivative of g is piecewise constant with a jump discontinuity at the interior knot. A plot of f and g is shown in Figure 1.3. It is hard to distinguish one from the other.

We note that

- The C^2 condition is equivalent to

$$p_{i-1}^{(j)}(x_i) = p_i^{(j)}(x_i), \quad j = 0, 1, 2, \quad i = 2, \dots, n.$$

- The extra boundary conditions D_2 or N are introduced to obtain a unique interpolant. Indeed counting requirements we have $3(n-1)$ C^2 conditions,

$n + 1$ conditions (1.2), and two boundary conditions, adding to $4n$. Since a cubic polynomial has four coefficients, this number is equal to the number of coefficients of the n polynomials p_1, \dots, p_n and give hope for uniqueness of the interpolant.

1.1.3 Give me a moment

Existence and uniqueness of a solution of the D_2 -spline problem hinges on the nonsingularity of a linear system of equations that we now derive. The unknowns are derivatives at the knots. Here we use second derivatives which are sometimes called **moments**. We start with the following lemma.

Lemma 1.3 (Representing each p_i using $(0, 2)$ interpolation)

Given $a < b$, $h = (b - a)/n$ with $n \geq 2$, $x_i = a + (i - 1)h$, and numbers y_i, μ_i for $i = 1, \dots, n + 1$. For $i = 1, \dots, n$ there are unique cubic polynomials p_i such that

$$p_i(x_i) = y_i, \quad p_i(x_{i+1}) = y_{i+1}, \quad p''_i(x_i) = \mu_i, \quad p''_i(x_{i+1}) = \mu_{i+1}. \quad (1.6)$$

Moreover,

$$p_i(x) = c_{i,1} + c_{i,2}(x - x_i) + c_{i,3}(x - x_i)^2 + c_{i,4}(x - x_i)^3 \quad i = 1, \dots, n, \quad (1.7)$$

where

$$c_{i,1} = y_i, \quad c_{i,2} = \frac{y_{i+1} - y_i}{h} - \frac{h}{3}\mu_i - \frac{h}{6}\mu_{i+1}, \quad c_{i,3} = \frac{\mu_i}{2}, \quad c_{i,4} = \frac{\mu_{i+1} - \mu_i}{6h}. \quad (1.8)$$

Proof. Consider p_i in the form (1.7) for some $1 \leq i \leq n$. Evoking (1.6) we find $p_i(x_i) = c_{i,1} = y_i$. Since $p''_i(x) = 2c_{i,3} + 6c_{i,4}(x - x_i)$ we obtain $c_{i,3}$ from $p''_i(x_i) = 2c_{i,3} = \mu_i$ (a moment), and then $c_{i,4}$ from $p''_i(x_{i+1}) = \mu_i + 6hc_{i,4} = \mu_{i+1}$. Finally we find $c_{i,2}$ by solving $p_i(x_{i+1}) = y_i + c_{i,2}h + \frac{\mu_i}{2}h^2 + \frac{\mu_{i+1} - \mu_i}{6h}h^3 = y_{i+1}$. For $j = 0, 1, 2, 3$ the shifted powers $(x - x_i)^j$ constitute a basis for cubic polynomials and the formulas (1.8) are unique by construction. It follows that p_i is unique. \square

Theorem 1.4 (Constructing a D_2 -spline)

Suppose for some moments μ_1, \dots, μ_{n+1} that each p_i is given as in Lemma 1.3 for $i = 1, \dots, n$. If in addition

$$\mu_{i-1} + 4\mu_i + \mu_{i+1} = \frac{6}{h^2}(y_{i+1} - 2y_i + y_{i-1}), \quad i = 2, \dots, n, \quad (1.9)$$

then the function g given by (1.4) solves a D_2 -spline problem.

Proof. Suppose for $1 \leq i \leq n$ that p_i is given as in Lemma 1.3 for some μ_1, \dots, μ_{n+1} . Consider the C^2 requirement. Since $p_{i-1}(x_i) = p_i(x_i) = y_i$ and $p''_{i-1}(x_i) = p''_i(x_i) = \mu_i$ for $i = 2, \dots, n$ it follows that $g \in C^2$ if and only if $p'_{i-1}(x_i) = p'_i(x_i)$ for $i = 2, \dots, n$. By (1.7)

$$\begin{aligned} p'_{i-1}(x_i) &= c_{i-1,2} + 2hc_{i-1,3} + 3h^2c_{i-1,4} \\ &= \frac{y_i - y_{i-1}}{h} - \frac{h}{3}\mu_{i-1} - \frac{h}{6}\mu_i + 2h\frac{\mu_{i-1}}{2} + 3h^2\frac{\mu_i - \mu_{i-1}}{6h} \\ &= \frac{y_i - y_{i-1}}{h} + \frac{h}{6}\mu_{i-1} + \frac{h}{3}\mu_i \\ p'_i(x_i) &= c_{i2} = \frac{y_{i+1} - y_i}{h} - \frac{h}{3}\mu_i - \frac{h}{6}\mu_{i+1}. \end{aligned} \tag{1.10}$$

A simple calculation shows that $p'_{i-1}(x_i) = p'_i(x_i)$ if and only if (1.9) holds.

Finally consider the function g given by (1.4). If (1.9) holds then $g \in C^2[a, b]$. By construction $g(x_i) = y_i$, $i = 1, \dots, n+1$, $g''(a) = p''_1(x_1) = \mu_1$ and $g''(b) = p''_{n+1}(x_{n+1}) = \mu_{n+1}$. It follows that g solves the D_2 -spline problem. \square

In order for the D_2 -spline to exist we need to show that μ_2, \dots, μ_n always can be determined from (1.9). For $n \geq 3$ and with μ_1 and μ_{n+1} given (1.9) can be written in the form

$$\left[\begin{array}{cccccc} 4 & 1 & & & & \\ 1 & 4 & 1 & & & \\ & \ddots & \ddots & \ddots & & \\ & & 1 & 4 & 1 & \\ & & & 1 & 4 & \end{array} \right] \left[\begin{array}{c} \mu_2 \\ \mu_3 \\ \vdots \\ \mu_{n-1} \\ \mu_n \end{array} \right] = \frac{6}{h^2} \left[\begin{array}{c} \delta^2 y_2 - \mu_1 \\ \delta^2 y_3 \\ \vdots \\ \delta^2 y_{n-1} \\ \delta^2 y_n - \mu_{n+1} \end{array} \right], \quad \delta^2 y_i := y_{i+1} - 2y_i + y_{i-1}. \tag{1.11}$$

This is a square linear system of equations. We recall (see Theorem 0.18) that a square system $\mathbf{Ax} = \mathbf{b}$ has a solution for all right hand sides \mathbf{b} if and only if the coefficient matrix \mathbf{A} is nonsingular, i.e., the homogeneous system $\mathbf{Ax} = \mathbf{0}$ only has the solution $\mathbf{x} = \mathbf{0}$. Moreover, the solution is unique. We need to show that the coefficient matrix in (1.11) is nonsingular.

We observe that the matrix in (1.11) is strictly diagonally dominant in accordance with the following definition.

Definition 1.5 (Strict diagonal dominance)

The matrix $\mathbf{A} = [a_{ij}] \in \mathbb{C}^{n \times n}$ is **strictly diagonally dominant** if

$$|a_{ii}| > \sum_{j \neq i} |a_{ij}|, \quad i = 1, \dots, n. \tag{1.12}$$

Theorem 1.6 (Strict diagonal dominance)

A strictly diagonally dominant matrix is nonsingular. Moreover, if $\mathbf{A} \in \mathbb{C}^{n \times n}$ is strictly diagonally dominant then the solution \mathbf{x} of $\mathbf{Ax} = \mathbf{b}$ is bounded as follows:

$$\max_{1 \leq i \leq n} |x_i| \leq \max_{1 \leq i \leq n} \left(\frac{|b_i|}{\sigma_i} \right), \text{ where } \sigma_i := |a_{ii}| - \sum_{j \neq i} |a_{ij}|. \quad (1.13)$$

Proof. We first show that the bound (1.13) holds for any solution \mathbf{x} . Choose k so that $|x_k| = \max_i |x_i|$. Then

$$|b_k| = |a_{kk}x_k + \sum_{j \neq k} a_{kj}x_j| \geq |a_{kk}||x_k| - \sum_{j \neq k} |a_{kj}||x_j| \geq |x_k|(|a_{kk}| - \sum_{j \neq k} |a_{kj}|),$$

and this implies $\max_{1 \leq i \leq n} |x_i| = |x_k| \leq \frac{|b_k|}{\sigma_k} \leq \max_{1 \leq i \leq n} \left(\frac{|b_i|}{\sigma_i} \right)$. For the nonsingularity, if $\mathbf{Ax} = \mathbf{0}$, then $\max_{1 \leq i \leq n} |x_i| \leq 0$ by (1.13), and so $\mathbf{x} = \mathbf{0}$. \square

Theorem 1.6 implies that the system (1.11) has a unique solution giving rise to a function g detailed in Lemma 1.3 and solving the D_2 -spline problem. For uniqueness suppose g_1 and g_2 are two D_2 -splines interpolating the same data. Then $g := g_1 - g_2$ is an N -spline satisfying (1.2) with $\mathbf{y} = \mathbf{0}$. The solution $[\mu_2, \dots, \mu_n]^T$ of (1.11) and also $\mu_1 = \mu_{n+1}$ are zero. It follows from (1.8) that all coefficients $c_{i,j}$ are zero. We conclude that $g = 0$ and $g_1 = g_2$.

Example 1.7 (Cubic B-spline)

For the N -spline with $[a, b] = [0, 4]$ and $\mathbf{y} = [0, \frac{1}{6}, \frac{2}{3}, \frac{1}{6}, 0]$ the linear system (1.9) takes the form

$$\begin{bmatrix} 4 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 4 \end{bmatrix} \begin{bmatrix} \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix} = \begin{bmatrix} 2 \\ -6 \\ 2 \end{bmatrix}.$$

The solution is $\mu_2 = \mu_4 = 1$, $\mu_3 = -2$. The knotset is $\{0, 1, 2, 3, 4\}$. Using (1.8) (cf. Exercise 1.18) we find

$$g(x) := \begin{cases} p_1(x) = \frac{1}{6}x^3, & \text{if } 0 \leq x < 1, \\ p_2(x) = \frac{1}{6} + \frac{1}{2}(x-1) + \frac{1}{2}(x-1)^2 - \frac{1}{2}(x-1)^3, & \text{if } 1 \leq x < 2, \\ p_3(x) = \frac{2}{3} - (x-2)^2 + \frac{1}{2}(x-2)^3, & \text{if } 2 \leq x < 3, \\ p_4(x) = \frac{1}{6} - \frac{1}{2}(x-3) + \frac{1}{2}(x-3)^2 - \frac{1}{6}(x-3)^3 = \frac{1}{6}(4-x)^3, & \text{if } 3 \leq x \leq 4, \end{cases} \quad (1.14)$$

A plot of this spline is shown in Figure 1.4. On $(0, 4)$ the function g equals the nonzero part of a C^2 cubic B-spline.

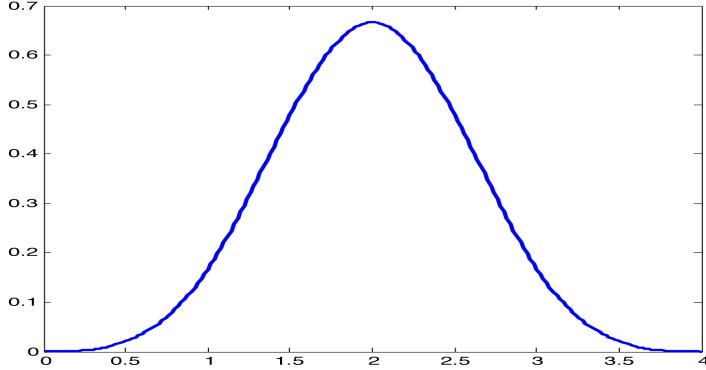


Figure 1.4: A cubic B-spline.

1.1.4 LU Factorization of a Tridiagonal System

To find the D^2 -spline g we have to solve the triangular system (1.11). Consider solving a general tridiagonal linear system $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{A} = \text{tridiag}(a_i, d_i, c_i) \in \mathbb{C}^{n \times n}$. Instead of using Gaussian elimination we can try to construct triangular matrices \mathbf{L} and \mathbf{U} such that the product $\mathbf{A} = \mathbf{LU}$ has the form

$$\begin{bmatrix} d_1 & c_1 & & \\ a_1 & d_2 & c_2 & \\ \ddots & \ddots & \ddots & \\ & a_{n-2} & d_{n-1} & c_{n-1} \\ & & a_{n-1} & d_n \end{bmatrix} = \begin{bmatrix} 1 & & & \\ l_1 & 1 & & \\ & \ddots & \ddots & \\ & & l_{n-1} & 1 \end{bmatrix} \begin{bmatrix} u_1 & c_1 & & \\ & \ddots & \ddots & \\ & & u_{n-1} & c_{n-1} \\ & & & u_n \end{bmatrix}. \quad (1.15)$$

If \mathbf{L} and \mathbf{U} can be determined and u_1, \dots, u_n are nonzero we can find \mathbf{x} by solving two simpler systems $\mathbf{Lz} = \mathbf{b}$ and $\mathbf{Ux} = \mathbf{z}$.

For $n = 3$ equation (1.15) takes the form

$$\begin{bmatrix} d_1 & c_1 & 0 \\ a_1 & d_2 & c_2 \\ 0 & a_2 & d_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ l_1 & 1 & 0 \\ 0 & l_2 & 1 \end{bmatrix} \begin{bmatrix} u_1 & c_1 & 0 \\ 0 & u_2 & c_2 \\ 0 & 0 & u_3 \end{bmatrix} = \begin{bmatrix} u_1 & c_1 & 0 \\ l_1 u_1 & l_1 c_1 + u_2 & c_2 \\ 0 & l_2 u_2 & l_2 c_2 + u_3 \end{bmatrix},$$

and the systems $\mathbf{Lz} = \mathbf{b}$ and $\mathbf{Ux} = \mathbf{z}$ can be written

$$\begin{bmatrix} 1 & 0 & 0 \\ l_1 & 1 & 0 \\ 0 & l_2 & 1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}, \quad \begin{bmatrix} u_1 & c_1 & 0 \\ 0 & u_2 & c_2 \\ 0 & 0 & u_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}.$$

Comparing elements we find

$$\begin{aligned} u_1 &= d_1, \quad l_1 = a_1/u_1, \quad u_2 = d_2 - l_1 c_1, \quad l_2 = a_2/u_2, \quad u_3 = d_3 - l_2 c_2, \\ z_1 &= b_1, \quad z_2 = b_2 - l_1 z_1, \quad z_3 = b_3 - l_2 z_2, \\ x_3 &= z_3/u_3, \quad x_2 = (z_2 - c_2 x_3)/u_2, \quad x_1 = (z_1 - c_1 x_2)/u_1. \end{aligned}$$

In general, if

$$u_1 = d_1, \quad l_k = a_k/u_k, \quad u_{k+1} = d_{k+1} - l_k c_k, \quad k = 1, 2, \dots, n-1, \quad (1.16)$$

then $\mathbf{A} = \mathbf{L}\mathbf{U}$. If u_1, u_2, \dots, u_{n-1} are nonzero then (1.16) is well defined. If in addition $u_n \neq 0$ then we can solve $\mathbf{L}\mathbf{z} = \mathbf{b}$ and $\mathbf{U}\mathbf{x} = \mathbf{z}$ for \mathbf{z} and \mathbf{x} . We formulate this as two algorithms.

Algorithm 1.8 (trifactor)

Vectors $\mathbf{l} \in \mathbb{C}^{n-1}$, $\mathbf{u} \in \mathbb{C}^n$ are computed from $\mathbf{a}, \mathbf{c} \in \mathbb{C}^{n-1}$, $\mathbf{d} \in \mathbb{C}^n$. This implements the LU factorization of a tridiagonal matrix.

```

1 function [l,u]=trifactor(a,d,c)
2 % [l,u]=trifactor(a,d,c)
3 u=d; l=a;
4 for k=1:length(a)
5     l(k)=a(k)/u(k);
6     u(k+1)=d(k+1)-l(k)*c(k);
7 end

```

Algorithm 1.9 (trisolve)

The solution \mathbf{x} of a tridiagonal system with r right hand sides is computed from a previous call to trifactor. Here $\mathbf{l} \in \mathbb{C}^{n-1}$ and $\mathbf{u} \in \mathbb{C}^n$ are output from trifactor and $\mathbf{b} \in \mathbb{C}^{n,r}$ for some $r \in \mathbb{N}$.

```

1 function x = trisolve (l,u,c,b)
2 % x = trisolve (l,u,c,b)
3 x=b;
4 n= size(b,1);
5 for k=2:n
6     x(:,k)=b(:,k)-l(:,k-1)*x(:,k-1);
7 end
8 x(:,n)=x(:,n)/u(n);
9 for k=(n-1):-1:1
10    x(:,k)=(x(:,k)-c(:,k)*x(:,k+1))/u(k);
11 end

```

Since division by zero can occur, the algorithms will not work in general, but for tridiagonal strictly diagonally dominant systems we have⁴

⁴It can be shown that any strictly diagonally dominant linear system has a unique LU factorization.

Theorem 1.10 (LU of a tridiagonal strictly dominant system)

A strictly diagonally dominant tridiagonal matrix has a unique LU-factorization of the form (1.15).

Proof. We show that the u_k 's in (1.16) are nonzero for $k = 1, \dots, n$. For this it is sufficient to show by induction that

$$|u_k| \geq \sigma_k + |c_k|, \quad \text{where, } \sigma_k := |d_k| - |a_{k-1}| - |c_k| > 0, \quad k = 1, \dots, n, \quad (1.17)$$

and where $a_0 := c_n := 0$. By assumption $|u_1| = |d_1| = \sigma_1 + |c_1|$. Suppose $|u_k| \geq \sigma_k + |c_k|$ for some $1 \leq k \leq n-1$. Then $|c_k|/|u_k| < 1$ and by (1.16) and strict diagonal dominance

$$\begin{aligned} |u_{k+1}| &= |d_{k+1} - l_k c_k| = |d_{k+1} - \frac{a_k c_k}{u_k}| \geq |d_{k+1}| - \frac{|a_k||c_k|}{|u_k|} \\ &\geq |d_{k+1}| - |a_k| = \sigma_{k+1} + |c_{k+1}|. \end{aligned} \quad (1.18)$$

□

Corollary 1.11 (Stability of the LU factorization)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is tridiagonal and strictly diagonally dominant with computed elements in the LU factorization given by (1.16). Then (1.17) holds, $u_1 = d_1$ and

$$|l_k| = \frac{|a_k|}{|u_k|} \leq \frac{|a_k|}{|d_k| - |a_{k-1}|}, \quad |u_{k+1}| \leq |d_{k+1}| + \frac{|a_k||c_k|}{|d_k| - |a_{k-1}|}, \quad k = 1, \dots, n-1. \quad (1.19)$$

Proof. Using (1.16) and (1.17) for $1 \leq k \leq n-1$ we find

$$|l_k| = \frac{|a_k|}{|u_k|} \leq \frac{|a_k|}{|d_k| - |a_{k-1}|}, \quad |u_{k+1}| \leq |d_{k+1}| + |l_k||c_k| \leq |d_{k+1}| + \frac{|a_k||c_k|}{|d_k| - |a_{k-1}|}.$$

□

- For a strictly diagonally dominant tridiagonal matrix it follows from Corollary 1.11 that the LU factorization algorithm `trifactor` is stable. We cannot have severe growth in the computed elements u_k and l_k .
- The number of arithmetic operations to compute the LU factorization of a tridiagonal matrix of order n using (1.16) is $3n - 3$, while the number of arithmetic operations for Algorithm `trisolve` is $r(5n - 4)$, where r is the number of right-hand sides. This means that the complexity to solve one

tridiagonal system is $O(n)$, or more precisely $8n - 7$, and this number only grows linearly⁵ with n .

1.1.5 Exercises for section 1.1

Exercise 1.12 (The shifted power basis is a basis)

Show that the polynomials $\{(x - x_i)^j\}_{0 \leq j \leq n}$ is a basis for polynomials of degree n .⁶

Exercise 1.13 (The natural spline, $n = 1$)

How can one define an N -spline when $n = 1$?

Exercise 1.14 (Bounding the moments)

Show that for the N -spline the solution of the linear system (1.11) is bounded as follows:⁷

$$\max_{2 \leq j \leq n} |\mu_j| \leq \frac{3}{h^2} \max_{2 \leq i \leq n} |y_{i+1} - 2y_i + y_{i-1}|.$$

Exercise 1.15 (Moment equations for 1. derivative boundary conditions)

Suppose instead of the D_2 boundary conditions we use D_1 conditions given by $g'(a) = s_1$ and $g'(b) = s_{n+1}$ for some given numbers s_1 and s_{n+1} . Show that the linear system for the moments of a D_1 -spline can be written

$$\begin{bmatrix} 2 & 1 & & & \\ 1 & 4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 4 & 1 \\ & & & 1 & 2 \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \\ \mu_{n+1} \end{bmatrix} = \frac{6}{h^2} \begin{bmatrix} y_2 - y_1 - hs_1 \\ \delta^2 y_2 \\ \delta^2 y_3 \\ \vdots \\ \delta^2 y_{n-1} \\ \delta^2 y_n \\ hs_{n+1} - y_{n+1} + y_n \end{bmatrix}, \quad (1.20)$$

where $\delta^2 y_i := y_{i+1} - 2y_i + y_{i-1}$, $i = 2, \dots, n$. Hint: Use (1.10) to compute $g'(x_1)$ and $g'(x_{n+1})$. Is g unique?

⁵We show in Section 2.2.2 that Gaussian elimination on a full $n \times n$ system is an $O(n^3)$ process

⁶Hint: consider an arbitrary polynomial of degree n and expanded it in Taylor series around x_i

⁷Hint, use Theorem 1.6

Exercise 1.16 (Minimal norm property of the natural spline)

Study proof of the following theorem⁸.

Theorem 1.17 (Minimal norm property of a cubic spline) Suppose g is an N -spline. Then

$$\int_a^b (g''(x))^2 dx \leq \int_a^b (h''(x))^2 dx$$

for all $h \in C^2[a, b]$ such that $h(x_i) = g(x_i)$, $i = 1, \dots, n + 1$.

Proof. Let h be any interpolant as in the theorem. We first show the orthogonality condition

$$\int_a^b g''e'' = 0, \quad e := h - g. \quad (1.21)$$

Integration by parts gives $\int_a^b g''e'' = [g''e']_a^b - \int_a^b g'''e'$. The first term is zero since g'' is continuous and $g''(b) = g''(a) = 0$. For the second term, since g''' is equal to a constant v_i on each subinterval (x_i, x_{i+1}) and $e(x_i) = 0$, for $i = 1, \dots, n + 1$

$$\int_a^b g'''e' = \sum_{i=1}^n \int_{x_i}^{x_{i+1}} g'''e' = \sum_{i=1}^n v_i \int_{x_i}^{x_{i+1}} e' = \sum_{i=1}^n v_i (e(x_{i+1}) - e(x_i)) = 0.$$

Writing $h = g + e$ and using (1.21)

$$\begin{aligned} \int_a^b (h'')^2 &= \int_a^b (g'' + e'')^2 \\ &= \int_a^b (g'')^2 + \int_a^b (e'')^2 + 2 \int_a^b g''e'' \\ &= \int_a^b (g'')^2 + \int_a^b (e'')^2 \geq \int_a^b (g'')^2 \end{aligned}$$

and the proof is complete. \square

⁸The name spline is inherited from a “physical analogue”, an elastic ruler that is used to draw smooth curves. Heavy weights, called **ducks**, are used to force the ruler to pass through, or near given locations. (Cf. Figure 1.5). The ruler will take a shape that minimizes its potential energy. Since the potential energy is proportional to the integral of the square of the curvature, and the curvature can be approximated by the second derivative it follows from Theorem 1.17 that the mathematical N -spline approximately models the physical spline.



Figure 1.5: A physical spline with ducks.

1.1.6 Computer exercises for section 1.1

Exercise 1.18 (Computing the D_2 -spline)

Let g be the D_2 -spline corresponding to an interval $[a, b]$, a vector $\mathbf{y} \in \mathbb{R}^{n+1}$ and μ_1, μ_{n+1} . The vector $\mathbf{x} = [x_1, \dots, x_n]$ and the coefficient matrix $\mathbf{C} \in \mathbb{R}^{n \times 4}$ in (1.7) are returned in the following algorithm. It uses algorithms 1.8 and 1.9 to solve the tridiagonal linear system.

Algorithm 1.19 (splineint)

```

1 function [x,C]=splineint(a,b,y,mu1,munp1)
2 y=y(:); n=length(y)-1;
3 h=(b-a)/n; x=a:h:b-h; c=ones(n-2,1);
4 [l,u]= trifactor(c,4*ones(n-1,1),c);
5 b1=6/h^2*(y(3:n+1)-2*y(2:n)+y(1:n-1));
6 b1(1)=b1(1)-mu1; b1(n-1)=b1(n-1)-munp1;
7 mu=[mu1; trisolve(l,u,c,b1); munp1];
8 C=zeros(4*n,1);
9 C(1:4:4*n-3)=y(1:n);
10 C(2:4:4*n-2)=(y(2:n+1)-y(1:n))/h-h*mu(1:n)/3-h*mu(2:n+1)/6;
11 C(3:4:4*n-1)=mu(1:n)/2;
12 C(4:4:4*n)=(mu(2:n+1)-mu(1:n))/(6*h);
13 C=reshape(C,4,n)';

```

Use the algorithm to compute the $c_{i,j}$ in Example 1.7.

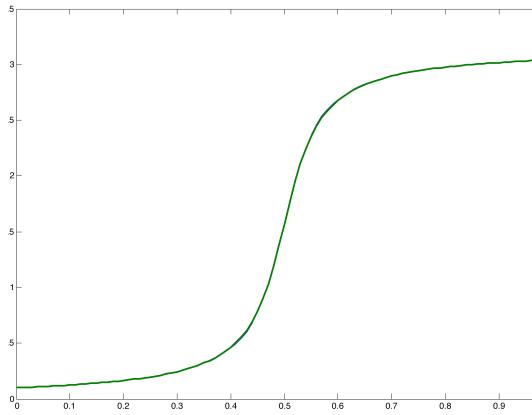


Figure 1.6: The cubic spline interpolating $f(x) = \arctan(10x) + \pi/2$ at 14 equidistant sites on $[-1, 1]$. The exact function is also shown.

Exercise 1.20 (Spline evaluation)

To plot a piecewise polynomial g in the form (1.4) we need to compute values $g(r_j)$ at a number of sites $\mathbf{r} = [r_1, \dots, r_m] \in \mathbb{R}^m$ for some reasonably large integer m . To determine $g(r_j)$ for some j we need to find an integer i_j so that $g(r_j) = p_{i_j}(r_j)$.

Given $k \in \mathbb{N}$, $\mathbf{t} = [t_1, \dots, t_k]$ and a real number x . We consider the problem of computing an integer i so that $i = 1$ if $x < t_2$, $i = k$ if $x \geq t_k$, and $t_i \leq x < t_{i+1}$ otherwise. If $\mathbf{x} \in \mathbb{R}^m$ is a vector then an m -vector \mathbf{i} should be computed, such that the j th component of \mathbf{i} gives the location of the j th component of \mathbf{x} . The following Matlab function determines $\mathbf{i} = [i_1, \dots, i_m]$. It uses the built in Matlab functions `length`, `min`, `sort`, `find`.

Algorithm 1.21 (`findsubintervals`)

```

1 function i = findsubintervals(t, x)
2 %i = findsubintervals(t, x)
3 k= length(t); m= length(x);
4 if k<2
5     i=ones(m,1);
6 else
7     t(1)=min(x(1),t(1))-1;
8     [~,j]=sort([t(:)',x(:)'']);
9     i=(find(j>k)-(1:m))';
10 end
```

Use `findsubintervals` and the following algorithm to make the plots in

Figure 1.6.

Algorithm 1.22 (splineval) Given the output \mathbf{x}, \mathbf{C} of Algorithm 1.19 defining a cubic spline g , and a vector \mathbf{X} . The vector $\mathbf{G} = g(\mathbf{X})$ is computed.

```

1 function [X,G]=splineval(x,C,X)
2 m=length(X);
3 i=findsubintervals(x,X);
4 G=zeros(m,1);
5 for j=1:m
6     k=i(j);
7     t=X(j)-x(k);
8     G(j)=[1,t,t^2,t^3]*C(k,:)';
9 end

```

1.2 A two point boundary value problem

Consider the simple **two point boundary value problem**

$$-u''(x) = f(x), \quad x \in [0, 1], \quad u(0) = 0, \quad u(1) = 0, \quad (1.22)$$

where f is a given continuous function on $[0, 1]$ and u is an unknown function. This problem is also known as the **one-dimensional (1D) Poisson problem**. In principle it is easy to solve (1.22) exactly. We just integrate f twice and determine the two integration constants so that the homogeneous boundary conditions $u(0) = u(1) = 0$ are satisfied. For example, if $f(x) = 1$ then $u(x) = x(x - 1)/2$ is the solution.

Suppose f cannot be integrated exactly. Problem (1.22) can then be solved approximately using the **finite difference method**. We need a difference approximation to the second derivative. If g is a function differentiable at x then

$$g'(x) = \lim_{h \rightarrow 0} \frac{g(x + \frac{h}{2}) - g(x - \frac{h}{2})}{h}$$

and applying this to a function u that is twice differentiable at x

$$\begin{aligned} u''(x) &= \lim_{h \rightarrow 0} \frac{u'(x + \frac{h}{2}) - u'(x - \frac{h}{2})}{h} = \lim_{h \rightarrow 0} \frac{\frac{u(x+h)-u(x)}{h} - \frac{u(x)-u(x-h)}{h}}{h} \\ &= \lim_{h \rightarrow 0} \frac{u(x+h) - 2u(x) + u(x-h)}{h^2}. \end{aligned}$$

To define the points where this difference approximation is used we choose a positive integer m , let $h := 1/(m+1)$ be the discretization parameter, and replace the interval $[0, 1]$ by grid points $x_j := jh$ for $j = 0, 1, \dots, m+1$. We then obtain

approximations v_j to the exact solution $u(x_j)$ for $j = 1, \dots, m$ by replacing the differential equation by the difference equation

$$\frac{-v_{j-1} + 2v_j - v_{j+1}}{h^2} = f(jh), \quad j = 1, \dots, m, \quad v_0 = v_{m+1} = 0.$$

Moving the h^2 factor to the right hand side this can be written as an $m \times m$ linear system

$$\mathbf{T}\mathbf{v} = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & \ddots & \ddots & \ddots \\ & & & 0 \\ & & -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_{m-1} \\ v_m \end{bmatrix} = h^2 \begin{bmatrix} f(h) \\ f(2h) \\ \vdots \\ f((m-1)h) \\ f(mh) \end{bmatrix} =: \mathbf{b}. \quad (1.23)$$

The matrix \mathbf{T} is called the **second derivative matrix** and will occur frequently in this book. It is our second example of a tridiagonal matrix, $\mathbf{T} = \text{tridiag}(a_i, d_i, c_i) \in \mathbb{R}^{m \times m}$, where in this case $a_i = c_i = -1$ and $d_i = 2$ for all i .

1.2.1 Diagonal dominance

We want to show that (1.23) has a unique solution. Note that \mathbf{T} is not strictly diagonally dominant. However, \mathbf{T} is weakly diagonally dominant in accordance with the following definition.

Definition 1.23 (Diagonal dominance)

*The matrix $\mathbf{A} = [a_{ij}] \in \mathbb{C}^{n \times n}$ is **weakly diagonally dominant** if*

$$|a_{ii}| \geq \sum_{j \neq i} |a_{ij}|, \quad i = 1, \dots, n. \quad (1.24)$$

We showed in Theorem 1.6 that a strictly diagonally dominant matrix is nonsingular. This is in general not true in the weakly diagonally dominant case. Consider the 3 matrices

$$\mathbf{A}_1 = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 1 \end{bmatrix}, \quad \mathbf{A}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{A}_3 = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}.$$

They are all weakly diagonally dominant, but \mathbf{A}_1 and \mathbf{A}_2 are singular, while \mathbf{A}_3 is nonsingular. Indeed, for \mathbf{A}_1 column two is the sum of columns one and three, \mathbf{A}_2 has a zero row, and $\det(\mathbf{A}_3) = 4 \neq 0$. It follows that for the nonsingularity and existence of an LU-factorization we need some conditions in addition to weak diagonal dominance. Here are some sufficient conditions.

Theorem 1.24 (Weak diagonal dominance)

Suppose $\mathbf{A} = \text{tridiag}(a_i, d_i, c_i) \in \mathbb{C}^{n \times n}$ is tridiagonal and weakly diagonally dominant. If in addition $|d_1| > |c_1|$ and $a_i \neq 0$ for $i = 1, \dots, n-2$, then \mathbf{A} has a unique LU factorization (1.15). If in addition $d_n \neq 0$, then \mathbf{A} is nonsingular.

Proof. The proof is similar to the proof of Theorem 1.6. The matrix \mathbf{A} has an LU factorization if the u_k 's in (1.16) are nonzero for $k = 1, \dots, n-1$. For this it is sufficient to show by induction that $|u_k| > |c_k|$ for $k = 1, \dots, n-1$. By assumption $|u_1| = |d_1| > |c_1|$. Suppose $|u_k| > |c_k|$ for some $1 \leq k \leq n-2$. Then $|c_k|/|u_k| < 1$ and by (1.16) and since $a_k \neq 0$

$$|u_{k+1}| = |d_{k+1} - l_k c_k| = |d_{k+1} - \frac{a_k c_k}{u_k}| \geq |d_{k+1}| - \frac{|a_k||c_k|}{|u_k|} > |d_{k+1}| - |a_k|. \quad (1.25)$$

This also holds for $k = n-1$ if $a_{n-1} \neq 0$. By (1.25) and weak diagonal dominance $|u_{k+1}| > |d_{k+1}| - |a_k| \geq |c_{k+1}|$ and it follows by induction that an LU factorization exists. It is unique since any LU factorization must satisfy (1.16). For the nonsingularity we need to show that $u_n \neq 0$. For then by Lemma 1.35, both \mathbf{L} and \mathbf{U} are nonsingular, and this is equivalent to $\mathbf{A} = \mathbf{LU}$ being nonsingular. If $a_{n-1} \neq 0$ then by (1.16) $|u_n| > |d_n| - |a_{n-1}| \geq 0$ by weak diagonal dominance, while if $a_{n-1} = 0$ then again by (1.25) $|u_n| \geq |d_n| > 0$. \square

Consider now the special system $\mathbf{T}\mathbf{v} = \mathbf{b}$ given by (1.23). The matrix \mathbf{T} is weakly diagonally dominant and satisfies the additional conditions in Theorem 1.24. Thus it is nonsingular and we can solve the system in $O(n)$ arithmetic operations using the algorithms `trifactor` and `trisolve`.

We could use the explicit inverse of \mathbf{T} , given in Exercise 1.26, to compute the solution of $\mathbf{T}\mathbf{v} = \mathbf{b}$ as $\mathbf{v} = \mathbf{T}^{-1}\mathbf{b}$. However this is not a good idea. In fact, all elements in \mathbf{T}^{-1} are nonzero and the calculation of $\mathbf{T}^{-1}\mathbf{b}$ requires $O(n^2)$ operations.

1.2.2 Exercises for section 1.2

Exercise 1.25 (LU factorization of 2. derivative matrix)

Show that $\mathbf{T} = \mathbf{LU}$, where

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ -\frac{1}{2} & 1 & \ddots & & \vdots \\ 0 & -\frac{2}{3} & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -\frac{m-1}{m} & 1 \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} 2 & -1 & 0 & \cdots & 0 \\ 0 & \frac{3}{2} & -1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \frac{m}{m-1} & -1 \\ 0 & \cdots & \cdots & 0 & \frac{m+1}{m} \end{bmatrix} \quad (1.26)$$

is the LU factorization of \mathbf{T} .

Exercise 1.26 (Inverse of the 2. derivative matrix)

Let $\mathbf{S} \in \mathbb{R}^{m \times m}$ have elements s_{ij} given by

$$s_{i,j} = s_{j,i} = \frac{1}{m+1} j(m+1-i), \quad 1 \leq j \leq i \leq m. \quad (1.27)$$

Show that $\mathbf{ST} = \mathbf{I}$ and conclude that $\mathbf{T}^{-1} = \mathbf{S}$.

Exercise 1.27 (Central difference approximation of 2. derivative)

Consider

$$\delta^2 f(x) := \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}, \quad h > 0, \quad f : [x-h, x+h] \rightarrow \mathbb{R}.$$

1. Show using Taylor expansion that if $f \in C^2[x-h, x+h]$ then for some η_2

$$\delta^2 f(x) = f''(\eta_2), \quad x-h < \eta_2 < x+h.$$

2. If $f \in C^4[x-h, x+h]$ then for some η_4

$$\delta^2 f(x) = f''(x) + \frac{h^2}{12} f^{(4)}(\eta_4), \quad x-h < \eta_4 < x+h.$$

$\delta^2 f(x)$ is known as the **central difference approximation** to the second derivative at x .

Exercise 1.28 (Two point boundary value problem)

We consider a finite difference method for the two point boundary value problem

$$\begin{aligned} -u''(x) + r(x)u'(x) + q(x)u(x) &= f(x), \text{ for } x \in [a, b], \\ u(a) &= g_0, \quad u(b) = g_1. \end{aligned} \quad (1.28)$$

We assume that the given functions f, q and r are continuous on $[a, b]$ and that $q(x) \geq 0$ for $x \in [a, b]$. It can then be shown that (1.28) has a unique solution u .

To solve (1.28) numerically we choose $m \in \mathbb{N}$, $h = (b-a)/(m+1)$, $x_j = a+jh$ for $j = 0, 1, \dots, m+1$ and solve the difference equation

$$\frac{-v_{j-1} + 2v_j - v_{j+1}}{h^2} + r(x_j) \frac{v_{j+1} - v_{j-1}}{2h} + q(x_j)v_j = f(x_j), \quad j = 1, \dots, m, \quad (1.29)$$

with $v_0 = g_0$ and $v_{m+1} = g_1$.

- (a) Show that (1.29) leads to a tridiagonal linear system $\mathbf{Av} = \mathbf{b}$, where $\mathbf{A} = \text{tridiag}(a_j, d_j, c_j) \in \mathbb{R}^{m \times m}$ has elements

$$a_j = -1 - \frac{h}{2}r(x_j), \quad c_j = -1 + \frac{h}{2}r(x_j), \quad d_j = 2 + h^2q(x_j),$$

and

$$b_j = \begin{cases} h^2f(x_1) - a_1g_0, & \text{if } j = 1, \\ h^2f(x_j), & \text{if } 2 \leq j \leq m-1, \\ h^2f(x_m) - c_mg_1, & \text{if } j = m. \end{cases}$$

- (b) Show that the linear system satisfies the conditions in Theorem 1.24 if the spacing h is so small that $\frac{h}{2}|r(x)| < 1$ for all $x \in [a, b]$.

- (c) Propose a method to find v_1, \dots, v_m .

Exercise 1.29 (Two point boundary value problem; computation)

- (a) Consider the problem (1.28) with $r = 0$, $f = q = 1$ and boundary conditions $u(0) = 1$, $u(1) = 0$. The exact solution is $u(x) = 1 - \sinh x / \sinh 1$. Write a computer program to solve (1.29) for $h = 0.1, 0.05, 0.025, 0.0125$, and compute the "error" $\max_{1 \leq j \leq m} |u(x_j) - v_j|$ for each h .
- (b) Make a combined plot of the solution u and the computed points v_j , $j = 0, \dots, m+1$ for $h = 0.1$.
- (c) One can show that the error is proportional to h^p for some integer p . Estimate p based on the error for $h = 0.1, 0.05, 0.025, 0.0125$.

1.3 An Eigenvalue Problem

Recall that if $\mathbf{A} \in \mathbb{C}^{n \times n}$ is a square matrix and $\mathbf{Ax} = \lambda\mathbf{x}$ for some nonzero $\mathbf{x} \in \mathbb{C}^n$, then $\lambda \in \mathbb{C}$ is called an **eigenvalue** and \mathbf{x} an **eigenvector**. We call (λ, \mathbf{x}) an **eigenpair** of \mathbf{A} .

1.3.1 The Buckling of a Beam

Consider a horizontal beam of length L located between 0 and L on the x -axis of the plane. We assume that the beam is fixed at $x = 0$ and $x = L$ and that a force F is applied at $(L, 0)$ in the direction towards the origin. This situation can be modeled by the boundary value problem

$$Ry''(x) = -Fy(x), \quad y(0) = y(L) = 0, \tag{1.30}$$

where $y(x)$ is the vertical displacement of the beam at x , and R is a constant defined by the rigidity of the beam. We can transform the problem to the unit interval $[0, 1]$ by considering the function $u : [0, 1] \rightarrow \mathbb{R}$ given by $u(t) := y(tL)$. Since $u''(t) = L^2 y''(tL)$, the problem (1.30) then becomes

$$u''(t) = -Ku(t), \quad u(0) = u(1) = 0, \quad K := \frac{FL^2}{R}. \quad (1.31)$$

Clearly $u = 0$ is a solution, but we can have nonzero solutions corresponding to certain values of the K known as eigenvalues. The corresponding function u is called an eigenfunction. If $F = 0$ then $K = 0$ and $u = 0$ is the only solution, but if the force is increased it will reach a critical value where the beam will buckle and maybe break. This critical value corresponds to the smallest eigenvalue of (1.31). With $u(t) = \sin(\pi t)$ we find $u''(t) = -\pi^2 u(t)$ and this u is a solution if $K = \pi^2$. It can be shown that this is the smallest eigenvalue of (1.31) and solving for F we find $F = \frac{\pi^2 R}{L^2}$.

We can approximate this eigenvalue numerically. Using the same finite difference approximation as in Section 1.2 we obtain

$$\frac{-v_{j-1} + 2v_j - v_{j+1}}{h^2} = Kv_j, \quad j = 1, \dots, m, \quad h = \frac{1}{m+1}, \quad v_0 = v_{m+1} = 0,$$

where $v_j \approx u(jh)$ for $j = 0, \dots, m+1$. If we define $\lambda := h^2 K$ then we obtain the equation

$$\mathbf{T}\mathbf{v} = \lambda\mathbf{v}, \quad \text{with } \mathbf{v} = [v_1, \dots, v_m]^T, \quad (1.32)$$

and $\mathbf{T} := \text{tridiag}_m(-1, 2, -1)$ is the matrix given by (1.23). The problem now is to determine the eigenvalues of \mathbf{T} . Normally we would need a numerical method to determine the eigenvalues of a matrix, but for this simple problem the eigenvalues can be determined exactly. We show in the next subsection that the smallest eigenvalue of (1.32) is given by $\lambda = 4 \sin^2(\pi h/2)$. Since $\lambda = h^2 K = \frac{h^2 FL^2}{R}$ we can solve for F to obtain

$$F = \frac{4 \sin^2(\pi h/2) R}{h^2 L^2}.$$

For small h this is a good approximation to the value $\frac{\pi^2 R}{L^2}$ we computed above.

Exercise 1.30 (Approximate force) Show that

$$F = \frac{4 \sin^2(\pi h/2) R}{h^2 L^2} = \frac{\pi^2 R}{L^2} + O(h^2).$$

1.3.2 The eigenpairs of the 1D test matrix

The second derivative matrix $\mathbf{T} = \text{tridiag}(-1, 2, -1)$ is a special case of the tridiagonal matrix

$$\mathbf{T}_1 := \text{tridiag}(a, d, a) \quad (1.33)$$

where $a, d \in \mathbb{R}$. We call this the **1D test matrix**. It is symmetric and strictly diagonally dominant if $|d| > 2|a|$.

We show that the eigenvectors are the columns of the **sine matrix** defined by

$$\mathbf{S} = \left[\sin \frac{jk\pi}{m+1} \right]_{j,k=1}^m \in \mathbb{R}^{m \times m}. \quad (1.34)$$

For $m = 3$,

$$\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3] = \begin{bmatrix} \sin \frac{\pi}{4} & \sin \frac{2\pi}{4} & \sin \frac{3\pi}{4} \\ \sin \frac{2\pi}{4} & \sin \frac{4\pi}{4} & \sin \frac{6\pi}{4} \\ \sin \frac{3\pi}{4} & \sin \frac{6\pi}{4} & \sin \frac{9\pi}{4} \end{bmatrix} = \begin{bmatrix} t & 1 & t \\ 1 & 0 & -1 \\ t & -1 & t \end{bmatrix}, \quad t := \frac{1}{\sqrt{2}}.$$

Lemma 1.31 (Eigenpairs of 1D test matrix)

Suppose $\mathbf{T}_1 = (t_{kj})_{k,j} = \text{tridiag}(a, d, a) \in \mathbb{R}^{m \times m}$ with $m \geq 2$, $a, d \in \mathbb{R}$, and let $h = 1/(m+1)$.

1. We have $\mathbf{T}_1 \mathbf{s}_j = \lambda_j \mathbf{s}_j$ for $j = 1, \dots, m$, where

$$\mathbf{s}_j = [\sin(j\pi h), \sin(2j\pi h), \dots, \sin(mj\pi h)]^T, \quad (1.35)$$

$$\lambda_j = d + 2a \cos(j\pi h). \quad (1.36)$$

2. The eigenvalues are distinct and the eigenvectors are orthogonal

$$\mathbf{s}_j^T \mathbf{s}_k = \frac{m+1}{2} \delta_{j,k}, \quad j, k = 1, \dots, m. \quad (1.37)$$

Proof. We find for $1 < k < m$

$$\begin{aligned} (\mathbf{T}_1 \mathbf{s}_j)_k &= \sum_{l=1}^m t_{k,l} \sin(lj\pi h) = a[\sin((k-1)j\pi h) + \sin((k+1)j\pi h)] + d \sin(kj\pi h) \\ &= 2a \cos(j\pi h) \sin(kj\pi h) + d \sin(kj\pi h) = \lambda_j s_{k,j}. \end{aligned}$$

This also holds for $k = 1, m$, and part 1 follows. Since $j\pi h = j\pi/(m+1) \in (0, \pi)$ for $j = 1, \dots, m$ and the cosine function is strictly monotone decreasing on $(0, \pi)$ the eigenvalues are distinct, and since \mathbf{T}_1 is symmetric it follows from Lemma 1.32 below that the eigenvectors \mathbf{s}_j are orthogonal. To finish the proof of (1.37) we compute

$$\begin{aligned} \mathbf{s}_j^T \mathbf{s}_j &= \sum_{k=1}^m \sin^2(kj\pi h) = \sum_{k=0}^m \sin^2(kj\pi h) = \frac{1}{2} \sum_{k=0}^m (1 - \cos(2kj\pi h)) \\ &= \frac{m+1}{2} - \frac{1}{2} \sum_{k=0}^m \cos(2kj\pi h) = \frac{m+1}{2}, \end{aligned}$$

since the last cosine sum is zero. We show this by summing a geometric series of complex exponentials. With $i = \sqrt{-1}$ we find

$$\sum_{k=0}^m \cos(2kj\pi h) + i \sum_{k=0}^m \sin(2kj\pi h) = \sum_{k=0}^m e^{2ikj\pi h} = \frac{e^{2i(m+1)j\pi h} - 1}{e^{2ij\pi h} - 1} = 0,$$

and (1.37) follows. \square

Recall that the conjugate transpose of a matrix is defined by $\mathbf{A}^* := \overline{\mathbf{A}}^T$, where $\overline{\mathbf{A}}$ is obtained from \mathbf{A} by taking the complex conjugate of all elements. A matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ is **Hermitian** if $\mathbf{A}^* = \mathbf{A}$. A real symmetric matrix is Hermitian.

Lemma 1.32 (Eigenpairs of a Hermitian matrix)

The eigenvalues of a Hermitian matrix are real. Moreover, eigenvectors corresponding to distinct eigenvalues are orthogonal.

Proof. Suppose $\mathbf{A}^* = \mathbf{A}$ and $\mathbf{Ax} = \lambda \mathbf{x}$ with $\mathbf{x} \neq 0$. We multiply both sides of $\mathbf{Ax} = \lambda \mathbf{x}$ from the left by \mathbf{x}^* and divide by $\mathbf{x}^* \mathbf{x}$ to obtain $\lambda = \frac{\mathbf{x}^* \mathbf{Ax}}{\mathbf{x}^* \mathbf{x}}$. Taking complex conjugates we find $\bar{\lambda} = \lambda^* = \frac{(\mathbf{x}^* \mathbf{Ax})^*}{(\mathbf{x}^* \mathbf{x})^*} = \frac{\mathbf{x}^* \mathbf{A}^* \mathbf{x}}{\mathbf{x}^* \mathbf{x}} = \frac{\mathbf{x}^* \mathbf{Ax}}{\mathbf{x}^* \mathbf{x}} = \lambda$, and λ is real. Suppose that (λ, \mathbf{x}) and (μ, \mathbf{y}) are two eigenpairs for \mathbf{A} with $\mu \neq \lambda$. Multiplying $\mathbf{Ax} = \lambda \mathbf{x}$ by \mathbf{y}^* gives

$$\lambda \mathbf{y}^* \mathbf{x} = \mathbf{y}^* \mathbf{Ax} = (\mathbf{x}^* \mathbf{A}^* \mathbf{y})^* = (\mathbf{x}^* \mathbf{Ay})^* = (\mu \mathbf{x}^* \mathbf{y})^* = \mu \mathbf{y}^* \mathbf{x},$$

using that μ is real. Since $\lambda \neq \mu$ it follows that $\mathbf{y}^* \mathbf{x} = 0$, which means that \mathbf{x} and \mathbf{y} are orthogonal. \square

Exercise 1.33 (Eigenpairs \mathbf{T} of order 2) Compute directly the eigenvalues and eigenvectors for \mathbf{T} when $n = 2$ and thus verify Lemma 1.31 in this case.

1.4 Block Multiplication and Triangular Matrices

Block multiplication is a powerful and essential tool for dealing with matrices. It will be used extensively in this book. We will also need some basic facts about triangular matrices.

1.4.1 Block multiplication

A rectangular matrix \mathbf{A} can be partitioned into submatrices by drawing horizontal lines between selected rows and vertical lines between selected columns. For

example, the matrix

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

can be partitioned as

$$(i) \quad \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} = \left[\begin{array}{c|cc} 1 & 2 & 3 \\ \hline 4 & 5 & 6 \\ 7 & 8 & 9 \end{array} \right], \quad (ii) \quad [\mathbf{a}_{:1}, \mathbf{a}_{:2}, \mathbf{a}_{:3}] = \left[\begin{array}{c|c|c} 1 & 2 & 3 \\ \hline 4 & 5 & 6 \\ 7 & 8 & 9 \end{array} \right],$$

$$(iii) \quad \begin{bmatrix} \mathbf{a}_{1:}^T \\ \mathbf{a}_{2:}^T \\ \mathbf{a}_{3:}^T \end{bmatrix} = \left[\begin{array}{ccc} 1 & 2 & 3 \\ \hline 4 & 5 & 6 \\ 7 & 8 & 9 \end{array} \right], \quad (iv) \quad [\mathbf{A}_{11}, \mathbf{A}_{12}] = \left[\begin{array}{c|cc} 1 & 2 & 3 \\ \hline 4 & 5 & 6 \\ 7 & 8 & 9 \end{array} \right].$$

In (i) the matrix \mathbf{A} is divided into four submatrices

$$\mathbf{A}_{11} = [1], \quad \mathbf{A}_{12} = [2, 3], \quad \mathbf{A}_{21} = \begin{bmatrix} 4 \\ 7 \end{bmatrix}, \quad \text{and} \quad \mathbf{A}_{22} = \begin{bmatrix} 5 & 6 \\ 8 & 9 \end{bmatrix},$$

while in (ii) and (iii) \mathbf{A} has been partitioned into columns and rows, respectively. The submatrices in a partition are often referred to as **blocks** and a partitioned matrix is sometimes called a **block matrix**.

In the following we assume that $\mathbf{A} \in \mathbb{C}^{m \times p}$ and $\mathbf{B} \in \mathbb{C}^{p \times n}$. Here are some rules and observations for block multiplication.

1. If $\mathbf{B} = [\mathbf{b}_{:1}, \dots, \mathbf{b}_{:n}]$ is partitioned into columns then the partition of the product \mathbf{AB} into columns is

$$\mathbf{AB} = [\mathbf{Ab}_{:1}, \mathbf{Ab}_{:2}, \dots, \mathbf{Ab}_{:n}].$$

In particular, if \mathbf{I} is the identity matrix of order p then

$$\mathbf{A} = \mathbf{AI} = \mathbf{A} [\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_p] = [\mathbf{Ae}_1, \mathbf{Ae}_2, \dots, \mathbf{Ae}_p]$$

and we see that column j of \mathbf{A} can be written \mathbf{Ae}_j for $j = 1, \dots, p$.

2. Similarly, if \mathbf{A} is partitioned into rows then

$$\mathbf{AB} = \begin{bmatrix} \mathbf{a}_{1:}^T \\ \mathbf{a}_{2:}^T \\ \vdots \\ \mathbf{a}_{m:}^T \end{bmatrix} \mathbf{B} = \begin{bmatrix} \mathbf{a}_{1:}^T \mathbf{B} \\ \mathbf{a}_{2:}^T \mathbf{B} \\ \vdots \\ \mathbf{a}_{m:}^T \mathbf{B} \end{bmatrix},$$

and taking $\mathbf{A} = \mathbf{I}$ it follows that row i of \mathbf{B} can be written $\mathbf{e}_i^T \mathbf{B}$ for $i = 1, \dots, m$.

3. It is often useful to write the matrix-vector product $\mathbf{A}\mathbf{x}$ as a linear combination of the columns of \mathbf{A}

$$\mathbf{A}\mathbf{x} = x_1 \mathbf{a}_{:1} + x_2 \mathbf{a}_{:2} + \cdots + x_p \mathbf{a}_{:p}.$$

4. If $\mathbf{B} = [\mathbf{B}_1, \mathbf{B}_2]$, where $\mathbf{B}_1 \in \mathbb{C}^{p \times r}$ and $\mathbf{B}_2 \in \mathbb{C}^{p \times (n-r)}$ then

$$\mathbf{A}[\mathbf{B}_1, \mathbf{B}_2] = [\mathbf{A}\mathbf{B}_1, \mathbf{A}\mathbf{B}_2].$$

This follows from Rule 1. by an appropriate grouping of columns.

5. If $\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \end{bmatrix}$, where $\mathbf{A}_1 \in \mathbb{C}^{k \times p}$ and $\mathbf{A}_2 \in \mathbb{C}^{(m-k) \times p}$ then

$$\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \end{bmatrix} \mathbf{B} = \begin{bmatrix} \mathbf{A}_1 \mathbf{B} \\ \mathbf{A}_2 \mathbf{B} \end{bmatrix}.$$

This follows from Rule 2. by a grouping of rows.

6. If $\mathbf{A} = [\mathbf{A}_1, \mathbf{A}_2]$ and $\mathbf{B} = \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \end{bmatrix}$, where $\mathbf{A}_1 \in \mathbb{C}^{m \times s}$, $\mathbf{A}_2 \in \mathbb{C}^{m \times (p-s)}$, $\mathbf{B}_1 \in \mathbb{C}^{s \times n}$ and $\mathbf{B}_2 \in \mathbb{C}^{(p-s) \times n}$ then

$$[\mathbf{A}_1, \mathbf{A}_2] \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \end{bmatrix} = [\mathbf{A}_1 \mathbf{B}_1 + \mathbf{A}_2 \mathbf{B}_2].$$

Indeed, $(\mathbf{AB})_{ij} = \sum_{k=1}^p a_{ik} b_{kj} = \sum_{k=1}^s a_{ik} b_{kj} + \sum_{k=s+1}^p a_{ik} b_{kj} = (\mathbf{A}_1 \mathbf{B}_1)_{ij} + (\mathbf{A}_2 \mathbf{B}_2)_{ij} = (\mathbf{A}_1 \mathbf{B}_1 + \mathbf{A}_2 \mathbf{B}_2)_{ij}$.

7. If $\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}$ and $\mathbf{B} = \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ \mathbf{B}_{21} & \mathbf{B}_{22} \end{bmatrix}$ then

$$\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ \mathbf{B}_{21} & \mathbf{B}_{22} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11} \mathbf{B}_{11} + \mathbf{A}_{12} \mathbf{B}_{21} & \mathbf{A}_{11} \mathbf{B}_{12} + \mathbf{A}_{12} \mathbf{B}_{22} \\ \mathbf{A}_{21} \mathbf{B}_{11} + \mathbf{A}_{22} \mathbf{B}_{21} & \mathbf{A}_{21} \mathbf{B}_{12} + \mathbf{A}_{22} \mathbf{B}_{22} \end{bmatrix},$$

provided the vertical partition in \mathbf{A} matches the horizontal one in \mathbf{B} , i.e. the number of columns in \mathbf{A}_{11} and \mathbf{A}_{21} equals the number of rows in \mathbf{B}_{11} and \mathbf{B}_{12} and the number of columns in \mathbf{A} equals the number of rows in \mathbf{B} . To show this we use Rule 4. to obtain

$$\mathbf{AB} = \left[\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{B}_{11} \\ \mathbf{B}_{21} \end{bmatrix}, \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{B}_{12} \\ \mathbf{B}_{22} \end{bmatrix} \right].$$

We complete the proof using Rules 5. and 6.

8. Consider finally the general case. If all the matrix products $\mathbf{A}_{ik} \mathbf{B}_{kj}$ in

$$\mathbf{C}_{ij} = \sum_{k=1}^s \mathbf{A}_{ik} \mathbf{B}_{kj}, \quad i = 1, \dots, p, \quad j = 1, \dots, q$$

are well defined then

$$\begin{bmatrix} A_{11} & \cdots & A_{1s} \\ \vdots & & \vdots \\ A_{p1} & \cdots & A_{ps} \end{bmatrix} \begin{bmatrix} B_{11} & \cdots & B_{1q} \\ \vdots & & \vdots \\ B_{s1} & \cdots & B_{sq} \end{bmatrix} = \begin{bmatrix} C_{11} & \cdots & C_{1q} \\ \vdots & & \vdots \\ C_{p1} & \cdots & C_{pq} \end{bmatrix}.$$

The requirements are that

- the number of columns in \mathbf{A} is equal to the number of rows in \mathbf{B} .
- the position of the vertical partition lines in \mathbf{A} has to match the position of the horizontal partition lines in \mathbf{B} . The horizontal lines in \mathbf{A} and the vertical lines in \mathbf{B} can be anywhere.

1.4.2 Triangular matrices

We need some basic facts about triangular matrices and we start with

Lemma 1.34 (Inverse of a block triangular matrix)

Suppose

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{0} & \mathbf{A}_{22} \end{bmatrix}$$

where \mathbf{A} , \mathbf{A}_{11} and \mathbf{A}_{22} are square matrices. Then \mathbf{A} is nonsingular if and only if both \mathbf{A}_{11} and \mathbf{A}_{22} are nonsingular. In that case

$$\mathbf{A}^{-1} = \begin{bmatrix} \mathbf{A}_{11}^{-1} & \mathbf{C} \\ \mathbf{0} & \mathbf{A}_{22}^{-1} \end{bmatrix}, \quad (1.38)$$

for some matrix \mathbf{C} .

Proof. Suppose \mathbf{A} is nonsingular. We partition $\mathbf{B} := \mathbf{A}^{-1}$ conformally with \mathbf{A} and have

$$\mathbf{B}\mathbf{A} = \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ \mathbf{B}_{21} & \mathbf{B}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{0} & \mathbf{A}_{22} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} = \mathbf{I}$$

Using block-multiplication we find

$$\mathbf{B}_{11}\mathbf{A}_{11} = \mathbf{I}, \quad \mathbf{B}_{21}\mathbf{A}_{11} = \mathbf{0}, \quad \mathbf{B}_{21}\mathbf{A}_{12} + \mathbf{B}_{22}\mathbf{A}_{22} = \mathbf{I}, \quad \mathbf{B}_{11}\mathbf{A}_{12} + \mathbf{B}_{12}\mathbf{A}_{22} = \mathbf{0}.$$

The first equation implies that \mathbf{A}_{11} is nonsingular, this in turn implies that $\mathbf{B}_{21} = \mathbf{0}\mathbf{A}_{11}^{-1} = \mathbf{0}$ in the second equation, and then the third equation simplifies to $\mathbf{B}_{22}\mathbf{A}_{22} = \mathbf{I}$. We conclude that also \mathbf{A}_{22} is nonsingular. From the fourth equation we find

$$\mathbf{B}_{12} = \mathbf{C} = -\mathbf{A}_{11}^{-1}\mathbf{A}_{12}\mathbf{A}_{22}^{-1}.$$

Conversely, if \mathbf{A}_{11} and \mathbf{A}_{22} are nonsingular then

$$\begin{bmatrix} \mathbf{A}_{11}^{-1} & -\mathbf{A}_{11}^{-1}\mathbf{A}_{12}\mathbf{A}_{22}^{-1} \\ \mathbf{0} & \mathbf{A}_{22}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{0} & \mathbf{A}_{22} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} = \mathbf{I}$$

and \mathbf{A} is nonsingular with the indicated inverse. \square

Consider now a triangular matrix.

Lemma 1.35 (Inverse of a triangular matrix)

An upper (lower) triangular matrix $\mathbf{A} = [a_{ij}] \in \mathbb{C}^{n \times n}$ is nonsingular if and only if the diagonal elements a_{ii} , $i = 1, \dots, n$ are nonzero. In that case the inverse is upper (lower) triangular with diagonal elements a_{ii}^{-1} , $i = 1, \dots, n$.

Proof. We use induction on n . The result holds for $n = 1$. The 1-by-1 matrix $\mathbf{A} = [a_{11}]$ is nonsingular if and only if $a_{11} \neq 0$ and in that case $\mathbf{A}^{-1} = [a_{11}^{-1}]$. Suppose the result holds for $n = k$ and let $\mathbf{A} \in \mathbb{C}^{(k+1) \times (k+1)}$ be upper triangular. We partition \mathbf{A} in the form

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_k & \mathbf{a}_k \\ \mathbf{0} & a_{k+1,k+1} \end{bmatrix}$$

and note that $\mathbf{A}_k \in \mathbb{C}^{k \times k}$ is upper triangular. By Lemma 1.34 \mathbf{A} is nonsingular if and only if \mathbf{A}_k and $(a_{k+1,k+1})$ are nonsingular and in that case

$$\mathbf{A}^{-1} = \begin{bmatrix} \mathbf{A}_k^{-1} & \mathbf{c} \\ \mathbf{0} & a_{k+1,k+1}^{-1} \end{bmatrix},$$

for some $\mathbf{c} \in \mathbb{C}^n$. By the induction hypothesis \mathbf{A}_k is nonsingular if and only if the diagonal elements a_{11}, \dots, a_{kk} of \mathbf{A}_k are nonzero and in that case \mathbf{A}_k^{-1} is upper triangular with diagonal elements a_{ii}^{-1} , $i = 1, \dots, k$. The result for \mathbf{A} follows. \square

Lemma 1.36 (Product of triangular matrices)

The product $\mathbf{C} = \mathbf{AB} = (c_{ij})$ of two upper (lower) triangular matrices $\mathbf{A} = (a_{ij})$ and $\mathbf{B} = (b_{ij})$ is upper (lower) triangular with diagonal elements $c_{ii} = a_{ii}b_{ii}$ for all i .

Proof. Exercise. \square

A matrix is called **unit triangular** if it is triangular with 1's on the diagonal.

Lemma 1.37 (Unit triangular matrices)

For a unit upper (lower) triangular matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$:

1. \mathbf{A} is nonsingular and the inverse is unit upper(lower) triangular.
2. The product of two unit upper (lower) triangular matrices is unit upper (lower) triangular.

Proof. 1. follows from Lemma 1.35, while Lemma 1.36 implies 2. \square

1.4.3 Exercises for section 1.4

Exercise 1.38 (Matrix element as a quadratic form)

For any matrix \mathbf{A} show that $a_{ij} = \mathbf{e}_i^T \mathbf{A} \mathbf{e}_j$ for all i, j .

Exercise 1.39 (Outer product expansion of a matrix)

For any matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$ show that $\mathbf{A} = \sum_{i=1}^m \sum_{j=1}^n a_{ij} \mathbf{e}_i \mathbf{e}_j^T$.

Exercise 1.40 (The product $\mathbf{A}^T \mathbf{A}$)

Let $\mathbf{B} = \mathbf{A}^T \mathbf{A}$. Explain why this product is defined for any matrix \mathbf{A} . Show that $b_{ij} = \mathbf{a}_{:i}^T \mathbf{a}_{:j}$ for all i, j .

Exercise 1.41 (Outer product expansion)

For $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{B} \in \mathbb{R}^{p \times n}$ show that

$$\mathbf{AB}^T = \mathbf{a}_{:1} \mathbf{b}_{:1}^T + \mathbf{a}_{:2} \mathbf{b}_{:2}^T + \cdots + \mathbf{a}_{:n} \mathbf{b}_{:n}^T.$$

This is called the **outer product expansion** of the columns of \mathbf{A} and \mathbf{B} .

Exercise 1.42 (System with many right hand sides; compact form)

Suppose $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{B} \in \mathbb{R}^{m \times p}$, and $\mathbf{X} \in \mathbb{R}^{n \times p}$. Show that

$$\mathbf{AX} = \mathbf{B} \iff \mathbf{Ax}_{:j} = \mathbf{b}_{:j}, \quad j = 1, \dots, p.$$

Exercise 1.43 (Block multiplication example)

Suppose $\mathbf{A} = [\mathbf{A}_1, \mathbf{A}_2]$ and $\mathbf{B} = \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{0} \end{bmatrix}$. When is $\mathbf{AB} = \mathbf{A}_1 \mathbf{B}_1$?

Exercise 1.44 (Another block multiplication example)

Suppose $\mathbf{A}, \mathbf{B}, \mathbf{C} \in \mathbb{R}^{n \times n}$ are given in block form by

$$\mathbf{A} := \begin{bmatrix} \lambda & \mathbf{a}^T \\ \mathbf{0} & \mathbf{A}_1 \end{bmatrix}, \quad \mathbf{B} := \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{B}_1 \end{bmatrix}, \quad \mathbf{C} := \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{C}_1 \end{bmatrix},$$

where $\mathbf{A}_1, \mathbf{B}_1, \mathbf{C}_1 \in \mathbb{R}^{(n-1) \times (n-1)}$. Show that

$$\mathbf{CAB} = \begin{bmatrix} \lambda & \mathbf{a}^T \mathbf{B}_1 \\ \mathbf{0} & \mathbf{C}_1 \mathbf{A}_1 \mathbf{B}_1 \end{bmatrix}.$$

1.5 Review Questions

- 1.4.1** How do we define nonsingularity of a matrix?
- 1.4.2** Define the second derivative matrix \mathbf{T} . How did we show that it is nonsingular?
- 1.4.3** Why do we not use the explicit inverse of \mathbf{T} to solve the linear system $\mathbf{T}\mathbf{x} = \mathbf{b}$?
- 1.4.4** What are the eigenpairs of the matrix \mathbf{T} ?
- 1.4.5** Why are the diagonal elements of a Hermitian matrix real?
- 1.4.6** Is the matrix $\begin{bmatrix} 1 & 1+i \\ 1+i & 2 \end{bmatrix}$ Hermitian? Symmetric?
- 1.4.7** Is a weakly diagonally dominant matrix nonsingular?
- 1.4.8** Is a strictly diagonally dominant matrix always nonsingular?
- 1.4.9** Does a tridiagonal matrix always have an LU factorization?

Chapter 2

Gaussian elimination and LU Factorizations



Carl Friedrich Gauss, 1777-1855 (left), Myrick Hascall Doolittle, 1830-1911 (right).

Numerical methods for solving systems of linear equations are often based on writing a matrix as a product of simpler matrices. Such a **factorization** is useful if the corresponding matrix problem for each of the factors is simple to solve, and extra numerical stability issues are not introduced. Examples of a factorization was encountered in Chapter 1 and we saw how an LU factorization can be used to solve certain tridiagonal systems in $O(n)$ operations. Other factorizations based on unitary matrices will be considered later in this book.

In this chapter we first consider Gaussian elimination. Gaussian elimination leads to an LU factorization of the coefficient matrix or more generally to a PLU factorization, if row interchanges are introduced. Here \mathbf{P} is a permutation matrix.

We also consider the general theory of LU factorizations.

2.1 Gaussian Elimination and LU-factorization

2.1.1 3 by 3 example

Gaussian elimination with row interchanges is the classical method for solving n linear equations in n unknowns⁹. We first recall how it works on a 3×3 system.

Example 2.1 (Gaussian elimination on a 3×3 system)

Consider a nonsingular system of three equations in three unknowns:

$$\begin{aligned} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 &= b_1^{(1)}, & \text{I} \\ a_{21}^{(1)}x_1 + a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 &= b_2^{(1)}, & \text{II} \\ a_{31}^{(1)}x_1 + a_{32}^{(1)}x_2 + a_{33}^{(1)}x_3 &= b_3^{(1)}. & \text{III.} \end{aligned}$$

To solve this system by Gaussian elimination suppose $a_{11}^{(1)} \neq 0$. We subtract $l_{21}^{(1)} := a_{21}^{(1)}/a_{11}^{(1)}$ times equation I from equation II and $l_{31}^{(1)} := a_{31}^{(1)}/a_{11}^{(1)}$ times equation I from equation III. The result is

$$\begin{aligned} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 &= b_1^{(1)}, & \text{I} \\ a_{22}^{(2)}x_2 + a_{23}^{(2)}x_3 &= b_2^{(2)}, & \text{II}' \\ a_{32}^{(2)}x_2 + a_{33}^{(2)}x_3 &= b_3^{(2)}, & \text{III}', \end{aligned}$$

where $b_i^{(2)} = b_i^{(1)} - l_{i1}^{(1)}b_1^{(1)}$ for $i = 2, 3$ and $a_{ij}^{(2)} = a_{ij}^{(1)} - l_{i1}^{(1)}a_{1j}^{(1)}$ for $i, j = 2, 3$. If $a_{11}^{(1)} = 0$ and $a_{21}^{(1)} \neq 0$ we first interchange equation I and equation II. If $a_{11}^{(1)} = a_{21}^{(1)} = 0$ we interchange equation I and III. Since the system is nonsingular the first column cannot be zero and an interchange is always possible.

If $a_{22}^{(2)} \neq 0$ we subtract $l_{32}^{(2)} := a_{32}^{(2)}/a_{22}^{(2)}$ times equation II from equation III to obtain

$$\begin{aligned} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 &= b_1^{(1)}, & \text{I} \\ a_{22}^{(2)}x_2 + a_{23}^{(2)}x_3 &= b_2^{(2)}, & \text{II}' \\ a_{33}^{(3)}x_3 &= b_3^{(3)}, & \text{III}'', \end{aligned}$$

where $a_{33}^{(3)} = a_{33}^{(2)} - l_{32}^{(2)}a_{23}^{(2)}$ and $b_3^{(3)} = b_3^{(2)} - l_{32}^{(2)}b_2^{(2)}$. If $a_{22}^{(2)} = 0$ then $a_{32}^{(2)} \neq 0$ (cf. Section 2.4) and we first interchange equation II and equation III. The reduced

⁹The method was known long before Gauss used it in 1809. It was further developed by Doolittle in 1881, see [7].

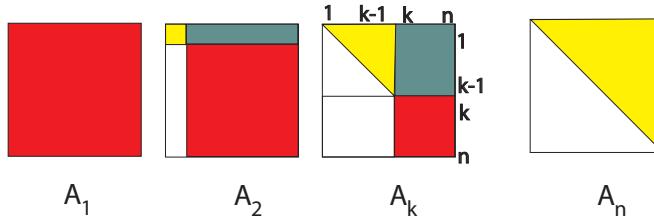


Figure 2.1: Gaussian elimination

system is easy to solve since it is upper triangular. Starting from the bottom and moving upwards we find

$$\begin{aligned} x_3 &= b_3^{(3)} / a_{33}^{(3)} \\ x_2 &= (b_2^{(2)} - a_{23}^{(2)} x_3) / a_{22}^{(2)} \\ x_1 &= (b_1^{(1)} - a_{12}^{(1)} x_2 - a_{13}^{(1)} x_3) / a_{11}^{(1)}. \end{aligned}$$

This is known as **back substitution**. If $a_{kk}^{(k)} \neq 0$, $k = 1, 2$ then

$$\begin{aligned} \mathbf{LU} &:= \begin{bmatrix} 1 & 0 & 0 \\ l_{21}^{(1)} & 1 & 0 \\ l_{31}^{(1)} & l_{32}^{(2)} & 1 \end{bmatrix} \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} \\ 0 & 0 & a_{33}^{(3)} \end{bmatrix} \\ &= \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} \\ l_{21}^{(1)} a_{11}^{(1)} & l_{21}^{(1)} a_{12}^{(1)} + a_{22}^{(2)} & l_{21}^{(1)} a_{13}^{(1)} + a_{23}^{(2)} \\ l_{31}^{(1)} a_{11}^{(1)} & l_{31}^{(1)} a_{12}^{(1)} + l_{32}^{(2)} a_{22}^{(2)} & l_{31}^{(1)} a_{13}^{(1)} + l_{32}^{(2)} a_{23}^{(2)} + a_{33}^{(3)} \end{bmatrix} = \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & a_{23}^{(1)} \\ a_{31}^{(1)} & a_{32}^{(1)} & a_{33}^{(1)} \end{bmatrix}. \end{aligned}$$

Thus Gaussian elimination leads to an LU factorization of the coefficient matrix $\mathbf{A}^{(1)}$ (cf. the proof of Theorem 2.5).

2.1.2 Gauss and LU

In **Gaussian elimination** without row interchanges we start with a linear system $\mathbf{Ax} = \mathbf{b}$ and generate a sequence of equivalent systems $\mathbf{A}^{(k)}\mathbf{x} = \mathbf{b}^{(k)}$ for $k = 1, \dots, n$, where $\mathbf{A}^{(1)} = \mathbf{A}$, $\mathbf{b}^{(1)} = \mathbf{b}$, and $\mathbf{A}^{(k)}$ has zeros under the diagonal in its first $k - 1$ columns. Thus $\mathbf{A}^{(n)}$ is upper triangular and the system $\mathbf{A}^{(n)}\mathbf{x} = \mathbf{b}^{(n)}$ is easy to solve. The process is illustrated in Figure 2.1.

The matrix $\mathbf{A}^{(k)}$ takes the form

$$\mathbf{A}^{(k)} = \left[\begin{array}{ccc|ccc|ccc} a_{1,1}^{(1)} & \cdots & a_{1,k-1}^{(1)} & a_{1,k}^{(1)} & \cdots & a_{1,j}^{(1)} & \cdots & a_{1,n}^{(1)} \\ \ddots & & \vdots & \vdots & & \vdots & & \vdots \\ & a_{k-1,k-1}^{(k-1)} & & a_{k-1,k}^{(k-1)} & \cdots & a_{k-1,j}^{(k-1)} & \cdots & a_{k-1,n}^{(k-1)} \\ \hline & a_{k,k}^{(k)} & \cdots & a_{k,j}^{(k)} & \cdots & a_{k,n}^{(k)} & & \\ & \vdots & & \vdots & & \vdots & & \\ & a_{i,k}^{(k)} & \cdots & a_{i,j}^{(k)} & \cdots & a_{i,n}^{(k)} & & \\ & \vdots & & \vdots & & \vdots & & \\ & a_{n,k}^{(k)} & \cdots & a_{n,j}^{(k)} & \cdots & a_{n,n}^{(k)} & & \end{array} \right]. \quad (2.1)$$

The process transforming $\mathbf{A}^{(k)}$ into $\mathbf{A}^{(k+1)}$ for $k = 1, \dots, n-1$ can be described as follows.

$$\begin{aligned} & \text{for } i = k+1 : n \\ & l_{ik}^{(k)} = a_{ik}^{(k)} / a_{kk}^{(k)} \\ & \text{for } j = k : n \\ & a_{ij}^{(k+1)} = a_{ij}^{(k)} - l_{ik}^{(k)} a_{kj}^{(k)} \end{aligned} \quad (2.2)$$

For $j = k$ it follows from (2.2) that $a_{ik}^{(k+1)} = a_{ik}^{(k)} - \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}} a_{kk}^{(k)} = 0$ for $i = k+1, \dots, n$. Thus $\mathbf{A}^{(k+1)}$ will have zeros under the diagonal in its first k columns and the elimination is carried one step further. The numbers $l_{ik}^{(k)}$ in (2.2) are called **multipliers**.

To characterize matrices for which Gaussian elimination with no row interchanges is possible we start with a definition.

Definition 2.2 (Principal submatrix)

For $k = 1, \dots, n$ the matrices $\mathbf{A}_{[k]} \in \mathbb{C}^{k \times k}$ given by

$$\mathbf{A}_{[k]} := \mathbf{A}(1:k, 1:k) = \begin{bmatrix} a_{11} & \cdots & a_{k1} \\ \vdots & & \vdots \\ a_{k1} & \cdots & a_{kk} \end{bmatrix}$$

are called the **leading principal submatrices** of $\mathbf{A} \in \mathbb{C}^{n \times n}$. More generally, a matrix $\mathbf{B} \in \mathbb{C}^{k \times k}$ is called a **principal submatrix** of \mathbf{A} if $\mathbf{B} = \mathbf{A}(\mathbf{r}, \mathbf{r})$, where $\mathbf{r} = [r_1, \dots, r_k]$ for some $1 \leq r_1 < \dots < r_k \leq n$. Thus,

$$b_{i,j} = a_{r_i, r_j}, \quad i, j = 1, \dots, k.$$

The determinant of a (leading) principal submatrix is called a **(leading) principal minor**.

A principal submatrix is leading if $r_j = j$ for $j = 1, \dots, k$. Also a principal submatrix is special in that it uses the same rows and columns of \mathbf{A} . For $k = 1$ The only principal submatrices of order $k = 1$ are the diagonal elements of \mathbf{A} .

Example 2.3 (Principal submatrices)

The principal submatrices of $\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ are

$$[1], [5], [9], [4 \ 2], [\frac{1}{7} \ \frac{3}{9}], [\frac{5}{8} \ \frac{6}{9}], \mathbf{A}.$$

The leading principal submatrices are

$$[1], [\frac{1}{4} \ \frac{2}{5}], \mathbf{A}.$$

Gaussian elimination with no row interchanges is valid if and only if the pivots $a_{kk}^{(k)}$ are nonzero for $k = 1, \dots, n - 1$.

Theorem 2.4 We have $a_{k,k}^{(k)} \neq 0$ for $k = 1, \dots, n - 1$ if and only if the leading principal submatrices $\mathbf{A}_{[k]}$ of \mathbf{A} are nonsingular for $k = 1, \dots, n - 1$. Moreover

$$\det(\mathbf{A}_{[k]}) = a_{11}^{(1)} a_{22}^{(2)} \cdots a_{kk}^{(k)}, \quad k = 1, \dots, n. \quad (2.3)$$

Proof. Let $\mathbf{B}_k = \mathbf{A}_{k-1}^{(k)}$ be the upper left $k - 1$ corner of $\mathbf{A}^{(k)}$ given by (2.1). Observe that the elements of \mathbf{B}_k are computed from \mathbf{A} by using only elements from $\mathbf{A}_{[k-1]}$. Since the determinant of a matrix does not change under the operation of subtracting a multiple of one row from another row the determinant of $\mathbf{A}_{[k]}$ equals the product of diagonal elements of \mathbf{B}_{k+1} and (2.3) follows. But then $a_{11}^{(1)} \cdots a_{kk}^{(k)} \neq 0$ for $k = 1, \dots, n - 1$ if and only if $\det(\mathbf{A}_{[k]}) \neq 0$ for $k = 1, \dots, n - 1$, or equivalently $\mathbf{A}_{[k]}$ is nonsingular for $k = 1, \dots, n - 1$. \square

Gaussian elimination is a way to compute the LU-factorization of the coefficient matrix.

Theorem 2.5 Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ and that $\mathbf{A}_{[k]}$ is nonsingular for $k = 1, \dots, n - 1$. Then Gaussian elimination with no row interchanges results in an LU-factorization of \mathbf{A} . In particular $\mathbf{A} = \mathbf{L}\mathbf{U}$, where

$$\mathbf{L} = \begin{bmatrix} 1 & & & \\ l_{21}^{(1)} & 1 & & \\ \vdots & & \ddots & \\ l_{n1}^{(1)} & l_{n2}^{(2)} & \cdots & 1 \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} a_{11}^{(1)} & \cdots & a_{1n}^{(1)} \\ & \ddots & \vdots \\ & & a_{nn}^n \end{bmatrix}, \quad (2.4)$$

where the $l_{ij}^{(j)}$ and $a_{ij}^{(i)}$ are given by (2.2).

Proof. From (2.2) we have for all i, j

$$l_{ik}^{(k)} a_{kj}^{(k)} = a_{ij}^{(k)} - a_{ij}^{(k+1)} \text{ for } k < \min(i, j), \text{ and } l_{ij}^{(k)} a_{jj}^{(j)} = a_{ij}^{(j)} \text{ for } i > j.$$

Thus for $i \leq j$ we find

$$(\mathbf{LU})_{ij} = \sum_{k=1}^n l_{ik}^{(k)} u_{kj} = \sum_{k=1}^{i-1} l_{ik}^{(k)} a_{kj}^{(k)} + a_{ij}^{(i)} = \sum_{k=1}^{i-1} (a_{ij}^{(k)} - a_{ij}^{(k+1)}) + a_{ij}^{(i)} = a_{ij}^{(1)} = a_{ij}, \quad (2.5)$$

while for $i > j$

$$(\mathbf{LU})_{ij} = \sum_{k=1}^n l_{ik}^{(k)} u_{kj} = \sum_{k=1}^{j-1} l_{ik}^{(k)} a_{kj}^{(k)} + l_{ij} a_{jj}^{(j)} = \sum_{k=1}^{j-1} (a_{ij}^{(k)} - a_{ij}^{(k+1)}) + a_{ij}^{(j)} = a_{ij}. \quad (2.6)$$

□

Note that this Theorem holds even if \mathbf{A} is singular. Since \mathbf{L} is nonsingular the matrix \mathbf{U} is then singular, and we must have $a_{nn}^n = 0$ when \mathbf{A} is singular.

2.2 Banded Triangular Systems

Once we have an LU-factorization of \mathbf{A} the system $\mathbf{Ax} = \mathbf{b}$ is solved in two steps. Since $\mathbf{LUx} = \mathbf{b}$ we have $\mathbf{Ly} = \mathbf{b}$, where $\mathbf{y} := \mathbf{Ux}$. We first solve $\mathbf{Ly} = \mathbf{b}$, for \mathbf{y} and then $\mathbf{Ux} = \mathbf{y}$ for \mathbf{x} .

2.2.1 Algorithms for triangular systems

A nonsingular triangular linear system $\mathbf{Ax} = \mathbf{b}$ is easy to solve. By Lemma 1.35 \mathbf{A} has nonzero diagonal elements. Consider first the lower triangular case. For $n = 3$ the system is

$$\begin{bmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}.$$

From the first equation we find $x_1 = b_1/a_{11}$. Solving the second equation for x_2 we obtain $x_2 = (b_2 - a_{21}x_1)/a_{22}$. Finally the third equation gives $x_3 = (b_3 - a_{31}x_1 - a_{32}x_2)/a_{33}$. This process is known as forward substitution. In general

$$x_k = \left(b_k - \sum_{j=1}^{k-1} a_{k,j} x_j \right) / a_{kk}, \quad k = 1, 2, \dots, n. \quad (2.7)$$

When \mathbf{A} is a lower triangular band matrix the number of arithmetic operations necessary to find \mathbf{x} can be reduced. Suppose \mathbf{A} is a lower triangular d -banded,

$$\begin{bmatrix} a_{11} & 0 & 0 & 0 & 0 \\ a_{21} & a_{22} & 0 & 0 & 0 \\ 0 & a_{32} & a_{33} & 0 & 0 \\ 0 & 0 & a_{43} & a_{44} & 0 \\ 0 & 0 & 0 & a_{54} & a_{55} \end{bmatrix}, \quad \begin{bmatrix} a_{11} & 0 & 0 & 0 & 0 \\ a_{21} & a_{22} & 0 & 0 & 0 \\ a_{31} & a_{32} & a_{33} & 0 & 0 \\ 0 & a_{42} & a_{43} & a_{44} & 0 \\ 0 & 0 & a_{53} & a_{54} & a_{55} \end{bmatrix}$$

Figure 2.2: Lower triangular 5×5 band matrices: $d = 1$ (left) and $d = 2$ right.

so that $a_{k,j} = 0$ for $j \notin \{l_k, l_k + 1, \dots, k\}$ for $k = 1, 2, \dots, n$, and where $l_k := \max(1, k-d)$, see Figure 2.2. For a lower triangular d -band matrix the calculation in (2.7) can be simplified as follows

$$x_k = (b_k - \sum_{j=l_k}^{k-1} a_{k,j}x_j)/a_{kk}, \quad k = 1, 2, \dots, n. \quad (2.8)$$

Note that (2.8) reduces to (2.7) if $d = n$. Letting $A(k, l_k : (k-1)) * x(l_k : (k-1))$ denote the sum $\sum_{j=l_k}^{k-1} a_{kj}x_j$ we arrive at the following algorithm, where the intial "r" in the name signals that this algorithm is row oriented. For each k we take the inner product of a part of a row with the already computed unknowns.

Algorithm 2.6 (forwardsolve)

Given a nonsingular lower triangular d -banded matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ and $\mathbf{b} \in \mathbb{C}^n$. An $\mathbf{x} \in \mathbb{C}^n$ is computed so that $\mathbf{Ax} = \mathbf{b}$.

```

1 function x=rforwardsolve(A,b,d)
2 n=length(b); x=b;
3 x(1)=b(1)/A(1,1);
4 for k=2:n
5     lk=max(1,k-d);
6     x(k)=(b(k)-A(k,(lk:(k-1))*x(lk:(k-1)))/A(k,k);
7 end

```

A system $\mathbf{Ax} = \mathbf{b}$, where \mathbf{A} is upper triangular must be solved by back substitution or 'bottom-up'. We first find x_n from the last equation and then move upwards for the remaining unknowns. For an upper triangular d -banded matrix this leads to the following algorithm.

Algorithm 2.7 (backsolve)

Given a nonsingular upper triangular d -banded matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ and $\mathbf{b} \in \mathbb{C}^n$. An $\mathbf{x} \in \mathbb{C}^n$ is computed so that $\mathbf{Ax} = \mathbf{b}$.

```

1 function x=rbacksolve (A, b ,d)
2 n=length(b); x=b;
3 x(n)=b(n)/A(n,n);
4 for k=n-1:-1:1
5     uk=min(n,k+d);
6     x(k)=(b(k)-A(k,(k+1):uk)*x((k+1):uk))/A(k,k);
7 end

```

Exercise 2.8 (Column oriented backsolve)

In this exercise we develop column oriented vectorized versions of forward and backward substitution. Consider the system $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{C}^{n \times n}$ is lower triangular. Suppose after $k - 1$ steps of the algorithm we have a reduced system in the form

$$\begin{bmatrix} a_{k,k} & 0 & \cdots & 0 \\ a_{k+1,k} & a_{k+1,k+1} & \cdots & 0 \\ \vdots & \ddots & \vdots & \\ a_{n,k} & \cdots & a_{n \times n} \end{bmatrix} \begin{bmatrix} x_k \\ x_{k+1} \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_k \\ b_{k+1} \\ \vdots \\ b_n \end{bmatrix}.$$

This system is of order $n - k + 1$. The unknowns are x_k, \dots, x_n .

a) We see that $x_k = b_k/a_{k,k}$ and eliminating x_k from the remaining equations we obtain a system of order $n - k$ with unknowns x_{k+1}, \dots, x_n

$$\begin{bmatrix} a_{k+1,k+1} & 0 & \cdots & 0 \\ a_{k+2,k+1} & a_{k+2,k+2} & \cdots & 0 \\ \vdots & \ddots & \vdots & \\ a_{n,k+1} & \cdots & a_{n,n} \end{bmatrix} \begin{bmatrix} x_{k+1} \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_{k+1} \\ \vdots \\ b_n \end{bmatrix} - x_k \begin{bmatrix} a_{k+1,k} \\ \vdots \\ a_{n,k} \end{bmatrix}.$$

Thus at the k th step, $k = 1, 2, \dots, n$ we set $x_k = b_k/A(k,k)$ and update b as follows:

$$b((k+1) : n) = b((k+1) : n) - x(k) * A((k+1) : n, k).$$

This leads to the following algorithm.

Algorithm 2.9 (Forward Solve (column oriented))

Given a nonsingular lower triangular d -banded matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ and $\mathbf{b} \in \mathbb{C}^n$. An $\mathbf{x} \in \mathbb{C}^n$ is computed so that $\mathbf{Ax} = \mathbf{b}$.

```

1 function x=cforwardsolve(A,b,d)
2 x=b; n=length(b);
3 for k=1:n-1
4   x(k)=b(k)/A(k,k); uk=min(n,k+d);
5   b((k+1):uk)=b((k+1):uk)-A((k+1):uk,k)*x(k);
6 end
7 x(n)=b(n)/A(n,n);
8 end

```

b) Suppose now $\mathbf{A} \in \mathbb{C}^{n \times n}$ is nonsingular, upper triangular, d -banded, and $\mathbf{b} \in \mathbb{C}^n$. Justify the following column oriented vectorized algorithms for solving $\mathbf{Ax} = \mathbf{b}$.

Algorithm 2.10 (Backsolve (column oriented))

Given a nonsingular upper triangular d -banded matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ and $\mathbf{b} \in \mathbb{C}^n$. An $\mathbf{x} \in \mathbb{C}^n$ is computed so that $\mathbf{Ax} = \mathbf{b}$.

```

1 function x=cbacksolve(A,b,d)
2 x=b; n=length(b);
3 for k=n:-1:2
4   x(k)=b(k)/A(k,k); lk=max(1,k-d);
5   b(1k:(k-1))=b(1k:(k-1))-A(1k:(k-1),k)*x(k);
6 end
7 x(1)=b(1)/A(1,1);
8 end

```

Exercise 2.11 (Computing the inverse of a triangular matrix)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is a nonsingular upper triangular matrix with upper triangular inverse $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_n]$. The k th column \mathbf{b}_k of \mathbf{B} is the solution of the linear systems $\mathbf{Ab}_k = \mathbf{e}_k$. Show that $b_k(k) = 1/a(k,k)$ for $k = 1, \dots, n$, and explain why we can find \mathbf{b}_k by solving the linear systems

$$\mathbf{A}((k+1):n, (k+1):n)\mathbf{b}_k((k+1):n) = -\mathbf{A}((k+1):n, k)b_k(k), \quad k = 1, \dots, n-1. \quad (2.9)$$

Is it possible to store the interesting part of \mathbf{b}_k in \mathbf{A} as soon as it is computed? What if \mathbf{A} is upper triangular?

$$\mathbf{A}(1:k, 1:k)\mathbf{b}_k(1:k) = \mathbf{I}(1:k, k), \quad k = n, n-1, \dots, 1, \quad \text{upper triangular} \quad (2.10)$$

2.2.2 Counting operations

It is useful to have a number which indicates the amount of work an algorithm requires. In this book we measure this by estimating the total number of arithmetic operations. We count both additions, subtractions, multiplications and divisions, but not work on indices. As an example we show that the calculation to find the LU factorization of a full matrix of order n using Gaussian elimination is exactly

$$N_{LU} := \frac{2}{3}n^3 - \frac{1}{2}n^2 - \frac{1}{6}n. \quad (2.11)$$

Let M, D, A, S be the number of multiplications, divisions, additions, and subtractions. In (2.2) the multiplications and subtractions occur in the calculation of $a_{ij}^{k+1} = a_{ij}^{(k)} - l_{ik}^{(k)}a_{kj}^{(k)}$ which is carried out $(n - k)^2$ times. Moreover, each calculation involves one subtraction and one multiplication. Thus we find $M + S = 2 \sum_{k=1}^{n-1} (n - k)^2 = 2 \sum_{m=1}^{n-1} m^2 = \frac{2}{3}n(n-1)(n-\frac{1}{2})$. For each k there are $n - k$ divisions giving a sum of $\sum_{m=1}^{n-1} (n - k) = \frac{1}{2}n(n - 1)$. Since there are no additions we obtain the total

$$M + D + A + S = \frac{2}{3}n(n-1)\left(n-\frac{1}{2}\right) + \frac{1}{2}n(n-1) = N_{LU}$$

given by (2.11).

We are only interested in N_{LU} when n is large and for such n the term $\frac{2}{3}n^3$ dominates. We therefore regularly ignore lower order terms and use **number of operations** both for the exact count and for the highest order term. We also say more loosely that the the number of operations is $O(n^3)$. We will use the number of operations counted in one of these ways as a measure of the **complexity of an algorithm** and say that the complexity of LU factorization of a full matrix is $O(n^3)$ or more precisely $\frac{2}{3}n^3$.

We will compare the number of arithmetic operations of many algorithms with the number of arithmetic operations of Gaussian elimination and define for $n \in \mathbb{N}$ the number G_n as follows:

Definition 2.12 ($G_n := \frac{2}{3}n^3$)
We define $G_n := \frac{2}{3}n^3$.

There is a quick way to arrive at the leading term $2n^3/3$. We only consider the operations contributing to this term. In (2.2) the leading term comes from the inner loop contributing to $M + S$. Then we replace sums by integrals letting the summation indices be continuous variables and adjust limits of integration in an insightful way to simplify the calculation. Thus,

$$M + S = 2 \sum_{k=1}^{n-1} (n - k)^2 \approx 2 \int_1^{n-1} (n - k)^2 dk \approx 2 \int_0^n (n - k)^2 dk = \frac{2}{3}n^3$$

and this is the correct leading term.

Consider next N_S , the number of forward plus backward substitutions. By (2.7) we obtain

$$N_S = 2 \sum_{k=1}^n (2k - 1) \approx 2 \int_1^n (2k - 1) dk \approx 4 \int_0^n k dk = 2n^2.$$

The last integral actually give the exact value for the sum in this case (cf. (2.14))

We see that LU factorization is an $O(n^3)$ process while solving a triangular system requires $O(n^2)$ arithmetic operations. Thus, if $n = 10^6$ and one arithmetic operation requires $c = 10^{-14}$ seconds of computing time then $cn^3 = 10^4$ seconds ≈ 3 hours and $cn^2 = 0.01$ second, giving dramatic differences in computing time.

Exercise 2.13 (Finite sums of integers)

Use induction on m , or some other method, to show that

$$1 + 2 + \cdots + m = \frac{1}{2}m(m + 1), \quad (2.12)$$

$$1^2 + 2^2 + \cdots + m^2 = \frac{1}{3}m(m + \frac{1}{2})(m + 1), \quad (2.13)$$

$$1 + 3 + 5 + \cdots + 2m - 1 = m^2, \quad (2.14)$$

$$1 * 2 + 2 * 3 + 3 * 4 + \cdots + (m - 1)m = \frac{1}{3}(m - 1)m(m + 1). \quad (2.15)$$

Exercise 2.14 (Multiplying triangular matrices)

Show that the matrix multiplication \mathbf{AB} can be done in $\frac{1}{3}n(2n^2 + 1) \approx G_n$ arithmetic operations when $\mathbf{A} \in \mathbb{R}^{n \times n}$ is lower triangular and $\mathbf{B} \in \mathbb{R}^{n \times n}$ is upper triangular. What about \mathbf{BA} ?

2.3 The LU and LDU Factorizations

Gaussian elimination without row interchanges is one way of computing an LU factorization of a matrix. There are other ways that can be advantageous for certain kind of problems. Here we consider the general theory of LU factorizations. Recall that $\mathbf{A} = \mathbf{LU}$ is an **LU factorization** of $\mathbf{A} \in \mathbb{C}^{n \times n}$ if $\mathbf{L} \in \mathbb{C}^{n \times n}$ is lower triangular and $\mathbf{U} \in \mathbb{C}^{n \times n}$ is upper triangular , i.e.,

$$\mathbf{L} = \begin{bmatrix} l_{1,1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ l_{n,1} & \cdots & l_{n,n} \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} u_{1,1} & \cdots & u_{1,n} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & u_{n,n} \end{bmatrix}.$$

To find an LU factorization there is one equation for each of the n^2 elements in \mathbf{A} , and \mathbf{L} and \mathbf{U} contain a total of $n^2 + n$ unknown elements. There are several ways to restrict the number of unknowns to n^2 .

L1U: $l_{ii} = 1$ all i ,

LU1: $u_{ii} = 1$ all i ,

LDU: $\mathbf{A} = \mathbf{LDU}$, $l_{ii} = u_{ii} = 1$ all i , $\mathbf{D} = \text{diag}(d_{11}, \dots, d_{nn})$.

2.3.1 Existence and uniqueness



Henry Jensen, 1915-1974 (left), Prescott Durand Crout, 1907-1984. Jensen worked on LU factorizations. His name is also associated with a very useful inequality (cf. Theorem 7.40).

Consider the L1U factorization. Three things can happen. An L1U factorization exists and is unique, it exists, but it is not unique, or it does not exist. The 2×2 case illustrates this.

Example 2.15 (L1U of 2×2 matrix)

Let $a, b, c, d \in \mathbb{C}$. An L1U factorization of $\mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ must satisfy the equations

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ l_1 & 1 \end{bmatrix} \begin{bmatrix} u_1 & u_2 \\ 0 & u_3 \end{bmatrix} = \begin{bmatrix} u_1 & u_2 \\ u_1 l_1 & u_2 l_1 + u_3 \end{bmatrix}$$

for the unknowns l_1 in \mathbf{L} and u_1, u_2, u_3 in \mathbf{U} . The equations are

$$u_1 = a, \quad u_2 = b, \quad al_1 = c, \quad bl_1 + u_3 = d. \quad (2.16)$$

These equations do not always have a solution. Indeed, the main problem is the equation $al_1 = c$. There are essentially three cases

1. $a \neq 0$: The matrix has a unique L1U factorization.

2. $a = c = 0$: The L1U factorization exists, but it is not unique. Any value for l_1 can be used.
3. $a = 0, c \neq 0$: No L1U factorization exists.

Consider the four matrices

$$\mathbf{A}_1 := \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}, \quad \mathbf{A}_2 := \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}, \quad \mathbf{A}_3 := \begin{bmatrix} 0 & 1 \\ 0 & 2 \end{bmatrix}, \quad \mathbf{A}_4 := \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

From the previous discussion it follows that \mathbf{A}_1 has a unique L1U factorization, \mathbf{A}_2 has no L1U factorization, \mathbf{A}_3 has an L1U factorization but it is not unique, and \mathbf{A}_4 has a unique L1U factorization even if it is singular.

In preparation for the main theorem about LU factorization we prove a simple lemma.

Lemma 2.16 (L1U of leading principal submatrices)

Suppose $\mathbf{A} = \mathbf{LU}$ is an L1U factorization of $\mathbf{A} \in \mathbb{C}^{n \times n}$. For $k = 1, \dots, n$ let $\mathbf{A}_{[k]}, \mathbf{L}_{[k]}, \mathbf{U}_{[k]}$ be the leading principal submatrices of $\mathbf{A}, \mathbf{L}, \mathbf{U}$, respectively. Then $\mathbf{A}_{[k]} = \mathbf{L}_{[k]}\mathbf{U}_{[k]}$ is an LU factorization of $\mathbf{A}_{[k]}$ for $k = 1, \dots, n$.

Proof. For $k = 1, \dots, n-1$ we partition $\mathbf{A} = \mathbf{LU}$ as follows:

$$\begin{bmatrix} \mathbf{A}_{[k]} & \mathbf{B}_k \\ \mathbf{C}_k & \mathbf{F}_k \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{[k]} & \mathbf{0} \\ \mathbf{M}_k & \mathbf{N}_k \end{bmatrix} \begin{bmatrix} \mathbf{U}_{[k]} & \mathbf{S}_k \\ \mathbf{0} & \mathbf{T}_k \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{[k]}\mathbf{U}_{[k]} & \mathbf{L}_{[k]}\mathbf{S}_k \\ \mathbf{M}_k\mathbf{U}_{[k]} & \mathbf{M}_k\mathbf{S}_k + \mathbf{N}_k\mathbf{T}_k \end{bmatrix}, \quad (2.17)$$

where $\mathbf{F}_k, \mathbf{N}_k, \mathbf{T}_k \in \mathbb{C}^{n-k, n-k}$. Comparing blocks we find $\mathbf{A}_{[k]} = \mathbf{L}_{[k]}\mathbf{U}_{[k]}$. Since $\mathbf{L}_{[k]}$ is unit lower triangular and $\mathbf{U}_{[k]}$ is upper triangular this is an L1U factorization of $\mathbf{A}_{[k]}$. \square

The following theorem gives a necessary and sufficient condition for existence of a unique LU factorization. The conditions are the same for the three factorizations L1U, LU1 and LDU.

Theorem 2.17 (LU Theorem)

A square matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ has a unique L1U (LU1, LDU) factorization if and only if the leading principal submatrices $\mathbf{A}_{[k]}$ of \mathbf{A} are nonsingular for $k = 1, \dots, n-1$.

Proof. Suppose $\mathbf{A}_{[k]}$ is nonsingular for $k = 1, \dots, n-1$. Under these conditions Gaussian elimination gives an L1U factorization (cf. Theorem 2.5). We give another proof here that in addition to showing uniqueness also gives alternative ways to compute the L1U factorization. The proofs for the LU1 and LDU factorizations are similar and left as exercises.

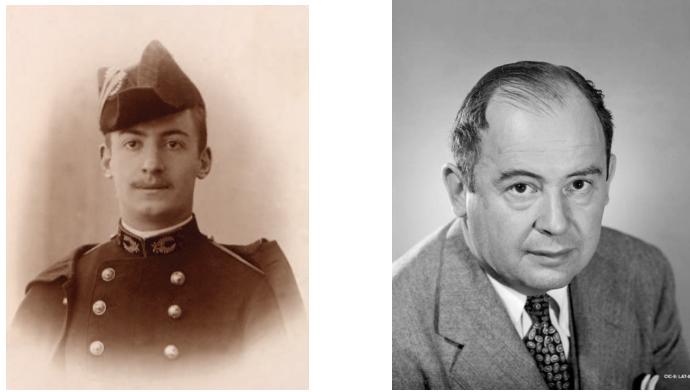


Figure 2.3: André-Louis Cholesky, 1875-1918 (left), John von Neumann, 1903-1957.

We use induction on n to show that \mathbf{A} has a unique L1U factorization. The result is clearly true for $n = 1$, since the unique L1U factorization of a 1×1 matrix is $[a_{11}] = [1][a_{11}]$. Suppose that $\mathbf{A}_{[n-1]}$ has a unique L1U factorization $\mathbf{A}_{[n-1]} = \mathbf{L}_{n-1}\mathbf{U}_{n-1}$, and that $\mathbf{A}_{[1]}, \dots, \mathbf{A}_{[n-1]}$ are nonsingular. By block multiplication

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{[n-1]} & \mathbf{c}_n \\ \mathbf{r}_n^T & a_{nn} \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{n-1} & \mathbf{0} \\ \mathbf{l}_n^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{U}_{n-1} & \mathbf{u}_n \\ 0 & u_{nn} \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{n-1}\mathbf{U}_{n-1} & \mathbf{L}_{n-1}\mathbf{u}_n \\ \mathbf{l}_n^T\mathbf{U}_{n-1} & \mathbf{l}_n^T\mathbf{u}_n + u_{nn} \end{bmatrix}, \quad (2.18)$$

if and only if $\mathbf{A}_{[n-1]} = \mathbf{L}_{n-1}\mathbf{U}_{n-1}$ and $\mathbf{l}_n, \mathbf{u}_n \in \mathbb{C}^{n-1}$ and $u_{nn} \in \mathbb{C}$ are determined from

$$\mathbf{U}_{n-1}^T \mathbf{l}_n = \mathbf{r}_n, \quad \mathbf{L}_{n-1} \mathbf{u}_n = \mathbf{c}_n, \quad u_{nn} = a_{nn} - \mathbf{l}_n^T \mathbf{u}_n. \quad (2.19)$$

Since $\mathbf{A}_{[n-1]}$ is nonsingular it follows that \mathbf{L}_{n-1} and \mathbf{U}_{n-1} are nonsingular and therefore $\mathbf{l}_n, \mathbf{u}_n$, and u_{nn} are uniquely given. Thus (2.18) gives a unique L1U factorization of \mathbf{A} .

Conversely, suppose \mathbf{A} has a unique L1U factorization $\mathbf{A} = \mathbf{LU}$. By Lemma 2.16 $\mathbf{A}_{[k]} = \mathbf{L}_{[k]}\mathbf{U}_{[k]}$ is an L1U factorization of $\mathbf{A}_{[k]}$ for $k = 1, \dots, n-1$. Suppose $\mathbf{A}_{[k]}$ is singular for some $k \leq n-1$. We will show that this leads to a contradiction. Let k be the smallest integer so that $\mathbf{A}_{[k]}$ is singular. Since $\mathbf{A}_{[j]}$ is nonsingular for $j \leq k-1$ it follows from what we have already shown that $\mathbf{A}_{[k]} = \mathbf{L}_{[k]}\mathbf{U}_{[k]}$ is the unique L1U factorization of $\mathbf{A}_{[k]}$. The matrix $\mathbf{U}_{[k]}$ is singular since $\mathbf{A}_{[k]}$ is singular and $\mathbf{L}_{[k]}$ is nonsingular. By (2.17) we have $\mathbf{U}_{[k]}^T \mathbf{M}_k^T = \mathbf{C}_k^T$. This can be written as $n-k$ linear systems for the columns of \mathbf{M}_k^T . By assumption \mathbf{M}_k^T exists, but since $\mathbf{U}_{[k]}^T$ is singular \mathbf{M}_k is not unique, a contradiction. \square

By combining the last two equations in (2.19) we obtain with $k = n$

$$\mathbf{U}_{k-1}^T \mathbf{l}_k = \mathbf{r}_k, \quad \begin{bmatrix} \mathbf{L}_{k-1} & \mathbf{0} \\ \mathbf{l}_k^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{u}_k \\ u_{kk} \end{bmatrix} = \begin{bmatrix} \mathbf{c}_k \\ a_{kk} \end{bmatrix}.$$

This can be used in an algorithm to compute the L1U factorization. Moreover, if \mathbf{A} is d -banded then the first $k-d$ components in \mathbf{r}_k and \mathbf{c}_k are zero so both \mathbf{L} and \mathbf{U} will be d -banded. Thus we can use the banded `rforwardsolve` Algorithm 2.6 to solve the lower triangular system $\mathbf{U}_{k-1}^T \mathbf{l}_k = \mathbf{r}_k$ for the k th row \mathbf{l}_k^T in \mathbf{L} and the k th column $[\mathbf{u}_k]$ in \mathbf{U} for $k = 2, \dots, n$. This leads to the following algorithm to compute the L1U factorization of a d -banded matrix. The algorithm will fail if the conditions in the LU theorem are not satisfied.

Algorithm 2.18 (L1U factorization)

```

1 function [L,U]=L1U(A,d)
2 n=length(A);
3 L=eye(n,n); U=zeros(n,n);U(1,1)=A(1,1);
4 for k=2:n
5   km=max(1,(k-d));
6   L(k,km:(k-1))=rforwardsolve(U(km:(k-1),km:(k-1))',A(k,km:(k-1))',d)';
7   U(km:k,k)=rforwardsolve(L(km:k,km:k),A(km:k,k),d);
8 end

```

For each k we essentially solve a lower triangular linear system of order d . Thus the number of arithmetic operation for this algorithm is $O(d^2n)$.

Exercise 2.19 (# operations for banded triangular systems)

Show that for $1 \leq d \leq n$ algorithms 2.18 requires exactly $N_{LU}(n, d) := (2d^2 + d)n - (d^2 + d)(8d + 1)/6 = O(d^2n)$ operations.¹⁰ In particular, for a full matrix $d = n - 1$ and we find $N_{LU}(n, n) = \frac{2}{3}n^3 - \frac{1}{2}n^2 - \frac{1}{6}n \approx G_n$ in agreement with the exact count (2.11) for Gaussian elimination, while for a tridiagonal matrix $N_{LU}(n, 1) = 3n - 3 = O(n)$.

Remark 2.20 (LU of upper triangular matrix)

A matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ can have an LU factorization even if $\mathbf{A}_{[k]}$ is singular for some $k < n$. By Theorem 3.3 such an LU factorization cannot be unique. An L1U factorization of an upper triangular matrix \mathbf{A} is $\mathbf{A} = \mathbf{I}\mathbf{A}$ so it always exists even if \mathbf{A} has zeros somewhere on the diagonal. By Lemma 1.35, if some a_{kk} is zero then $\mathbf{A}_{[k]}$ is singular and the L1U factorization is not unique. In particular, for the zero matrix any unit lower triangular matrix can be used as \mathbf{L} in an L1U factorization.

¹⁰Hint: Consider the cases $2 \leq k \leq d$ and $d + 1 \leq k \leq n$ separately.

Exercise 2.21 (L1U and LU1)

Show that the matrix A_3 in Example 2.15 has no LU1 or LDU factorization. Give an example of a matrix that has an LU1 factorization, but no LDU or L1U factorization.

Exercise 2.22 (LU of nonsingular matrix)

Show that the following is equivalent for a nonsingular matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$.

1. \mathbf{A} has an LDU factorization.
2. \mathbf{A} has an L1U factorization.
3. \mathbf{A} has an LU1 factorization.

Exercise 2.23 (Row interchange)

Show that $\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$ has a unique L1U factorization. Note that we have only interchanged rows in Example 2.15.

Exercise 2.24 (LU and determinant)

Suppose \mathbf{A} has an L1U factorization $\mathbf{A} = \mathbf{L}\mathbf{U}$. Show that $\det(\mathbf{A}_{[k]}) = u_{11}u_{22} \cdots u_{kk}$ for $k = 1, \dots, n$.

Exercise 2.25 (Diagonal elements in U)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ and $\mathbf{A}_{[k]}$ is nonsingular for $k = 1, \dots, n-1$. Use Exercise 2.24 to show that the diagonal elements u_{kk} in the L1U factorization are

$$u_{11} = a_{11}, \quad u_{kk} = \frac{\det(\mathbf{A}_{[k]})}{\det(\mathbf{A}_{[k-1]})}, \text{ for } k = 2, \dots, n. \quad (2.20)$$

Exercise 2.26 (Proof of LDU theorem)

Give a proof of the LU theorem for the LDU case.

Exercise 2.27 (Proof of LU1 theorem)

Give a proof of the LU theorem for the LU1 case.

2.3.2 Block LU factorization

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is a block matrix of the form

$$\mathbf{A} := \begin{bmatrix} \mathbf{A}_{11} & \cdots & \mathbf{A}_{1m} \\ \vdots & & \vdots \\ \mathbf{A}_{m1} & \cdots & \mathbf{A}_{mm} \end{bmatrix}, \quad (2.21)$$

where each diagonal block \mathbf{A}_{ii} is square. We call the factorization

$$\mathbf{A} = \mathbf{L}\mathbf{U} = \begin{bmatrix} \mathbf{I} & & & \\ \mathbf{L}_{21} & \mathbf{I} & & \\ \vdots & & \ddots & \\ \mathbf{L}_{m1} & \cdots & \mathbf{L}_{m,m-1} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{11} & \cdots & \mathbf{U}_{1m} \\ \mathbf{U}_{21} & \cdots & \mathbf{U}_{2m} \\ \ddots & \ddots & \vdots \\ & & \mathbf{U}_{mm} \end{bmatrix} \quad (2.22)$$

a **block LU factorization** of \mathbf{A} . Here the i th diagonal blocks \mathbf{I} and \mathbf{U}_{ii} in \mathbf{L} and \mathbf{U} have the same size as \mathbf{A}_{ii} , the i th diagonal block in \mathbf{A} . Block LU1 and block LDU factorizations are defined similarly.

The results for element-wise LU factorization carry over to block LU factorization as follows.

Theorem 2.28 (Block LU theorem)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is a block matrix of the form (2.21). Then \mathbf{A} has a unique block LU factorization (2.22) if and only if the leading principal block submatrices

$$\mathbf{A}_{\{k\}} := \begin{bmatrix} \mathbf{A}_{11} & \cdots & \mathbf{A}_{1k} \\ \vdots & & \vdots \\ \mathbf{A}_{k1} & \cdots & \mathbf{A}_{kk} \end{bmatrix}$$

are nonsingular for $k = 1, \dots, m - 1$.

Proof. Suppose $\mathbf{A}_{\{k\}}$ is nonsingular for $k = 1, \dots, m - 1$. Following the proof in Theorem 2.17 suppose $\mathbf{A}_{\{m-1\}}$ has a unique block LU factorization $\mathbf{A}_{\{m-1\}} = \mathbf{L}_{\{m-1\}}\mathbf{U}_{\{m-1\}}$, and that $\mathbf{A}_{\{1\}}, \dots, \mathbf{A}_{\{m-1\}}$ are nonsingular. Then $\mathbf{L}_{\{m-1\}}$ and $\mathbf{U}_{\{m-1\}}$ are nonsingular and

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} \mathbf{A}_{\{m-1\}} & \mathbf{B} \\ \mathbf{C}^T & \mathbf{A}_{mm} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{L}_{\{m-1\}} & \mathbf{0} \\ \mathbf{C}^T \mathbf{U}_{\{m-1\}}^{-1} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{\{m-1\}} & \mathbf{L}_{\{m-1\}}^{-1} \mathbf{B} \\ 0 & \mathbf{A}_{mm} - \mathbf{C}^T \mathbf{U}_{\{m-1\}}^{-1} \mathbf{L}_{\{m-1\}}^{-1} \mathbf{B} \end{bmatrix}, \end{aligned} \quad (2.23)$$

is a block LU factorization of \mathbf{A} . It is unique by derivation. Conversely, suppose \mathbf{A} has a unique block LU factorization $\mathbf{A} = \mathbf{L}\mathbf{U}$. Then as in Lemma 2.16 it is

easily seen that $\mathbf{A}_{\{k\}} = \mathbf{L}_{\{k\}} \mathbf{U}_{\{k\}}$ is the unique block LU factorization of $\mathbf{A}_{[k]}$ for $k = 1, \dots, m$. The rest of the proof is similar to the proof of Theorem 2.17. \square

Remark 2.29 (Comparing LU and block LU)

The number of arithmetic operations for the block LU factorization is the same as for the ordinary LU factorization. An advantage of the block method is that it combines many of the operations into matrix operations.

Remark 2.30 (A block LU is not an LU)

Note that (2.22) is not an LU factorization of \mathbf{A} since the \mathbf{U}_{ii} 's are not upper triangular in general. To relate the block LU factorization to the usual LU factorization we assume that each \mathbf{U}_{ii} has an LU factorization $\mathbf{U}_{ii} = \tilde{\mathbf{L}}_{ii} \tilde{\mathbf{U}}_{ii}$. Then $\mathbf{A} = \hat{\mathbf{L}} \hat{\mathbf{U}}$, where $\hat{\mathbf{L}} := \mathbf{L} \operatorname{diag}(\tilde{\mathbf{L}}_{ii})$ and $\hat{\mathbf{U}} := \operatorname{diag}(\tilde{\mathbf{L}}_{ii}^{-1}) \mathbf{U}$, and this is an ordinary LU factorization of \mathbf{A} .

Exercise 2.31 (Making block LU into LU)

Show that $\hat{\mathbf{L}}$ is unit lower triangular and $\hat{\mathbf{U}}$ is upper triangular.

2.4 The PLU-Factorization

Theorem 2.4 shows that Gaussian elimination can fail on a nonsingular system. A simple example is $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$. We show here that any nonsingular linear system can be solved by Gaussian elimination if we incorporate row interchanges.

2.4.1 Pivoting

Interchanging two rows (and/or two columns) during Gaussian elimination is known as **pivoting**. The element which is moved to the diagonal position (k, k) is called the **pivot element** or **pivot** for short, and the row containing the pivot is called the **pivot row**. Gaussian elimination with row pivoting can be described as follows.

1. **Choose** $r_k \geq k$ so that $a_{r_k, k}^{(k)} \neq 0$.
2. **Interchange** rows r_k and k of $\mathbf{A}^{(k)}$.
3. **Eliminate** by computing $l_{ik}^{(k)}$ and $a_{ij}^{(k+1)}$ using (2.2).

To show that Gaussian elimination can always be carried to completion by using suitable row interchanges suppose by induction on k that $\mathbf{A}^{(k)}$ is nonsingular. Since $\mathbf{A}^{(1)} = \mathbf{A}$ this holds for $k = 1$. By Lemma 1.34 the lower right diagonal block

in $\mathbf{A}^{(k)}$ is nonsingular. But then at least one entry in the first column of that block must be nonzero and it follows that r_k exists so that $a_{r_k,k}^{(k)} \neq 0$. But then $\mathbf{A}^{(k+1)}$ is nonsingular since it is computed from $\mathbf{A}^{(k)}$ using row operations preserving the nonsingularity. We conclude that $\mathbf{A}^{(k)}$ is nonsingular for $k = 1, \dots, n$.

2.4.2 Permutation matrices

Row interchanges can be described in terms of permutation matrices.

Definition 2.32 A **permutation matrix** is a matrix of the form

$$\mathbf{P} = \mathbf{I}(:, \mathbf{p}) = [\mathbf{e}_{i_1}, \mathbf{e}_{i_2}, \dots, \mathbf{e}_{i_n}] \in \mathbb{R}^{n,n},$$

where $\mathbf{e}_{i_1}, \dots, \mathbf{e}_{i_n}$ is a permutation of the unit vectors $\mathbf{e}_1, \dots, \mathbf{e}_n \in \mathbb{R}^n$.

Every permutation $\mathbf{p} = [i_1, \dots, i_n]^T$ of the integers $1, 2, \dots, n$ gives rise to a permutation matrix and vice versa. Post-multiplying a matrix \mathbf{A} by a permutation matrix results in a permutation of the columns, while pre-multiplying by a permutation matrix gives a permutation of the rows. In symbols

$$\mathbf{AP} = \mathbf{A}(:, \mathbf{p}), \quad \mathbf{P}^T \mathbf{A} = \mathbf{A}(\mathbf{p}, :). \quad (2.24)$$

Indeed, $\mathbf{AP} = (\mathbf{A}\mathbf{e}_{i_1}, \dots, \mathbf{A}\mathbf{e}_{i_n}) = \mathbf{A}(:, \mathbf{p})$ and $\mathbf{P}^T \mathbf{A} = (\mathbf{A}^T \mathbf{P})^T = (\mathbf{A}^T(:, \mathbf{p}))^T = \mathbf{A}(\mathbf{p}, :)$.

Since $\mathbf{P}^T \mathbf{P} = \mathbf{I}$ the inverse of \mathbf{P} is equal to its transpose, $\mathbf{P}^{-1} = \mathbf{P}^T$ and $\mathbf{PP}^T = \mathbf{I}$ as well. We will use a particularly simple permutation matrix.

Definition 2.33 We define a **(j,k)-Interchange matrix** \mathbf{I}_{jk} by interchanging column j and k of the identity matrix.

Since $\mathbf{I}_{jk} = \mathbf{I}_{kj}$, and we obtain the identity by applying \mathbf{I}_{jk} twice, we see that $\mathbf{I}_{jk}^2 = \mathbf{I}$ and an interchange matrix is symmetric and equal to its own inverse. Pre-multiplying a matrix by an interchange matrix interchanges two rows of the matrix, while post-multiplication interchanges two columns.

We can keep track of the row interchanges using **pivot vectors** \mathbf{p}_k . We define

$$\mathbf{p} := \mathbf{p}_n, \text{ where } \mathbf{p}_1 := [1, 2, \dots, n]^T, \text{ and } \mathbf{p}_{k+1} := \mathbf{I}_{r_k, k} \mathbf{p}_k \text{ for } k = 1, \dots, n-1. \quad (2.25)$$

We obtain \mathbf{p}_{k+1} from \mathbf{p}_k by interchanging the entries r_k and k in \mathbf{p}_k . In particular, since $r_k \geq k$, the first $k-1$ components in \mathbf{p}_k and \mathbf{p}_{k+1} are the same.

There is a close relation between the pivot vectors \mathbf{p}_k and the corresponding interchange matrices $\mathbf{P}_k := \mathbf{I}_{r_k, k}$. Since $\mathbf{P}_k \mathbf{I}(\mathbf{p}_k, :) = \mathbf{I}(\mathbf{P}_k \mathbf{p}_k, :) = \mathbf{I}(\mathbf{p}_{k+1}, :)$ we obtain

$$\mathbf{P}^T = \mathbf{P}_{n-1} \cdots \mathbf{P}_1 = \mathbf{I}(\mathbf{p}, :), \quad \mathbf{P} = \mathbf{P}_1 \mathbf{P}_2 \cdots \mathbf{P}_{n-1} = \mathbf{I}(:, \mathbf{p}). \quad (2.26)$$

Instead of interchanging the rows of \mathbf{A} during elimination we can keep track of the ordering of the rows using the pivot vectors \mathbf{p}_k . The Gaussian elimination in Section 2.1 with row pivoting starting with $a_{ij}^{(1)} = a_{ij}$ can be described as follows:

$$\begin{aligned} \mathbf{p} &= [1, \dots, n]^T; \\ \text{for } k &= 1 : n - 1 \\ &\quad \text{choose } r_k \geq k \text{ so that } a_{p_{r_k}, k}^{(k)} \neq 0. \\ \mathbf{p} &= I_{r_k, k} \mathbf{p} \\ \text{for } i &= k + 1 : n \\ a_{p_i, k}^{(k)} &= a_{p_i, k}^{(k)} / a_{p_k, k}^{(k)} \\ \text{for } j &= k : n \\ a_{p_i, j}^{(k+1)} &= a_{p_i, j}^{(k)} - a_{p_i, k}^{(k)} a_{p_k, j}^{(k)} \end{aligned} \tag{2.27}$$

This leads to the following factorization:

Theorem 2.34 *Gaussian elimination with row pivoting on a nonsingular matrix $\mathbf{A} \in \mathbb{C}^{n,n}$ leads to the factorization $\mathbf{A} = \mathbf{PLU}$, where \mathbf{P} is a permutation matrix, \mathbf{L} is lower triangular with ones on the diagonal, and \mathbf{U} is upper triangular. More explicitly, $\mathbf{P} = \mathbf{I}(:, \mathbf{p})$, where $\mathbf{p} = \mathbf{I}_{r_{n-1}, n-1} \cdots \mathbf{I}_{r_1, 1} [1, \dots, n]^T$, and*

$$\mathbf{L} = \begin{bmatrix} 1 & & & \\ a_{p_2, 1}^{(1)} & 1 & & \\ \vdots & & \ddots & \\ a_{p_n, 1}^{(1)} & a_{p_n, 2}^{(2)} & \cdots & 1 \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} a_{p_1, 1}^{(1)} & \cdots & a_{p_1, n}^{(1)} \\ \ddots & \ddots & \vdots \\ & & a_{p_n, n}^{(n)} \end{bmatrix}. \tag{2.28}$$

Proof. The proof is analogous to the proof for LU-factorization without pivoting. From (2.27) we have for all i, j

$$a_{p_i, k}^{(k)} a_{p_k, j}^{(k)} = a_{p_i, j}^{(k)} - a_{p_i, j}^{(k+1)} \text{ for } k < \min(i, j), \text{ and } a_{p_i, j}^{(k)} a_{p_j, j}^{(j)} = a_{p_i, j}^{(j)} \text{ for } i > j.$$

Thus for $i \leq j$ we find

$$\begin{aligned} (\mathbf{LU})_{ij} &= \sum_{k=1}^n l_{i,k} u_{kj} = \sum_{k=1}^{i-1} a_{p_i, k}^{(k)} a_{p_k, j}^{(k)} + a_{p_i, j}^{(i)} \\ &= \sum_{k=1}^{i-1} (a_{p_i, j}^{(k)} - a_{p_i, j}^{(k+1)}) + a_{p_i, j}^{(i)} = a_{p_i, j}^{(1)} = a_{p_i, j} = (\mathbf{P}^T \mathbf{A})_{ij}, \end{aligned}$$

while for $i > j$

$$\begin{aligned} (\mathbf{LU})_{ij} &= \sum_{k=1}^n l_{ik}^{(k)} u_{kj} = \sum_{k=1}^{j-1} a_{p_i,k}^{(k)} a_{p_k,j}^{(k)} + a_{p_i,j}^{(k)} a_{p_j,j}^{(j)} \\ &= \sum_{k=1}^{j-1} (a_{p_i,j}^{(k)} - a_{p_i,j}^{(k+1)}) + a_{p_i,j}^{(j)} = a_{p_i,j}^{(1)} = a_{p_i,j} = (\mathbf{P}^T \mathbf{A})_{ij}. \end{aligned}$$

□

The PLU factorization can also be written $\mathbf{P}^T \mathbf{A} = \mathbf{L}\mathbf{U}$. This shows that for a nonsingular matrix there is a permutation of the rows of \mathbf{A} so that the permuted matrix has an LU factorization.

2.4.3 Pivot strategies

In (2.27) we did not say which of the nonzero elements we should choose. A common strategy is to choose the largest element. This is known as **partial pivoting**

$$|a_{r_k,k}^{(k)}| := \max\{|a_{i,k}^{(k)}| : k \leq i \leq n\}$$

with r_k the smallest such index in case of a tie. The following example illustrating that small pivots should be avoided.

Example 2.35 Applying Gaussian elimination without row interchanges to the linear system

$$\begin{aligned} 10^{-4}x_1 + 2x_2 &= 4 \\ x_1 + x_2 &= 3 \end{aligned}$$

we obtain the upper triangular system

$$\begin{aligned} 10^{-4}x_1 + 2x_2 &= 4 \\ (1 - 2 \times 10^{-4})x_2 &= 3 - 4 \times 10^{-4} \end{aligned}$$

The exact solution is

$$x_2 = \frac{-39997}{-19999} \approx 2, \quad x_1 = \frac{4 - 2x_2}{10^{-4}} = \frac{20000}{19999} \approx 1.$$

Suppose we round the result of each arithmetic operation to three digits. The solutions $\text{fl}(x_1)$ and $\text{fl}(x_2)$ computed in this way is

$$\text{fl}(x_2) = 2, \quad \text{fl}(x_1) = 0.$$

The computed value 0 of x_1 is completely wrong. Suppose instead we apply Gaussian elimination to the same system, but where we have interchanged the equations. The system is

$$\begin{aligned}x_1 + x_2 &= 3 \\10^{-4}x_1 + 2x_2 &= 4\end{aligned}$$

and we obtain the upper triangular system

$$\begin{aligned}x_1 + x_2 &= 3 \\(2 - 10^{-4})x_2 &= 4 - 3 \times 10^{-4}\end{aligned}$$

Now the solution is computed as follows

$$x_2 = \frac{3.9997}{1.9999} \approx 2, \quad x_1 = 3 - x_2 \approx 1.$$

In this case rounding each calculation to three digits produces $\text{fl}(x_1) = 1$ and $\text{fl}(x_2) = 2$ which is quite satisfactory since it is the exact solution rounded to three digits.

Related to partial pivoting is **scaled partial pivoting**. Here r_k is the smallest index such that

$$\frac{|a_{r_k,k}^{(k)}|}{s_k} := \max \left\{ \frac{|a_{i,k}^{(k)}|}{s_k} : k \leq i \leq n \right\}, \quad s_k := \max_{1 \leq j \leq n} |a_{kj}|.$$

This can sometimes give more accurate results if the coefficient matrix have coefficients of wildly different sizes. Note that the scaling factors s_k are computed using the initial matrix.

It also is possible to interchange both rows and columns. The choice

$$a_{r_k,s_k}^{(k)} := \max \{ |a_{i,j}^{(k)}| : k \leq i, j \leq n \}$$

with r_k, s_k the smallest such indices in case of a tie, is known as **complete pivoting**. Complete pivoting is known to be more numerically stable than partial pivoting, but requires a lot of search and is seldom used in practice.

Exercise 2.36 (Using PLU for A^T)

Suppose we know the PLU factors P, L, U in a PLU factorization $A = PLU$ of $A \in \mathbb{R}^{n,n}$. Explain how we can solve the system $A^T x = b$ economically.

Exercise 2.37 (Using PLU for determinant)

Suppose we know the PLU factors P, L, U in a PLU factorization $A = PLU$ of $A \in \mathbb{R}^{n,n}$. Explain how we can use this to compute the determinant of A .

Exercise 2.38 (Using PLU for A^{-1})

Suppose the factors P, L, U in a PLU factorization of $A \in \mathbb{R}^{n,n}$ are known. Use Exercise 2.14 to show that it takes approximately $2G_n$ arithmetic operations to compute $A^{-1} = U^{-1}L^{-1}P^T$. Here we have not counted the final multiplication with P^T which amounts to n row interchanges.

2.5 Review Questions

2.5.1 When is a triangular matrix nonsingular?

2.5.2 What is the general condition for Gaussian elimination without row interchanges to be well defined?

2.5.3 What is the content of the LU theorem?

2.5.4 Approximately how many arithmetic operations are needed for

- the multiplication of two square matrices?
- The LU factorization of a matrix?
- the solution of $Ax = b$, when A is triangular?

2.5.5 What is a PLU factorization? When does it exist?

2.5.6 What is complete pivoting?

Chapter 3

LDL* Factorization and Positive definite Matrices

Recall that a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ is **Hermitian** if $\mathbf{A}^* = \mathbf{A}$, i.e., $a_{ji} = \bar{a}_{ij}$ for all i, j . A real Hermitian matrix is symmetric. Note that the diagonal elements of a Hermitian matrix must be real.

3.1 The LDL* factorization

There are special versions of the LU factorization for Hermitian and positive definite matrices. The most important ones are

LDL*: LDU with $\mathbf{U} = \mathbf{L}^*$ and $\mathbf{D} \in \mathbb{R}^{n \times n}$ is a diagonal matrix,

LL*: LU with $\mathbf{U} = \mathbf{L}^*$ and $l_{ii} > 0$ all i .

A matrix \mathbf{A} having an LDL* factorization must be Hermitian since \mathbf{D} is real so that $\mathbf{A}^* = (\mathbf{LDL}^*)^* = \mathbf{LD}^*\mathbf{L}^* = \mathbf{A}$. The LL* factorization is called a **Cholesky factorization**.

Example 3.1 (LDL* of 2×2 Hermitian matrix)

Let $a, d \in \mathbb{R}$ and $b \in \mathbb{C}$. An LDL* factorization of a 2×2 Hermitian matrix must satisfy the equations

$$\begin{bmatrix} a & \bar{b} \\ b & d \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ l_1 & 1 \end{bmatrix} \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix} \begin{bmatrix} 1 & \bar{l}_1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} d_1 & d_1\bar{l}_1 \\ d_1l_1 & d_1|l_1|^2 + d_2 \end{bmatrix}$$

for the unknowns l_1 in \mathbf{L} and d_1, d_2 in \mathbf{D} . They are determined from

$$d_1 = a, \quad al_1 = b, \quad d_2 = d - a|l_1|^2. \quad (3.1)$$

There are essentially three cases

1. $a \neq 0$: The matrix has a unique LDL^* factorization. Note that d_1 and d_2 are real.
2. $a = b = 0$: The LDL^* factorization exists, but it is not unique. Any value for l_1 can be used.
3. $a = 0, b \neq 0$: No LDL^* factorization exists.

Lemma 2.16 carries over to the Hermitian case.

Lemma 3.2 (LDL* of leading principal sub matrices)

Suppose $\mathbf{A} = \mathbf{LDL}^*$ is an LDL^* factorization of $\mathbf{A} \in \mathbb{C}^{n \times n}$. For $k = 1, \dots, n$ let $\mathbf{A}_{[k]}, \mathbf{L}_{[k]}$ and $\mathbf{D}_{[k]}$ be the leading principal submatrices of \mathbf{A}, \mathbf{L} and \mathbf{D} , respectively. Then $\mathbf{A}_{[k]} = \mathbf{L}_{[k]} \mathbf{D}_{[k]} \mathbf{L}_{[k]}^*$ is an LDL^* factorization of $\mathbf{A}_{[k]}$ for $k = 1, \dots, n$.

Proof. For $k = 1, \dots, n-1$ we partition $\mathbf{A} = \mathbf{LDL}^*$ as follows:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{[k]} & \mathbf{B}_k \\ \mathbf{B}_k & \mathbf{F}_k \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{[k]} & \mathbf{0} \\ \mathbf{M}_k & \mathbf{N}_k \end{bmatrix} \begin{bmatrix} \mathbf{D}_{[k]} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_k \end{bmatrix} \begin{bmatrix} \mathbf{L}_{[k]}^* & \mathbf{M}_k^* \\ \mathbf{0} & \mathbf{N}_k^* \end{bmatrix} = \mathbf{LDU}, \quad (3.2)$$

where $\mathbf{F}_k, \mathbf{N}_k, \mathbf{E}_k \in \mathbb{C}^{n-k, n-k}$. Block multiplication gives $\mathbf{A}_{[k]} = \mathbf{L}_{[k]} \mathbf{D}_{[k]} \mathbf{L}_{[k]}^*$. Since $\mathbf{L}_{[k]}$ is unit lower triangular and $\mathbf{D}_{[k]}$ is real and diagonal this is an LDL^* factorization of $\mathbf{A}_{[k]}$. \square

Theorem 3.3 (LDL* theorem)

The matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ has a unique LDL^* factorization if and only if $\mathbf{A} = \mathbf{A}^*$ and $\mathbf{A}_{[k]}$ is nonsingular for $k = 1, \dots, n-1$.

Proof. We essentially repeat the proof of Theorem 2.17 incorporating the necessary changes. Suppose $\mathbf{A}^* = \mathbf{A}$ and that $\mathbf{A}_{[k]}$ is nonsingular for $k = 1, \dots, n-1$. Note that $\mathbf{A}_{[k]}^* = \mathbf{A}_{[k]}$ for $k = 1, \dots, n$. We use induction on n to show that \mathbf{A} has a unique LDL^* factorization. The result is clearly true for $n = 1$, since the unique LDL^* factorization of a 1-by-1 matrix is $[a_{11}] = [1][a_{11}][1]$ and a_{11} is real since $\mathbf{A}^* = \mathbf{A}$. Suppose that $\mathbf{A}_{[n-1]}$ has a unique LDL^* factorization $\mathbf{A}_{[n-1]} = \mathbf{L}_{n-1} \mathbf{D}_{n-1} \mathbf{L}_{n-1}^*$, and that $\mathbf{A}_{[1]}, \dots, \mathbf{A}_{[n-1]}$ are nonsingular. By definition \mathbf{D}_{n-1} is real. Using block multiplication

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} \mathbf{A}_{[n-1]} & \mathbf{a}_n \\ \mathbf{a}_n^* & a_{nn} \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{n-1} & \mathbf{0} \\ \mathbf{l}_n^* & 1 \end{bmatrix} \begin{bmatrix} \mathbf{D}_{n-1} & \mathbf{0} \\ 0 & d_{nn} \end{bmatrix} \begin{bmatrix} \mathbf{L}_{n-1}^* & \mathbf{l}_n \\ \mathbf{0}^* & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{L}_{n-1} \mathbf{D}_{n-1} \mathbf{L}_{n-1}^* & \mathbf{L}_{n-1} \mathbf{D}_{n-1} \mathbf{l}_n \\ \mathbf{l}_n^* \mathbf{D}_{n-1} \mathbf{L}_{n-1}^* & \mathbf{l}_n^* \mathbf{D}_{n-1} \mathbf{l}_n + d_{nn} \end{bmatrix} \end{aligned} \quad (3.3)$$

if and only if $\mathbf{A}_{[n-1]} = \mathbf{L}_{n-1} \mathbf{D}_{n-1} \mathbf{L}_{n-1}^*$, and

$$\mathbf{a}_n = \mathbf{L}_{n-1} \mathbf{D}_{n-1} \mathbf{l}_n, \quad a_{nn} = \mathbf{l}_n^* \mathbf{D}_{n-1} \mathbf{l}_n + d_{nn}. \quad (3.4)$$

Thus we obtain an LDL^* factorization of \mathbf{A} that is unique since \mathbf{L}_{n-1} and \mathbf{D}_{n-1} are nonsingular. Also d_{nn} is real since a_{nn} and \mathbf{D}_{n-1} are real.

For the converse we use Lemma 3.2 in the same way as Lemma 2.16 was used to prove Theorem 2.17. \square

Here is an analog of Algorithm 2.18 that tries to compute the LDL^* factorization of a matrix. It uses the upper part of the matrix.

Algorithm 3.4 (LDL* factorization)

```

1 function [L,dg]=LDL(A,d)
2 n=length(A);
3 L=eye(n,n); dg=zeros(n,1); dg(1)=A(1,1);
4 for k=2:n
5     m=rforwardsolve(L(1:k-1,1:k-1),A(1:k-1,k),d);
6     L(k,1:k-1)=m./dg(1:k-1);
7     dg(k)=A(k,k)-L(k,1:k-1)*m;
8 end

```

The number of arithmetic operations for the LDL^* factorization is approximately $\frac{1}{2}G_n$, half the number of operations needed for the LU factorization. Indeed, in the L1U factorization we needed to solve two triangular systems to find the vectors \mathbf{s} and \mathbf{m} , while only one such system is needed to find \mathbf{x} in the symmetric case (3.3). The work to find d_{nn} is $O(n)$ and does not contribute to the highest order term.

Example 3.5 (A factorization) Is the factorization

$$\begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1/3 & 1 \end{bmatrix} \begin{bmatrix} 3 & 0 \\ 0 & 8/3 \end{bmatrix} \begin{bmatrix} 1 & 1/3 \\ 0 & 1 \end{bmatrix}$$

an LDL^* factorization?

3.2 Positive Definite and Semidefinite Matrices

Real symmetric positive definite matrices occur often in scientific computing. In this section we study some properties of such matrices. We consider the complex case as well. The real non-symmetric case is considered in a separate section.

Definition 3.6 (Positiv definite matrix)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is a square Hermitian matrix. The function $f : \mathbb{C}^n \rightarrow \mathbb{R}$ given by

$$f(\mathbf{x}) = \mathbf{x}^* \mathbf{A} \mathbf{x} = \sum_{i=1}^n \sum_{j=1}^n a_{ij} \bar{x}_i x_j$$

is called a **quadratic form**. Note that f is real valued. Indeed, $\overline{f(\mathbf{x})} = \overline{\mathbf{x}^* \mathbf{A} \mathbf{x}} = (\mathbf{x}^* \mathbf{A} \mathbf{x})^* = \mathbf{x}^* \mathbf{A}^* \mathbf{x} = f(\mathbf{x})$ since \mathbf{A} is Hermitian. We say that \mathbf{A} is

- (i) **positive definite** if $\mathbf{A}^* = \mathbf{A}$ and $\mathbf{x}^* \mathbf{A} \mathbf{x} > 0$ for all nonzero $\mathbf{x} \in \mathbb{C}^n$.
- (ii) **positive semidefinite** if $\mathbf{A}^* = \mathbf{A}$ and $\mathbf{x}^* \mathbf{A} \mathbf{x} \geq 0$ for all $\mathbf{x} \in \mathbb{C}^n$.
- (iii) **negative (semi)definite** if $\mathbf{A}^* = \mathbf{A}$ and $-\mathbf{A}$ is positive (semi)definite.

We observe that

1. The zero-matrix is positive semidefinite, while the unit matrix is positive definite.
2. The matrix \mathbf{A} is positive definite if and only if it is positive semidefinite and $\mathbf{x}^* \mathbf{A} \mathbf{x} = 0 \implies \mathbf{x} = \mathbf{0}$.
3. A positive definite matrix \mathbf{A} is nonsingular. For if $\mathbf{A} \mathbf{x} = \mathbf{0}$ then $\mathbf{x}^* \mathbf{A} \mathbf{x} = 0$ and this implies that $\mathbf{x} = \mathbf{0}$.
4. If \mathbf{A} is real then it is enough to show definiteness for real vectors only. Indeed, if $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{A}^T = \mathbf{A}$ and $\mathbf{x}^T \mathbf{A} \mathbf{x} > 0$ for all nonzero $\mathbf{x} \in \mathbb{R}^n$ then $\mathbf{z}^* \mathbf{A} \mathbf{z} > 0$ for all nonzero $\mathbf{z} \in \mathbb{C}^n$. For if $\mathbf{z} = \mathbf{x} + i\mathbf{y} \neq \mathbf{0}$ with $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ then

$$\begin{aligned}\mathbf{z}^* \mathbf{A} \mathbf{z} &= (\mathbf{x} - iy)^T \mathbf{A} (\mathbf{x} + iy) = \mathbf{x}^T \mathbf{A} \mathbf{x} - iy^T \mathbf{A} \mathbf{x} + ix^T \mathbf{A} \mathbf{y} - i^2 \mathbf{y}^T \mathbf{A} \mathbf{y} \\ &= \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{y}^T \mathbf{A} \mathbf{y}\end{aligned}$$

and this is positive since at least one of the real vectors \mathbf{x}, \mathbf{y} is nonzero.

Example 3.7 (Gradient and hessian)

Symmetric positive definite matrices is important in nonlinear optimization. Consider (cf. (B.1)) the gradient ∇f and hessian Hf of a function $f : \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}$

$$\nabla f(\mathbf{x}) = \begin{bmatrix} \frac{\partial f(\mathbf{x})}{\partial x_1} \\ \vdots \\ \frac{\partial f(\mathbf{x})}{\partial x_n} \end{bmatrix} \in \mathbb{R}^n, \quad Hf(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_1} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_n} \end{bmatrix} \in \mathbb{R}^{n \times n}.$$

We assume that f has continuous first and second partial derivatives on Ω .

Under suitable conditions on the domain Ω it is shown in advanced calculus texts that if $\nabla f(\mathbf{x}) = \mathbf{0}$ and $Hf(\mathbf{x})$ is symmetric positive definite then \mathbf{x} is a local minimum for f . This can be shown using the second-order Taylor expansion (B.2). Moreover, \mathbf{x} is a local maximum if $\nabla f(\mathbf{x}) = \mathbf{0}$ and $Hf(\mathbf{x})$ is negative definite.

A matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$ has rank n or equivalently linearly independent columns if and only if $\mathbf{Ax} = \mathbf{0} \implies \mathbf{x} = \mathbf{0}$. The Euclidian norm on \mathbb{C}^n is given by

$$\|\mathbf{y}\|_2 := \sqrt{\mathbf{y}^* \mathbf{y}} = \sum_{j=1}^n |y_j|^2, \quad \mathbf{y} = [y_1, \dots, y_n]^T \in \mathbb{C}^n. \quad (3.5)$$

Lemma 3.8 (The matrix $\mathbf{A}^* \mathbf{A}$)

The matrix $\mathbf{A}^* \mathbf{A}$ is positive semidefinite for any $m, n \in \mathbb{N}$ and $\mathbf{A} \in \mathbb{C}^{m \times n}$. It is positive definite if and only if \mathbf{A} has linearly independent columns.

Proof. Clearly $\mathbf{A}^* \mathbf{A}$ is Hermitian. The rest follows from the relation $\mathbf{x} \mathbf{A}^* \mathbf{A} \mathbf{x} = \|\mathbf{Ax}\|_2^2$ (cf. (3.5)). Clearly $\mathbf{A}^* \mathbf{A}$ is positive semidefinite. We have $\mathbf{x} \mathbf{A}^* \mathbf{A} \mathbf{x} = 0 \iff \|\mathbf{Ax}\|_2 = 0 \iff \mathbf{Ax} = \mathbf{0}$. This implies that $\mathbf{x} = 0$ if $\mathbf{A}^* \mathbf{A}$ either has linearly independent columns or is positive definite. \square

Lemma 3.9 (\mathbf{T} is symmetric positive definite)

The second derivative matrix $\mathbf{T} = \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{n \times n}$ is symmetric positive definite.

Proof. Clearly \mathbf{T} is symmetric. For any $\mathbf{x} \in \mathbb{R}^n$

$$\begin{aligned} \mathbf{x}^T \mathbf{T} \mathbf{x} &= 2 \sum_{i=1}^n x_i^2 - \sum_{i=1}^{n-1} x_i x_{i+1} - \sum_{i=2}^n x_{i-1} x_i \\ &= \sum_{i=1}^{n-1} x_i^2 - 2 \sum_{i=1}^{n-1} x_i x_{i+1} + \sum_{i=1}^{n-1} x_{i+1}^2 + x_1^2 + x_n^2 \\ &= x_1^2 + x_n^2 + \sum_{i=1}^{n-1} (x_{i+1} - x_i)^2. \end{aligned}$$

Thus $\mathbf{x}^T \mathbf{T} \mathbf{x} \geq 0$ and if $\mathbf{x}^T \mathbf{T} \mathbf{x} = 0$ then $x_1 = x_n = 0$ and $x_i = x_{i+1}$ for $i = 1, \dots, n-1$ which implies that $\mathbf{x} = 0$. Hence \mathbf{T} is positive definite. \square

3.2.1 The Cholesky factorization.

Recall that a **principal submatrix** $\mathbf{B} = \mathbf{A}(\mathbf{r}, \mathbf{r}) \in \mathbb{C}^{k \times k}$ of a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ has elements $b_{i,j} = a_{r_i, r_j}$ for $i, j = 1, \dots, k$, where $1 \leq r_1 < \dots < r_k \leq n$. It is a **leading principal submatrix**, denoted $\mathbf{A}_{[k]}$ if $\mathbf{r} = [1, 2, \dots, k]^T$.

Lemma 3.10 (Submatrices)

Any principal sub matrix of a positive (semi)definite matrix is positive (semi)definite.

Proof. Suppose the submatrix $\mathbf{B} = \mathbf{A}(\mathbf{r}, \mathbf{r})$ is defined by the rows and columns $\mathbf{r} = [r_1, \dots, r_k]^T$ of \mathbf{A} . Let $\mathbf{X} = [\mathbf{e}_{r_1}, \dots, \mathbf{e}_{r_k}] \in \mathbb{C}^{n \times k}$. Then $\mathbf{B} := \mathbf{X}^* \mathbf{A} \mathbf{X}$. Let $\mathbf{y} \in \mathbb{C}^k$ and set $\mathbf{x} := \mathbf{X} \mathbf{y} \in \mathbb{C}^n$. Then $\mathbf{y}^* \mathbf{B} \mathbf{y} = \mathbf{y}^* \mathbf{X}^* \mathbf{A} \mathbf{X} \mathbf{y} = \mathbf{x}^* \mathbf{A} \mathbf{x}$. This is nonnegative if \mathbf{A} is positive semidefinite and positive if \mathbf{A} is positive definite and \mathbf{X} has linearly independent columns. For then \mathbf{x} is nonzero if \mathbf{y} is nonzero. \square

Theorem 3.11 (LDL* and LL*)

The following is equivalent for a Hermitian matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$.

1. \mathbf{A} is positive definite,
2. \mathbf{A} has an LDL* factorization with positive diagonal elements in \mathbf{D} ,
3. \mathbf{A} has a Cholesky factorization.

If the Cholesky factorization exists it is unique.

Proof.

1 \implies 2: Suppose \mathbf{A} is positive definite. By Lemma 3.10 the leading principal submatrices $\mathbf{A}_{[k]} \in \mathbb{C}^{k \times k}$ are positive definite and by Lemma 3.17 they are nonsingular for $k = 1, \dots, n - 1$. By Theorem 2.17 \mathbf{A} has a unique LDL* factorization $\mathbf{A} = \mathbf{L} \mathbf{D} \mathbf{L}^*$. The i th diagonal element in \mathbf{D} is positive, $d_{ii} = \mathbf{e}_i^* \mathbf{D} \mathbf{e}_i = \mathbf{e}_i^* \mathbf{L}^{-1} \mathbf{A} \mathbf{L}^{-1*} \mathbf{e}_i = \mathbf{x}_i^* \mathbf{A} \mathbf{x}_i > 0$. Indeed, $\mathbf{x}_i := \mathbf{L}^{-1*} \mathbf{e}_i$ is nonzero since \mathbf{L}^{-1*} is nonsingular.

2 \implies 3: Suppose if \mathbf{A} has an LDL* factorization $\mathbf{A} = \mathbf{L} \mathbf{D} \mathbf{L}^*$ with positive diagonal elements d_{ii} in \mathbf{D} . Then $\mathbf{A} = \mathbf{S} \mathbf{S}^*$, where $\mathbf{S} := \mathbf{L} \mathbf{D}^{1/2}$ and $\mathbf{D}^{1/2} := \text{diag}(\sqrt{d_{11}}, \dots, \sqrt{d_{nn}})$, and this is a Cholesky factorization of \mathbf{A} .

3 \implies 1: Suppose \mathbf{A} has a Cholesky factorization $\mathbf{A} = \mathbf{L} \mathbf{L}^*$. Since \mathbf{L} has positive diagonal elements it is nonsingular and \mathbf{A} is positive definite by Lemma 3.8.

For uniqueness suppose $\mathbf{L} \mathbf{L}^* = \mathbf{S} \mathbf{S}^*$ are two Cholesky factorizations of the positive definite matrix \mathbf{A} . Since \mathbf{A} is nonsingular both \mathbf{L} and \mathbf{S} are nonsingular. Then $\mathbf{S}^{-1} \mathbf{L} = \mathbf{S}^* \mathbf{L}^{-1*}$, where by Lemma 1.35 $\mathbf{S}^{-1} \mathbf{L}$ is lower triangular and $\mathbf{S}^* \mathbf{L}^{-1*}$ is upper triangular, with diagonal elements ℓ_{ii}/s_{ii} and s_{ii}/ℓ_{ii} , respectively. But then both matrices must be equal to the same diagonal matrix and $\ell_{ii}^2 = s_{ii}^2$. By positivity $\ell_{ii} = s_{ii}$ and we conclude that $\mathbf{S}^{-1} \mathbf{L} = \mathbf{I} = \mathbf{S}^* \mathbf{L}^{-T}$ which means that $\mathbf{L} = \mathbf{S}$. \square

A Cholesky factorization can also be written in the equivalent form $\mathbf{A} = \mathbf{R}^* \mathbf{R}$, where $\mathbf{R} = \mathbf{L}^*$ is upper triangular with positive diagonal elements.

Example 3.12 (2 \times 2)

The matrix $\mathbf{A} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$ has an LDL^* - and a Cholesky-factorization given by

$$\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & \frac{3}{2} \end{bmatrix} \begin{bmatrix} 1 & -\frac{1}{2} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \sqrt{2} & 0 \\ -1/\sqrt{2} & \sqrt{3/2} \end{bmatrix} \begin{bmatrix} \sqrt{2} & -1/\sqrt{2} \\ 0 & \sqrt{3/2} \end{bmatrix}.$$

There are many good algorithms for finding the Cholesky factorization of a matrix, see [6]. The following version for finding the factorization of a matrix with bandwidth d uses the LDL^* factorization algorithm 3.4. Only the upper part of \mathbf{A} is used. The algorithm uses the Matlab command `diag`.

Algorithm 3.13 (bandcholesky)

```

1 function L=bandcholesky (A,d)
2 %L=bandcholesky (A,d)
3 [L,dg]=LDL(A,d);
4 L=L*diag(sqrt(dg));

```

As for the LDL^* factorization the leading term in an operation count for a band matrix is $O(d^2n)$. When d is small this is a considerable saving compared to the count $\frac{1}{2}G_n = n^3/3$ for a full matrix.

3.2.2 Positive definite and positive semidefinite criteria

Not all Hermitian matrices are positive definite, and sometimes we can tell just by glancing at the matrix that it cannot be positive definite. Here are some necessary conditions. We consider only the real case.

Theorem 3.14 (Necessary conditions for positive definite)

If $\mathbf{A} \in \mathbb{R}^{n \times n}$ is positive definite then for all i, j with $i \neq j$

1. $a_{ii} > 0$,
2. $|a_{ij}| < (a_{ii} + a_{jj})/2$,
3. $|a_{ij}| < \sqrt{a_{ii}a_{jj}}$.

Proof. Clearly $a_{ii} = \mathbf{e}_i^T \mathbf{A} \mathbf{e}_i > 0$ and Part 1 follows. If $\alpha \mathbf{e}_i + \beta \mathbf{e}_j \neq 0$ then

$$0 < (\alpha \mathbf{e}_i + \beta \mathbf{e}_j)^T \mathbf{A} (\alpha \mathbf{e}_i + \beta \mathbf{e}_j) = \alpha^2 a_{ii} + \beta^2 a_{jj} + 2\alpha\beta a_{ij}. \quad (3.6)$$

Taking $\alpha = 1, \beta = \pm 1$ we obtain $a_{ii} + a_{jj} \pm 2a_{ij} > 0$ and this implies Part 2. Taking $\alpha = -a_{ij}, \beta = a_{ii}$ in (3.6) we find

$$0 < a_{ij}^2 a_{ii} + a_{ii}^2 a_{jj} - 2a_{ij}^2 a_{ii} = a_{ii}(a_{ii}a_{jj} - a_{ij}^2).$$

Since $a_{ii} > 0$ Part 3 follows. \square

Example 3.15 (Not positive definite)

Consider the matrices

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}, \quad \mathbf{A}_2 = \begin{bmatrix} 1 & 2 \\ 2 & 2 \end{bmatrix}, \quad \mathbf{A}_3 = \begin{bmatrix} -2 & 1 \\ 1 & 2 \end{bmatrix}.$$

Here \mathbf{A}_1 and \mathbf{A}_3 are not positive definite, since a diagonal element is not positive. \mathbf{A}_2 is not positive definite since neither Part 2 nor Part 3 in Theorem 3.14 are satisfied.

The matrix $\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ enjoys all the necessary conditions in Theorem 3.14. But to decide if it is positive definite it is nice to have sufficient conditions as well.

We start by considering eigenvalues of a positive (semi)definite matrix.

Lemma 3.16 (positive eigenvalues)

A matrix is positive (semi)definite if and only if it is Hermitian and all its eigenvalues are positive (nonnegative).

Proof. Suppose \mathbf{A} is positive (semi)definite. Then \mathbf{A} is Hermitian by definition, and if $\mathbf{Ax} = \lambda\mathbf{x}$ and \mathbf{x} is nonzero, then $\mathbf{x}^*\mathbf{Ax} = \lambda\mathbf{x}^*\mathbf{x}$. This implies that $\lambda > 0 (\geq 0)$ since \mathbf{A} is positive (semi)definite and $\mathbf{x}^*\mathbf{x} = \|\mathbf{x}\|_2^2 > 0$. Conversely, suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is Hermitian with positive (nonnegative) eigenvalues $\lambda_1, \dots, \lambda_n$. By Theorem 5.38 (the spectral theorem) there is a matrix $\mathbf{U} \in \mathbb{C}^{n \times n}$ with $\mathbf{U}^*\mathbf{U} = \mathbf{U}\mathbf{U}^* = \mathbf{I}$ such that $\mathbf{U}^*\mathbf{AU} = \text{diag}(\lambda_1, \dots, \lambda_n)$. Let $\mathbf{x} \in \mathbb{C}^n$ and define $\mathbf{z} := \mathbf{U}^*\mathbf{x} = [z_1, \dots, z_n]^T \in \mathbb{C}^n$. Then $\mathbf{x} = \mathbf{U}\mathbf{U}^*\mathbf{x} = \mathbf{U}\mathbf{z}$ and by the spectral theorem

$$\mathbf{x}^*\mathbf{Ax} = \mathbf{z}^*\mathbf{U}^*\mathbf{AU}\mathbf{z} = \mathbf{z}^*\text{diag}(\lambda_1, \dots, \lambda_n)\mathbf{z} = \sum_{j=1}^n \lambda_j |z_j|^2 \geq 0.$$

It follows that \mathbf{A} is positive semidefinite. Since \mathbf{U}^* is nonsingular we see that $\mathbf{z} = \mathbf{U}^*\mathbf{x}$ is nonzero if \mathbf{x} is nonzero, and therefore \mathbf{A} is positive definite. \square

Lemma 3.17 (positive semidefinite and nonsingular)

A matrix is positive definite if and only if it is positive semidefinite and nonsingular.

Proof. If \mathbf{A} is positive definite then it is positive semidefinite and if $\mathbf{Ax} = \mathbf{0}$ then $\mathbf{x}^*\mathbf{Ax} = 0$ which implies that $\mathbf{x} = \mathbf{0}$. Conversely, if \mathbf{A} is positive semidefinite then it is Hermitian with nonnegative eigenvalues. Since it is nonsingular all eigenvalues are positive showing that \mathbf{A} is positive definite. \square

The following necessary and sufficient conditions can be used to decide if a matrix is positive definite.

Theorem 3.18 (Positive definite characterization)

The following statements are equivalent for a Hermitian matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$.

1. \mathbf{A} is positive definite.
2. \mathbf{A} has only positive eigenvalues.
3. All leading principal submatrices have a positive determinant.
4. $\mathbf{A} = \mathbf{B}\mathbf{B}^*$ for a nonsingular $\mathbf{B} \in \mathbb{C}^{n \times n}$.

Proof.

1 \iff 2: This follows from Lemma 3.16.

1 \implies 3: A positive definite matrix has positive eigenvalues, and since the determinant of a matrix equals the product of its eigenvalues (cf. Theorem 0.38) the determinant is positive. Every leading principal submatrix of a positive definite matrix is positive definite (cf. Lemma 3.10) and therefore has a positive determinant.

3 \implies 4: Since a leading principal submatrix has a positive determinant it is nonsingular and Theorem 2.17 implies that \mathbf{A} has a unique LDL^* factorization $\mathbf{A} = \mathbf{L}\mathbf{D}\mathbf{L}^*$. By Lemma 3.2 $\mathbf{A}_{[k]} = \mathbf{L}_{[k]}\mathbf{D}_{[k]}\mathbf{L}_{[k]}^*$ is an LDL^* factorization of $\mathbf{A}_{[k]}$, $k = 1, \dots, n$. But then for all k

$$\det(\mathbf{A}_{[k]}) = \det(\mathbf{L}_{[k]}) \det(\mathbf{D}_{[k]}) \det(\mathbf{L}_{[k]}^*) = \det(\mathbf{D}_{[k]}) = d_{11} \cdots d_{kk} > 0.$$

But then \mathbf{D} has positive diagonal elements and we have $\mathbf{A} = \mathbf{B}\mathbf{B}^*$, where $\mathbf{B} := \mathbf{L}\mathbf{D}^{1/2}$.

4 \implies 1: This follows from Lemma 3.8. \square

Example 3.19 (Positive definite characterization)

Consider the symmetric matrix $\mathbf{A} := \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix}$.

1. We have $\mathbf{x}^T \mathbf{A} \mathbf{x} = 2x_1^2 + 2x_2^2 + (x_1 + x_2)^2 > 0$ for all nonzero \mathbf{x} showing that \mathbf{A} is positive definite.
2. The eigenvalues of \mathbf{A} are $\lambda_1 = 2$ and $\lambda_2 = 4$. They are positive showing that \mathbf{A} is positive definite since it is symmetric.
3. We find $\det(\mathbf{A}_{[1]}) = 3$ and $\det(\mathbf{A}_{[2]}) = 8$ showing again that \mathbf{A} is positive definite.

4. Finally \mathbf{A} is positive definite since by Example 3.5 we have

$$\mathbf{A} = \mathbf{B}\mathbf{B}^*, \quad \mathbf{B} = \begin{bmatrix} 1 & 0 \\ 1/3 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{3} & 0 \\ 0 & \sqrt{8/3} \end{bmatrix}.$$

Exercise 3.20 (Positive definite characterizations)

Show directly that all 4 characterizations in Theorem 3.18 hold for the matrix

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}.$$

3.3 Semi-Cholesky factorization of a banded matrix

A Hermitian positive semidefinite matrix has a factorization that is similar to the Cholesky factorization.

Definition 3.21 (Semi-Cholesky factorization)

A factorization $\mathbf{A} = \mathbf{L}\mathbf{L}^*$ where \mathbf{L} is lower triangular with nonnegative diagonal elements is called a **semi-Cholesky factorization**.

Note that a semi-Cholesky factorization of a positive definite matrix is necessarily a Cholesky factorization. For if \mathbf{A} is positive definite then it is nonsingular and then \mathbf{L} must be nonsingular. Thus the diagonal elements of \mathbf{L} cannot be zero.

Theorem 3.22 (Characterization, semi-Cholesky factorization)

A matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ has a semi-Cholesky factorization $\mathbf{A} = \mathbf{L}\mathbf{L}^*$ if and only if it is symmetric positive semidefinite.

Proof. If $\mathbf{A} = \mathbf{L}\mathbf{L}^*$ is a semi-Cholesky factorization then $\mathbf{x}^*\mathbf{A}\mathbf{x} = \|\mathbf{L}^*\mathbf{x}\|_2^2 \geq 0$ and \mathbf{A} is positive semidefinite. For the converse we use induction on n . A positive semidefinite matrix of order one has a semi-Cholesky factorization since the one and only element in \mathbf{A} is nonnegative. Suppose any symmetric positive semidefinite matrix of order $n - 1$ has a semi-Cholesky factorization and suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is symmetric positive semidefinite. We partition \mathbf{A} as follows

$$\mathbf{A} = \begin{bmatrix} \alpha & \mathbf{v}^* \\ \mathbf{v} & \mathbf{B} \end{bmatrix}, \quad \alpha \in \mathbb{C}, \quad \mathbf{v} \in \mathbb{C}^{n-1}, \quad \mathbf{B} \in \mathbb{C}^{(n-1) \times (n-1)}. \quad (3.7)$$

There are two cases. Suppose first $\alpha = \mathbf{e}_1^* \mathbf{A} \mathbf{e}_1 > 0$. We claim that $\mathbf{C} := \mathbf{B} - \mathbf{v}\mathbf{v}^*/\alpha$ is symmetric positive semidefinite. \mathbf{C} is symmetric. To show that \mathbf{C} is positive

semidefinite we consider any $\mathbf{y} \in \mathbb{C}^{n-1}$ and define $\mathbf{x}^* := [-\mathbf{v}^* \mathbf{y}/\alpha, \mathbf{y}^*] \in \mathbb{C}^n$. Then

$$\begin{aligned} 0 \leq \mathbf{x}^* \mathbf{A} \mathbf{x} &= [-\mathbf{v}^* \mathbf{y}/\alpha, \mathbf{y}^*] \begin{bmatrix} \alpha & \mathbf{v}^* \\ \mathbf{v} & \mathbf{B} \end{bmatrix} \begin{bmatrix} -\mathbf{v}^* \mathbf{y}/\alpha \\ \mathbf{y} \end{bmatrix} \\ &= [0, -(\mathbf{v}^* \mathbf{y}) \mathbf{v}^*/\alpha + \mathbf{y}^* \mathbf{B}] \begin{bmatrix} -\mathbf{v}^* \mathbf{y}/\alpha \\ \mathbf{y} \end{bmatrix} \\ &= -(\mathbf{v}^* \mathbf{y})(\mathbf{v}^* \mathbf{y})/\alpha + \mathbf{y}^* \mathbf{B} \mathbf{y} = \mathbf{y}^* \mathbf{C} \mathbf{y}, \end{aligned} \quad (3.8)$$

since $(\mathbf{v}^* \mathbf{y}) \mathbf{v}^* \mathbf{y} = (\mathbf{v}^* \mathbf{y})^* \mathbf{v}^* \mathbf{y} = \mathbf{y}^* \mathbf{v} \mathbf{v}^* \mathbf{y}$. So $\mathbf{C} \in \mathbb{C}^{(n-1) \times (n-1)}$ is symmetric positive semidefinite and by the induction hypothesis it has a semi-Cholesky factorization $\mathbf{C} = \mathbf{L}_1 \mathbf{L}_1^*$. The matrix

$$\mathbf{L}^* := \begin{bmatrix} \beta & \mathbf{v}^* / \beta \\ \mathbf{0} & \mathbf{L}_1^* \end{bmatrix}, \quad \beta := \sqrt{\alpha}, \quad (3.9)$$

is upper triangular with nonnegative diagonal elements and

$$\mathbf{L} \mathbf{L}^* = \begin{bmatrix} \beta & \mathbf{0} \\ \mathbf{v} / \beta & \mathbf{L}_1 \end{bmatrix} \begin{bmatrix} \beta & \mathbf{v}^* / \beta \\ \mathbf{0} & \mathbf{L}_1^* \end{bmatrix} = \begin{bmatrix} \alpha & \mathbf{v}^* \\ \mathbf{v} & \mathbf{B} \end{bmatrix} = \mathbf{A}$$

is a semi-Cholesky factorization of \mathbf{A} .

If $\alpha = 0$ then¹¹ $\mathbf{v} = \mathbf{0}$. Moreover, $\mathbf{B} \in \mathbb{C}^{(n-1) \times (n-1)}$ in (3.7) is positive semidefinite and therefore has a semi-Cholesky factorization $\mathbf{B} = \mathbf{L}_1 \mathbf{L}_1^*$. But then $\mathbf{L} \mathbf{L}^*$, where $\mathbf{L} = \begin{bmatrix} 0 & \mathbf{0}^* \\ \mathbf{0} & \mathbf{L}_1 \end{bmatrix}$ is a semi-Cholesky factorization of \mathbf{A} . Indeed, \mathbf{L} is lower triangular and

$$\mathbf{L} \mathbf{L}^* = \begin{bmatrix} 0 & \mathbf{0}^* \\ \mathbf{0} & \mathbf{L}_1 \end{bmatrix} \begin{bmatrix} 0 & \mathbf{0}^* \\ \mathbf{0} & \mathbf{L}_1^* \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{0}^* \\ \mathbf{0} & \mathbf{B} \end{bmatrix} = \mathbf{A}.$$

□

Recall that a matrix \mathbf{A} is d -banded if $a_{ij} = 0$ for $|i - j| > d$. A (semi-) Cholesky factorization preserves bandwidth.

Theorem 3.23 (Bandwidth semi-Cholesky factor)

The semi-Cholesky factor \mathbf{L} given by (3.9) has the same bandwidth as \mathbf{A} .

Proof. Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is d -banded. Then $\mathbf{v}^* = [\mathbf{u}^*, \mathbf{0}^*]$ in (3.7), where $\mathbf{u} \in \mathbb{C}^d$, and therefore $\mathbf{C} := \mathbf{B} - \mathbf{v} \mathbf{v}^*/\alpha$ differs from \mathbf{B} only in the upper left $d \times d$ corner. It follows that \mathbf{C} has the same bandwidth as \mathbf{B} and \mathbf{A} . By induction on

¹¹A semidefinite version of Part 3 in Lemma 3.14 shows that $|a_{ij}| \leq \sqrt{a_{ii} a_{jj}}$. Thus, if $a_{ii} = 0$ then $a_{ij} = 0$ for all j .

n , $\mathbf{C} = \mathbf{L}_1 \mathbf{L}_1^*$, where \mathbf{L}_1^* has the same bandwidth as \mathbf{C} . But then \mathbf{L} in (3.9) has the same bandwidth as \mathbf{A} . \square

Consider now implementing an algorithm based on the previous discussion. Since \mathbf{A} is symmetric we only need to use the lower part of \mathbf{A} . The first column of \mathbf{L} is $[\beta, \mathbf{v}^*/\beta]^*$ if $\alpha > 0$. If $\alpha = 0$ then by 4 in Lemma 3.14 the first column of \mathbf{A} is zero and this is also the first column of \mathbf{L} . We obtain

```

if  $A(1, 1) > 0$ 
     $A(1, 1) = \sqrt{A(1, 1)}$ 
     $A(2 : n, 1) = A(2 : n, 1)/A(1, 1)$ 
    for  $j = 2 : n$ 
         $A(j : n, j) = A(j : n, j) - A(j, 1) * A(j : n, 1)$ 

```

(3.10)

Here we store the first column of \mathbf{L} in the first column of \mathbf{A} and the lower part of $\mathbf{C} = \mathbf{B} - \mathbf{v}\mathbf{v}^*/\alpha$ in the lower part of $A(2 : n, 2 : n)$.

The code can be made more efficient when \mathbf{A} is a d -banded matrix. We simply replace all occurrences of n by $\min(i + d, n)$. Continuing the reduction we arrive at the following algorithm.

Algorithm 3.24 (bandsemi-cholesky)

Suppose \mathbf{A} is positive semidefinite. A lower triangular matrix \mathbf{L} is computed so that $\mathbf{A} = \mathbf{L}\mathbf{L}^*$. This is the Cholesky factorization of \mathbf{A} if \mathbf{A} is positive definite and a semi-Cholesky factorization of \mathbf{A} otherwise. The algorithm uses the Matlab command `tril`.

```

1 function L=bandsemicholeskyL(A,d)
2 %L=bandsemicholeskyL(A,d)
3 n=length(A);
4 for k=1:n
5     if A(k,k)>0
6         kp=min(n,k+d);
7         A(k,k)=sqrt(A(k,k));
8         A((k+1):kp,k)=A((k+1):kp,k)/A(k,k);
9         for j=k+1:kp
10             A(j:kp,j)=A(j:kp,j)-A(j,k)*A(j:kp,k);
11         end
12     else
13         A(k:kp,k)=zeros(kp-k+1,1);
14     end
15 end
16 L=tril(A);

```

In the algorithm we overwrite the lower triangle of \mathbf{A} with the elements of

L. Column k of \mathbf{L} is zero for those k where $\ell_{kk} = 0$. We reduce round-off noise by forcing those rows to be zero. In the semidefinite case no update is necessary and we “do nothing”.

Deciding when a diagonal element is zero can be a problem in floating point arithmetic.

We end the section with some necessary and sufficient conditions for a matrix to be positive semidefinite.

Theorem 3.25 (Positive semidefinite characterization)

The following is equivalent for a Hermitian matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$.

1. \mathbf{A} is positive semidefinite.
2. \mathbf{A} has only nonnegative eigenvalues.
3. All principal submatrices have a nonnegative determinant.
4. $\mathbf{A} = \mathbf{B}\mathbf{B}^*$ for some $\mathbf{B} \in \mathbb{C}^{n \times n}$.

Proof.

1 \iff 2: This follows from Lemma 3.16.

1. \iff 3.: follows from Theorem 3.22 below, while 1. \Rightarrow 4. is a consequence of Theorem 3.10. To prove 4. \Rightarrow 1. one first shows that $\epsilon\mathbf{I} + \mathbf{A}$ is symmetric positive definite for all $\epsilon > 0$ (Cf. page 567 of [26]). But then $\mathbf{x}^*\mathbf{A}\mathbf{x} = \lim_{\epsilon \rightarrow 0} \mathbf{x}^*(\epsilon\mathbf{I} + \mathbf{A})\mathbf{x} \geq 0$ for all $\mathbf{x} \in \mathbb{C}^n$. \square

Example 3.26 (Positive semidefinite characterization)

Consider the symmetric matrix $\mathbf{A} := \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$.

1. We have $\mathbf{x}^*\mathbf{A}\mathbf{x} = x_1^2 + x_2^2 + x_1x_2 + x_2x_1 = (x_1 + x_2)^2 \geq 0$ for all $\mathbf{x} \in \mathbb{R}^2$ showing that \mathbf{A} is positive semidefinite.
2. The eigenvalues of \mathbf{A} are $\lambda_1 = 2$ and $\lambda_2 = 0$ and they are nonnegative showing that \mathbf{A} is positive semidefinite since it is symmetric.
3. There are three principal submatrices, and they have determinants $\det([a_{11}]) = 1$, $\det([a_{22}]) = 1$ and $\det(\mathbf{A}) = 0$ and showing again that \mathbf{A} is positive semidefinite.
4. Finally \mathbf{A} is positive semidefinite since $\mathbf{A} = \mathbf{B}\mathbf{B}^*$, where $\mathbf{B} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}$.

In part 4 of Theorem 3.25 we require nonnegativity of all principal minors, while only positivity of leading principal minors was required for positive definite

matrices (cf. Theorem 3.18). To see that nonnegativity of the leading principal minors is not enough consider the matrix $\mathbf{A} := \begin{bmatrix} 0 & 0 \\ 0 & -1 \end{bmatrix}$. The leading principal minors are nonnegative, but \mathbf{A} is not positive semidefinite.

3.4 The non-symmetric real case

In this section we say that a matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ is **positive semidefinite** if $\mathbf{x}^* \mathbf{A} \mathbf{x} \geq 0$ for all $\mathbf{x} \in \mathbb{R}^n$ and **positive definite** if $\mathbf{x}^* \mathbf{A} \mathbf{x} > 0$ for all nonzero $\mathbf{x} \in \mathbb{R}^n$. Thus we do not require \mathbf{A} to be symmetric. This means that some of the eigenvalues can be complex (cf. Example 3.28). Note that a non-symmetric positive definite matrix is nonsingular, but in Exercise 3.29 you show that the converse is not true.

We have the following theorem.

Theorem 3.27 (The non-symmetric case)

Suppose $\mathbf{A} \in \mathbb{R}^{n,n}$ is positive definite.

1. Every principal submatrix of \mathbf{A} is positive definite,
2. \mathbf{A} has a unique LU factorization,
3. the real eigenvalues of \mathbf{A} are positive,
4. $\det(\mathbf{A}) > 0$,
5. $a_{ii}a_{jj} > a_{ij}a_{ji}$, for $i \neq j$.

Proof.

1. The proof is the same as for Lemma 3.10.
2. Since all leading submatrices are positive definite they are nonsingular and the result follows from the LU Theorem 2.17.
3. Suppose (λ, \mathbf{x}) is an eigenpair of \mathbf{A} and that λ is real. Since \mathbf{A} is real we can choose \mathbf{x} to be real. Multiplying $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ by \mathbf{x}^T and solving for λ we find $\lambda = \frac{\mathbf{x}^T \mathbf{A} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} > 0$.
4. The determinant of \mathbf{A} equals the product of its eigenvalues. The eigenvalues are either real and positive or occur in complex conjugate pairs. The product of two nonzero complex conjugate numbers is positive.
5. The principal submatrix $\begin{bmatrix} a_{ii} & a_{ij} \\ a_{ji} & a_{jj} \end{bmatrix}$ has a positive determinant.

□

Example 3.28 (2×2 positive definite)

A non-symmetric positive definite matrix can have complex eigenvalues. The family of matrices

$$\mathbf{A}[a] := \begin{bmatrix} 2 & 2-a \\ a & 1 \end{bmatrix}, \quad a \in \mathbb{R}$$

is positive definite for any $a \in \mathbb{R}$. Indeed

$$\mathbf{x}^T \mathbf{A} \mathbf{x} = 2x_1^2 + (2-a)x_1x_2 + ax_2x_1 + x_2^2 = x_1^2 + (x_1 + x_2)^2 > 0.$$

The eigenvalues of $\mathbf{A}[a]$ are positive for $a \in [1 - \frac{\sqrt{5}}{2}, 1 + \frac{\sqrt{5}}{2}]$ and complex for other values of a .

Exercise 3.29 (A counterexample)

In the non-symmetric case a nonsingular positive semidefinite matrix is not necessarily positive definite. Show this by considering the matrix $\mathbf{A} := \begin{bmatrix} 1 & 0 \\ -2 & 1 \end{bmatrix}$.

3.5 Review Questions

3.5.1 What is the content of the LDL* theorem?

3.5.2 Is $\mathbf{A}^* \mathbf{A}$ always positive definite?

3.5.3 What class of matrices has a Cholesky factorization?

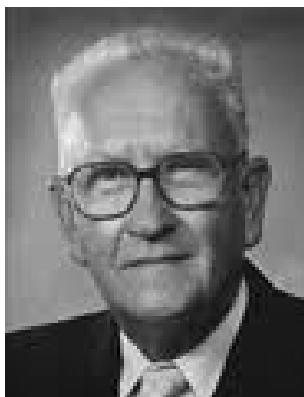
3.5.4 What is the bandwidth of the Cholesky factor of a band matrix?

3.5.5 For a symmetric matrix give 3 conditions that are equivalent to positive definiteness.

3.5.6 What class of matrices has a semi-Cholesky factorization?

Chapter 4

Orthonormal and Unitary Transformations



Alston Scott Householder, 1904-1993 (left), James Hardy Wilkinson, 1919-1986 (right). Householder and Wilkinson are two of the founders of modern numerical analysis and scientific computing.

In Gaussian elimination and LU factorization we solve a linear system by transforming it to triangular form. These are not the only kind of transformations that can be used for such a task. Matrices with orthonormal columns, called unitary matrices can be used to reduce a square matrix to upper triangular form and more generally a rectangular matrix to upper triangular (also called upper trapezoidal) form. This lead to a decomposition of a rectangular matrix known as a **QR decomposition** and a reduced form which we refer to as a **QR factorization**. The QR decomposition and factorization will be used in later chapters to solve least squares- and eigenvalue problems.

Unitary transformations have the advantage that they preserve the Euclidian norm of a vector. This means that when a unitary transformation is applied to an inaccurate vector then the error will not grow. Thus a unitary transformation is numerically stable. We consider two classes of unitary transformations known as Householder- and Givens transformations, respectively.

4.1 Inner products, orthogonality and unitary matrices

An **inner product** or **scalar product** in a vector space is a function mapping pairs of vectors into a scalar.

4.1.1 Real and complex inner products

Definition 4.1 (Inner product)

An **inner product** in a complex vector space \mathcal{V} is a function $\mathcal{V} \times \mathcal{V} \rightarrow \mathbb{C}$ satisfying for all $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{V}$ and all $a, b \in \mathbb{C}$ the following conditions:

1. $\langle \mathbf{x}, \mathbf{x} \rangle \geq 0$ with equality if and only if $\mathbf{x} = \mathbf{0}$. (positivity)
2. $\langle \mathbf{x}, \mathbf{y} \rangle = \overline{\langle \mathbf{y}, \mathbf{x} \rangle}$ (skew symmetry)
3. $\langle a\mathbf{x} + b\mathbf{y}, \mathbf{z} \rangle = a\langle \mathbf{x}, \mathbf{z} \rangle + b\langle \mathbf{y}, \mathbf{z} \rangle$. (linearity)

The pair $(\mathcal{V}, \langle \cdot, \cdot \rangle)$ is called an **inner product space**.

Note the complex conjugate in 2. We find

$$\langle \mathbf{x}, a\mathbf{y} + b\mathbf{z} \rangle = \bar{a}\langle \mathbf{x}, \mathbf{y} \rangle + \bar{b}\langle \mathbf{x}, \mathbf{z} \rangle, \quad \langle a\mathbf{x}, a\mathbf{y} \rangle = |a|^2 \langle \mathbf{x}, \mathbf{y} \rangle. \quad (4.1)$$

An **inner product** in a real vector space \mathcal{V} is real valued function satisfying Properties 1,2,3 in Definition 4.1, where we can replace skew symmetry by symmetry

$$\langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{y}, \mathbf{x} \rangle \text{ (symmetry).}$$

In the real case we have linearity in the second variable since we can remove the complex conjugates in (4.1).

The **standard inner product** in \mathbb{C}^n is given by

$$\langle \mathbf{x}, \mathbf{y} \rangle := \mathbf{y}^* \mathbf{x} = \mathbf{x}^T \bar{\mathbf{y}} = \sum_{j=1}^n x_j \bar{y}_j.$$

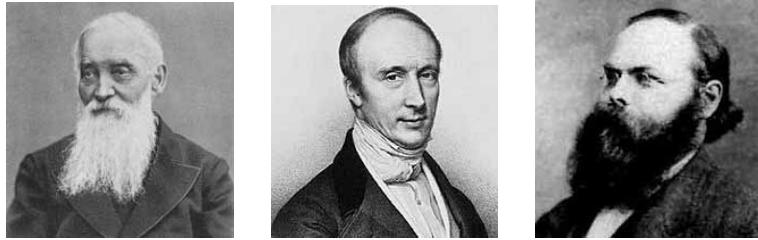
It is clearly an inner product in \mathbb{C}^n .

The function

$$\|\cdot\| : \mathcal{V} \rightarrow \mathbb{R}, \quad \mathbf{x} \mapsto \|\mathbf{x}\| := \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} \quad (4.2)$$

is called the **inner product norm**.

The inner product norm for the standard inner product is the Euclidian norm $\|\mathbf{x}\| = \|\mathbf{x}\|_2 = \sqrt{\mathbf{x}^* \mathbf{x}}$.



Viktor Yakovlevich Bunyakovsky, 1804-1889 (left), Augustin-Louis Cauchy, 1789-1857 (center), Karl Hermann Amandus Schwarz, 1843-1921 (right). The name Bunyakovsky is also associated with the Cauchy-Schwarz inequality.

The following inequality holds for any inner product.

Theorem 4.2 (Cauchy-Schwarz inequality)

For any \mathbf{x}, \mathbf{y} in a real or complex inner product space

$$|\langle \mathbf{x}, \mathbf{y} \rangle| \leq \|\mathbf{x}\| \|\mathbf{y}\|, \quad (4.3)$$

with equality if and only if \mathbf{x} and \mathbf{y} are linearly dependent.

Proof. If $\mathbf{y} = \mathbf{0}$ then $\langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{x}, 0\mathbf{y} \rangle = 0\langle \mathbf{x}, \mathbf{y} \rangle = 0$ and $\|\mathbf{y}\| = 0$. Thus the inequality holds with equality, and \mathbf{x} and \mathbf{y} are linearly dependent. So assume $\mathbf{y} \neq \mathbf{0}$. Define

$$\mathbf{z} := \mathbf{x} - a\mathbf{y}, \quad a := \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\langle \mathbf{y}, \mathbf{y} \rangle}.$$

Then $\langle \mathbf{z}, \mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle - a\langle \mathbf{y}, \mathbf{y} \rangle = 0$ so that by 2. and (4.1)

$$\langle a\mathbf{y}, \mathbf{z} \rangle + \langle \mathbf{z}, a\mathbf{y} \rangle = a\overline{\langle \mathbf{z}, \mathbf{y} \rangle} + \bar{a}\langle \mathbf{z}, \mathbf{y} \rangle = 0. \quad (4.4)$$

But then

$$\begin{aligned} \|\mathbf{x}\|^2 &= \langle \mathbf{x}, \mathbf{x} \rangle = \langle \mathbf{z} + a\mathbf{y}, \mathbf{z} + a\mathbf{y} \rangle \\ &\stackrel{(4.4)}{=} \langle \mathbf{z}, \mathbf{z} \rangle + \langle a\mathbf{y}, a\mathbf{y} \rangle \stackrel{(4.1)}{=} \|\mathbf{z}\|^2 + |a|^2 \|\mathbf{y}\|^2 \\ &\geq |a|^2 \|\mathbf{y}\|^2 = \frac{|\langle \mathbf{x}, \mathbf{y} \rangle|^2}{\|\mathbf{y}\|^2}. \end{aligned}$$

Multiplying by $\|\mathbf{y}\|^2$ gives (4.3). We have equality if and only if $\mathbf{z} = \mathbf{0}$, which means that \mathbf{x} and \mathbf{y} are linearly dependent. \square

Theorem 4.3 (Inner product norm)

For all \mathbf{x}, \mathbf{y} in an inner product space and all a in \mathbb{C} we have

1. $\|\mathbf{x}\| \geq 0$ with equality if and only if $\mathbf{x} = \mathbf{0}$. (positivity)
2. $\|a\mathbf{x}\| = |a| \|\mathbf{x}\|$. (homogeneity)
3. $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$. (subadditivity)

In general a function $\|\cdot\| : \mathbb{C}^n \rightarrow \mathbb{R}$ that satisfies these properties is called a **vector norm**. A class of vector norms called p -norms will be studied in Chapter 7.

Proof. The first statement is an immediate consequence of positivity, while the second one follows from (4.1). Expanding $\|\mathbf{x} + a\mathbf{y}\|^2 = \langle \mathbf{x} + a\mathbf{y}, \mathbf{x} + a\mathbf{y} \rangle$ using (4.1) we obtain

$$\|\mathbf{x} + a\mathbf{y}\|^2 = \|\mathbf{x}\|^2 + a\langle \mathbf{y}, \mathbf{x} \rangle + \bar{a}\langle \mathbf{x}, \mathbf{y} \rangle + |a|^2\|\mathbf{y}\|^2, \quad a \in \mathbb{C}, \quad \mathbf{x}, \mathbf{y} \in \mathcal{V}. \quad (4.5)$$

Now (4.5) with $a = 1$ and the Cauchy-Schwarz inequality implies

$$\|\mathbf{x} + \mathbf{y}\|^2 \leq \|\mathbf{x}\|^2 + 2\|\mathbf{x}\|\|\mathbf{y}\| + \|\mathbf{y}\|^2 = (\|\mathbf{x}\| + \|\mathbf{y}\|)^2.$$

Taking square roots completes the proof. \square

In the real case the Cauchy-Schwarz inequality implies that $-1 \leq \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{x}\|\|\mathbf{y}\|} \leq 1$ for nonzero \mathbf{x} and \mathbf{y} , so there is a unique angle θ in $[0, \pi]$ such that

$$\cos \theta = \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{x}\|\|\mathbf{y}\|}. \quad (4.6)$$

This defines the **angle** between vectors in a real inner product space.

Exercise 4.4 (The $\mathbf{A}^* \mathbf{A}$ inner product)

Suppose $\mathbf{A} \in \mathbb{C}^{m \times n}$ has linearly independent columns. Show that $\langle \mathbf{x}, \mathbf{y} \rangle := \mathbf{x}^* \mathbf{A}^* \mathbf{A} \mathbf{y}$ defines an inner product on \mathbb{C}^n .

Exercise 4.5 (Angle between vectors in complex case)

Show that in the complex case there is a unique angle θ in $[0, \pi/2]$ such that

$$\cos \theta = \frac{|\langle \mathbf{x}, \mathbf{y} \rangle|}{\|\mathbf{x}\|\|\mathbf{y}\|}. \quad (4.7)$$

4.1.2 Orthogonality

Definition 4.6 (Orthogonality)

Two vectors \mathbf{x}, \mathbf{y} in a real or complex inner product space are **orthogonal** or **perpendicular**, denoted as $\mathbf{x} \perp \mathbf{y}$, if $\langle \mathbf{x}, \mathbf{y} \rangle = 0$. The vectors are **orthonormal** if in addition $\|\mathbf{x}\| = \|\mathbf{y}\| = 1$.

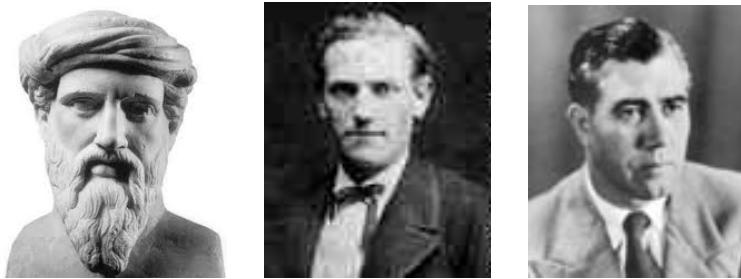
From the definitions (4.6), (4.7) of angle θ between two vectors in \mathbb{R}^n or \mathbb{C}^n it follows that $\mathbf{x} \perp \mathbf{y}$ if and only if $\theta = \pi/2$.

Theorem 4.7 (Pythagoras)

For a real or complex inner product space

$$\|\mathbf{x} + \mathbf{y}\|^2 = \|\mathbf{x}\|^2 + \|\mathbf{y}\|^2, \quad \text{if } \mathbf{x} \perp \mathbf{y}. \quad (4.8)$$

Proof. We set $a = 1$ in (4.5) and use the orthogonality. \square



Pythagoras of Samos, BC 570-BC 495 (left), Jørgen Pedersen Gram, 1850-1916 (center), Erhard Schmidt, 1876-1959 (right).

Definition 4.8 (Orthogonal- and orthonormal bases)

A set of nonzero vectors $\{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ in a subspace \mathcal{S} of a real or complex inner product space is an **orthogonal basis** for \mathcal{S} if it is a basis for \mathcal{S} and $\langle \mathbf{v}_i, \mathbf{v}_j \rangle = 0$ for $i \neq j$. It is an **orthonormal basis** for \mathcal{S} if it is a basis for \mathcal{S} and $\langle \mathbf{v}_i, \mathbf{v}_j \rangle = \delta_{ij}$ for all i, j .

A basis for a subspace of an inner product space can be turned into an orthogonal- or orthonormal basis for the subspace by the following construction.

Theorem 4.9 (Gram-Schmidt)

Let $\{\mathbf{s}_1, \dots, \mathbf{s}_k\}$ be a basis for a real or complex inner product space $(\mathcal{S}, \langle \cdot, \cdot \rangle)$. Define

$$\mathbf{v}_1 := \mathbf{s}_1, \quad \mathbf{v}_j := \mathbf{s}_j - \sum_{i=1}^{j-1} \frac{\langle \mathbf{s}_j, \mathbf{v}_i \rangle}{\langle \mathbf{v}_i, \mathbf{v}_i \rangle} \mathbf{v}_i, \quad j = 2, \dots, k. \quad (4.9)$$

Then $\{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ is an orthogonal basis for \mathcal{S} and the normalized vectors

$$\{\mathbf{u}_1, \dots, \mathbf{u}_k\} := \left\{ \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|}, \dots, \frac{\mathbf{v}_k}{\|\mathbf{v}_k\|} \right\}$$

form an orthonormal basis for \mathcal{S} .

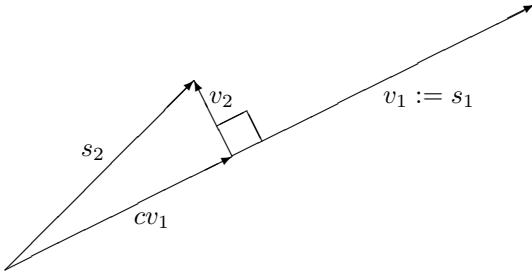


Figure 4.1: The construction of \mathbf{v}_1 and \mathbf{v}_2 in Gram-Schmidt. The constant c is given by $c := \langle \mathbf{s}_2, \mathbf{v}_1 \rangle / \langle \mathbf{v}_1, \mathbf{v}_1 \rangle$.

Proof. To show that $\{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ is an orthogonal basis for \mathcal{S} we use induction on k . Define subspaces $S_j := \text{span}\{\mathbf{s}_1, \dots, \mathbf{s}_j\}$ for $j = 1, \dots, k$. Clearly $\mathbf{v}_1 = \mathbf{s}_1$ is an orthogonal basis for S_1 . Suppose for some $j \geq 2$ that $\mathbf{v}_1, \dots, \mathbf{v}_{j-1}$ is an orthogonal basis for S_{j-1} and let \mathbf{v}_j be given by (4.9) as a linear combination of \mathbf{s}_j and $\mathbf{v}_1, \dots, \mathbf{v}_{j-1}$. Now each of these \mathbf{v}_i is a linear combination of $\mathbf{s}_1, \dots, \mathbf{s}_i$, and we obtain $\mathbf{v}_j = \sum_{i=1}^j a_i \mathbf{s}_i$ for some a_0, \dots, a_j with $a_j = 1$. Since $\mathbf{s}_1, \dots, \mathbf{s}_j$ are linearly independent and $a_j \neq 0$ we deduce that $\mathbf{v}_j \neq 0$. By the induction hypothesis

$$\langle \mathbf{v}_j, \mathbf{v}_l \rangle = \langle \mathbf{s}_j, \mathbf{v}_l \rangle - \sum_{i=1}^{j-1} \frac{\langle \mathbf{s}_j, \mathbf{v}_i \rangle}{\langle \mathbf{v}_i, \mathbf{v}_i \rangle} \langle \mathbf{v}_i, \mathbf{v}_l \rangle = \langle \mathbf{s}_j, \mathbf{v}_l \rangle - \frac{\langle \mathbf{s}_j, \mathbf{v}_l \rangle}{\langle \mathbf{v}_l, \mathbf{v}_l \rangle} \langle \mathbf{v}_l, \mathbf{v}_l \rangle = 0$$

for $l = 1, \dots, j-1$. Thus $\mathbf{v}_1, \dots, \mathbf{v}_j$ is an orthogonal basis for S_j .

If $\{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ is an orthogonal basis for \mathcal{S} then clearly $\{\mathbf{u}_1, \dots, \mathbf{u}_k\}$ is an orthonormal basis for \mathcal{S} . \square

Sometimes we want to extend an orthogonal basis for a subspace to an orthogonal basis for a larger space.

Theorem 4.10 (Orthogonal Extension of basis)

Suppose $\mathcal{S} \subset \mathcal{T}$ are finite dimensional subspaces of a vector space \mathcal{V} . An orthogonal basis for \mathcal{S} can always be extended to an orthogonal basis for \mathcal{T} .

Proof. Suppose $\dim \mathcal{S} := k < \dim \mathcal{T} = n$. Using Theorem 0.10 we first extend an orthogonal basis $\mathbf{s}_1, \dots, \mathbf{s}_k$ for \mathcal{S} to a basis $\mathbf{s}_1, \dots, \mathbf{s}_k, \mathbf{s}_{k+1}, \dots, \mathbf{s}_n$ for \mathcal{T} , and then apply the Gram-Schmidt process to this basis obtaining an orthogonal basis $\mathbf{v}_1, \dots, \mathbf{v}_n$ for \mathcal{T} . This is an extension of the basis for \mathcal{S} since $\mathbf{v}_i = \mathbf{s}_i$ for $i = 1, \dots, k$. We show this by induction. Clearly $\mathbf{v}_1 = \mathbf{s}_1$. Suppose for some $2 \leq$

$r < k$ that $\mathbf{v}_j = \mathbf{s}_j$ for $j = 1, \dots, r - 1$. Consider (4.9) for $j = r$. Since $\langle \mathbf{s}_r, \mathbf{v}_i \rangle = \langle \mathbf{s}_r, \mathbf{s}_i \rangle = 0$ for $i < r$ we obtain $\mathbf{v}_r = \mathbf{s}_r$. \square

Letting $\mathcal{S} = \text{span}(\mathbf{s}_1, \dots, \mathbf{s}_k)$ and \mathcal{T} be \mathbb{R}^n or \mathbb{C}^n we obtain

Corollary 4.11 (Extending orthogonal vectors to a basis)

For $1 \leq k < n$ a set $\{\mathbf{s}_1, \dots, \mathbf{s}_k\}$ of nonzero orthogonal vectors in \mathbb{R}^n or \mathbb{C}^n can be extended to an orthogonal basis for the whole space.

4.1.3 Unitary and orthogonal matrices

In the rest of this chapter orthogonality is in terms of the **standard inner product in \mathbb{C}^n** given by $\langle \mathbf{x}, \mathbf{y} \rangle := \mathbf{y}^* \mathbf{x} = \sum_{j=1}^n x_j \bar{y}_j$.

A matrix $\mathbf{U} \in \mathbb{C}^{n \times n}$ is **unitary** if $\mathbf{U}^* \mathbf{U} = \mathbf{I}$. If \mathbf{U} is real then $\mathbf{U}^T \mathbf{U} = \mathbf{I}$ and \mathbf{U} is called an **orthogonal matrix**. Unitary and orthogonal matrices have orthonormal columns.

If $\mathbf{U}^* \mathbf{U} = \mathbf{I}$ the matrix \mathbf{U} is nonsingular, $\mathbf{U}^{-1} = \mathbf{U}^*$ and $\mathbf{U} \mathbf{U}^* = \mathbf{I}$ as well. Moreover, both the columns and rows of a unitary matrix of order n form orthonormal bases for \mathbb{C}^n . We also note that the product of two unitary matrices is unitary. Indeed if $\mathbf{U}_1^* \mathbf{U}_1 = \mathbf{I}$ and $\mathbf{U}_2^* \mathbf{U}_2 = \mathbf{I}$ then $(\mathbf{U}_1 \mathbf{U}_2)^* (\mathbf{U}_1 \mathbf{U}_2) = \mathbf{U}_2^* \mathbf{U}_1^* \mathbf{U}_1 \mathbf{U}_2 = \mathbf{I}$.

Theorem 4.12 (Unitary matrix)

The matrix $\mathbf{U} \in \mathbb{C}^{n \times n}$ is unitary if and only if $\langle \mathbf{U}\mathbf{x}, \mathbf{U}\mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$. In particular, if \mathbf{U} is unitary then $\|\mathbf{U}\mathbf{x}\|_2 = \|\mathbf{x}\|_2$ for all $\mathbf{x} \in \mathbb{C}^n$.

Proof. If $\mathbf{U}^* \mathbf{U} = \mathbf{I}$ and $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$ then

$$\langle \mathbf{U}\mathbf{x}, \mathbf{U}\mathbf{y} \rangle = (\mathbf{U}\mathbf{y})^* (\mathbf{U}\mathbf{x}) = \mathbf{y}^* \mathbf{U}^* \mathbf{U}\mathbf{x} = \mathbf{y}^* \mathbf{x} = \langle \mathbf{x}, \mathbf{y} \rangle.$$

Conversely, if $\langle \mathbf{U}\mathbf{x}, \mathbf{U}\mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$ then $\mathbf{U}^* \mathbf{U} = \mathbf{I}$ since for $i, j = 1, \dots, n$

$$(\mathbf{U}^* \mathbf{U})_{i,j} = \mathbf{e}_i^T \mathbf{U}^* \mathbf{U} \mathbf{e}_j = (\mathbf{U} \mathbf{e}_i)^* (\mathbf{U} \mathbf{e}_j) = \langle \mathbf{U} \mathbf{e}_j, \mathbf{U} \mathbf{e}_i \rangle = \langle \mathbf{e}_j, \mathbf{e}_i \rangle = \mathbf{e}_i^* \mathbf{e}_j = \delta_{i,j}.$$

The last part of the theorem follows immediately by taking $\mathbf{y} = \mathbf{x}$: \square

4.2 The Householder Transformation

We consider now a unitary matrix with many useful properties.

Definition 4.13 (Householder transformation)

A matrix $\mathbf{H} \in \mathbb{C}^{n \times n}$ of the form

$$\mathbf{H} := \mathbf{I} - \mathbf{u}\mathbf{u}^*, \text{ where } \mathbf{u} \in \mathbb{C}^n \text{ and } \mathbf{u}^* \mathbf{u} = 2$$

is called a **Householder transformation**. The name **elementary reflector** is also used.

In the real case and for $n = 2$ we find $\mathbf{H} = \begin{bmatrix} 1-u_1^2 & -u_1 u_2 \\ -u_2 u_1 & 1-u_2^2 \end{bmatrix}$. A Householder transformation is Hermitian and unitary. Indeed, $\mathbf{H}^* = (\mathbf{I} - \mathbf{u}\mathbf{u}^*)^* = \mathbf{H}$ and

$$\mathbf{H}^* \mathbf{H} = \mathbf{H}^2 = (\mathbf{I} - \mathbf{u}\mathbf{u}^*)(\mathbf{I} - \mathbf{u}\mathbf{u}^*) = \mathbf{I} - 2\mathbf{u}\mathbf{u}^* + \mathbf{u}(\mathbf{u}^*\mathbf{u})\mathbf{u}^* = \mathbf{I}.$$

In the real case \mathbf{H} is symmetric and orthonormal.

There are several ways to represent a Householder transformation. Householder used $\mathbf{I} - 2\mathbf{u}\mathbf{u}^*$, where $\mathbf{u}^*\mathbf{u} = 1$. For any nonzero $\mathbf{v} \in \mathbb{R}^n$ the matrix

$$\mathbf{H} := \mathbf{I} - 2 \frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}} \quad (4.10)$$

is a Householder transformation. Indeed, $\mathbf{H} = \mathbf{I} - \mathbf{u}\mathbf{u}^T$, where $\mathbf{u} := \sqrt{2} \frac{\mathbf{v}}{\|\mathbf{v}\|_2}$ has length $\sqrt{2}$.

Two vectors in \mathbb{R}^n of the same length can be mapped into each other by a Householder transformation.

Lemma 4.14 Suppose $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ with $\|\mathbf{x}\|_2 = \|\mathbf{y}\|_2$ and $\mathbf{v} := \mathbf{x} - \mathbf{y} \neq \mathbf{0}$. Then $(\mathbf{I} - 2 \frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}})\mathbf{x} = \mathbf{y}$.

Proof. Since $\mathbf{x}^T\mathbf{x} = \mathbf{y}^T\mathbf{y}$ we have

$$\mathbf{v}^T\mathbf{v} = (\mathbf{x} - \mathbf{y})^T(\mathbf{x} - \mathbf{y}) = 2\mathbf{x}^T\mathbf{x} - 2\mathbf{y}^T\mathbf{x} = 2\mathbf{v}^T\mathbf{x}. \quad (4.11)$$

But then $(\mathbf{I} - 2 \frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}})\mathbf{x} = \mathbf{x} - \frac{2\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}}\mathbf{v} = \mathbf{x} - \mathbf{v} = \mathbf{y}$. \square

A geometric interpretation of this lemma is shown in Figure 4.2. We have

$$\mathbf{H} = \mathbf{I} - \frac{2\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}} = \mathbf{P} - \frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}}, \text{ where } \mathbf{P} := \mathbf{I} - \frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}},$$

and

$$\mathbf{P}\mathbf{x} = \mathbf{x} - \frac{\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}}\mathbf{v} \stackrel{(4.11)}{=} \mathbf{x} - \frac{1}{2}\mathbf{v} = \frac{1}{2}(\mathbf{x} + \mathbf{y}).$$

It follows that $\mathbf{H}\mathbf{x}$ is the reflected image of \mathbf{x} . The mirror $\mathcal{M} := \{\mathbf{w} \in \mathbb{R}^n : \mathbf{w}^T\mathbf{v} = 0\}$ contains the vector $(\mathbf{x} + \mathbf{y})/2$ and has normal $\mathbf{x} - \mathbf{y}$.

Example 4.15 (Reflector)

Suppose $\mathbf{x} := [1, 0, 1]^T$ and $\mathbf{y} := [-1, 0, 1]^T$. Then $\mathbf{v} = \mathbf{x} - \mathbf{y} = [2, 0, 0]^T$ and

$$\mathbf{H} := \mathbf{I} - \frac{2\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \frac{2}{4} \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 2 & 0 & 0 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

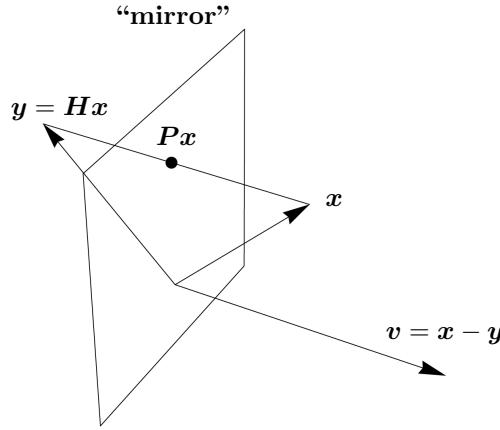


Figure 4.2: The Householder transformation in Example 4.15

$$\mathbf{P} := \mathbf{I} - \frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \frac{1}{4} \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 2 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$\mathcal{M} := \{\mathbf{w} \in \mathbb{R}^3 : \mathbf{w}^T \mathbf{v} = 0\} = \left\{ \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} : 2w_1 = 0 \right\}$$

is the yz plane (cf. Figure 4.2), $\mathbf{Hx} = [-1, 0, 1]^T = \mathbf{y}$, and $\mathbf{Px} = [0, 0, 1]^T = (\mathbf{x} + \mathbf{y})/2 \in \mathcal{M}$.

Householder transformations can be used to produce zeros in vectors. In the following Theorem we map any vector in \mathbb{C}^n into a multiple of the first unit vector.

Theorem 4.16 (Zeros in vectors)

For any $\mathbf{x} \in \mathbb{C}^n$ there is a Householder transformation $\mathbf{H} \in \mathbb{C}^{n \times n}$ such that

$$\mathbf{Hx} = a\mathbf{e}_1, \quad a = -\rho\|\mathbf{x}\|_2, \quad \rho := \begin{cases} x_1/|x_1|, & \text{if } x_1 \neq 0, \\ 1, & \text{otherwise.} \end{cases}$$

Proof. If $\mathbf{x} = 0$ then $a = 0$, any \mathbf{u} with $\|\mathbf{u}\|_2 = \sqrt{2}$ will do, and we choose $\mathbf{u} := \sqrt{2}\mathbf{e}_1$ in this case. Suppose $\mathbf{x} \neq 0$. we define

$$\mathbf{u} := \frac{\mathbf{z} + \mathbf{e}_1}{\sqrt{1 + z_1}}, \quad \text{where } \mathbf{z} := \bar{\rho}\mathbf{x}/\|\mathbf{x}\|_2. \quad (4.12)$$

Since $|\rho| = 1$ we have $\rho\|\mathbf{x}\|_2 \mathbf{z} = |\rho|^2 \mathbf{x} = \mathbf{x}$. Moreover, $\|\mathbf{z}\|_2 = 1$ and $z_1 = |x_1|/\|\mathbf{x}\|_2$ is real so that $\mathbf{u}^* \mathbf{u} = \frac{(\mathbf{z} + \mathbf{e}_1)^*(\mathbf{z} + \mathbf{e}_1)}{1+z_1} = \frac{2+2z_1}{1+z_1} = 2$. Finally,

$$\begin{aligned}\mathbf{H}\mathbf{x} &= \mathbf{x} - (\mathbf{u}^* \mathbf{x}) \mathbf{u} = \rho\|\mathbf{x}\|_2 (\mathbf{z} - (\mathbf{u}^* \mathbf{z}) \mathbf{u}) = \rho\|\mathbf{x}\|_2 (\mathbf{z} - \frac{(\mathbf{z}^* + \mathbf{e}_1^*) \mathbf{z}}{1+z_1} (\mathbf{z} + \mathbf{e}_1)) \\ &= \rho\|\mathbf{x}\|_2 (\mathbf{z} - (\mathbf{z} + \mathbf{e}_1)) = -\rho\|\mathbf{x}\|_2 \mathbf{e}_1 = a \mathbf{e}_1.\end{aligned}$$

□

The formulas in Theorem 4.16 are implemented in the following algorithm adapted from [28]. To any given $\mathbf{x} \in \mathbb{C}^n$ a number a and a vector \mathbf{u} with $\mathbf{u}^* \mathbf{u} = 2$ is computed so that $(\mathbf{I} - \mathbf{u}\mathbf{u}^*)\mathbf{x} = a\mathbf{e}_1$.

Algorithm 4.17 (Generate a Householder transformation)

```

1 function [u,a]=housegen(x)
2 a=norm(x);
3 if a==0
4   u=x; u(1)=sqrt(2); return;
5 end
6 if x(1)==0
7   r=1;
8 else
9   r=x(1)/abs(x(1));
10 end
11 u=conj(r)*x/a;
12 u(1)=u(1)+1;
13 u=u/sqrt(u(1));
14 a=-r*a;
15 end
```

Note that

- In Theorem 4.16 the first component of \mathbf{z} is $z_1 = |x_1|/\|\mathbf{x}\|_2 \geq 0$. Since $\|\mathbf{z}\|_2 = 1$ we have $1 \leq 1 + z_1 \leq 2$. It follows that we avoid cancelation error when computing $1 + z_1$ and \mathbf{u} and a are computed in a numerically stable way.
- In order to compute $\mathbf{H}\mathbf{x}$ for a vector \mathbf{x} we do not need to form the matrix \mathbf{H} . Indeed, $\mathbf{H}\mathbf{x} = (\mathbf{I} - \mathbf{u}\mathbf{u}^*)\mathbf{x} = \mathbf{x} - (\mathbf{u}^* \mathbf{x}) \mathbf{u}$. If $\mathbf{u}, \mathbf{x} \in \mathbb{R}^m$ this requires $2m$ operations to find $\mathbf{u}^T \mathbf{x}$, m operations for $(\mathbf{u}^T \mathbf{x}) \mathbf{u}$ and m operations for the final subtraction of the two vectors, a total of $4m$ arithmetic operations. If $\mathbf{A} \in \mathbb{R}^{m \times n}$ then $4mn$ operations are required for $\mathbf{H}\mathbf{A} = \mathbf{A} - (\mathbf{u}^T \mathbf{A}) \mathbf{u}$, i.e., $4m$ operations for each of the n columns of \mathbf{A} .
- Householder transformations can also be used to zero out only the lower part of a vector. Suppose $\mathbf{x}^T := [\mathbf{y}, \mathbf{z}]^T$, where $\mathbf{y} \in \mathbb{C}^k$, $\mathbf{z} \in \mathbb{C}^{n-k}$ for some

$1 \leq k < n$. The command $[\hat{\mathbf{u}}, a] := \text{housegen}(\mathbf{z})$ defines a Householder transformation $\hat{\mathbf{H}} = \mathbf{I} - \hat{\mathbf{u}}\hat{\mathbf{u}}^*$ so that $\hat{\mathbf{H}}\mathbf{z} = a\mathbf{e}_1$. With $\mathbf{u}^T := [\mathbf{0}, \hat{\mathbf{u}}]^T \in \mathbb{C}^n$ we see that $\mathbf{u}^*\mathbf{u} = \hat{\mathbf{u}}^*\hat{\mathbf{u}} = 2$, and

$$\mathbf{H}\mathbf{x} = \begin{bmatrix} \mathbf{y} \\ a\mathbf{e}_1 \end{bmatrix}, \text{ where } \mathbf{H} := \mathbf{I} - \mathbf{u}\mathbf{u}^* = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} - \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{u}} \end{bmatrix} \begin{bmatrix} \mathbf{0} & \hat{\mathbf{u}}^* \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{H}} \end{bmatrix},$$

defines a Householder transformation that produces zeros in the lower part of \mathbf{x} .

Exercise 4.18 (What does algorithm housegen do when $\mathbf{x} = \mathbf{e}_1$?)

Determine \mathbf{H} in Algorithm 4.17 when $\mathbf{x} = \mathbf{e}_1$.

Exercise 4.19 (Examples of Householder transformations)

If $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ with $\|\mathbf{x}\|_2 = \|\mathbf{y}\|_2$ and $\mathbf{v} := \mathbf{x} - \mathbf{y} \neq \mathbf{0}$ then it follows from Example 4.15 that $(\mathbf{I} - 2\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}})\mathbf{x} = \mathbf{y}$. Use this to construct a Householder transformation \mathbf{H} such that $\mathbf{H}\mathbf{x} = \mathbf{y}$ in the following cases.

a) $\mathbf{x} = \begin{bmatrix} 3 \\ 4 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 5 \\ 0 \end{bmatrix}.$

b) $\mathbf{x} = \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix}.$

Exercise 4.20 (2×2 Householder transformation)

Show that a real 2×2 Householder transformation can be written in the form

$$\mathbf{H} = \begin{bmatrix} -\cos \phi & \sin \phi \\ \sin \phi & \cos \phi \end{bmatrix}.$$

Find $\mathbf{H}\mathbf{x}$ if $\mathbf{x} = [\cos \phi, \sin \phi]^T$.

4.3 Householder Triangulation

We say that a matrix $\mathbf{R} \in \mathbb{C}^{m \times n}$ is **upper trapezoidal**, if $r_{i,j} = 0$ for $j < i$ and $i = 2, 3, \dots, m$. Upper trapezoidal matrices corresponding to $m < n$, $m = n$, and $m > n$ look as follows:

$$\begin{bmatrix} x & x & x & x \\ 0 & x & x & x \\ 0 & 0 & x & x \end{bmatrix}, \quad \begin{bmatrix} x & x & x & x \\ 0 & x & x & x \\ 0 & 0 & x & x \\ 0 & 0 & 0 & x \end{bmatrix}, \quad \begin{bmatrix} x & x & x \\ 0 & x & x \\ 0 & 0 & x \\ 0 & 0 & 0 \end{bmatrix}.$$

In this section we consider a method for bringing a matrix to upper trapezoidal form using Householder transformations. We treat the cases $m > n$ and $m \leq n$ separately and consider first $m > n$. We describe how to find a sequence $\mathbf{H}_1, \dots, \mathbf{H}_n$ of Householder transformations such that

$$\mathbf{A}_{n+1} := \mathbf{H}_n \mathbf{H}_{n-1} \cdots \mathbf{H}_1 \mathbf{A} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix} = \mathbf{R},$$

and where \mathbf{R}_1 is upper triangular. We define

$$\mathbf{A}_1 := \mathbf{A}, \quad \mathbf{A}_{k+1} = \mathbf{H}_k \mathbf{A}_k, \quad k = 1, 2, \dots, n.$$

Suppose \mathbf{A}_k is upper trapezoidal in its first $k - 1$ columns (which is true for $k = 1$)

$$\mathbf{A}_k = \left[\begin{array}{ccc|ccccc} a_{1,1}^{(1)} & \cdots & a_{1,k-1}^{(1)} & a_{1,k}^{(1)} & \cdots & a_{1,j}^{(1)} & \cdots & a_{1,n}^{(1)} \\ \ddots & & \vdots & \vdots & & \vdots & & \vdots \\ & a_{k-1,k-1}^{(k-1)} & & a_{k-1,k}^{(k-1)} & \cdots & a_{k-1,j}^{(k-1)} & \cdots & a_{k-1,n}^{(k-1)} \\ \hline & & & a_{k,k}^{(k)} & \cdots & a_{k,j}^{(k)} & \cdots & a_{k,n}^{(k)} \\ & & & \vdots & & \vdots & & \vdots \\ & a_{i,k}^{(k)} & \cdots & a_{i,j}^{(k)} & \cdots & a_{i,n}^{(k)} & & \\ & \vdots & & \vdots & & \vdots & & \\ & a_{m,k}^{(k)} & \cdots & a_{m,j}^{(k)} & \cdots & a_{m,n}^{(k)} & & \end{array} \right] \quad (4.13)$$

$$= \begin{bmatrix} \mathbf{B}_k & \mathbf{C}_k \\ \mathbf{0} & \mathbf{D}_k \end{bmatrix}.$$

Let $\hat{\mathbf{H}}_k := \mathbf{I} - \hat{\mathbf{u}}_k \hat{\mathbf{u}}_k^*$ be a Householder transformation that maps the first column $[a_{k,k}^{(k)}, \dots, a_{m,k}^{(k)}]^T$ of \mathbf{D}_k to a multiple of \mathbf{e}_1 , $\hat{\mathbf{H}}_k(\mathbf{D}_k \mathbf{e}_1) = a_k \mathbf{e}_1$. Using Algorithm 4.17 we have $[\hat{\mathbf{u}}_k, a_k] = \text{housegen}(\mathbf{D}_k \mathbf{e}_1)$. Then $\mathbf{H}_k := \begin{bmatrix} \mathbf{I}_{k-1} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{H}}_k \end{bmatrix}$ is a Householder transformation and

$$\mathbf{A}_{k+1} := \mathbf{H}_k \mathbf{A}_k = \begin{bmatrix} \mathbf{B}_k & \mathbf{C}_k \\ \mathbf{0} & \hat{\mathbf{H}}_k \mathbf{D}_k \end{bmatrix} = \begin{bmatrix} \mathbf{B}_{k+1} & \mathbf{C}_{k+1} \\ \mathbf{0} & \mathbf{D}_{k+1} \end{bmatrix},$$

where $\mathbf{B}_{k+1} \in \mathbb{C}^{k \times k}$ is upper triangular and $\mathbf{D}_{k+1} \in \mathbb{C}^{(m-k) \times (n-k)}$. Thus \mathbf{A}_{k+1} is upper trapezoidal in its first k columns and the reduction has been carried one step further. At the end $\mathbf{R} := \mathbf{A}_{n+1} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix}$, where \mathbf{R}_1 is upper triangular.

The process can also be applied to $\mathbf{A} \in \mathbb{C}^{m \times n}$ if $m \leq n$. If $m = 1$ then \mathbf{A} is already in upper trapezoidal form. Suppose $m > 1$. In this case $m-1$ Householder transformations will suffice and $\mathbf{H}_{m-1} \cdots \mathbf{H}_1 \mathbf{A}$ is upper trapezoidal.

In an algorithm we can store most of the vector $\hat{\mathbf{u}}_k = [u_{kk}, \dots, u_{mk}]^T$ and the matrix and \mathbf{A}_k in \mathbf{A} . However, the elements $u_{k,k}$ and $a_k = r_{k,k}$ have to

compete for the diagonal in \mathbf{A} . For $m = 4$ and $n = 3$ the two possibilities look as follows:

$$\mathbf{A} = \begin{bmatrix} u_{11} & r_{12} & r_{13} \\ u_{21} & u_{22} & r_{23} \\ u_{31} & u_{32} & u_{33} \\ u_{41} & u_{42} & u_{43} \end{bmatrix} \text{ or } \mathbf{A} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ u_{21} & r_{22} & r_{23} \\ u_{31} & u_{32} & r_{33} \\ u_{41} & u_{42} & u_{43} \end{bmatrix}.$$

Whatever alternative is chosen, if the looser is needed, it has to be stored in a separate vector. In the following algorithm we store $a_k = r_{k,k}$ in \mathbf{A} . We also apply the Householder transformations to a second matrix \mathbf{B} . We will see that the algorithm can be used to solve linear systems and least squares problems with right hand side(s) \mathbf{B} and to compute the product of the Householder transformations by choosing $\mathbf{B} = \mathbf{I}$.

Algorithm 4.21 (Householder triangulation)

Suppose $\mathbf{A} \in \mathbb{C}^{m \times n}$, $\mathbf{B} \in \mathbb{C}^{m \times r}$ and let $s := \min(n, m - 1)$. The algorithm uses housegen to compute Householder transformations $\mathbf{H}_1, \dots, \mathbf{H}_s$ such that $\mathbf{R} = \mathbf{H}_s \cdots \mathbf{H}_1 \mathbf{A}$ is upper trapezoidal and $\mathbf{C} = \mathbf{H}_s \cdots \mathbf{H}_1 \mathbf{B}$. If \mathbf{B} is the empty matrix then \mathbf{C} is the empty matrix with m rows and 0 columns.

```

1 function [R,C] = housetriang(A,B)
2 [m,n]=size(A); r=size(B,2); A=[A,B];
3 for k=1:min(n,m-1)
4     [v,A(k,k)]=housegen(A(k:m,k));
5     C=A(k:m,k+1:n+r); A(k:m,k+1:n+r)=C-v*(v'*C);
6 end
7 R=triu(A(:,1:n)); C=A(:,n+1:n+r);

```

Here $v = \hat{\mathbf{u}}_k$ and the update is computed as $\hat{\mathbf{H}}_k \mathbf{C} = (\mathbf{I} - vv^*)\mathbf{C} = \mathbf{C} - v(v^*\mathbf{C})$. The Matlab command triu extracts the upper triangular part of \mathbf{A} introducing zeros in rows $n + 1, \dots, m$.

4.3.1 The number of arithmetic operations

The bulk of the work in Algorithm 4.21 is the computation of $\mathbf{C} - v * (v^* * \mathbf{C})$ for each k . Since in Algorithm 4.21, $\mathbf{C} \in \mathbb{C}^{(m-k+1) \times (n+r-k)}$ and $m \geq n$ the cost of computing the update $\mathbf{C} - v * (v^T * \mathbf{C})$ in the real case is $4(m - k + 1)(n + r - k)$ arithmetic operations. This implies that the work in Algorithm 4.21 can be estimated as

$$\int_0^n 4(m - k)(n + r - k)dk = 2m(n + r)^2 - \frac{2}{3}(n + r)^3. \quad (4.14)$$

For $m = n$ and $r = 0$ this gives $4n^3/3 = 2G_n$ for the number of arithmetic operations to bring a matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ to upper triangular form using Householder transformations.

4.3.2 Solving linear systems using unitary transformations

Consider now the linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$, where \mathbf{A} is square. Using Algorithm 4.21 we obtain an upper triangular system $\mathbf{R}\mathbf{x} = \mathbf{c}$ that is upper triangular and nonsingular if \mathbf{A} is nonsingular. Thus, it can be solved by back substitution and we have a method for solving linear systems that is an alternative to Gaussian elimination. The two methods are similar since they both reduce \mathbf{A} to upper triangular form using certain transformations and they both work for nonsingular systems.

Which method is better? Here is a short discussion.

- Advantages with Householder:
 - Row interchanges are not necessary, but see [6].
 - Numerically stable.
- Advantages with Gauss
 - Half the number of arithmetic operations compared to Householder.
 - Row interchanges are often not necessary.
 - Usually stable (but no guarantee).

Linear systems can be constructed where Gaussian elimination will fail numerically even if row interchanges are used, see [36]. On the other hand the transformations used in Householder triangulation are unitary so the method is quite stable. So why is Gaussian elimination more popular than Householder triangulation? One reason is that the number of arithmetic operations in (4.14) when $m = n$ is $4n^3/3 = 2G_n$, which is twice the number for Gaussian elimination. Numerical stability can be a problem with Gaussian elimination, but years and years of experience shows that it works well for most practical problems and pivoting is often not necessary. Also Gaussian elimination often wins for banded and sparse problems.

4.4 The QR Decomposition and QR Factorization

Gaussian elimination without row interchanges results in an LU factorization $\mathbf{A} = \mathbf{L}\mathbf{U}$ of $\mathbf{A} \in \mathbb{R}^{n \times n}$. Consider Householder triangulation of \mathbf{A} . Applying Algorithm 4.21 gives $\mathbf{R} = \mathbf{H}_{n-1} \cdots \mathbf{H}_1 \mathbf{A}$ implying the factorization $\mathbf{A} = \mathbf{Q}\mathbf{R}$, where $\mathbf{Q} = \mathbf{H}_1 \cdots \mathbf{H}_{n-1}$ is orthonormal and \mathbf{R} is upper triangular. This is known as a QR-factorization of \mathbf{A} .

4.4.1 Existence

For a rectangular matrix we define the following.

Definition 4.22 (QR decomposition)

Let $\mathbf{A} \in \mathbb{C}^{m \times n}$ with $m, n \in \mathbb{N}$. We say that $\mathbf{A} = \mathbf{Q}\mathbf{R}$ is a **QR decomposition** of \mathbf{A} if $\mathbf{Q} \in \mathbb{C}^{m,m}$ is square and unitary and $\mathbf{R} \in \mathbb{C}^{m \times n}$ is upper trapezoidal. If $m \geq n$ then \mathbf{R} takes the form

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0}_{m-n,n} \end{bmatrix}$$

where $\mathbf{R}_1 \in \mathbb{C}^{n \times n}$ is upper triangular and $\mathbf{0}_{m-n,n}$ is the zero matrix with $m - n$ rows and n columns. For $m \geq n$ we call $\mathbf{A} = \mathbf{Q}_1\mathbf{R}_1$ a **QR factorization** of \mathbf{A} if $\mathbf{Q}_1 \in \mathbb{C}^{m \times n}$ has orthonormal columns and $\mathbf{R}_1 \in \mathbb{C}^{n \times n}$ is upper triangular.

Suppose $m \geq n$. A QR factorization is obtained from a QR decomposition $\mathbf{A} = \mathbf{Q}\mathbf{R}$ by simply using the first n columns of \mathbf{Q} and the first n rows of \mathbf{R} . Indeed, if we partition \mathbf{Q} as $[\mathbf{Q}_1, \mathbf{Q}_2]$ and $\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix}$, where $\mathbf{Q}_1 \in \mathbb{R}^{m \times n}$ and $\mathbf{R}_1 \in \mathbb{R}^{n \times n}$ then $\mathbf{A} = \mathbf{Q}_1\mathbf{R}_1$ is a QR factorization of \mathbf{A} . On the other hand a QR factorization $\mathbf{A} = \mathbf{Q}_1\mathbf{R}_1$ of \mathbf{A} can be turned into a QR decomposition by extending the set of columns $\{\mathbf{q}_1, \dots, \mathbf{q}_n\}$ of \mathbf{Q}_1 into an orthonormal basis $\{\mathbf{q}_1, \dots, \mathbf{q}_n, \mathbf{q}_{n+1}, \dots, \mathbf{q}_m\}$ for \mathbb{R}^m and adding $m - n$ rows of zeros to \mathbf{R}_1 . We then obtain the QR decomposition $\mathbf{A} = \mathbf{Q}\mathbf{R}$, where $\mathbf{Q} = [\mathbf{q}_1, \dots, \mathbf{q}_m]$ and $\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix}$.

Example 4.23 (QR decomposition and factorization)

An example of a QR decomposition is

$$\mathbf{A} = \begin{bmatrix} 1 & 3 & 1 \\ 1 & 3 & 7 \\ 1 & -1 & -4 \\ 1 & -1 & 2 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \times \begin{bmatrix} 2 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 6 \\ 0 & 0 & 0 \end{bmatrix} = \mathbf{Q}\mathbf{R},$$

while a QR factorization $\mathbf{A} = \mathbf{Q}_1\mathbf{R}_1$ is obtained by dropping the last column of \mathbf{Q} and the last row of \mathbf{R} , so that

$$\mathbf{A} = \frac{1}{2} \begin{bmatrix} 1 & 1 & -1 \\ 1 & 1 & 1 \\ 1 & -1 & -1 \\ 1 & -1 & 1 \end{bmatrix} \times \begin{bmatrix} 2 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 6 \end{bmatrix} = \mathbf{Q}_1\mathbf{R}_1.$$

Consider existence and uniqueness.

Theorem 4.24 (Existence of QR decomposition)

Any matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$ with $m, n \in \mathbb{N}$ has a QR decomposition.

Proof. If $m = 1$ then \mathbf{A} is already in upper trapezoidal form and $\mathbf{A} = [1]\mathbf{A}$ is a QR decomposition of \mathbf{A} . Suppose $m > 1$ and set $s := \min(m - 1, n)$. Note that

the function `housegen(x)` returns the vector \mathbf{u} in a Householder transformation for any vector \mathbf{x} . With $\mathbf{B} = \mathbf{I}$ in Algorithm 4.21 we obtain $\mathbf{R} = \mathbf{C}\mathbf{A}$ and $\mathbf{C} = \mathbf{H}_s \cdots \mathbf{H}_2\mathbf{H}_1$. Thus $\mathbf{A} = \mathbf{Q}\mathbf{R}$ is a QR decomposition of \mathbf{A} since $\mathbf{Q} := \mathbf{C}^* = \mathbf{H}_1 \cdots \mathbf{H}_s$ is a product of unitary matrices and therefore unitary. \square

Theorem 4.25 (Uniqueness of QR factorization)

If $m \geq n$ and \mathbf{A} is real then the QR factorization is unique if \mathbf{A} has linearly independent columns and \mathbf{R} has positive diagonal elements.

Proof. Let $\mathbf{A} = \mathbf{Q}_1\mathbf{R}_1$ be a QR factorization of \mathbf{A} . Now $\mathbf{A}^T\mathbf{A} = \mathbf{R}_1^T\mathbf{Q}_1^T\mathbf{Q}_1\mathbf{R}_1 = \mathbf{R}_1^T\mathbf{R}_1$. Since $\mathbf{A}^T\mathbf{A}$ is symmetric positive definite the matrix \mathbf{R}_1 is nonsingular, and if its diagonal elements are positive this is the Cholesky factorization of $\mathbf{A}^T\mathbf{A}$. Since the Cholesky factorization is unique it follows that \mathbf{R}_1 is unique and since necessarily $\mathbf{Q}_1 = \mathbf{A}\mathbf{R}_1^{-1}$, it must also be unique. \square

Example 4.26 (QR decomposition and factorization)

Consider finding the QR decomposition and factorization of the matrix $\mathbf{A} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$ using the method of the uniqueness proof of Theorem 4.24. We find $\mathbf{B} := \mathbf{A}^T\mathbf{A} = \begin{bmatrix} 5 & -4 \\ -4 & 5 \end{bmatrix}$. The Cholesky factorization of $\mathbf{B} = \mathbf{R}^T\mathbf{R}$ is given by $\mathbf{R} = \frac{1}{\sqrt{5}} \begin{bmatrix} 5 & -4 \\ 0 & 3 \end{bmatrix}$. Now $\mathbf{R}^{-1} = \frac{1}{3\sqrt{5}} \begin{bmatrix} 3 & 4 \\ 0 & 5 \end{bmatrix}$ so $\mathbf{Q} = \mathbf{A}\mathbf{R}^{-1} = \frac{1}{\sqrt{5}} \begin{bmatrix} 2 & 1 \\ -1 & 2 \end{bmatrix}$. Since \mathbf{A} is square $\mathbf{A} = \mathbf{Q}\mathbf{R}$ is both the QR decomposition and QR factorization of \mathbf{A} .

The QR factorization can be used to prove a classical determinant inequality.

Theorem 4.27 (Hadamard's inequality)

For any $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n] \in \mathbb{C}^{n \times n}$ we have

$$|\det(\mathbf{A})| \leq \prod_{j=1}^n \|\mathbf{a}_j\|_2. \quad (4.15)$$

Equality holds if and only if \mathbf{A} has a zero column or the columns of \mathbf{A} are orthogonal.

Proof. Let $\mathbf{A} = \mathbf{Q}\mathbf{R}$ be a QR factorization of \mathbf{A} . Since

$$1 = \det(\mathbf{I}) = \det(\mathbf{Q}^*\mathbf{Q}) = \det(\mathbf{Q}^*)\det(\mathbf{Q}) = \det(\mathbf{Q})^* \det(\mathbf{Q}) = |\det(\mathbf{Q})|^2$$

we have $|\det(\mathbf{Q})| = 1$. Let $\mathbf{R} = [\mathbf{r}_1, \dots, \mathbf{r}_n]$. Then $(\mathbf{A}^*\mathbf{A})_{jj} = \|\mathbf{a}_j\|_2^2 = (\mathbf{R}^*\mathbf{R})_{jj} = \|\mathbf{r}_j\|_2^2$, and

$$|\det(\mathbf{A})| = |\det(\mathbf{Q}\mathbf{R})| = |\det(\mathbf{R})| = \prod_{j=1}^n |r_{jj}| \leq \prod_{j=1}^n \|\mathbf{r}_j\|_2 = \prod_{j=1}^n \|\mathbf{a}_j\|_2.$$

The inequality is proved. We clearly have equality if \mathbf{A} has a zero column, for then both sides of (4.15) are zero. Suppose the columns are nonzero. We have equality if and only if $r_{jj} = \|\mathbf{r}_j\|_2$ for $j = 1, \dots, n$. This happens if and only if \mathbf{R} is diagonal. But then $\mathbf{A}^* \mathbf{A} = \mathbf{R}^* \mathbf{R}$ is diagonal, which means that the columns of \mathbf{A} are orthogonal. \square

Exercise 4.28 (QR decomposition)

$$\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 0 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{Q} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} 2 & 2 \\ 0 & 2 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

Show that \mathbf{Q} is orthonormal and that \mathbf{QR} is a QR decomposition of \mathbf{A} . Find a QR factorization of \mathbf{A} .

Exercise 4.29 (Householder triangulation)

a) Let

$$\mathbf{A} := \begin{bmatrix} 1 & 0 & 1 \\ -2 & -1 & 0 \\ 2 & 2 & 1 \end{bmatrix}.$$

Find Householder transformations $\mathbf{H}_1, \mathbf{H}_2 \in \mathbb{R}^{3 \times 3}$ such that $\mathbf{H}_2 \mathbf{H}_1 \mathbf{A}$ is upper triangular.

b) Find the QR factorization of \mathbf{A} , when \mathbf{R} has positive diagonal elements.

4.4.2 QR and Gram-Schmidt

The Gram-Schmidt orthogonalization of the columns of \mathbf{A} can be used to find the QR factorization of \mathbf{A} .

Theorem 4.30 (QR and Gram-Schmidt)

Suppose $\mathbf{A} \in \mathbb{R}^{m \times n}$ has rank n and let $\mathbf{v}_1, \dots, \mathbf{v}_n$ be the result of applying Gram Schmidt to the columns $\mathbf{a}_1, \dots, \mathbf{a}_n$ of \mathbf{A} , i.e.,

$$\mathbf{v}_1 = \mathbf{a}_1, \quad \mathbf{v}_j = \mathbf{a}_j - \sum_{i=1}^{j-1} \frac{\mathbf{a}_j^T \mathbf{v}_i}{\mathbf{v}_i^T \mathbf{v}_i} \mathbf{v}_i, \quad \text{for } j = 2, \dots, n. \quad (4.16)$$

Let

$$\mathbf{Q}_1 := [\mathbf{q}_1, \dots, \mathbf{q}_n], \quad \mathbf{q}_j := \frac{\mathbf{v}_j}{\|\mathbf{v}_j\|_2}, \quad j = 1, \dots, n \text{ and}$$

$$\mathbf{R}_1 := \begin{bmatrix} \|\mathbf{v}_1\|_2 & \mathbf{a}_2^T \mathbf{q}_1 & \mathbf{a}_3^T \mathbf{q}_1 & \cdots & \mathbf{a}_{n-1}^T \mathbf{q}_1 & \mathbf{a}_n^T \mathbf{q}_1 \\ 0 & \|\mathbf{v}_2\|_2 & \mathbf{a}_3^T \mathbf{q}_2 & \cdots & \mathbf{a}_{n-1}^T \mathbf{q}_2 & \mathbf{a}_n^T \mathbf{q}_2 \\ 0 & 0 & \|\mathbf{v}_3\|_2 & \cdots & \mathbf{a}_{n-1}^T \mathbf{q}_3 & \mathbf{a}_n^T \mathbf{q}_3 \\ \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \ddots & \ddots & \|\mathbf{v}_{n-1}\|_2 & \mathbf{a}_n^T \mathbf{q}_{n-1} & 0 & \|\mathbf{v}_n\|_2 \end{bmatrix}. \quad (4.17)$$

Then $\mathbf{A} = \mathbf{Q}_1 \mathbf{R}_1$ is the unique QR factorization of \mathbf{A} .

Proof. Let \mathbf{Q}_1 and \mathbf{R}_1 be given by (4.17). The matrix \mathbf{Q}_1 is well defined and has orthonormal columns, since $\{\mathbf{q}_1, \dots, \mathbf{q}_n\}$ is an orthonormal basis for $\text{span}(\mathbf{A})$ by Theorem 4.9. By (4.16)

$$\mathbf{a}_j = \mathbf{v}_j + \sum_{i=1}^{j-1} \frac{\mathbf{a}_j^T \mathbf{v}_i}{\mathbf{v}_i^T \mathbf{v}_i} \mathbf{v}_i = r_{jj} \mathbf{q}_j + \sum_{i=1}^{j-1} \mathbf{q}_i r_{ij} = \mathbf{Q}_1 \mathbf{R}_1 \mathbf{e}_j, \quad j = 1, \dots, n.$$

Clearly \mathbf{R}_1 has positive diagonal elements and the factorization is unique. \square

Example 4.31 (QR using Gram-Schmidt)

Consider finding the QR decomposition and factorization of the matrix $\mathbf{A} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} = [\mathbf{a}_1, \mathbf{a}_2]$ using Gram-Schmidt. Using (4.16) we find $\mathbf{v}_1 = \mathbf{a}_1$ and $\mathbf{v}_2 = \mathbf{a}_2 - \frac{\mathbf{a}_2^T \mathbf{v}_1}{\mathbf{v}_1^T \mathbf{v}_1} \mathbf{v}_1 = \frac{3}{5} \begin{bmatrix} 1 \\ 2 \end{bmatrix}$. Thus $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2]$, where $\mathbf{q}_1 = \frac{1}{\sqrt{5}} \begin{bmatrix} 2 \\ -1 \end{bmatrix}$ and $\mathbf{q}_2 = \frac{1}{\sqrt{5}} \begin{bmatrix} 1 \\ 2 \end{bmatrix}$. By (4.17) we find

$$\mathbf{R}_1 = \mathbf{R} = \begin{bmatrix} \|\mathbf{v}_1\|_2 & \mathbf{a}_2^T \mathbf{q}_1 \\ 0 & \|\mathbf{v}_2\|_2 \end{bmatrix} = \frac{1}{\sqrt{5}} \begin{bmatrix} 5 & -4 \\ 0 & 3 \end{bmatrix}$$

and this agrees with what we found in Example 4.26.

Exercise 4.32 (QR using Gram-Schmidt, II)

Construct \mathbf{Q}_1 and \mathbf{R}_1 in Example 4.23 using Gram-Schmidt orthogonalization.

Warning. The Gram-Schmidt orthogonalization process should not be used to compute the QR factorization numerically. The columns of \mathbf{Q}_1 computed in floating point arithmetic using Gram-Schmidt orthogonalization will often be far from orthogonal. There is a modified version of Gram-Schmidt which behaves better numerically, see [3]. Here we only considered Householder transformations (cf. Algorithm 4.21).

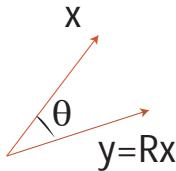


Figure 4.3: A plane rotation.

4.5 Givens Rotations

In some applications, the matrix we want to triangulate has a special structure. Suppose for example that $A \in \mathbb{R}^{n \times n}$ is square and upper Hessenberg as illustrated by a **Wilkinson diagram** for $n = 4$

$$A = \begin{bmatrix} x & x & x & x \\ x & x & x & x \\ 0 & x & x & x \\ 0 & 0 & x & x \end{bmatrix}.$$

Only one element in each column needs to be annihilated and a full Householder transformation will be inefficient. In this case we can use a simpler transformation.

Definition 4.33 (Givens rotation, plane rotation)

A **plane rotation** (also called a **Given's rotation**) is a matrix $P \in \mathbb{R}^{2,2}$ of the form

$$P := \begin{bmatrix} c & s \\ -s & c \end{bmatrix}, \text{ where } c^2 + s^2 = 1.$$

A plane rotation is orthonormal and there is a unique angle $\theta \in [0, 2\pi)$ such that $c = \cos \theta$ and $s = \sin \theta$. Moreover, the identity matrix is a plane rotation corresponding to $\theta = 0$.

Exercise 4.34 (Plane rotation)

Show that if $x = \begin{bmatrix} r \cos \alpha \\ r \sin \alpha \end{bmatrix}$ then $Px = \begin{bmatrix} r \cos(\alpha - \theta) \\ r \sin(\alpha - \theta) \end{bmatrix}$. Thus P rotates a vector x in the plane an angle θ clockwise. See Figure 4.3.

Suppose

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \neq \mathbf{0}, \quad c := \frac{x_1}{r}, \quad s := \frac{x_2}{r}, \quad r := \|x\|_2.$$

Then

$$Px = \frac{1}{r} \begin{bmatrix} x_1 & x_2 \\ -x_2 & x_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} x_1^2 + x_2^2 \\ 0 \end{bmatrix} = \begin{bmatrix} r \\ 0 \end{bmatrix},$$

and we have introduced a zero in \mathbf{x} . We can take $\mathbf{P} = \mathbf{I}$ when $\mathbf{x} = \mathbf{0}$.

For an n -vector $\mathbf{x} \in \mathbb{R}^n$ and $1 \leq i < j \leq n$ we define a **rotation in the i, j -plane** as a matrix $\mathbf{P}_{ij} = (p_{kl}) \in \mathbb{R}^{n \times n}$ by $p_{kl} = \delta_{kl}$ except for positions ii, jj, ij, ji , which are given by

$$\begin{bmatrix} p_{ii} & p_{ij} \\ p_{ji} & p_{jj} \end{bmatrix} = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}, \text{ where } c^2 + s^2 = 1.$$

Thus, for $n = 4$,

$$\mathbf{P}_{1,2} = \begin{bmatrix} c & s & 0 & 0 \\ -s & c & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{P}_{13} = \begin{bmatrix} c & 0 & s & 0 \\ 0 & 1 & 0 & 0 \\ -s & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{P}_{23} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & s & c & 0 \\ 0 & -s & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$



Karl Adolf Hessenberg, 1904-1959 (left), James Wallace Givens, Jr, 1910-1993 (right)

Premultiplying a matrix by a rotation in the i, j -plane changes only rows i and j of the matrix, while post multiplying the matrix by such a rotation only changes column i and j . In particular, if $\mathbf{B} = \mathbf{P}_{ij}\mathbf{A}$ and $\mathbf{C} = \mathbf{A}\mathbf{P}_{ij}$ then $\mathbf{B}(k, :) = \mathbf{A}(k, :)$, $\mathbf{C}(:, k) = \mathbf{A}(:, k)$ for all $k \neq i, j$ and

$$\begin{bmatrix} \mathbf{B}(i, :) \\ \mathbf{B}(j, :) \end{bmatrix} = \begin{bmatrix} c & s \\ -s & c \end{bmatrix} \begin{bmatrix} \mathbf{A}(i, :) \\ \mathbf{A}(j, :) \end{bmatrix}, \quad [\mathbf{C}(:, i) \ \mathbf{C}(:, j)] = [\mathbf{A}(:, i) \ \mathbf{A}(:, j)] \begin{bmatrix} c & s \\ -s & c \end{bmatrix}. \quad (4.18)$$

Givens rotations can be used as an alternative to Householder transformations for solving linear systems. It can be shown that for a dense system of order n the number of arithmetic operations is asymptotically $2n^3$, corresponding to the work of 3 Gaussian eliminations, while, the work using Householder transformations corresponds to 2 Gaussian eliminations. However, for matrices with a special structure Givens rotations can be used to advantage. As an example consider an upper Hessenberg matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$. It can be transformed to upper triangular

form using rotations $P_{i,i+1}$ for $i = 1, \dots, n - 1$. For $n = 4$ the process can be illustrated as follows.

$$\mathbf{A} = \begin{bmatrix} x & x & x & x \\ x & x & x & x \\ 0 & x & x & x \\ 0 & 0 & x & x \end{bmatrix} \xrightarrow{\mathbf{P}_{12}} \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ \mathbf{0} & x & x & x \\ 0 & x & x & x \\ 0 & 0 & x & x \end{bmatrix} \xrightarrow{\mathbf{P}_{23}} \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ 0 & r_{22} & r_{23} & r_{24} \\ 0 & 0 & x & x \\ 0 & 0 & x & x \end{bmatrix} \xrightarrow{\mathbf{P}_{34}} \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ 0 & r_{22} & r_{23} & r_{24} \\ 0 & 0 & r_{33} & r_{34} \\ 0 & 0 & 0 & r_{44} \end{bmatrix}.$$

For an algorithm see Exercise 4.35. This reduction is used in the QR-method discussed in Chapter 14.

Exercise 4.35 (Solving upper Hessenberg system using rotations)

Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be upper Hessenberg and nonsingular, and let $\mathbf{b} \in \mathbb{R}^n$. The following algorithm solves the linear system $\mathbf{Ax} = \mathbf{b}$ using rotations $\mathbf{P}_{k,k+1}$ for $k = 1, \dots, n - 1$. It uses the back solve algorithm 2.7. Determine the number of arithmetic operations of this algorithm.

Algorithm 4.36 (Upper Hessenberg linear system)

```

1 function x=rothesstri(A,b)
2 n=length(A); A=[A b];
3 for k=1:n-1
4     r=norm([A(k,k),A(k+1,k)]);
5     if r>0
6         c=A(k,k)/r; s=A(k+1,k)/r;
7         A([k k+1],k+1:n+1)=[c s;-s c]*A([k k+1],k+1:n+1);
8     end
9     A(k,k)=r; A(k+1,k)=0;
10 end
11 x=rbacksolve(A(:,1:n),A(:,n+1),n);

```

4.6 Review Questions

4.6.1 What is a Householder transformation?

4.6.2 Why are they good for numerical work?

4.6.3 What are the main differences between solving a linear system by Gaussian elimination and Householder transformations?

4.6.4 What are the differences between a QR decomposition and a QR factorization?

4.6.5 Does any matrix have a QR decomposition?

4.6.6 What is a Givens transformation?

Part II

Some matrix theory and least squares

Chapter 5

Eigenpairs and Similarity Transformations

We have seen that a Hermitian matrix is positive definite if and only if it has positive eigenvalues. Eigenvalues and some related quantities called singular values occur in many branches of applied mathematics and are also needed for a deeper study of linear systems. In this and the next chapter we study eigenvalues and singular values. Recall that if $A \in \mathbb{C}^{n \times n}$ is a square matrix, $\lambda \in \mathbb{C}$ and $x \in \mathbb{C}^n$ then (λ, x) is an **eigenpair** for A if $Ax = \lambda x$ and x is nonzero. The scalar λ is called an **eigenvalue** and x is said to be an **eigenvector**. The set of eigenvalues is called the **spectrum** of A and is denoted by $\sigma(A)$. For example, $\sigma(I) = \{1, \dots, 1\} = \{1\}$. The eigenvalues are the roots of the **characteristic polynomial** of A given for $\lambda \in \mathbb{C}$ by

$$\pi_A(\lambda) = \det(A - \lambda I).$$

The equation $\det(A - \lambda I) = 0$ is called the **characteristic equation** of A . Equivalently the characteristic equation can be written $\det(\lambda I - A) = 0$.

5.1 Defective and nondefective matrices

For the eigenvectors we will see that it is important to know if the eigenvectors of a matrix of order n form a basis for \mathbb{C}^n . We say that A is:

- **defective** if the eigenvectors do not form a basis,
- **nondefective** if the eigenvectors form a basis,

We have the following sufficient condition for a matrix to be nondefective.

Theorem 5.1 (Distinct eigenvalues)

A matrix with distinct eigenvalues is nondefective.

Proof. The proof is by contradiction. Suppose \mathbf{A} has linearly dependent eigenpairs $(\lambda_k, \mathbf{x}_k)$, $k = 1, \dots, n$ and let m be the smallest integer such that $\{\mathbf{x}_1, \dots, \mathbf{x}_m\}$ is linearly dependent. Thus $\sum_{j=1}^m c_j \mathbf{x}_j = \mathbf{0}$, where at least one c_j is nonzero. Since the value of the sum is zero there must be at least two c_j 's that are nonzero so we have $m \geq 2$. We find

$$\sum_{j=1}^m c_j \mathbf{x}_j = \mathbf{0} \Rightarrow \sum_{j=1}^m c_j \mathbf{A} \mathbf{x}_j = \sum_{j=1}^m c_j \lambda_j \mathbf{x}_j = \mathbf{0}.$$

From the last relation we subtract $\sum_{j=1}^{m-1} c_j \lambda_m \mathbf{x}_j = \mathbf{0}$ and find $\sum_{j=1}^{m-1} c_j (\lambda_j - \lambda_m) \mathbf{x}_j = \mathbf{0}$. But since $\lambda_j - \lambda_m \neq 0$ for $j = 1, \dots, m-1$ and at least one $c_j \neq 0$ for $j < m$ we see that $\{\mathbf{x}_1, \dots, \mathbf{x}_{m-1}\}$ is linearly dependent, contradicting the minimality of m . \square

If some of the eigenvalues occur with multiplicity higher than one then the matrix can be either defective or nondefective.

Example 5.2 (Defective and nondefective matrices)

Consider the matrices

$$\mathbf{I} := \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{J} := \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}.$$

Since $\mathbf{I}\mathbf{x} = \mathbf{x}$ and $\lambda_1 = \lambda_2 = 1$ any vector $\mathbf{x} \in \mathbb{C}^2$ is an eigenvector for \mathbf{I} . In particular the two unit vectors \mathbf{e}_1 and \mathbf{e}_2 are eigenvectors and form an orthonormal basis for \mathbb{C}^2 . We conclude that the identity matrix is nondefective. The matrix \mathbf{J} also has the eigenvalue one with multiplicity two, but since $\mathbf{J}\mathbf{x} = \mathbf{x}$ if and only if $x_2 = 0$, any eigenvector must be a multiple of \mathbf{e}_1 . Thus \mathbf{J} is defective.

If the eigenvectors $\mathbf{x}_1, \dots, \mathbf{x}_n$ form a basis for \mathbb{C}^n then any $\mathbf{x} \in \mathbb{C}^n$ can be written

$$\mathbf{x} = \sum_{j=1}^n c_j \mathbf{x}_j \text{ for some scalars } c_1, \dots, c_n.$$

We call this an **eigenvector expansion** of \mathbf{x} . By definition any nondefective matrix has an eigenvector expansion.

Example 5.3 (Eigenvector expansion example)

The eigenpairs of $\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$ are $(1, [1, 1]^T)$ and $(3, [1, -1]^T)$. Any $\mathbf{x} = [x_1, x_2]^T \in \mathbb{C}^2$ has the eigenvector expansion

$$\mathbf{x} = \frac{x_1 + x_2}{2} \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \frac{x_1 - x_2}{2} \begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

5.1.1 Similarity transformations

We need a transformation that can be used to simplify a matrix without changing the eigenvalues.

Definition 5.4 (Similar matrices)

*Two matrices $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n \times n}$ are said to be **similar** if there is a nonsingular matrix $\mathbf{S} \in \mathbb{C}^{n \times n}$ such that $\mathbf{B} = \mathbf{S}^{-1}\mathbf{AS}$. The transformation $\mathbf{A} \rightarrow \mathbf{B}$ is called a **similarity transformation**. The columns of \mathbf{S} are denoted by $\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_n$.*

We note that

- Similar matrices have the same eigenvalues, they even have the same characteristic polynomial. Indeed, by the product rule for determinants $\det(\mathbf{AC}) = \det(\mathbf{A})\det(\mathbf{C})$ so that

$$\begin{aligned}\pi_{\mathbf{B}}(\lambda) &= \det(\mathbf{S}^{-1}\mathbf{AS} - \lambda\mathbf{I}) = \det(\mathbf{S}^{-1}(\mathbf{A} - \lambda\mathbf{I})\mathbf{S}) \\ &= \det(\mathbf{S}^{-1})\det(\mathbf{A} - \lambda\mathbf{I})\det(\mathbf{S}) = \det(\mathbf{S}^{-1}\mathbf{S})\det(\mathbf{A} - \lambda\mathbf{I}) = \pi_{\mathbf{A}}(\lambda),\end{aligned}$$

since $\det(\mathbf{I}) = 1$.

- (λ, \mathbf{x}) is an eigenpair for $\mathbf{S}^{-1}\mathbf{AS}$ if and only if (λ, \mathbf{Sx}) is an eigenpair for \mathbf{A} . In fact $(\mathbf{S}^{-1}\mathbf{AS})\mathbf{x} = \lambda\mathbf{x}$ if and only if $\mathbf{A}(\mathbf{Sx}) = \lambda(\mathbf{Sx})$.
- If $\mathbf{S}^{-1}\mathbf{AS} = \mathbf{D} = \text{diag}(\lambda_1, \dots, \lambda_n)$ we can partition $\mathbf{AS} = \mathbf{SD}$ by columns to obtain $[\mathbf{As}_1, \dots, \mathbf{As}_n] = [\lambda_1\mathbf{s}_1, \dots, \lambda_n\mathbf{s}_n]$. Thus the columns of \mathbf{S} are eigenvectors of \mathbf{A} . Moreover, \mathbf{A} is nondefective since \mathbf{S} is nonsingular. Conversely, if \mathbf{A} is nondefective then it can be diagonalized by a similarity transformation $\mathbf{S}^{-1}\mathbf{AS}$, where the columns of \mathbf{S} are eigenvectors of \mathbf{A} .
- For any square matrices $\mathbf{A}, \mathbf{C} \in \mathbb{C}^{n \times n}$ the two products \mathbf{AC} and \mathbf{CA} have the same characteristic polynomial. More generally, for rectangular matrices $\mathbf{A} \in \mathbb{C}^{m \times n}$ and $\mathbf{C} \in \mathbb{C}^{n \times m}$, with say $m > n$, the bigger matrix has $m - n$ extra zero eigenvalues

$$\pi_{\mathbf{AC}}(\lambda) = \lambda^{m-n}\pi_{\mathbf{CA}}(\lambda), \quad \lambda \in \mathbb{C}. \quad (5.1)$$

To show this define for any $m, n \in \mathbb{N}$ block triangular matrices of order $n+m$ by

$$\mathbf{E} := \begin{bmatrix} \mathbf{AC} & \mathbf{0} \\ \mathbf{C} & \mathbf{0} \end{bmatrix}, \quad \mathbf{F} := \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{C} & \mathbf{CA} \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} \mathbf{I} & \mathbf{A} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}.$$

The matrix \mathbf{S} is nonsingular with $\mathbf{S}^{-1} = \begin{bmatrix} \mathbf{I} & -\mathbf{A} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}$. Moreover, $\mathbf{ES} = \mathbf{SF}$ so \mathbf{E} and \mathbf{F} are similar and therefore have the same characteristic polynomials that are the products of the characteristic polynomial of the diagonal blocks. But then $\pi_{\mathbf{E}}(\lambda) = \lambda^n\pi_{\mathbf{AC}}(\lambda) = \pi_{\mathbf{F}}(\lambda) = \lambda^m\pi_{\mathbf{CA}}(\lambda)$. This implies the statements for $m \geq n$.

Exercise 5.5 (Eigenvalues of a block triangular matrix)

What are the eigenvalues of the matrix

$$\begin{bmatrix} 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 3 \end{bmatrix} \in \mathbb{R}^{8,8}? \quad (5.2)$$

Exercise 5.6 (Characteristic polynomial of transpose)

We have $\det(\mathbf{B}^T) = \det(\mathbf{B})$ and $\det(\overline{\mathbf{B}}) = \overline{\det(\mathbf{B})}$ for any square matrix \mathbf{B} . Use this to show that

1. $\pi_{\mathbf{A}^T} = \pi_{\mathbf{A}}$,
2. $\pi_{\mathbf{A}^*}(\bar{\lambda}) = \overline{\pi_{\mathbf{A}}(\lambda)}$.

Exercise 5.7 (Characteristic polynomial of inverse)

Suppose (λ, \mathbf{x}) is an eigenpair for $\mathbf{A} \in \mathbb{C}^{n \times n}$. Show that

1. If \mathbf{A} is nonsingular then $(\lambda^{-1}, \mathbf{x})$ is an eigenpair for \mathbf{A}^{-1} .
2. (λ^k, \mathbf{x}) is an eigenpair for \mathbf{A}^k for $k \in \mathbb{Z}$.

Exercise 5.8 (The power of the eigenvector expansion)

Show that if $\mathbf{A} \in \mathbb{C}^{n \times n}$ is nondefective with eigenpairs $(\lambda_j, \mathbf{x}_j)$, $j = 1, \dots, n$ then for any $\mathbf{x} \in \mathbb{C}^n$ and $k \in \mathbb{N}$

$$\mathbf{A}^k \mathbf{x} = \sum_{j=1}^n c_j \lambda_j^k \mathbf{x}_j \text{ for some scalars } c_1, \dots, c_n. \quad (5.3)$$

Show that if \mathbf{A} is nonsingular then (5.3) holds for all $k \in \mathbb{Z}$.

Exercise 5.9 (Eigenvalues of an idempotent matrix)

Let $\lambda \in \sigma(\mathbf{A})$ where $\mathbf{A}^2 = \mathbf{A} \in \mathbb{C}^{n \times n}$. Show that $\lambda = 0$ or $\lambda = 1$. (A matrix is called **idempotent** if $\mathbf{A}^2 = \mathbf{A}$).

Exercise 5.10 (Eigenvalues of a nilpotent matrix)

Let $\lambda \in \sigma(\mathbf{A})$ where $\mathbf{A}^k = 0$ for some $k \in \mathbb{N}$. Show that $\lambda = 0$. (A matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ such that $\mathbf{A}^k = 0$ for some $k \in \mathbb{N}$ is called **nilpotent**).

Exercise 5.11 (Eigenvalues of a unitary matrix)

Let $\lambda \in \sigma(\mathbf{A})$, where $\mathbf{A}^* \mathbf{A} = \mathbf{I}$. Show that $|\lambda| = 1$.

Exercise 5.12 (Nonsingular approximation of a singular matrix)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is singular. Then we can find $\epsilon_0 > 0$ such that $\mathbf{A} + \epsilon\mathbf{I}$ is nonsingular for all $\epsilon \in \mathbb{C}$ with $|\epsilon| < \epsilon_0$. Hint: $\det(\mathbf{A}) = \lambda_1 \lambda_2 \cdots \lambda_n$, where λ_i are the eigenvalues of \mathbf{A} .

Exercise 5.13 (Companion matrix)

For $q_0, \dots, q_{n-1} \in \mathbb{C}$ let $p(\lambda) = \lambda^n + q_{n-1}\lambda^{n-1} + \cdots + q_0$ be a polynomial of degree n in λ . We derive two matrices that have $(-1)^n p$ as its characteristic polynomial.

- a) Show that $p = (-1)^n \pi_{\mathbf{A}}$ where

$$\mathbf{A} = \begin{bmatrix} -q_{n-1} & -q_{n-2} & \cdots & -q_1 & -q_0 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}.$$

\mathbf{A} is called the **companion matrix** of f .

- b) Show that $p = (-1)^n \pi_{\mathbf{B}}$ where

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & \cdots & 0 & -q_0 \\ 1 & 0 & \cdots & 0 & -q_1 \\ 0 & 1 & \cdots & 0 & -q_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -q_{n-1} \end{bmatrix}.$$

Thus \mathbf{B} can also be regarded as a companion matrix for p .

5.2 Geometric multiplicity of eigenvalues and the Jordan Form

We have seen that a nondefective matrix can be diagonalized by its eigenvectors, while a defective matrix does not enjoy this property. The following question arises. How close to a diagonal matrix can we reduce a general matrix by a similarity transformation? We give one answer to this question, called the Jordan form, or the Jordan canonical form, in Theorem 5.19. For a proof, see for example [15]. The Jordan form is an important tool in matrix analysis and it has applications to systems of differential equations, see [14]. We first take a closer look at the multiplicity of eigenvalues.

5.2.1 Algebraic and geometric multiplicity of eigenvalues

Linear independence of eigenvectors depends on the multiplicity of the eigenvalues in a nontrivial way. For multiple eigenvalues we need to distinguish between two kinds of multiplicities.

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ has k distinct eigenvalues $\lambda_1, \dots, \lambda_k$ with multiplicities a_1, \dots, a_k so that

$$\pi_{\mathbf{A}}(\lambda) := \det(\mathbf{A} - \lambda \mathbf{I}) = (\lambda_1 - \lambda)^{a_1} \cdots (\lambda_k - \lambda)^{a_k}, \quad \lambda_i \neq \lambda_j, \quad i \neq j, \quad \sum_{i=1}^k a_i = n. \quad (5.4)$$

The positive integer $a_i = a(\lambda_i) = a_{\mathbf{A}}(\lambda_i)$ is called the **multiplicity**, or more precisely the **algebraic multiplicity** of the eigenvalue λ_i . The multiplicity of an eigenvalue is simple (double, triple) if a_i is equal to one (two, three).

To define a second kind of multiplicity we consider for each $\lambda \in \sigma(\mathbf{A})$ the nullspace

$$\ker(\mathbf{A} - \lambda \mathbf{I}) := \{\mathbf{x} \in \mathbb{C}^n : (\mathbf{A} - \lambda \mathbf{I})\mathbf{x} = \mathbf{0}\} \quad (5.5)$$

of $\mathbf{A} - \lambda \mathbf{I}$. The nullspace is a subspace of \mathbb{C}^n consisting of all eigenvectors of \mathbf{A} corresponding to the eigenvalue λ . The dimension of the subspace must be at least one since $\mathbf{A} - \lambda \mathbf{I}$ is singular.

Definition 5.14 (Geometric multiplicity)

The **geometric multiplicity** $g = g(\lambda) = g_{\mathbf{A}}(\lambda)$ of an eigenvalue λ of \mathbf{A} is the dimension of the nullspace $\ker(\mathbf{A} - \lambda \mathbf{I})$.

Example 5.15 (Geometric multiplicity)

The $n \times n$ identity matrix \mathbf{I} has the eigenvalue $\lambda = 1$ with $\pi_{\mathbf{I}}(\lambda) = (1 - \lambda)^n$. Since $\mathbf{I} - \lambda \mathbf{I}$ is the zero matrix when $\lambda = 1$, the nullspace of $\mathbf{I} - \lambda \mathbf{I}$ is all of n -space and it follows that $a = g = n$. On the other hand we saw in Example 5.2 that the matrix $\mathbf{J} := \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$ has the eigenvalue $\lambda = 1$ with $a = 2$ and any eigenvector is a multiple of \mathbf{e}_1 . Thus $g = 1$.

Theorem 5.16 (Geometric multiplicity of similar matrices)

Similar matrices have the same eigenvalues with the same algebraic and geometric multiplicities.

Proof. Similar matrices have the same characteristic polynomials and only the invariance of geometric multiplicity needs to be shown. Suppose $\lambda \in \sigma(\mathbf{A})$, $\dim \ker(\mathbf{S}^{-1} \mathbf{A} \mathbf{S} - \lambda \mathbf{I}) = k$, and $\dim \ker(\mathbf{A} - \lambda \mathbf{I}) = \ell$. We need to show that $k = \ell$. Suppose $\mathbf{v}_1, \dots, \mathbf{v}_k$ is a basis for $\ker(\mathbf{S}^{-1} \mathbf{A} \mathbf{S} - \lambda \mathbf{I})$. Then $\mathbf{S}^{-1} \mathbf{A} \mathbf{S} \mathbf{v}_i = \lambda \mathbf{v}_i$

or $\mathbf{ASv}_i = \lambda \mathbf{Sv}_i$, $i = 1, \dots, k$. But then $\{\mathbf{Sv}_1, \dots, \mathbf{Sv}_k\} \subset \ker(\mathbf{A} - \lambda \mathbf{I})$, which implies that $k \leq \ell$. Similarly, if $\mathbf{w}_1, \dots, \mathbf{w}_\ell$ is a basis for $\ker(\mathbf{A} - \lambda \mathbf{I})$ then $\{\mathbf{S}^{-1}\mathbf{w}_1, \dots, \mathbf{S}^{-1}\mathbf{w}_\ell\} \subset \ker(\mathbf{S}^{-1}\mathbf{AS} - \lambda \mathbf{I})$, which implies that $k \geq \ell$. We conclude that $k = \ell$. \square

Exercise 5.17 (Find eigenpair example)

Find eigenvalues and eigenvectors of $\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 2 & 3 \\ 0 & 0 & 2 \end{bmatrix}$. Is \mathbf{A} defective?

5.2.2 The Jordan form



Marie Ennemond Camille Jordan, 1838–1922 (left), William Rowan Hamilton, 1805–1865 (right).

Definition 5.18 (Jordan block)

A **Jordan block** of order m , denoted $\mathbf{J}_m(\lambda)$ is an $m \times m$ matrix of the form

$$\mathbf{J}_m(\lambda) := \begin{bmatrix} \lambda & 1 & 0 & \cdots & 0 & 0 \\ 0 & \lambda & 1 & \cdots & 0 & 0 \\ 0 & 0 & \lambda & \cdots & 0 & 0 \\ \vdots & & \vdots & & \ddots & \\ 0 & 0 & 0 & \cdots & \lambda & i \\ 0 & 0 & 0 & \cdots & 0 & \lambda \end{bmatrix} = \lambda \mathbf{I}_m + \mathbf{E}_m, \quad \mathbf{E}_m := \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & & \vdots & & \ddots & \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}. \quad (5.6)$$

A 3×3 Jordan block has the form $\mathbf{J}_3(\lambda) = \begin{bmatrix} \lambda & 1 & 0 \\ 0 & \lambda & 1 \\ 0 & 0 & \lambda \end{bmatrix}$. Since a Jordan block is upper triangular λ is an eigenvalue of $\mathbf{J}_m(\lambda)$ and any eigenvector must be a multiple of e_1 . Indeed, if $\mathbf{J}_m(\lambda)\mathbf{v} = \lambda\mathbf{v}$ for some $\mathbf{v} = [v_1, \dots, v_m]$ then $v_2 = \dots = v_m = 0$. Thus, the eigenvalue λ of $\mathbf{J}_m(\lambda)$ have algebraic multiplicity $a = m$ and geometric multiplicity $g = 1$.

The Jordan form is a decomposition of a matrix into Jordan blocks.

Theorem 5.19 (The Jordan form of a matrix)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ has k distinct eigenvalues $\lambda_1, \dots, \lambda_k$ of algebraic multiplicities a_1, \dots, a_k and geometric multiplicities g_1, \dots, g_k . There is a nonsingular matrix $\mathbf{S} \in \mathbb{C}^{n \times n}$ such that

$$\mathbf{J} := \mathbf{S}^{-1} \mathbf{A} \mathbf{S} = \text{diag}(\mathbf{U}_1, \dots, \mathbf{U}_k), \text{ with } \mathbf{U}_i \in \mathbb{C}^{a_i \times a_i}, \quad (5.7)$$

where each \mathbf{U}_i is block diagonal having g_i Jordan blocks along the diagonal

$$\mathbf{U}_i = \text{diag}(\mathbf{J}_{m_{i,1}}(\lambda_i), \dots, \mathbf{J}_{m_{i,g_i}}(\lambda_i)). \quad (5.8)$$

Here $m_{i,1}, \dots, m_{i,g_i}$ are integers and they are unique if they are ordered so that $m_{i,1} \geq m_{i,2} \geq \dots \geq m_{i,g_i}$. Moreover, $a_i = \sum_{j=1}^{g_i} m_{i,j}$ for all i .

We note that

1. The matrices \mathbf{S} and \mathbf{J} in (5.7) are called **Jordan factors**. We also call \mathbf{J} the **Jordan form** of \mathbf{A} .
2. The columns of \mathbf{S} are called **principal vectors**. They satisfy the matrix equation $\mathbf{AS} = \mathbf{SJ}$.
3. Each \mathbf{U}_i is upper triangular with the eigenvalue λ_i on the diagonal and consists of g_i Jordan blocks. These Jordan blocks can be taken in any order and it is customary to refer to any such block diagonal matrix as the Jordan form of \mathbf{A} .

Example 5.20 (Jordan form)

As an example consider the Jordan form

$$\mathbf{J} := \text{diag}(\mathbf{U}_1, \mathbf{U}_2) = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 2 & 1 \\ 0 & 0 & 2 \\ & & & 2 & 1 \\ & & & 0 & 2 \\ & & & & 2 \\ & & & & 3 & 1 \\ & & & & 0 & 3 \end{bmatrix} \in \mathbb{R}^{8 \times 8}. \quad (5.9)$$

We encountered this matrix in Exercise 5.5. The eigenvalues together with their algebraic and geometric multiplicities can be read off directly from the Jordan form.

- $\mathbf{U}_1 = \text{diag}(\mathbf{J}_3(2), \mathbf{J}_2(2), \mathbf{J}_1(2))$ and $\mathbf{U}_2 = \mathbf{J}_2(3)$.
- 2 is an eigenvalue of algebraic multiplicity 6 and geometric multiplicity 3, the number of Jordan blocks corresponding to $\lambda = 2$.
- 3 is an eigenvalue of algebraic multiplicity 2 and geometric multiplicity 1.

The columns of $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_8]$ are called **generalized eigenvectors** of \mathbf{A} . They are found from the columns of \mathbf{J} as follows

$$\begin{aligned}\mathbf{As}_1 &= 2\mathbf{s}_1, & \mathbf{As}_2 &= \mathbf{s}_1 + 2\mathbf{s}_2, & \mathbf{As}_3 &= \mathbf{s}_2 + 2\mathbf{s}_3, \\ \mathbf{As}_4 &= 2\mathbf{s}_4, & \mathbf{As}_5 &= \mathbf{s}_4 + 2\mathbf{s}_5, \\ \mathbf{As}_6 &= 2\mathbf{s}_6, \\ \mathbf{As}_7 &= 3\mathbf{s}_7, & \mathbf{As}_8 &= \mathbf{s}_7 + 3\mathbf{s}_8.\end{aligned}$$

We see that the generalized eigenvector corresponding to the first column in a Jordan block is an eigenvector of \mathbf{A} . The remaining generalized eigenvectors are not eigenvectors.

The matrix

$$\mathbf{J} := \begin{bmatrix} 3 & 1 & & \\ 0 & 3 & & \\ & 2 & 1 & \\ & 0 & 2 & \\ & & 2 & \\ & & 2 & 1 & 0 \\ & & 0 & 2 & 1 \\ & & 0 & 0 & 2 \end{bmatrix}$$

is also a Jordan form of \mathbf{A} . In any Jordan form of this \mathbf{A} the sizes of the 4 Jordan blocks $\mathbf{J}_3(2), \mathbf{J}_2(2), \mathbf{J}_1(2), \mathbf{J}_2(3)$ are uniquely given.

The Jordan form implies¹²

Corollary 5.21 (Geometric multiplicity)

We have

1. The geometric multiplicity of an eigenvalue is always bounded above by the algebraic multiplicity of the eigenvalue.
2. The number of linearly independent eigenvectors of a matrix equals the sum of the geometric multiplicities of the eigenvalues.
3. A matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ has n linearly independent eigenvectors if and only if the algebraic and geometric multiplicity of all eigenvalues are the same.

Proof.

1. The algebraic multiplicity a_i of an eigenvalue λ_i is equal to the size of the corresponding \mathbf{U}_i . Moreover each \mathbf{U}_i contains g_i Jordan blocks of size $m_{i,j} \geq 1$. Thus $g_i \leq a_i$.
2. Since \mathbf{A} and \mathbf{J} are similar the geometric multiplicities of the eigenvalues of these matrices are the same, and it is enough to prove statement 2 for the

¹²Corollary 5.21 can also be shown without using the Jordan form, see [16].

Jordan factor \mathbf{J} . We show this only for the matrix \mathbf{J} given by (5.9). The general case should then be clear. There are only 4 eigenvectors of \mathbf{J} , namely $\mathbf{e}_1, \mathbf{e}_4, \mathbf{e}_6, \mathbf{e}_7$ corresponding to the 4 Jordan blocks. These 4 vectors are clearly linearly independent. Moreover there are $k = 2$ distinct eigenvalues and $g_1 + g_2 = 3 + 1 = 4$.

3. Since $g_i \leq a_i$ for all i and $\sum_i a_i = n$ we have $\sum_i g_i = n$ if and only if $a_i = g_i$ for $i = 1, \dots, k$.

□

Exercise 5.22 (Jordan example)

For the Jordan form of the matrix $\mathbf{A} = \begin{bmatrix} 3 & 0 & -1 \\ -4 & 1 & -2 \\ -4 & 0 & -1 \end{bmatrix}$ we have $\mathbf{J} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$. Find \mathbf{S} .

Exercise 5.23 (A nilpotent matrix)

Show that $(\mathbf{J}_m(\lambda) - \lambda \mathbf{I})^r = \begin{bmatrix} \mathbf{0} & \mathbf{I}_{m-r} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$ for $1 \leq r \leq m-1$ and conclude that $(\mathbf{J}_m(\lambda) - \lambda \mathbf{I})^m = \mathbf{0}$.

Exercise 5.24 (Properties of the Jordan form)

Let \mathbf{J} be the Jordan form of a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ as given in Theorem 5.19. Then for $r = 0, 1, 2, \dots, m = 2, 3, \dots$, and any $\lambda \in \mathbb{C}$

1. $\mathbf{A}^r = \mathbf{S} \mathbf{J}^r \mathbf{S}^{-1}$,
2. $\mathbf{J}^r = \text{diag}(\mathbf{U}_1^r, \dots, \mathbf{U}_k^r)$,
3. $\mathbf{U}_i^r = \text{diag}(\mathbf{J}_{m_{i,1}}(\lambda_i)^r, \dots, \mathbf{J}_{m_{i,g_i}}(\lambda_i)^r)$,
4. $\mathbf{J}_m(\lambda)^r = (\mathbf{E}_m + \lambda \mathbf{I}_m)^r = \sum_{k=0}^{\min\{r,m-1\}} \binom{r}{k} \lambda^{r-k} \mathbf{E}_m^k$.

Exercise 5.25 (Powers of a Jordan block)

Find \mathbf{J}^{100} and \mathbf{A}^{100} for the matrix in Exercise 5.22.

Exercise 5.26 (The minimal polynomial)

Let \mathbf{J} be the Jordan form of a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ as given in Theorem 5.19. The polynomial

$$\mu_{\mathbf{A}}(\lambda) := \prod_{i=1}^k (\lambda_i - \lambda)^{m_i} \text{ where } m_i := \max_{1 \leq j \leq g_i} m_{i,j}, \quad (5.10)$$

is called the **minimal polynomial** of \mathbf{A} . We define the matrix polynomial $\mu_{\mathbf{A}}(\mathbf{A})$ by replacing the factors $\lambda_i - \lambda$ by $\lambda_i \mathbf{I} - \mathbf{A}$.

1. We have $\pi_A(\lambda) = \prod_{i=1}^k \prod_{j=1}^{g_i} (\lambda_i - \lambda)^{m_{i,j}}$. Use this to show that the minimal polynomial divides the characteristic polynomial, i.e., $\pi_A = \mu_A \nu_A$ for some polynomial ν_A .
2. Show that $\mu_A(A) = \mathbf{0} \iff \mu_A(J) = \mathbf{0}$.
3. (can be difficult) Use Exercises 5.23, 5.24 and the maximality of m_i to show that $\mu_A(A) = \mathbf{0}$. Thus a matrix satisfies its minimal equation. Finally show that the degree of any polynomial p such that $p(A) = \mathbf{0}$ is at least as large as the degree of the minimal polynomial.
4. Use 2. to show the **Cayley-Hamilton Theorem** which says that a matrix satisfies its characteristic equation $\pi_A(A) = \mathbf{0}$.

Exercise 5.27 (Big Jordan example)

Find the Jordan form of the matrix

$$A = \frac{1}{9} \begin{bmatrix} 10 & 16 & -8 & -5 & 6 & 1 & -3 & 4 \\ -7 & 32 & -7 & -10 & 12 & 2 & -6 & 8 \\ -6 & 12 & 12 & -15 & 18 & 3 & -9 & 12 \\ -5 & 10 & -5 & -2 & 24 & 4 & -12 & 16 \\ -4 & 8 & -4 & -16 & 30 & 14 & -15 & 20 \\ -3 & 6 & -3 & -12 & 9 & 24 & -9 & 24 \\ -2 & 4 & -2 & -8 & 6 & -2 & 15 & 28 \\ -1 & 2 & -1 & -4 & 3 & -1 & -6 & 41 \end{bmatrix}. \quad (5.11)$$

5.3 The Schur decomposition and normal matrices



Issai Schur, 1875-1941 (left), John William Strutt (Lord Rayleigh), 1842-1919 (right) who is a famous English physicist. The Rayleigh quotient (see Section 5.4.1) is named after him.

5.3.1 The Schur decomposition

We turn now to **unitary similarity transformations** $S^{-1}AS$, where $S = U$ is unitary. Thus $S^{-1} = U^*$ and a unitary similarity transformation takes the form U^*AU .

5.3.2 Unitary and orthogonal matrices

Although not every matrix can be diagonalized it can be brought into **triangular form** by a **unitary** similarity transformation.

Theorem 5.28 (Schur decomposition)

*For each $A \in \mathbb{C}^{n \times n}$ there exists a unitary matrix $U \in \mathbb{C}^{n \times n}$ such that $R := U^*AU$ is upper triangular.*

The matrices U and R in the Schur decomposition are called **Schur factors**. We call $A = URU^*$ the **Schur factorization** of A .

Proof. We use induction on n . For $n = 1$ the matrix U is the 1×1 identity matrix. Assume that the theorem is true for matrices of order k and suppose $A \in \mathbb{C}^{n \times n}$, where $n := k + 1$. Let (λ_1, v_1) be an eigenpair for A with $\|v_1\|_2 = 1$. By Theorem 4.10 we can extend v_1 to an orthonormal basis $\{v_1, v_2, \dots, v_n\}$ for \mathbb{C}^n . The matrix $V := [v_1, \dots, v_n] \in \mathbb{C}^{n \times n}$ is unitary, and

$$V^*AVe_1 = V^*Av_1 = \lambda_1 V^*v_1 = \lambda_1 e_1.$$

It follows that

$$V^*AV = \left[\begin{array}{c|c} \lambda_1 & \mathbf{x}^* \\ \mathbf{0} & M \end{array} \right], \text{ for some } M \in \mathbb{C}^{k \times k} \text{ and } \mathbf{x} \in \mathbb{C}^k. \quad (5.12)$$

By the induction hypothesis there is a unitary matrix $W_1 \in \mathbb{C}^{(n-1) \times (n-1)}$ such that $W_1^*MW_1$ is upper triangular. Define

$$W = \left[\begin{array}{c|c} 1 & \mathbf{0}^* \\ \mathbf{0} & W_1 \end{array} \right] \text{ and } U = VW.$$

Then W and U are unitary and

$$\begin{aligned} U^*AU &= W^*(V^*AV)W = \left[\begin{array}{c|c} 1 & \mathbf{0}^* \\ \mathbf{0} & W_1^* \end{array} \right] \left[\begin{array}{c|c} \lambda_1 & \mathbf{x}^* \\ \mathbf{0} & M \end{array} \right] \left[\begin{array}{c|c} 1 & \mathbf{0}^* \\ \mathbf{0} & W_1 \end{array} \right] \\ &= \left[\begin{array}{c|c} \lambda_1 & \mathbf{x}^*W_1 \\ \mathbf{0} & W_1^*MW_1 \end{array} \right] \end{aligned}$$

is upper triangular. \square

If \mathbf{A} has complex eigenvalues then \mathbf{U} will be complex even if \mathbf{A} is real. The following is a real version of Theorem 5.28.

Theorem 5.29 (Schur form, real eigenvalues)

For each $\mathbf{A} \in \mathbb{R}^{n \times n}$ with real eigenvalues there exists an orthogonal matrix $\mathbf{U} \in \mathbb{R}^{n \times n}$ such that $\mathbf{U}^T \mathbf{A} \mathbf{U}$ is upper triangular.

Proof. Consider the proof of Theorem 5.28. Since \mathbf{A} and λ_1 are real the eigenvector \mathbf{v}_1 is real and the matrix \mathbf{W} is real and $\mathbf{W}^T \mathbf{W} = \mathbf{I}$. By the induction hypothesis \mathbf{V} is real and $\mathbf{V}^T \mathbf{V} = \mathbf{I}$. But then also $\mathbf{U} = \mathbf{V} \mathbf{W}$ is real and $\mathbf{U}^T \mathbf{U} = \mathbf{I}$. \square

A real matrix with some complex eigenvalues can only be reduced to block triangular form by a real unitary similarity transformation. We consider this in Section 5.3.4.

Exercise 5.30 (Schur decomposition example)

Show that a Schur decomposition of $\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 3 & 2 \end{bmatrix}$ is $\mathbf{U}^T \mathbf{A} \mathbf{U} = \begin{bmatrix} -1 & -1 \\ 0 & 4 \end{bmatrix}$, where $\mathbf{U} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$.

Exercise 5.31 (Deflation example)

By using the unitary transformation \mathbf{V} on the $n \times n$ matrix \mathbf{A} , we obtain a matrix \mathbf{M} of order $n-1$. \mathbf{M} has the same eigenvalues as \mathbf{A} except λ . Thus we can find another eigenvalue of \mathbf{A} by working with a smaller matrix \mathbf{M} . This is an example of a **deflation** technique which is very useful in numerical work.

The second derivative matrix $\mathbf{T} := \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$ has an eigenpair $(2, \mathbf{x}_1)$, where $\mathbf{x}_1 = [-1, 0, 1]^T$. Find the remaining eigenvalues using deflation. Hint: We can extend \mathbf{x}_1 to a basis $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$ for \mathbb{R}^3 by defining $\mathbf{x}_2 = [0, 1, 0]^T$, $\mathbf{x}_3 = [1, 0, 1]^T$. This is already an orthogonal basis and normalizing we obtain the orthogonal matrix

$$\mathbf{V} = \begin{bmatrix} -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 1 & 0 \\ \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \end{bmatrix}.$$

We obtain (5.12) with $\lambda = 2$ and

$$\mathbf{M} = \begin{bmatrix} 2 & -\sqrt{2} \\ -\sqrt{2} & 2 \end{bmatrix}.$$

We can now find the remaining eigenvalues of \mathbf{A} from the 2×2 matrix \mathbf{M} .

5.3.3 Normal matrices

A matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ is **normal** if $\mathbf{A}^* \mathbf{A} = \mathbf{A} \mathbf{A}^*$.

Examples of normal matrices are

1. $\mathbf{A}^* = \mathbf{A}$, (Hermitian)
2. $\mathbf{A}^* = -\mathbf{A}$, (Skew-Hermitian)
3. $\mathbf{A}^* = \mathbf{A}^{-1}$, (Unitary)
4. $\mathbf{A} = \text{diag}(d_1, \dots, d_n)$. (Diagonal)

If \mathbf{A} is diagonal then

$$\mathbf{A}^* \mathbf{A} = \text{diag}(\overline{d_1}d_1, \dots, \overline{d_n}d_n) = \text{diag}(|d_1|^2, \dots, |d_n|^2) = \mathbf{A} \mathbf{A}^*,$$

and \mathbf{A} is normal. The 2. derivative matrix \mathbf{T} in (1.23) is normal. The eigenvalues of a normal matrix can be complex (cf. Exercise 5.35). However in the Hermitian case the eigenvalues are real (cf. Lemma 1.32).

The following theorem shows that \mathbf{A} has a set of orthonormal eigenvectors if and only if it is normal.

Theorem 5.32 (Spectral theorem for normal matrices)

A matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ is normal if and only if there exists a unitary matrix $\mathbf{U} \in \mathbb{C}^{n \times n}$ such that $\mathbf{U}^ \mathbf{A} \mathbf{U} = \mathbf{D}$ is diagonal. If $\mathbf{D} = \text{diag}(\lambda_1, \dots, \lambda_n)$ and $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_n]$ then $(\lambda_j, \mathbf{u}_j)$, $j = 1, \dots, n$ are orthonormal eigenpairs for \mathbf{A} .*

Proof. If $\mathbf{B} = \mathbf{U}^* \mathbf{A} \mathbf{U}$, with \mathbf{B} diagonal, and $\mathbf{U}^* \mathbf{U} = \mathbf{I}$, then $\mathbf{A} = \mathbf{U} \mathbf{B} \mathbf{U}^*$ and

$$\begin{aligned}\mathbf{A} \mathbf{A}^* &= (\mathbf{U} \mathbf{B} \mathbf{U}^*)(\mathbf{U} \mathbf{B}^* \mathbf{U}^*) = \mathbf{U} \mathbf{B} \mathbf{B}^* \mathbf{U}^* \text{ and} \\ \mathbf{A}^* \mathbf{A} &= (\mathbf{U} \mathbf{B}^* \mathbf{U}^*)(\mathbf{U} \mathbf{B} \mathbf{U}^*) = \mathbf{U} \mathbf{B}^* \mathbf{B} \mathbf{U}^*.\end{aligned}$$

Now $\mathbf{B} \mathbf{B}^* = \mathbf{B}^* \mathbf{B}$ since \mathbf{B} is diagonal, and \mathbf{A} is normal.

Conversely, suppose $\mathbf{A}^* \mathbf{A} = \mathbf{A} \mathbf{A}^*$. By Theorem 5.28 we can find \mathbf{U} with $\mathbf{U}^* \mathbf{U} = \mathbf{I}$ such that $\mathbf{B} := \mathbf{U}^* \mathbf{A} \mathbf{U}$ is upper triangular. Since \mathbf{A} is normal \mathbf{B} is normal. Indeed,

$$\mathbf{B} \mathbf{B}^* = \mathbf{U}^* \mathbf{A} \mathbf{U} \mathbf{U}^* \mathbf{A}^* \mathbf{U} = \mathbf{U}^* \mathbf{A} \mathbf{A}^* \mathbf{U} = \mathbf{U}^* \mathbf{A}^* \mathbf{A} \mathbf{U} = \mathbf{B}^* \mathbf{B}.$$

The proof is complete if we can show that an upper triangular normal matrix \mathbf{B} must be diagonal. The diagonal elements in $\mathbf{E} := \mathbf{B}^* \mathbf{B}$ and $\mathbf{F} := \mathbf{B} \mathbf{B}^*$ are given by

$$e_{ii} = \sum_{k=1}^n \bar{b}_{ki} b_{ki} = \sum_{k=1}^i |b_{ki}|^2 \text{ and } f_{ii} = \sum_{k=1}^n b_{ik} \bar{b}_{ik} = \sum_{k=i}^n |b_{ik}|^2.$$

The result now follows by equating e_{ii} and f_{ii} for $i = 1, 2, \dots, n$. In particular for $i = 1$ we have $|b_{11}|^2 = |b_{11}|^2 + |b_{12}|^2 + \dots + |b_{1n}|^2$, so $b_{1k} = 0$ for $k = 2, 3, \dots, n$. Suppose \mathbf{B} is diagonal in its first $i - 1$ rows so that $b_{jk} = 0$ for $j = 1, \dots, i-1$, $k = j+1, \dots, n$. Then

$$e_{ii} = \sum_{k=1}^i |b_{ki}|^2 = |b_{ii}|^2 = \sum_{k=i}^n |b_{ik}|^2 = f_{ii}$$

and it follows that $b_{ik} = 0$, $k = i+1, \dots, n$. By induction on the rows we see that \mathbf{B} is diagonal. The last part of the theorem follows from Section 5.1.1. \square

Example 5.33 The orthogonal diagonalization of $\mathbf{A} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$ is $\mathbf{U}^T \mathbf{A} \mathbf{U} = \text{diag}(1, 3)$, where $\mathbf{U} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

Exercise 5.34 (Skew-Hermitian matrix)

Suppose $\mathbf{C} = \mathbf{A} + i\mathbf{B}$, where $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}$. Show that \mathbf{C} is skew-Hermitian if and only if $\mathbf{A}^T = -\mathbf{A}$ and $\mathbf{B}^T = \mathbf{B}$.

Exercise 5.35 (Eigenvalues of a skew-Hermitian matrix)

Show that any eigenvalue of a skew-Hermitian matrix is purely imaginary.

Exercise 5.36 (Eigenvector expansion using orthogonal eigenvectors)

Show that if the eigenpairs $(\lambda_1, \mathbf{u}_1), \dots, (\lambda_n, \mathbf{u}_n)$ of $\mathbf{A} \in \mathbb{C}^{n \times n}$ are orthogonal, i.e., $\mathbf{u}_j^* \mathbf{u}_k = 0$ for $j \neq k$ then the eigenvector expansions of $\mathbf{x}, \mathbf{Ax} \in \mathbb{C}^n$ take the form

$$\mathbf{x} = \sum_{j=1}^n c_j \mathbf{u}_j, \quad \mathbf{Ax} = \sum_{j=1}^n c_j \lambda_j \mathbf{u}_j, \quad \text{where } c_j = \frac{\mathbf{u}_j^* \mathbf{x}}{\mathbf{u}_j^* \mathbf{u}_j}. \quad (5.13)$$

5.3.4 The quasi triangular form

How far can we reduce a real matrix \mathbf{A} with some complex eigenvalues by a real unitary similarity transformation? To study this we note that the complex eigenvalues of a real matrix occur in conjugate pairs, $\lambda = \mu + i\nu$, $\bar{\lambda} = \mu - i\nu$, where μ, ν are real. The real 2×2 matrix

$$\mathbf{M} = \begin{bmatrix} \mu & \nu \\ -\nu & \mu \end{bmatrix} \quad (5.14)$$

has eigenvalues $\lambda = \mu + i\nu$ and $\bar{\lambda} = \mu - i\nu$.

Definition 5.37 (Quasi-triangular matrix)

We say that a matrix is **quasi-triangular** if it is block triangular with only 1×1 and 2×2 blocks on the diagonal. Moreover, no 2×2 block should have real eigenvalues.

As an example consider the matrix

$$\mathbf{R} := \begin{bmatrix} \mathbf{D}_1 & \mathbf{R}_{1,2} & \mathbf{R}_{1,3} \\ \mathbf{0} & \mathbf{D}_2 & \mathbf{R}_{2,3} \\ \mathbf{0} & \mathbf{0} & \mathbf{D}_3 \end{bmatrix}, \quad \mathbf{D}_1 := \begin{bmatrix} 2 & 1 \\ -1 & 2 \end{bmatrix}, \quad \mathbf{D}_2 := [1], \quad \mathbf{D}_3 := \begin{bmatrix} 3 & 2 \\ -1 & 1 \end{bmatrix}.$$

Since \mathbf{R} is block triangular $\pi_{\mathbf{R}} = \pi_{\mathbf{D}_1}\pi_{\mathbf{D}_2}\pi_{\mathbf{D}_3}$. We find

$$\pi_{\mathbf{D}_1}(\lambda) = \pi_{\mathbf{D}_3}(\lambda) = \lambda^2 - 4\lambda + 5, \quad \pi_{\mathbf{D}_2}(\lambda) = \lambda - 1,$$

and the eigenvalues \mathbf{D}_1 and \mathbf{D}_3 are $\lambda_1 = 2+i$, $\lambda_2 = 2-i$, while \mathbf{D}_2 obviously has the eigenvalue $\lambda = 1$.

Any $\mathbf{A} \in \mathbb{R}^{n \times n}$ can be reduced to quasi-triangular form by a real orthogonal similarity transformation. For a proof see [31]. We will encounter the quasi triangular form in Chapter 14.

5.4 Hermitian Matrices

The special cases where \mathbf{A} is Hermitian, or real and symmetric, deserve special attention.

Theorem 5.38 (Spectral theorem, complex form)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is Hermitian. Then \mathbf{A} has real eigenvalues $\lambda_1, \dots, \lambda_n$. Moreover, there is a unitary matrix $\mathbf{U} \in \mathbb{C}^{n \times n}$ such that $\mathbf{U}^* \mathbf{A} \mathbf{U} = \text{diag}(\lambda_1, \dots, \lambda_n)$. For any such \mathbf{U} the columns $\{\mathbf{u}_1, \dots, \mathbf{u}_n\}$ of \mathbf{U} are orthonormal eigenvectors of \mathbf{A} and $\mathbf{A} \mathbf{u}_j = \lambda_j \mathbf{u}_j$ for $j = 1, \dots, n$.

Proof. That the eigenvalues are real was shown in Lemma 1.32. The rest follows from Theorem 5.32. \square

There is also a real version.

Theorem 5.39 (Spectral Theorem (real form))

Suppose $\mathbf{A} \in \mathbb{R}^{n \times n}$ is symmetric. Then \mathbf{A} has real eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_n$. There is an orthogonal matrix $\mathbf{U} \in \mathbb{R}^{n \times n}$ such that $\mathbf{U}^T \mathbf{A} \mathbf{U} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$. For any such \mathbf{U} the columns $\{\mathbf{u}_1, \dots, \mathbf{u}_n\}$ of \mathbf{U} are orthonormal eigenvectors of \mathbf{A} and $\mathbf{A} \mathbf{u}_j = \lambda_j \mathbf{u}_j$ for $j = 1, \dots, n$.

Proof. Since a real symmetric matrix has real eigenvalues and eigenvectors this follows from Theorem 5.38. \square

5.4.1 The Rayleigh Quotient

The Rayleigh quotient is an important tool when studying eigenvalues.

Definition 5.40 (Rayleigh quotient)

For $\mathbf{A} \in \mathbb{C}^{n \times n}$ and a nonzero \mathbf{x} the number

$$R(\mathbf{x}) = R_{\mathbf{A}}(\mathbf{x}) := \frac{\mathbf{x}^* \mathbf{A} \mathbf{x}}{\mathbf{x}^* \mathbf{x}}$$

is called a **Rayleigh quotient**.

If (λ, \mathbf{x}) is an eigenpair for \mathbf{A} then $R(\mathbf{x}) = \frac{\mathbf{x}^* \mathbf{A} \mathbf{x}}{\mathbf{x}^* \mathbf{x}} = \lambda$.

Equation (5.15) in the following lemma shows that the Rayleigh quotient of a normal matrix is a **convex combination** of its eigenvalues.

Lemma 5.41 (Convex combination of the eigenvalues)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is normal with orthonormal eigenpairs $(\lambda_j, \mathbf{u}_j)$, $j = 1, 2, \dots, n$. Then the Rayleigh quotient is a convex combination of the eigenvalues of \mathbf{A}

$$R_{\mathbf{A}}(\mathbf{x}) = \frac{\sum_{i=1}^n \lambda_i |c_i|^2}{\sum_{j=1}^n |c_j|^2}, \quad \mathbf{x} \neq \mathbf{0}, \quad \mathbf{x} = \sum_{j=1}^n c_j \mathbf{u}_j. \quad (5.15)$$

Proof. By orthonormality of the eigenvectors $\mathbf{x}^* \mathbf{x} = \sum_{i=1}^n \sum_{j=1}^n \bar{c}_i \bar{u}_i c_j u_j = \sum_{j=1}^n |c_j|^2$. Similarly, $\mathbf{x}^* \mathbf{A} \mathbf{x} = \sum_{i=1}^n \sum_{j=1}^n \bar{c}_i \bar{u}_i c_j \lambda_j u_j = \sum_{i=1}^n \lambda_i |c_i|^2$. and (5.15) follows. This is clearly a combination of nonnegative quantities and a convex combination since $\sum_{i=1}^n |c_i|^2 / \sum_{j=1}^n |c_j|^2 = 1$. \square

5.4.2 Minmax Theorems

There are some useful characterizations of the eigenvalues of a Hermitian matrix. First we show

Theorem 5.42 (Minmax)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is Hermitian with eigenvalues $\lambda_1, \dots, \lambda_n$, ordered so that $\lambda_1 \geq \dots \geq \lambda_n$. Let $1 \leq k \leq n$. For any subspace \mathcal{S} of \mathbb{C}^n of dimension $n - k + 1$

$$\lambda_k \leq \max_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} R(\mathbf{x}), \quad (5.16)$$

with equality for $\mathcal{S} = \tilde{\mathcal{S}} := \text{span}(\mathbf{u}_k, \dots, \mathbf{u}_n)$ and $\mathbf{x} = \mathbf{u}_k$. Here $(\lambda_j, \mathbf{u}_j)$, $1 \leq j \leq n$ are orthonormal eigenpairs for \mathbf{A} .

Proof. Let \mathcal{S} be any subspace of \mathbb{C}^n of dimension $n - k + 1$ and define $\mathcal{S}' := \text{span}(\mathbf{u}_1, \dots, \mathbf{u}_k)$. We need to find $\mathbf{y} \in \mathcal{S}$ so that $R(\mathbf{y}) \geq \lambda_k$. Now $\mathcal{S} + \mathcal{S}' := \{\mathbf{s} + \mathbf{s}' : \mathbf{s} \in \mathcal{S}, \mathbf{s}' \in \mathcal{S}'\}$ is a subspace of \mathbb{C}^n and by (7)

$$\dim(\mathcal{S} \cap \mathcal{S}') = \dim(\mathcal{S}) + \dim(\mathcal{S}') - \dim(\mathcal{S} + \mathcal{S}') \geq (n - k + 1) + k - n = 1.$$

It follows that $\mathcal{S} \cap \mathcal{S}'$ is nonempty. Let $\mathbf{y} \in \mathcal{S} \cap \mathcal{S}' = \sum_{j=1}^k c_j \mathbf{u}_j$ with $\sum_{j=1}^k |c_j|^2 = 1$. Defining $c_j = 0$ for $k+1 \leq j \leq n$, we obtain by Lemma 5.41

$$\max_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} R(\mathbf{x}) \geq R(\mathbf{y}) = \sum_{j=1}^n \lambda_j |c_j|^2 = \sum_{j=1}^k \lambda_j |c_j|^2 \geq \sum_{j=1}^k \lambda_k |c_j|^2 = \lambda_k,$$

and (5.16) follows. If $\mathbf{y} \in \tilde{\mathcal{S}}$, say $\mathbf{y} = \sum_{j=k}^n d_j \mathbf{u}_j$ with $\sum_{j=k}^n |d_j|^2 = 1$ then again by Lemma 5.41 $R(\mathbf{y}) = \sum_{j=k}^n \lambda_j |d_j|^2 \leq \lambda_k$, and since $\mathbf{y} \in \tilde{\mathcal{S}}$ is arbitrary we have $\max_{\substack{\mathbf{x} \in \tilde{\mathcal{S}} \\ \mathbf{x} \neq \mathbf{0}}} R(\mathbf{x}) \leq \lambda_k$ and equality in (5.16) follows for $\mathcal{S} = \tilde{\mathcal{S}}$. Moreover, $R(\mathbf{u}_k) = \lambda_k$. \square

There is also a maxmin version of this result.

Theorem 5.43 (Maxmin)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is Hermitian with eigenvalues $\lambda_1, \dots, \lambda_n$, ordered so that $\lambda_1 \geq \dots \geq \lambda_n$. Let $1 \leq k \leq n$. For any subspace \mathcal{S} of \mathbb{C}^n of dimension k

$$\lambda_k \geq \min_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} R(\mathbf{x}), \quad (5.17)$$

with equality for $\mathcal{S} = \tilde{\mathcal{S}} := \text{span}(\mathbf{u}_1, \dots, \mathbf{u}_k)$ and $\mathbf{x} = \mathbf{u}_k$. Here $(\lambda_j, \mathbf{u}_j)$, $1 \leq j \leq n$ are orthonormal eigenpairs for \mathbf{A} .

Proof. The proof is very similar to the proof of Theorem 5.42. We define $\mathcal{S}' := \text{span}(\mathbf{u}_k, \dots, \mathbf{u}_n)$ and show that $R(\mathbf{y}) \leq \lambda_k$ for some $\mathbf{y} \in \mathcal{S} \cap \mathcal{S}'$. It is easy to see that $R(\mathbf{y}) \geq \lambda_k$ for any $\mathbf{y} \in \tilde{\mathcal{S}}$. \square

These theorems immediately lead to classical minmax and maxmin characterizations.

Corollary 5.44 (The Courant-Fischer Theorem)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is Hermitian with eigenvalues $\lambda_1, \dots, \lambda_n$, ordered so that $\lambda_1 \geq \dots \geq \lambda_n$. Then

$$\lambda_k = \min_{\dim(\mathcal{S})=n-k+1} \max_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} R(\mathbf{x}) = \max_{\dim(\mathcal{S})=k} \min_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} R(\mathbf{x}), \quad k = 1, \dots, n. \quad (5.18)$$

Using Theorem 5.42 we can prove inequalities of eigenvalues without knowing the eigenvectors and we can get both upper and lower bounds.

Theorem 5.45 (Eigenvalue perturbation for Hermitian matrices)

Let $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n \times n}$ be Hermitian with eigenvalues $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_n$ and $\beta_1 \geq \beta_2 \geq \dots \geq \beta_n$. Then

$$\alpha_k + \varepsilon_n \leq \beta_k \leq \alpha_k + \varepsilon_1, \text{ for } k = 1, \dots, n, \quad (5.19)$$

where $\varepsilon_1 \geq \varepsilon_2 \geq \dots \geq \varepsilon_n$ are the eigenvalues of $\mathbf{E} := \mathbf{B} - \mathbf{A}$.

Proof. Since \mathbf{E} is a difference of Hermitian matrices it is Hermitian and the eigenvalues are real. Let (α_j, \mathbf{u}_j) , $j = 1, \dots, n$ be orthonormal eigenpairs for \mathbf{A} and let $\mathcal{S} := \text{span}\{\mathbf{u}_k, \dots, \mathbf{u}_n\}$. By Theorem 5.42 we obtain

$$\beta_k \leq \max_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} R_{\mathbf{B}}(\mathbf{x}) \leq \max_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} R_{\mathbf{A}}(\mathbf{x}) + \max_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} R_{\mathbf{E}}(\mathbf{x}) \leq \max_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} R_{\mathbf{A}}(\mathbf{x}) + \max_{\substack{\mathbf{x} \in \mathbb{C}^n \\ \mathbf{x} \neq \mathbf{0}}} R_{\mathbf{E}}(\mathbf{x}) = \alpha_k + \varepsilon_1,$$

and this proves the upper inequality. For the lower one we define $\mathbf{D} := -\mathbf{E}$ and observe that $-\varepsilon_n$ is the largest eigenvalue of \mathbf{D} . Since $\mathbf{A} = \mathbf{B} + \mathbf{D}$ it follows from the result just proved that $\alpha_k \leq \beta_k - \varepsilon_n$, which is the same as the lower inequality. \square

In many applications of this result the eigenvalues of the matrix \mathbf{E} will be small and then the theorem states that the eigenvalues of \mathbf{B} are close to those of \mathbf{A} . Moreover, it associates a unique eigenvalue of \mathbf{A} with each eigenvalue of \mathbf{B} .

Exercise 5.46 (Eigenvalue perturbation for Hermitian matrices)

Show that in Theorem 5.45, if \mathbf{E} is symmetric positive semidefinite then $\beta_i \geq \alpha_i$.

5.4.3 The Hoffman-Wielandt Theorem

We can also give a bound involving all eigenvalues. The following theorem shows that the eigenvalue problem for a normal matrix is well conditioned.

Theorem 5.47 (Hoffman-Wielandt Theorem)

Suppose $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n \times n}$ are both normal matrices with eigenvalues $\lambda_1, \dots, \lambda_n$ and μ_1, \dots, μ_n , respectively. Then there is a permutation i_1, \dots, i_n of $1, 2, \dots, n$ such that

$$\sum_{j=1}^n |\mu_{i_j} - \lambda_j|^2 \leq \sum_{i=1}^n \sum_{j=1}^n |a_{ij} - b_{ij}|^2. \quad (5.20)$$

For a proof of this theorem see [[30], p. 190]. For a Hermitian matrix we can use the identity permutation if we order both set of eigenvalues in nonincreasing or nondecreasing order.

Exercise 5.48 (Hoffman-Wielandt)

Show that (5.20) does not hold for the matrices $\mathbf{A} := \begin{bmatrix} 0 & 0 \\ 0 & 4 \end{bmatrix}$ and $\mathbf{B} := \begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix}$. Why does this not contradict the Hoffman-Wielandt theorem?

5.5 Left Eigenvectors

Definition 5.49 (Left and right eigenpairs)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is a square matrix, $\lambda \in \mathbb{C}$ and $\mathbf{y} \in \mathbb{C}^n$. We say that (λ, \mathbf{y}) is a **left eigenpair** for \mathbf{A} if $\mathbf{y}^* \mathbf{A} = \lambda \mathbf{y}^*$ or equivalently $\mathbf{A}^* \mathbf{y} = \bar{\lambda} \mathbf{y}$, and \mathbf{y} is nonzero. We say that (λ, \mathbf{y}) is a **right eigenpair** for \mathbf{A} if $\mathbf{A} \mathbf{y} = \lambda \mathbf{y}$ and \mathbf{y} is nonzero. If (λ, \mathbf{y}) is a left eigenpair then λ is called a **left eigenvalue** and \mathbf{y} a **left eigenvector**. Similarly if (λ, \mathbf{y}) is a right eigenpair then λ is called a **right eigenvalue** and \mathbf{y} a **right eigenvector**.

In this book an eigenpair will always mean a right eigenpair. A left eigenvector is an eigenvector of \mathbf{A}^* . If λ is a left eigenvalue of \mathbf{A} then $\bar{\lambda}$ is an eigenvalue of \mathbf{A}^* and then λ is an eigenvalue of \mathbf{A} (cf. Exercise 5.7). Thus left and right eigenvalues are identical, but left and right eigenvectors are in general different. For a Hermitian matrix the right and left eigenpairs are the same.

Using right and left linearly independent eigenpairs we get some useful eigenvector expansions.

Theorem 5.50 (Biorthogonal eigenvector expansion)

If $\mathbf{A} \in \mathbb{C}^{n \times n}$ has linearly independent right eigenvectors $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ then there exists a set of left eigenvectors $\{\mathbf{y}_1, \dots, \mathbf{y}_n\}$ with $\mathbf{y}_i^* \mathbf{x}_j = \delta_{i,j}$. Conversely, if $\mathbf{A} \in \mathbb{C}^{n \times n}$ has linearly independent left eigenvectors $\{\mathbf{y}_1, \dots, \mathbf{y}_n\}$ then there exists a set of right eigenvectors $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ with $\mathbf{y}_i^* \mathbf{x}_j = \delta_{i,j}$. For any scaling of these sets we have the eigenvector expansions

$$\mathbf{v} = \sum_{j=1}^n \frac{\mathbf{y}_j^* \mathbf{v}}{\mathbf{y}_j^* \mathbf{x}_j} \mathbf{x}_j = \sum_{k=1}^n \frac{\mathbf{x}_k^* \mathbf{v}}{\mathbf{y}_k^* \mathbf{x}_k} \mathbf{y}_k. \quad (5.21)$$

Proof. For any right eigenpairs $(\lambda_1, \mathbf{x}_1), \dots, (\lambda_n, \mathbf{x}_n)$ and left eigenpairs $(\lambda_1, \mathbf{y}_1), \dots, (\lambda_n, \mathbf{y}_n)$ of \mathbf{A} we have $\mathbf{AX} = \mathbf{XD}$, $\mathbf{Y}^* \mathbf{A} = \mathbf{DY}^*$, where

$$\mathbf{X} := [\mathbf{x}_1, \dots, \mathbf{x}_n], \quad \mathbf{Y} := [\mathbf{y}_1, \dots, \mathbf{y}_n], \quad \mathbf{D} := \text{diag}(\lambda_1, \dots, \lambda_n).$$

Suppose \mathbf{X} is nonsingular. Then $\mathbf{AX} = \mathbf{XD} \implies \mathbf{A} = \mathbf{XDX}^{-1} \implies \mathbf{X}^{-1} \mathbf{A} = \mathbf{DX}^{-1}$ and it follows that $\mathbf{Y}^* := \mathbf{X}^{-1}$ contains a collection of left eigenvectors such that $\mathbf{Y}^* \mathbf{X} = \mathbf{I}$. Thus the columns of \mathbf{Y} are linearly independent and $\mathbf{y}_i^* \mathbf{x}_j = \delta_{i,j}$. Similarly, if \mathbf{Y} is nonsingular then $\mathbf{AY}^{-*} = \mathbf{Y}^{-*} \mathbf{D}$ and it follows that $\mathbf{X} := \mathbf{Y}^{-*}$ contains a collection of linearly independent right eigenvectors such that $\mathbf{Y}^* \mathbf{X} = \mathbf{I}$. If $\mathbf{v} = \sum_{j=1}^n c_j \mathbf{x}_j$ then $\mathbf{y}_i^* \mathbf{v} = \sum_{j=1}^n c_j \mathbf{y}_i^* \mathbf{x}_j = c_i \mathbf{y}_i^* \mathbf{x}_i$, so $c_i =$

$y_i^*v/y_i^*x_i$ for $i = 1, \dots, n$ and the first expansion in (5.21) follows. The second expansion follows similarly. \square

For a Hermitian matrix the right eigenvectors $\{x_1, \dots, x_n\}$ are also left eigenvectors and (5.21) takes the form

$$v = \sum_{j=1}^n \frac{x_j^* v}{x_j^* x_j} x_j. \quad (5.22)$$

Exercise 5.51 (Biorthogonal expansion)

Determine right and left eigenpairs for the matrix $A := \begin{bmatrix} 3 & 1 \\ 2 & 2 \end{bmatrix}$ and the two expansions in (5.21) for any $v \in \mathbb{R}^2$.

5.5.1 Biorthogonality

Left- and right eigenvectors corresponding to distinct eigenvalues are orthogonal.

Theorem 5.52 (Biorthogonality)

Suppose (μ, y) and (λ, x) are left and right eigenpairs of $A \in \mathbb{C}^{n \times n}$. If $\lambda \neq \mu$ then $y^*x = 0$.

Proof. Using the eigenpair relation in two ways we obtain $y^*Ax = \lambda y^*x = \mu y^*x$ and we conclude that $y^*x = 0$. \square

Right and left eigenvectors corresponding to the same eigenvalue are sometimes orthogonal, sometimes not.

Theorem 5.53 (Simple eigenvalue)

Suppose (λ, x) and (λ, y) are right and left eigenpairs of $A \in \mathbb{C}^{n \times n}$. If λ has algebraic multiplicity one then $y^*x \neq 0$.

Proof. Assume that $\|x\|_2 = 1$. We have (cf. (5.12))

$$V^*AV = \left[\begin{array}{c|c} \lambda & z^* \\ \hline \mathbf{0} & M \end{array} \right],$$

where V is unitary and $Ve_1 = x$. We show that if $y^*x = 0$ then λ is also an eigenvalue of M contradicting the multiplicity assumption of λ . Let $u := V^*y$. Then

$$(V^*A^*V)u = V^*A^*y = \bar{\lambda}V^*y = \bar{\lambda}u,$$

so $(\bar{\lambda}, u)$ is an eigenpair of V^*A^*V . But then $y^*x = u^*V^*Ve_1 = u^*e_1$. Suppose that $u^*e_1 = 0$, i.e., $u = \begin{bmatrix} 0 \\ v \end{bmatrix}$ for some nonzero $v \in \mathbb{C}^{n-1}$. Then

$$V^*A^*Vu = \left[\begin{array}{c|c} \bar{\lambda} & \mathbf{0}^* \\ \hline z & M^* \end{array} \right] \begin{bmatrix} 0 \\ v \end{bmatrix} = \begin{bmatrix} 0 \\ M^*v \end{bmatrix} = \bar{\lambda} \begin{bmatrix} 0 \\ v \end{bmatrix}$$

and λ is an eigenvalue of \mathbf{M} . \square

The case with multiple eigenvalues is more complicated. For example, the matrix $\mathbf{A} := \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$ has one eigenvalue $\lambda = 1$ of algebraic multiplicity two, one right eigenvector $\mathbf{x} = \mathbf{e}_1$ and one left eigenvector $\mathbf{y} = \mathbf{e}_2$. Thus \mathbf{x} and \mathbf{y} are orthogonal.

Exercise 5.54 (Generalized Rayleigh quotient)

For $\mathbf{A} \in \mathbb{C}^{n \times n}$ and any $\mathbf{y}, \mathbf{x} \in \mathbb{C}^n$ with $\mathbf{y}^* \mathbf{x} \neq 0$ the quantity $R(\mathbf{y}, \mathbf{x}) = R_{\mathbf{A}}(\mathbf{y}, \mathbf{x}) := \frac{\mathbf{y}^* \mathbf{A} \mathbf{x}}{\mathbf{y}^* \mathbf{x}}$ is called a **generalized Rayleigh quotient** for \mathbf{A} . Show that if (λ, \mathbf{x}) is a right eigenpair for \mathbf{A} then $R(\mathbf{y}, \mathbf{x}) = \lambda$ for any \mathbf{y} with $\mathbf{y}^* \mathbf{x} \neq 0$. Also show that if (λ, \mathbf{y}) is a left eigenpair for \mathbf{A} then $R(\mathbf{y}, \mathbf{x}) = \lambda$ for any \mathbf{x} with $\mathbf{y}^* \mathbf{x} \neq 0$.

5.6 Review Questions

- 5.6.1 Does \mathbf{A} , \mathbf{A}^T and \mathbf{A}^* have the same eigenvalues? What about $\mathbf{A}^* \mathbf{A}$ and $\mathbf{A} \mathbf{A}^*$?
- 5.6.2 What is the geometric multiplicity of an eigenvalue? Can it be bigger than the algebraic multiplicity?
- 5.6.3 What is the Jordan form of a matrix?
- 5.6.4 What are the eigenvalues of a diagonal matrix?
- 5.6.5 What are the Schur factors of a matrix?
- 5.6.6 What is a quasi-triangular matrix?
- 5.6.7 Give some classes of normal matrices. Why are normal matrices important?
- 5.6.8 State the Courant-Fischer theorem.
- 5.6.9 State the Hoffman-Wielandt theorem for Hermitian matrices.
- 5.6.10 What is a left eigenvector of a matrix?

Chapter 6

The Singular Value Decomposition

The singular value decomposition is useful both for theory and practice. Some of its applications include solving over-determined equations, principal component analysis in statistics, numerical determination of the rank of a matrix, algorithms used in search engines, and the theory of matrices.

We know from Theorem 5.32 that a square matrix \mathbf{A} can be diagonalized by a unitary similarity transformation if and only if it is normal, that is $\mathbf{A}^*\mathbf{A} = \mathbf{A}\mathbf{A}^*$. In particular, if $\mathbf{A} \in \mathbb{C}^{n \times n}$ is normal then it has a set of orthonormal eigenpairs $(\lambda_1, \mathbf{u}_1), \dots, (\lambda_n, \mathbf{u}_n)$. Letting $\mathbf{U} := [\mathbf{u}_1, \dots, \mathbf{u}_n] \in \mathbb{C}^{n \times n}$ and $\mathbf{D} := \text{diag}(\lambda_1, \dots, \lambda_n)$ we have the spectral decomposition

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{U}^*, \text{ where } \mathbf{U}^*\mathbf{U} = \mathbf{I}. \quad (6.1)$$

The singular value decomposition (SVD) is a decomposition of a matrix in the form $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^*$, where \mathbf{U} and \mathbf{V} are unitary, and Σ is a nonnegative diagonal matrix, i.e., $\Sigma_{ij} = 0$ for all $i \neq j$ and $\sigma_i := \Sigma_{ii} \geq 0$ for all i . The diagonal elements σ_i are called **singular values**, while the columns of \mathbf{U} and \mathbf{V} are called **singular vectors**. To be a singular value decomposition the singular values should be ordered, i.e., $\sigma_i \geq \sigma_{i+1}$ for all i .

Example 6.1 (SVD)

The following is a singular value decomposition of a rectangular matrix.

$$\mathbf{A} = \frac{1}{15} \begin{bmatrix} 14 & 2 \\ 4 & 22 \\ 16 & 13 \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 1 & 2 & 2 \\ 2 & -2 & 1 \\ 2 & 1 & -2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \frac{1}{5} \begin{bmatrix} 3 & 4 \\ 4 & -3 \end{bmatrix} = \mathbf{U}\Sigma\mathbf{V}^*. \quad (6.2)$$

Indeed, \mathbf{U} and \mathbf{V} are unitary since the columns (singular vectors) are orthonormal, and Σ is a nonnegative diagonal matrix with singular values $\sigma_1 = 2$ and $\sigma_2 = 1$.

Exercise 6.2 (SVD1)

Show that the decomposition

$$\mathbf{A} := \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \mathbf{U}\mathbf{D}\mathbf{U}^T \quad (6.3)$$

is both a spectral decomposition and a singular value decomposition.

Exercise 6.3 (SVD2)

Show that the decomposition

$$\mathbf{A} := \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} =: \mathbf{U}\Sigma\mathbf{V}^T \quad (6.4)$$

is a singular value decomposition. Show that \mathbf{A} is defective so it cannot be diagonalized by any similarity transformation.

6.1 The SVD always exists

The singular value decomposition is closely related to the eigenpairs of $\mathbf{A}^*\mathbf{A}$ and $\mathbf{A}\mathbf{A}^*$.

6.1.1 The matrices $\mathbf{A}^*\mathbf{A}$, $\mathbf{A}\mathbf{A}^*$

Theorem 6.4 (The matrices $\mathbf{A}^*\mathbf{A}$, $\mathbf{A}\mathbf{A}^*$)

Suppose $m, n \in \mathbb{N}$ and $\mathbf{A} \in \mathbb{C}^{m \times n}$.

1. The matrices $\mathbf{A}^*\mathbf{A} \in \mathbb{C}^{n \times n}$ and $\mathbf{A}\mathbf{A}^* \in \mathbb{C}^{m \times m}$ have the same nonzero eigenvalues with the same algebraic multiplicities. Moreover the extra eigenvalues of the larger matrix are all zero.
2. The matrices $\mathbf{A}^*\mathbf{A} \in \mathbb{C}^{n \times n}$ and $\mathbf{A}\mathbf{A}^* \in \mathbb{C}^{m \times m}$ are Hermitian with nonnegative eigenvalues.
3. Let $(\lambda_j, \mathbf{v}_j)$ be orthonormal eigenpairs for $\mathbf{A}^*\mathbf{A}$ with

$$\lambda_1 \geq \dots \geq \lambda_r > 0 = \lambda_{r+1} = \dots = \lambda_n.$$

Then $\{\mathbf{A}\mathbf{v}_1, \dots, \mathbf{A}\mathbf{v}_r\}$ is an orthogonal basis for the column space $\text{span}(\mathbf{A}) := \{\mathbf{A}\mathbf{y} \in \mathbb{C}^m : \mathbf{y} \in \mathbb{C}^n\}$ and $\{\mathbf{v}_{r+1}, \dots, \mathbf{v}_n\}$ is an orthonormal basis for the nullspace $\ker(\mathbf{A}) := \{\mathbf{y} \in \mathbb{C}^n : \mathbf{A}\mathbf{y} = \mathbf{0}\}$.

4. Let $(\lambda_j, \mathbf{u}_j)$ be orthonormal eigenpairs for $\mathbf{A}\mathbf{A}^*$. If $\lambda_j > 0$, $j = 1, \dots, r$ and $\lambda_j = 0$, $j = r+1, \dots, m$ then $\{\mathbf{A}^*\mathbf{u}_1, \dots, \mathbf{A}^*\mathbf{u}_r\}$ is an orthogonal basis for the column space $\text{span}(\mathbf{A}^*)$ and $\{\mathbf{u}_{r+1}, \dots, \mathbf{u}_m\}$ is an orthonormal basis for the nullspace $\ker(\mathbf{A}^*)$.

5. The rank of \mathbf{A} equals the number of positive eigenvalues of $\mathbf{A}^*\mathbf{A}$ and $\mathbf{A}\mathbf{A}^*$.

Proof.

1. The characteristic polynomials of the two matrices are closely related as stated in (5.1) and shown there.
2. Clearly $\mathbf{A}^*\mathbf{A}$ and $\mathbf{A}\mathbf{A}^*$ are Hermitian. If $\mathbf{A}^*\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$ with $\mathbf{v} \neq \mathbf{0}$, then

$$\lambda = \frac{\mathbf{v}^*\mathbf{A}^*\mathbf{A}\mathbf{v}}{\mathbf{v}^*\mathbf{v}} = \frac{\|\mathbf{A}\mathbf{v}\|_2^2}{\|\mathbf{v}\|_2^2} \geq 0 \quad (6.5)$$

and the eigenvalues of $\mathbf{A}^*\mathbf{A}$ are nonnegative. By part 1 also $\mathbf{A}\mathbf{A}^*$ has nonnegative eigenvalues.

3. By orthonormality of $\mathbf{v}_1, \dots, \mathbf{v}_n$ we have $(\mathbf{A}\mathbf{v}_j)^*\mathbf{A}\mathbf{v}_k = \mathbf{v}_j^*\mathbf{A}^*\mathbf{A}\mathbf{v}_k = \lambda_k \mathbf{v}_j^*\mathbf{v}_k = 0$ for $j \neq k$, showing that $\mathbf{A}\mathbf{v}_1, \dots, \mathbf{A}\mathbf{v}_n$ are orthogonal vectors. Moreover, (6.5) implies that $\mathbf{A}\mathbf{v}_1, \dots, \mathbf{A}\mathbf{v}_r$ are nonzero and $\mathbf{A}\mathbf{v}_j = \mathbf{0}$ for $j = r+1, \dots, n$. In particular, the elements of $\{\mathbf{A}\mathbf{v}_1, \dots, \mathbf{A}\mathbf{v}_r\}$ and $\{\mathbf{v}_{r+1}, \dots, \mathbf{v}_n\}$ are linearly independent vectors in $\text{span}(\mathbf{A})$ and $\ker(\mathbf{A})$, respectively. The proof will be complete once it is shown that $\text{span}(\mathbf{A}) \subset \text{span}(\mathbf{A}\mathbf{v}_1, \dots, \mathbf{A}\mathbf{v}_r)$ and $\ker(\mathbf{A}) \subset \text{span}(\mathbf{v}_{r+1}, \dots, \mathbf{v}_n)$. Suppose $\mathbf{x} \in \text{span}(\mathbf{A})$. Then $\mathbf{x} = \mathbf{A}\mathbf{y}$ for some $\mathbf{y} \in \mathbb{C}^n$. Let $\mathbf{y} = \sum_{j=1}^n c_j \mathbf{v}_j$ be an eigenvector expansion of \mathbf{y} . Since $\mathbf{A}\mathbf{v}_j = \mathbf{0}$ for $j = r+1, \dots, n$ we obtain $\mathbf{x} = \mathbf{A}\mathbf{y} = \sum_{j=1}^n c_j \mathbf{A}\mathbf{v}_j = \sum_{j=1}^r c_j \mathbf{A}\mathbf{v}_j \in \text{span}(\mathbf{A}\mathbf{v}_1, \dots, \mathbf{A}\mathbf{v}_r)$. Finally, if $\mathbf{y} = \sum_{j=1}^n c_j \mathbf{v}_j \in \ker(\mathbf{A})$, then we have $\mathbf{A}\mathbf{y} = \sum_{j=1}^r c_j \mathbf{A}\mathbf{v}_j = \mathbf{0}$, and $c_1 = \dots = c_r = 0$ since $\mathbf{A}\mathbf{v}_1, \dots, \mathbf{A}\mathbf{v}_r$ are linearly independent. But then $\mathbf{y} = \sum_{j=r+1}^n c_j \mathbf{v}_j \in \text{span}(\mathbf{v}_{r+1}, \dots, \mathbf{v}_n)$.
4. Since $\mathbf{A}\mathbf{A}^* = \mathbf{B}^*\mathbf{B}$ with $\mathbf{B} := \mathbf{A}^*$ this follows from part 3 with $\mathbf{A} = \mathbf{B}$.
5. By part 1 and 2 $\mathbf{A}^*\mathbf{A}$ and $\mathbf{A}\mathbf{A}^*$ have the same number r of positive eigenvalues and by part 3 and 4 r is the rank of \mathbf{A} .

□

The following theorem shows, in a constructive way, that any matrix has a singular value decomposition.

Theorem 6.5 (Existence of SVD)

Suppose for $m, n, r \in \mathbb{N}$ that $\mathbf{A} \in \mathbb{C}^{m \times n}$ has rank r , and that $(\lambda_j, \mathbf{v}_j)$ are orthonormal eigenpairs for $\mathbf{A}^*\mathbf{A}$ with $\lambda_1 \geq \dots \geq \lambda_r > 0 = \lambda_{r+1} = \dots = \lambda_n$. Define

1. $\mathbf{V} := [\mathbf{v}_1, \dots, \mathbf{v}_n] \in \mathbb{C}^{n \times n}$,

2. $\Sigma \in \mathbb{R}^{m \times n}$ is a diagonal matrix with diagonal elements $\sigma_j := \sqrt{\lambda_j}$ for $j = 1, \dots, \min(m, n)$,
3. $\mathbf{U} := [\mathbf{u}_1, \dots, \mathbf{u}_m] \in \mathbb{C}^{m \times m}$, where $\mathbf{u}_j = \sigma_j^{-1} \mathbf{A} \mathbf{v}_j$ for $j = 1, \dots, r$ and $\mathbf{u}_{r+1}, \dots, \mathbf{u}_m$ is an extension of $\mathbf{u}_1, \dots, \mathbf{u}_r$ to an orthonormal basis $\mathbf{u}_1, \dots, \mathbf{u}_m$ for \mathbb{C}^m .

Then $\mathbf{A} = \mathbf{U} \Sigma \mathbf{V}^*$ is a singular value decomposition of \mathbf{A} .

Proof. Let $\mathbf{U}, \Sigma, \mathbf{V}$ be as in the theorem. The vectors $\mathbf{u}_1, \dots, \mathbf{u}_r$ are orthonormal since $\mathbf{A} \mathbf{v}_1, \dots, \mathbf{A} \mathbf{v}_r$ are orthogonal and $\sigma_j = \|\mathbf{A} \mathbf{v}_j\|_2 > 0$, $j = 1, \dots, r$ by (6.5). But then \mathbf{U} and \mathbf{V} are unitary and Σ is a nonnegative diagonal matrix. Moreover,

$$\mathbf{U} \Sigma = \mathbf{U} [\sigma_1 \mathbf{e}_1, \dots, \sigma_r \mathbf{e}_r, 0, \dots, 0] = [\sigma_1 \mathbf{u}_1, \dots, \sigma_r \mathbf{u}_r, 0, \dots, 0] = [\mathbf{A} \mathbf{v}_1, \dots, \mathbf{A} \mathbf{v}_n].$$

Thus $\mathbf{U} \Sigma = \mathbf{A} \mathbf{V}$ and since \mathbf{V} is unitary we find $\mathbf{U} \Sigma \mathbf{V}^* = \mathbf{A} \mathbf{V} \mathbf{V}^* = \mathbf{A}$ and we have an SVD of \mathbf{A} with $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$. \square

Example 6.6 (Find SVD)

Derive the SVD in (6.2).

Discussion: We have $\mathbf{A} = \frac{1}{15} \begin{bmatrix} 14 & 2 \\ 4 & 22 \\ 16 & 13 \end{bmatrix}$ and eigenpairs of

$$\mathbf{B} := \mathbf{A}^T \mathbf{A} = \frac{1}{25} \begin{bmatrix} 52 & 36 \\ 36 & 73 \end{bmatrix}$$

are found from

$$\mathbf{B} \begin{bmatrix} 3 \\ 4 \end{bmatrix} = 4 \begin{bmatrix} 3 \\ 4 \end{bmatrix}, \quad \mathbf{B} \begin{bmatrix} 4 \\ -3 \end{bmatrix} = 1 \begin{bmatrix} 4 \\ -3 \end{bmatrix}.$$

Thus $\sigma_1 = 2$, $\sigma_2 = 1$, and $\mathbf{V} = \frac{1}{5} \begin{bmatrix} 3 & 4 \\ 4 & -3 \end{bmatrix}$. Now $\mathbf{u}_1 = \mathbf{A} \mathbf{u}/\sigma_1 = [1, 2, 2]^T/3$,

$\mathbf{u}_2 = \mathbf{A} \mathbf{v}_2/\sigma_2 = [2, -2, 1]^T/3$. For an SVD we also need \mathbf{u}_3 which is any vector of length one orthogonal to \mathbf{u}_1 and \mathbf{u}_2 . $\mathbf{u}_3 = [2, 1, -2]^T/3$ is such a vector and we obtain the singular value decomposition

$$\mathbf{A} = \frac{1}{3} \begin{bmatrix} 1 & 2 & 2 \\ 2 & -2 & 1 \\ 2 & 1 & -2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \frac{1}{5} \begin{bmatrix} 3 & 4 \\ 4 & -3 \end{bmatrix}. \quad (6.6)$$

Exercise 6.7 (SVD examples)

Find the singular value decomposition of the following matrices

$$(a) \mathbf{A} = \begin{bmatrix} 3 \\ 4 \end{bmatrix}.$$

$$(b) \mathbf{A} = \begin{bmatrix} 1 & 1 \\ 2 & 2 \\ 2 & 2 \end{bmatrix}.$$

Exercise 6.8 (More SVD examples)

Find the singular value decomposition of the following matrices

- (a) $\mathbf{A} = \mathbf{e}_1$ the first unit vector in \mathbb{R}^m .
- (b) $\mathbf{A} = \mathbf{e}_n^T$ the last unit vector in \mathbb{R}^n .
- (c) $\mathbf{A} = \begin{bmatrix} -1 & 0 \\ 0 & 3 \end{bmatrix}$.

Exercise 6.9 (Singular values of a normal matrix)

Show that

1. *the singular values of a normal matrix are the absolute values of its eigenvalues,*
2. *the singular values of a symmetric positive semidefinite matrix are its eigenvalues.*

Exercise 6.10 (The matrices $\mathbf{A}^* \mathbf{A}$, $\mathbf{A} \mathbf{A}^*$ and SVD)

Show the following:

If $\mathbf{A} = \mathbf{U} \Sigma \mathbf{V}$ is a singular value decomposition of $\mathbf{A} \in \mathbb{C}^{m \times n}$ then

1. $\mathbf{A}^* \mathbf{A} = \mathbf{V} \text{diag}(\sigma_1^2, \dots, \sigma_n^2) \mathbf{V}^*$ is a spectral decomposition of $\mathbf{A}^* \mathbf{A}$.
2. $\mathbf{A} \mathbf{A}^* = \mathbf{U} \text{diag}(\sigma_1^2, \dots, \sigma_m^2) \mathbf{U}^*$ is a spectral decomposition of $\mathbf{A} \mathbf{A}^*$.
3. *The columns of \mathbf{U} are orthonormal eigenvectors of $\mathbf{A} \mathbf{A}^*$.*
4. *The columns of \mathbf{V} are orthonormal eigenvectors of $\mathbf{A}^* \mathbf{A}$.*

6.2 Further properties of SVD

We first consider a reduced SVD that is often convenient.

6.2.1 The singular value factorization

Suppose $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^*$ is a singular value decomposition of \mathbf{A} of rank r . Consider the block partitions

$$\begin{aligned}\mathbf{U} &= [\mathbf{U}_1, \mathbf{U}_2] \in \mathbb{C}^{m \times m}, \quad \mathbf{U}_1 := [\mathbf{u}_1, \dots, \mathbf{u}_r], \quad \mathbf{U}_2 := [\mathbf{u}_{r+1}, \dots, \mathbf{u}_m], \\ \mathbf{V} &= [\mathbf{V}_1, \mathbf{V}_2] \in \mathbb{C}^{n \times n}, \quad \mathbf{V}_1 := [\mathbf{v}_1, \dots, \mathbf{v}_r], \quad \mathbf{V}_2 := [\mathbf{v}_{r+1}, \dots, \mathbf{v}_n], \\ \Sigma &= \begin{bmatrix} \Sigma_1 & \mathbf{0}_{r,n-r} \\ \mathbf{0}_{m-r,r} & \mathbf{0}_{m-r,n-r} \end{bmatrix} \in \mathbb{R}^{m \times n}, \text{ where } \Sigma_1 := \text{diag}(\sigma_1, \dots, \sigma_r).\end{aligned}\quad (6.7)$$

Thus Σ_1 contains the r positive singular values on the diagonal and for $k, l \geq 0$ the symbol $\mathbf{0}_{k,l} = []$ denotes the empty matrix if $k = 0$ or $l = 0$, and the zero matrix with k rows and l columns otherwise. We obtain by block multiplication a reduced factorization

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^* = \mathbf{U}_1\Sigma_1\mathbf{V}_1^*. \quad (6.8)$$

As an example:

$$\begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \end{bmatrix}.$$

Definition 6.11 (SVF)

Let $m, n, r \in \mathbb{N}$ and suppose $\mathbf{A} \in \mathbb{C}^{m \times n}$ with $1 \leq r \leq \min(m, n)$. A **singular value factorization (SVF)** is a factorization of $\mathbf{A} \in \mathbb{C}^{m \times n}$ of the form $\mathbf{A} = \mathbf{U}_1\Sigma_1\mathbf{V}_1^*$, where $\mathbf{U}_1 \in \mathbb{C}^{m \times r}$ and $\mathbf{V}_1 \in \mathbb{C}^{n \times r}$ have orthonormal columns, and $\Sigma_1 \in \mathbb{R}^{r \times r}$ is a diagonal matrix with $\sigma_1 \geq \dots \geq \sigma_r > 0$.

An SVD and an SVF of a matrix \mathbf{A} are closely related.

1. Let \mathbf{A} have rank r and let $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^*$ be an SVD of \mathbf{A} . Then $\mathbf{A} = \mathbf{U}_1\Sigma_1\mathbf{V}_1^*$ is an SVF of \mathbf{A} . Moreover, $\mathbf{U}_1, \mathbf{V}_1$ contain the first r columns of \mathbf{U}, \mathbf{V} and Σ_1 is a diagonal matrix with the positive singular values on the diagonal.
2. Conversely, suppose $\mathbf{A} = \mathbf{U}_1\Sigma_1\mathbf{V}_1^*$ is a singular value factorization of \mathbf{A} with $\Sigma_1 \in \mathbb{R}^{r \times r}$. Extend \mathbf{U}_1 and \mathbf{V}_1 in any way to unitary matrices $\mathbf{U} \in \mathbb{C}^{m \times m}$ and $\mathbf{V} \in \mathbb{C}^{n \times n}$, and let Σ be given by (6.7). Then $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^*$ is an SVD of \mathbf{A} . Moreover, r is uniquely given as the rank of \mathbf{A} .
3. If $\mathbf{A} = [\mathbf{u}_1, \dots, \mathbf{u}_r] \text{ diag}(\sigma_1, \dots, \sigma_r) [\mathbf{v}_1, \dots, \mathbf{v}_r]^*$ is a singular value factorization of \mathbf{A} then

$$\mathbf{A} = \sum_{j=1}^r \sigma_j \mathbf{u}_j \mathbf{v}_j^*. \quad (6.9)$$

This is known as the **outer product form** of the SVF.

4. We note that a nonsingular square matrix has full rank and only positive singular values. Thus the SVD and SVF are the same for a nonsingular matrix.

Example 6.12 ($r < n < m$)

Find the SVF and SVD of

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 0 \end{bmatrix}.$$

Discussion: Eigenpairs of

$$\mathbf{B} := \mathbf{A}^T \mathbf{A} = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}$$

are derived from

$$\mathbf{B} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = 4 \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \mathbf{B} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = 0 \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

and we find $\sigma_1 = 2$, $\sigma_2 = 0$. Thus $r = 1$, $m = 3$, $n = 2$ and

$$\Sigma = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad \Sigma_1 = [2], \quad \mathbf{V} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

We find $\mathbf{u}_1 = \mathbf{Av}_1/\sigma_1 = \mathbf{s}_1/\sqrt{2}$, where $\mathbf{s}_1 = [1, 1, 0]^T$, and the SVF of \mathbf{A} is given by

$$\mathbf{A} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} [2] \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \end{bmatrix}.$$

To find an SVD we need to extend \mathbf{u}_1 to an orthonormal basis for \mathbb{R}^3 . We first extend \mathbf{s}_1 to a basis $\{\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3\}$ for \mathbb{R}^3 , apply the Gram-Schmidt orthogonalization process to $\{\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3\}$, and then normalize. Choosing the basis

$$\mathbf{s}_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{s}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{s}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix},$$

we find from (4.9)

$$\mathbf{w}_1 = \mathbf{s}_1, \quad \mathbf{w}_2 = \mathbf{s}_2 - \frac{\mathbf{s}_2^T \mathbf{w}_1}{\mathbf{w}_1^T \mathbf{w}_1} \mathbf{w}_1 = \begin{bmatrix} -1/2 \\ 1/2 \\ 0 \end{bmatrix}, \quad \mathbf{w}_3 = \mathbf{s}_3 - \frac{\mathbf{s}_3^T \mathbf{w}_1}{\mathbf{w}_1^T \mathbf{w}_1} \mathbf{w}_1 - \frac{\mathbf{s}_3^T \mathbf{w}_2}{\mathbf{w}_2^T \mathbf{w}_2} \mathbf{w}_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

Normalizing the \mathbf{w}_i 's we obtain $\mathbf{u}_1 = \mathbf{w}_1/\|\mathbf{w}_1\|_2 = [1/\sqrt{2}, 1/\sqrt{2}, 0]^T$, $\mathbf{u}_2 = \mathbf{w}_2/\|\mathbf{w}_2\|_2 = [-1/\sqrt{2}, 1/\sqrt{2}, 0]^T$, and $\mathbf{u}_3 = \mathbf{w}_3/\|\mathbf{w}_3\|_2 = [0, 0, 1]^T$. Therefore, $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T$, where

$$\mathbf{U} := \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} & 0 \\ 1/\sqrt{2} & 1/\sqrt{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{3,3}, \quad \Sigma := \begin{bmatrix} 2 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{3,2}, \quad \mathbf{V} := \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \in \mathbb{R}^{2,2}.$$

The method we used to find the singular value decomposition in the examples and exercises can be suitable for hand calculation with small matrices, but it is not appropriate as a basis for a general purpose numerical method. In particular, the Gram-Schmidt orthogonalization process is not numerically stable, and forming $\mathbf{A}^*\mathbf{A}$ can lead to extra errors in the computation. Standard computer implementations of the singular value decomposition ([31]) first reduces \mathbf{A} to bidiagonal form and then use an adapted version of the QR algorithm where the matrix $\mathbf{A}^*\mathbf{A}$ is not formed. The QR algorithm is discussed in Chapter 14.

Exercise 6.13 (Nonsingular matrix)

Derive the SVF and SVD of the matrix¹³ $\mathbf{A} = \frac{1}{25} \begin{bmatrix} 11 & 48 \\ 48 & 39 \end{bmatrix}$. Also find its spectral decomposition $\mathbf{U}\mathbf{D}\mathbf{U}^T$. The matrix \mathbf{A} is normal, but the spectral decomposition is not an SVD. Why?

Exercise 6.14 (Full row rank)

Find¹⁴ the SVF and SVD of

$$\mathbf{A} := \frac{1}{15} \begin{bmatrix} 14 & 4 & 16 \\ 2 & 22 & 13 \end{bmatrix} \in \mathbb{R}^{2 \times 3}.$$

6.2.2 SVD and the Four Fundamental Subspaces

The singular vectors form orthonormal bases for the four fundamental subspaces $\text{span}(\mathbf{A})$, $\ker(\mathbf{A})$, $\text{span}(\mathbf{A}^*)$, and $\ker(\mathbf{A}^*)$.

Theorem 6.15 (Singular vectors and orthonormal bases)

For positive integers m, n let $\mathbf{A} \in \mathbb{C}^{m \times n}$ have rank r and a singular value decomposition $\mathbf{A} = [\mathbf{u}_1, \dots, \mathbf{u}_m]\Sigma[\mathbf{v}_1, \dots, \mathbf{v}_n]^* = \mathbf{U}\Sigma\mathbf{V}^*$. Then the singular vectors satisfy

$$\begin{aligned} \mathbf{A}\mathbf{v}_i &= \sigma_i \mathbf{u}_i, \quad i = 1, \dots, r, & \mathbf{A}\mathbf{v}_i &= 0, \quad i = r + 1, \dots, n, \\ \mathbf{A}^*\mathbf{u}_i &= \sigma_i \mathbf{v}_i, \quad i = 1, \dots, r, & \mathbf{A}^*\mathbf{u}_i &= 0, \quad i = r + 1, \dots, m. \end{aligned} \tag{6.10}$$

Moreover,

1. $\{\mathbf{u}_1, \dots, \mathbf{u}_r\}$ is an orthonormal basis for $\text{span}(\mathbf{A})$,
2. $\{\mathbf{u}_{r+1}, \dots, \mathbf{u}_m\}$ is an orthonormal basis for $\ker(\mathbf{A}^*)$,
3. $\{\mathbf{v}_1, \dots, \mathbf{v}_r\}$ is an orthonormal basis for $\text{span}(\mathbf{A}^*)$,
4. $\{\mathbf{v}_{r+1}, \dots, \mathbf{v}_n\}$ is an orthonormal basis for $\ker(\mathbf{A})$.

¹³Answer: $\mathbf{A} = \frac{1}{5} \begin{bmatrix} 3 & -4 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix} \frac{1}{5} \begin{bmatrix} 3 & 4 \\ 4 & -3 \end{bmatrix}$.

¹⁴Hint: Take the transpose of the matrix in (6.2)

Proof. If $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^*$ then $\mathbf{AV} = \mathbf{U}\Sigma$, or in terms of the block partition (6.7) $\mathbf{A}[\mathbf{V}_1, \mathbf{V}_2] = [\mathbf{U}_1, \mathbf{U}_2] \begin{bmatrix} \Sigma_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$. But then $\mathbf{AV}_1 = \mathbf{U}_1\Sigma_1$, $\mathbf{AV}_2 = \mathbf{0}$, and this implies the first part of (6.10). Taking conjugate transpose of $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^*$ gives $\mathbf{A}^* = \mathbf{V}\Sigma^T\mathbf{U}^*$ or $\mathbf{A}^*\mathbf{U} = \mathbf{V}\Sigma^T$. Using the block partition as before we obtain the last part of (6.10).

It follows from Theorem 6.4 that $\{\mathbf{Av}_1, \dots, \mathbf{Av}_r\}$ is an orthogonal basis for $\text{span}(\mathbf{A})$, $\{\mathbf{A}^*\mathbf{u}_1, \dots, \mathbf{A}^*\mathbf{u}_r\}$ is an orthogonal basis for $\text{span}(\mathbf{A}^*)$, $\{\mathbf{v}_{r+1}, \dots, \mathbf{v}_m\}$ is an orthonormal basis for $\ker(\mathbf{A})$ and $\{\mathbf{u}_{r+1}, \dots, \mathbf{u}_m\}$ is an orthonormal basis for $\ker(\mathbf{A}^*)$. By (6.10) $\{\mathbf{u}_1, \dots, \mathbf{u}_r\}$ is an orthonormal basis for $\text{span}(\mathbf{A})$ and $\{\mathbf{v}_1, \dots, \mathbf{v}_r\}$ is an orthonormal basis for $\text{span}(\mathbf{A}^*)$. \square

Exercise 6.16 (Counting dimensions of fundamental subspaces)

Suppose $\mathbf{A} \in \mathbb{C}^{m \times n}$. Show using SVD that

1. $\text{rank}(\mathbf{A}) = \text{rank}(\mathbf{A}^*)$.
2. $\text{rank}(\mathbf{A}) + \text{null}(\mathbf{A}) = n$,
3. $\text{rank}(\mathbf{A}) + \text{null}(\mathbf{A}^*) = m$,

where $\text{null}(\mathbf{A})$ is defined as the dimension of $\ker(\mathbf{A})$.

Exercise 6.17 (Rank and nullity relations)

Use Theorem 6.4 to show that for any $\mathbf{A} \in \mathbb{C}^{m \times n}$

1. $\text{rank } \mathbf{A} = \text{rank}(\mathbf{A}^*\mathbf{A}) = \text{rank}(\mathbf{A}\mathbf{A}^*)$,
2. $\text{null}(\mathbf{A}^*\mathbf{A}) = \text{null } \mathbf{A}$, and $\text{null}(\mathbf{A}\mathbf{A}^*) = \text{null}(\mathbf{A}^*)$.

Exercise 6.18 (Orthonormal bases example)

Let \mathbf{A} and \mathbf{B} be as in Example 6.6. Give orthonormal bases for $\text{span}(\mathbf{B})$ and $\ker(\mathbf{B})$.

Exercise 6.19 (Some spanning sets)

Show for any $\mathbf{A} \in \mathbb{C}^{m \times n}$ that $\text{span}(\mathbf{A}^*\mathbf{A}) = \text{span}(\mathbf{V}_1) = \text{span}(\mathbf{A}^*)$

Exercise 6.20 (Singular values and eigenpair of composite matrix)

Let $\mathbf{A} \in \mathbb{C}^{m \times n}$ with $m \geq n$ have singular values $\sigma_1, \dots, \sigma_n$, left singular vectors $\mathbf{u}_1, \dots, \mathbf{u}_m \in \mathbb{C}^m$, and right singular vectors $\mathbf{v}_1, \dots, \mathbf{v}_n \in \mathbb{C}^n$. Show that the matrix

$$\mathbf{C} := \begin{bmatrix} \mathbf{0} & \mathbf{A} \\ \mathbf{A}^* & \mathbf{0} \end{bmatrix}$$

has the $n + m$ eigenpairs

$$\{(\sigma_1, \mathbf{p}_1), \dots, (\sigma_n, \mathbf{p}_n), (-\sigma_1, \mathbf{q}_1), \dots, (-\sigma_n, \mathbf{q}_n), (0, \mathbf{r}_{n+1}), \dots, (0, \mathbf{r}_m)\},$$

where

$$\mathbf{p}_i = \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_i \end{bmatrix}, \quad \mathbf{q}_i = \begin{bmatrix} \mathbf{u}_i \\ -\mathbf{v}_i \end{bmatrix}, \quad \mathbf{r}_j = \begin{bmatrix} \mathbf{u}_j \\ \mathbf{0} \end{bmatrix}, \text{ for } i = 1, \dots, n \text{ and } j = n + 1, \dots, m.$$

6.2.3 A Geometric Interpretation

The singular value decomposition and factorization give insight into the geometry of a linear transformation. Consider the linear transformation $\mathbf{T} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ given by $\mathbf{Tz} := \mathbf{Az}$ where $\mathbf{A} \in \mathbb{R}^{m \times n}$. Assume that $\text{rank}(\mathbf{A}) = n$. The function \mathbf{T} maps the unit sphere $\mathcal{S} := \{\mathbf{z} \in \mathbb{R}^n : \|\mathbf{z}\|_2 = 1\}$ onto an ellipsoid $\mathcal{E} := \mathbf{AS} = \{\mathbf{Az} : \mathbf{z} \in \mathcal{S}\}$ in \mathbb{R}^m .

Theorem 6.21 (SVF ellipse)

Suppose $\mathbf{A} \in \mathbb{R}^{m,n}$ has rank $r = n$, and let $\mathbf{A} = \mathbf{U}_1 \Sigma_1 \mathbf{V}_1^T$ be a singular value factorization of \mathbf{A} . Then

$$\mathcal{E} = \mathbf{U}_1 \tilde{\mathcal{E}} \text{ where } \tilde{\mathcal{E}} := \{\mathbf{y} = [y_1, \dots, y_n]^T \in \mathbb{R}^n : \frac{y_1^2}{\sigma_1^2} + \dots + \frac{y_n^2}{\sigma_n^2} = 1\}.$$

Proof. Suppose $\mathbf{z} \in \mathcal{S}$. Now $\mathbf{Az} = \mathbf{U}_1 \Sigma_1 \mathbf{V}_1^T \mathbf{z} = \mathbf{U}_1 \mathbf{y}$, where $\mathbf{y} := \Sigma_1 \mathbf{V}_1^T \mathbf{z}$. Since $\text{rank}(\mathbf{A}) = n$ it follows that $\mathbf{V}_1 = \mathbf{V}$ is square so that $\mathbf{V}_1 \mathbf{V}_1^T = \mathbf{I}$. But then $\mathbf{V}_1 \Sigma_1^{-1} \mathbf{y} = \mathbf{z}$ and we obtain

$$1 = \|\mathbf{z}\|_2^2 = \|\mathbf{V}_1 \Sigma_1^{-1} \mathbf{y}\|_2^2 = \|\Sigma_1^{-1} \mathbf{y}\|_2^2 = \frac{y_1^2}{\sigma_1^2} + \dots + \frac{y_n^2}{\sigma_n^2}.$$

This implies that $\mathbf{y} \in \tilde{\mathcal{E}}$. Finally, $\mathbf{x} = \mathbf{Az} = \mathbf{U}_1 \Sigma_1 \mathbf{V}_1^T \mathbf{z} = \mathbf{U}_1 \mathbf{y}$, where $\mathbf{y} \in \tilde{\mathcal{E}}$ implies that $\mathcal{E} = \mathbf{U}_1 \tilde{\mathcal{E}}$. \square

The equation $1 = \frac{y_1^2}{\sigma_1^2} + \dots + \frac{y_n^2}{\sigma_n^2}$ describes an ellipsoid in \mathbb{R}^n with semiaxes of length σ_j along the unit vectors \mathbf{e}_j for $j = 1, \dots, n$. Since the orthonormal transformation $\mathbf{U}_1 \mathbf{y} \rightarrow \mathbf{x}$ preserves length, the image $\mathcal{E} = \mathbf{AS}$ is a rotated ellipsoid with semiaxes along the left singular vectors $\mathbf{u}_j = \mathbf{U} \mathbf{e}_j$, of length σ_j , $j = 1, \dots, n$. Since $\mathbf{Av}_j = \sigma_j \mathbf{u}_j$, for $j = 1, \dots, n$ the right singular vectors defines points in \mathcal{S} that are mapped onto the semiaxes of \mathcal{E} .

Example 6.22 (Ellipse)

Consider the transformation $\mathbf{A} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ given by the matrix

$$\mathbf{A} := \frac{1}{25} \begin{bmatrix} 11 & 48 \\ 48 & 39 \end{bmatrix}$$

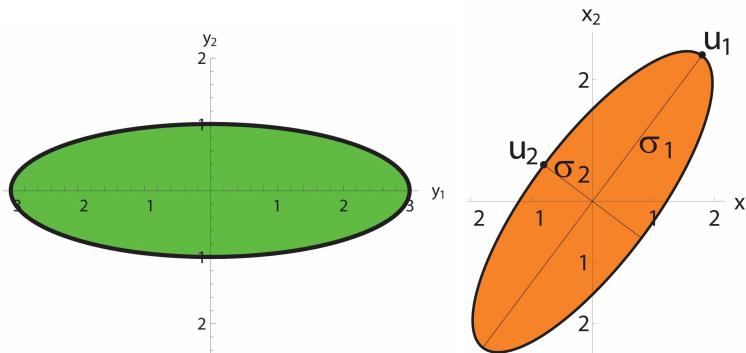


Figure 6.1: The ellipse $y_1^2/9 + y_2^2 = 1$ (left) and the rotated ellipse \mathbf{AS} (right).

in Example 6.13. Recall that $\sigma_1 = 3$, $\sigma_2 = 1$, $\mathbf{u}_1 = [3, 4]^T/5$ and $\mathbf{u}_2 = [-4, 3]^T/5$. The ellipses $y_1^2/\sigma_1^2 + y_2^2/\sigma_2^2 = 1$ and $\mathcal{E} = \mathbf{AS} = \mathbf{U}_1 \tilde{\mathcal{E}}$ are shown in Figure 6.1. Since $\mathbf{y} = \mathbf{U}_1^T \mathbf{x} = [3/5x_1 + 4/5x_2, -4/5x_1 + 3/5x_2]^T$, the equation for the ellipse on the right is

$$\frac{(\frac{3}{5}x_1 + \frac{4}{5}x_2)^2}{9} + \frac{(-\frac{4}{5}x_1 + \frac{3}{5}x_2)^2}{1} = 1,$$

6.3 Determining the Rank of a Matrix Numerically

In many elementary linear algebra courses a version of Gaussian elimination, called Gauss-Jordan elimination, is used to determine the rank of a matrix. To carry this out by hand for a large matrix can be a Herculean task and using a computer and floating point arithmetic the result will not be reliable. Entries, which in the final result should have been zero, will have nonzero values because of round-off errors. As an alternative we can use the singular value decomposition to determine rank. Although success is not at all guaranteed, the result will be more reliable than if Gauss-Jordan elimination is used.

By Theorem 6.5 the rank of a matrix is equal to the number of nonzero singular values and if we have computed the singular values, then all we have to do is to count the nonzero ones. The problem however is the same as for Gaussian elimination. Due to round-off errors none of the computed singular values are likely to be zero.

6.3.1 The Frobenius norm

This commonly occurring matrix norm will be used here in a discussion of how many of the computed singular values can possibly be considered to be zero. The

Frobenius norm, of a matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$ is defined by

$$\|\mathbf{A}\|_F := \left(\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2 \right)^{1/2}. \quad (6.12)$$

There is a relation between the Frobenius norm of a matrix and its singular values. First we derive some elementary properties of this norm. A systematic study of matrix norms is given in the next chapter.

Lemma 6.23 (Frobenius norm properties)

For any $m, n \in \mathbb{N}$ and any matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$

1. $\|\mathbf{A}^*\|_F = \|\mathbf{A}\|_F$,
2. $\|\mathbf{A}\|_F^2 = \sum_{j=1}^n \|\mathbf{a}_{:j}\|_2^2$,
3. $\|\mathbf{U}\mathbf{A}\|_F = \|\mathbf{A}\mathbf{V}\|_F = \|\mathbf{A}\|_F$ for any unitary matrices $\mathbf{U} \in \mathbb{C}^{m \times m}$ and $\mathbf{V} \in \mathbb{C}^{n \times n}$,
4. $\|\mathbf{AB}\|_F \leq \|\mathbf{A}\|_F \|\mathbf{B}\|_F$ for any $\mathbf{B} \in \mathbb{C}^{n, k}$, $k \in \mathbb{N}$,
5. $\|\mathbf{Ax}\|_2 \leq \|\mathbf{A}\|_F \|\mathbf{x}\|_2$, for all $\mathbf{x} \in \mathbb{C}^n$.

Proof.

1. $\|\mathbf{A}^*\|_F^2 = \sum_{j=1}^n \sum_{i=1}^m |\bar{a}_{ij}|^2 = \sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2 = \|\mathbf{A}\|_F^2$.
2. This follows since the Frobenius norm is the Euclidian norm of a vector, $\|\mathbf{A}\|_F := \|\text{vec}(\mathbf{A})\|_2$, where $\text{vec}(\mathbf{A}) \in \mathbb{C}^{mn}$ is the vector obtained by stacking the columns of \mathbf{A} on top of each other.
3. Recall that if $\mathbf{U}^* \mathbf{U} = I$ then $\|\mathbf{Ux}\|_2 = \|\mathbf{x}\|_2$ for all $\mathbf{x} \in \mathbb{C}^n$. Applying this to each column $\mathbf{a}_{:j}$ of \mathbf{A} we find $\|\mathbf{U}\mathbf{A}\|_F^2 \stackrel{2}{=} \sum_{j=1}^n \|\mathbf{U}\mathbf{a}_{:j}\|_2^2 = \sum_{j=1}^n \|\mathbf{a}_{:j}\|_2^2 \stackrel{2}{=} \|\mathbf{A}\|_F^2$. Similarly, since $\mathbf{VV}^* = I$ we find $\|\mathbf{AV}\|_F \stackrel{1}{=} \|\mathbf{V}^* \mathbf{A}^*\|_F = \|\mathbf{A}^*\|_F \stackrel{1}{=} \|\mathbf{A}\|_F$.
4. Using the Cauchy-Schwarz inequality and 2. we obtain

$$\|\mathbf{AB}\|_F^2 = \sum_{i=1}^m \sum_{j=1}^k |\mathbf{a}_{i:}^T \mathbf{b}_{:j}|^2 \leq \sum_{i=1}^m \sum_{j=1}^k \|\mathbf{a}_{i:}\|_2^2 \|\mathbf{b}_{:j}\|_2^2 = \|\mathbf{A}\|_F^2 \|\mathbf{B}\|_F^2.$$

5. Since $\|\mathbf{v}\|_F = \|\mathbf{v}\|_2$ for a vector this follows by taking $k = 1$ and $\mathbf{B} = \mathbf{x}$ in 4.

□

Theorem 6.24 (Frobenius norm and singular values)

We have $\|\mathbf{A}\|_F = \sqrt{\sigma_1^2 + \dots + \sigma_n^2}$, where $\sigma_1, \dots, \sigma_n$ are the singular values of \mathbf{A} .

Proof. Using Lemma 6.23 we find $\|\mathbf{A}\|_F \stackrel{3.}{=} \|\mathbf{U}^* \mathbf{A} \mathbf{V}\|_F = \|\Sigma\|_F = \sqrt{\sigma_1^2 + \dots + \sigma_n^2}$.
□

6.3.2 Low rank approximation

Suppose $m \geq n \geq 1$ and $\mathbf{A} \in \mathbb{C}^{m \times n}$ has a singular value decomposition $\mathbf{A} = \mathbf{U} \begin{bmatrix} \mathbf{D} \\ \mathbf{0} \end{bmatrix} \mathbf{V}^*$, where $\mathbf{D} = \text{diag}(\sigma_1, \dots, \sigma_n)$. We choose $\epsilon > 0$ and let $1 \leq r \leq n$ be the smallest integer such that $\sigma_{r+1}^2 + \dots + \sigma_n^2 < \epsilon^2$. Define $\mathbf{A}' := \mathbf{U} \begin{bmatrix} \mathbf{D}' \\ \mathbf{0} \end{bmatrix} \mathbf{V}^*$, where $\mathbf{D}' := \text{diag}(\sigma_1, \dots, \sigma_r, 0, \dots, 0) \in \mathbb{R}^{n \times n}$. By Lemma 6.23

$$\|\mathbf{A} - \mathbf{A}'\|_F = \|\mathbf{U} \begin{bmatrix} \mathbf{D} - \mathbf{D}' \\ \mathbf{0} \end{bmatrix} \mathbf{V}^*\|_F = \| \begin{bmatrix} \mathbf{D} - \mathbf{D}' \\ \mathbf{0} \end{bmatrix} \|_F = \sqrt{\sigma_{r+1}^2 + \dots + \sigma_n^2} < \epsilon.$$

Thus, if ϵ is small then \mathbf{A} is near a matrix \mathbf{A}' of rank r . This can be used to determine rank numerically. We choose an r such that $\sqrt{\sigma_{r+1}^2 + \dots + \sigma_n^2}$ is “small”. Then we postulate that $\text{rank}(\mathbf{A}) = r$ since \mathbf{A} is close to a matrix of rank r .

The following theorem shows that of all $m \times n$ matrices of rank r , \mathbf{A}' is closest to \mathbf{A} measured in the Frobenius norm.

Theorem 6.25 (Best low rank approximation)

Suppose $\mathbf{A} \in \mathbb{R}^{m \times n}$ has singular values $\sigma_1 \geq \dots \geq \sigma_n \geq 0$. For any $r \leq \text{rank}(\mathbf{A})$ we have

$$\|\mathbf{A} - \mathbf{A}'\|_F = \min_{\substack{\mathbf{B} \in \mathbb{R}^{m \times n} \\ \text{rank}(\mathbf{B})=r}} \|\mathbf{A} - \mathbf{B}\|_F = \sqrt{\sigma_{r+1}^2 + \dots + \sigma_n^2}.$$

For the proof of this theorem we refer to p. 322 of [31].

Exercise 6.26 (Rank example)

Consider the singular value decomposition

$$\mathbf{A} := \begin{bmatrix} 0 & 3 & 3 \\ 4 & 1 & -1 \\ 4 & 1 & -1 \\ 0 & 3 & 3 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} 6 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{2}{3} & \frac{2}{3} & \frac{1}{3} \\ \frac{2}{3} & -\frac{1}{3} & -\frac{2}{3} \\ \frac{1}{3} & -\frac{2}{3} & \frac{2}{3} \end{bmatrix}$$

- (a) Give orthonormal bases for $\text{span}(\mathbf{A})$, $\text{span}(\mathbf{A}^T)$, $\ker(\mathbf{A})$, $\ker(\mathbf{A}^T)$ and $\text{span}(\mathbf{A})^\perp$.

- (b) Explain why for all matrices $\mathbf{B} \in \mathbb{R}^{4,3}$ of rank one we have $\|\mathbf{A} - \mathbf{B}\|_F \geq 6$.
- (c) Give a matrix \mathbf{A}_1 of rank one such that $\|\mathbf{A} - \mathbf{A}_1\|_F = 6$.

Exercise 6.27 (Another rank example)

Let \mathbf{A} be the $n \times n$ matrix that for $n = 4$ takes the form

$$\mathbf{A} = \begin{bmatrix} 1 & -1 & -1 & -1 \\ 0 & 1 & -1 & -1 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Thus \mathbf{A} is upper triangular with diagonal elements one and all elements above the diagonal equal to -1 . Let \mathbf{B} be the matrix obtained from \mathbf{A} by changing the $(n, 1)$ element from zero to -2^{2-n} .

- (a) Show that $\mathbf{Bx} = \mathbf{0}$, where $\mathbf{x} := [2^{n-2}, 2^{n-3}, \dots, 2^0, 1]^T$. Conclude that \mathbf{B} is singular, $\det(\mathbf{A}) = 1$, and $\|\mathbf{A} - \mathbf{B}\|_F = 2^{2-n}$. Thus even if $\det(\mathbf{A})$ is not small the Frobenius norm of $\mathbf{A} - \mathbf{B}$ is small for large n , and the matrix \mathbf{A} is very close to being singular for large n .
- (b) Use Theorem 6.25 to show that the smallest singular value σ_n of \mathbf{A} is bounded above by 2^{2-n} .

6.4 Review Questions

6.4.1 Consider an SVD and an SVF of a matrix \mathbf{A} .

- What are the singular values of \mathbf{A} ?
- how is the SVD defined?
- how can we find an SVF if we know an SVD?
- how can we find an SVD if we know an SVF?
- what are the relations between the singular vectors?
- which singular vectors form bases for $\text{span}(\mathbf{A})$ and $\ker(\mathbf{A}^*)$?

6.4.2 How are the Frobenius norm and singular values related?

Chapter 7

Norms and perturbation theory for linear systems

To measure the size of vector and matrices we use norms.

7.1 Vector Norms

Definition 7.1 (Vector norm)

A **(vector) norm** in a real (resp. complex) vector space \mathcal{V} is a function $\|\cdot\| : \mathcal{V} \rightarrow \mathbb{R}$ that satisfies for all \mathbf{x}, \mathbf{y} in \mathcal{V} and all a in \mathbb{R} (resp. \mathbb{C})

$$1. \quad \|\mathbf{x}\| \geq 0 \text{ with equality if and only if } \mathbf{x} = \mathbf{0}. \quad (\text{positivity})$$

$$2. \quad \|a\mathbf{x}\| = |a| \|\mathbf{x}\|. \quad (\text{homogeneity})$$

$$3. \quad \|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|. \quad (\text{subadditivity})$$

The triple $(\mathcal{V}, \mathbb{R}, \|\cdot\|)$ (resp. $(\mathcal{V}, \mathbb{C}, \|\cdot\|)$) is called a **normed vector space** and the inequality 3. is called the **triangle inequality**.

In this book the vector space will be one of $\mathbb{R}^n, \mathbb{C}^n$ or one of the matrix spaces $\mathbb{R}^{m \times n}$, or $\mathbb{C}^{m \times n}$. Vector addition is defined by element wise addition and scalar multiplication is defined by multiplying every element by the scalar.

In this book we will use the following family of vector norms on $\mathcal{V} = \mathbb{C}^n$ and $\mathcal{V} = \mathbb{R}^n$.



Otto Ludwig Hölder, 1859-1937 (left), Hermann Minkowski, 1864-1909 (right).

Definition 7.2 (Vector p-norms)

We define for $p \geq 1$ and $\mathbf{x} \in \mathbb{R}^n$ or $\mathbf{x} \in \mathbb{C}^n$ the **p -norms** by

$$\|\mathbf{x}\|_p := \left(\sum_{j=1}^n |x_j|^p \right)^{1/p}, \quad (7.1)$$

$$\|\mathbf{x}\|_\infty := \max_{1 \leq j \leq n} |x_j|. \quad (7.2)$$

The most important cases are $p = 1, 2, \infty$:

$$1. \quad \|\mathbf{x}\|_1 := \sum_{j=1}^n |x_j|, \quad (\text{the one-norm or } l_1\text{-norm})$$

$$2. \quad \|\mathbf{x}\|_2 := \sqrt{\sum_{j=1}^n |x_j|^2}, \quad (\text{the two-norm, } l_2\text{-norm, or Euclidian norm})$$

$$3. \quad \|\mathbf{x}\|_\infty := \max_{1 \leq j \leq n} |x_j|, \quad (\text{the infinity-norm, } l_\infty\text{-norm, or max norm})$$

Some remarks are in order.

1. That the Euclidian norm is a vector norm follows from Theorem 4.3. In Section 7.4, we show that the p -norms are vector norms for $1 \leq p \leq \infty$.
2. The triangle inequality $\|\mathbf{x} + \mathbf{y}\|_p \leq \|\mathbf{x}\|_p + \|\mathbf{y}\|_p$ is called **Minkowski's inequality**.
3. To prove it one first establishes **Hölder's inequality**

$$\sum_{j=1}^n |x_j y_j| \leq \|\mathbf{x}\|_p \|\mathbf{y}\|_q, \quad \frac{1}{p} + \frac{1}{q} = 1, \quad \mathbf{x}, \mathbf{y} \in \mathbb{C}^n. \quad (7.3)$$

The relation $\frac{1}{p} + \frac{1}{q} = 1$ means that if $p = 1$ then $q = \infty$ and if $p = 2$ then $q = 2$, and the Hölder's inequality is the same as the Cauchy-Schwarz inequality (cf. Theorem 4.2) for the Euclidian norm.

4. The infinity norm is related to the other p -norms by

$$\lim_{p \rightarrow \infty} \|\mathbf{x}\|_p = \|\mathbf{x}\|_\infty \text{ for all } \mathbf{x} \in \mathbb{C}^n. \quad (7.4)$$

5. The equation (7.4) clearly holds for $\mathbf{x} = \mathbf{0}$. For $\mathbf{x} \neq \mathbf{0}$ we write

$$\|\mathbf{x}\|_p := \|\mathbf{x}\|_\infty \left(\sum_{j=1}^n \left(\frac{|x_j|}{\|\mathbf{x}\|_\infty} \right)^p \right)^{1/p}.$$

Now each term in the sum is not greater than one and at least one term is equal to one, and we obtain

$$\|\mathbf{x}\|_\infty \leq \|\mathbf{x}\|_p \leq n^{1/p} \|\mathbf{x}\|_\infty, \quad p \geq 1. \quad (7.5)$$

Since $\lim_{p \rightarrow \infty} n^{1/p} = 1$ for any $n \in \mathbb{N}$ we see that (7.4) follows.

We return now to the general case.

Definition 7.3 (Equivalent norms)

We say that two norms $\|\cdot\|$ and $\|\cdot\|'$ on \mathcal{V} are **equivalent** if there are positive constants m and M such that for all vectors $\mathbf{x} \in \mathcal{V}$ we have

$$m\|\mathbf{x}\|' \leq \|\mathbf{x}\| \leq M\|\mathbf{x}\|'. \quad (7.6)$$

By (7.5) the p - and ∞ -norms are equivalent for any $p \geq 1$. This result is generalized in the following theorem.

Theorem 7.4 (Basic properties of vector norms)

The following holds for a normed vector space $(\mathcal{V}, \mathbb{C}, \|\cdot\|)$.

1. $\|\mathbf{x} - \mathbf{y}\| \geq |\|\mathbf{x}\| - \|\mathbf{y}\||$, for all $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$ (inverse triangle inequality).
2. The vector norm is a continuous function $\mathcal{V} \rightarrow \mathbb{R}$.
3. All vector norms on \mathcal{V} are equivalent provided \mathcal{V} is finite dimensional.

Proof.

1. Since $\|\mathbf{x}\| = \|\mathbf{x} - \mathbf{y} + \mathbf{y}\| \leq \|\mathbf{x} - \mathbf{y}\| + \|\mathbf{y}\|$ we obtain $\|\mathbf{x} - \mathbf{y}\| \geq \|\mathbf{x}\| - \|\mathbf{y}\|$. By symmetry $\|\mathbf{x} - \mathbf{y}\| = \|\mathbf{y} - \mathbf{x}\| \geq \|\mathbf{y}\| - \|\mathbf{x}\|$ and we obtain the inverse triangle inequality.

2. This follows from the inverse triangle inequality.
3. The following proof can be skipped by those who do not have the necessary background in advanced calculus. Define the $\|\cdot\|'$ unit sphere

$$\mathcal{S} := \{\mathbf{y} \in \mathcal{V} : \|\mathbf{y}\|' = 1\}.$$

The set \mathcal{S} is a closed and bounded set and the function $f : \mathcal{S} \rightarrow \mathbb{R}$ given by $f(\mathbf{y}) = \|\mathbf{y}\|$ is continuous by what we just showed. Therefore f attains its minimum and maximum value on \mathcal{S} . Thus, there are positive constants m and M such that

$$m \leq \|\mathbf{y}\| \leq M, \quad \mathbf{y} \in \mathcal{S}. \quad (7.7)$$

For any $\mathbf{x} \in \mathcal{V}$ one has $\mathbf{y} := \mathbf{x}/\|\mathbf{x}\|' \in \mathcal{S}$, and (7.6) follows if we apply (7.7) to these \mathbf{y} .

□

7.2 Matrix Norms

For simplicity we consider only norms on the vector space $(\mathbb{C}^{m \times n}, \mathbb{C})$. All results also holds for $(\mathbb{R}^{m \times n}, \mathbb{R})$. A matrix norm is simply a norm on these vector spaces. The Frobenius norm

$$\|\mathbf{A}\|_F := \left(\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2 \right)^{1/2}$$

is a matrix norm. Indeed, writing all elements in \mathbf{A} in a string of length mn we see that the Frobenius norm is the 2-norm on the Euclidian space \mathbb{C}^{mn} .

Adapting Theorem 7.4 to the matrix situation gives

Theorem 7.5 (Matrix norm equivalence)

All matrix norms on $\mathbb{C}^{m \times n}$ are equivalent. Thus, if $\|\cdot\|$ and $\|\cdot\|'$ are two matrix norms on $\mathbb{C}^{m \times n}$ then there are positive constants μ and M such that

$$\mu \|\mathbf{A}\| \leq \|\mathbf{A}\|' \leq M \|\mathbf{A}\|$$

holds for all $\mathbf{A} \in \mathbb{C}^{m \times n}$. Moreover, a matrix norm is a continuous function.

Any vector norm $\|\cdot\|_V$ on \mathbb{C}^{mn} defines a matrix norm on $\mathbb{C}^{m \times n}$ given by $\|\mathbf{A}\| := \|\text{vec}(\mathbf{A})\|_V$, where $\text{vec}(\mathbf{A}) \in \mathbb{C}^{mn}$ is the vector obtained by stacking the columns of \mathbf{A} on top of each other. In particular, to the p vector norms for $p = 1, 2, \infty$, we have the corresponding **sum norm**, **Frobenius norm**, and **max norm** defined by

$$\|\mathbf{A}\|_S := \sum_{i=1}^m \sum_{j=1}^n |a_{ij}|, \quad \|\mathbf{A}\|_F := \left(\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2 \right)^{1/2}, \quad \|\mathbf{A}\|_M := \max_{i,j} |a_{ij}|. \quad (7.8)$$



Ferdinand Georg Frobenius, (1849-1917).

Of these norms the Frobenius norm is the most useful. Some of its properties were derived in Lemma 6.23 and Theorem 6.24.

7.2.1 Consistent and subordinate matrix norms

Since matrices can be multiplied it is useful to have an analogue of subadditivity for matrix multiplication. For square matrices the product \mathbf{AB} is defined in a fixed space $\mathbb{C}^{n \times n}$, while in the rectangular case matrix multiplication combines matrices in different spaces. The following definition captures this distinction.

Definition 7.6 (Consistent matrix norms)

A matrix norm is called consistent on $\mathbb{C}^{n \times n}$ if

$$4. \|\mathbf{AB}\| \leq \|\mathbf{A}\| \|\mathbf{B}\| \quad (\text{submultiplicativity})$$

holds for all $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n \times n}$. A matrix norm is consistent if it is defined on $\mathbb{C}^{m \times n}$ for all $m, n \in \mathbb{N}$, and 4. holds for all matrices \mathbf{A}, \mathbf{B} for which the product \mathbf{AB} is defined.

Clearly the Frobenius norm is defined for all $m, n \in \mathbb{N}$. From Lemma 6.23 it follows that the Frobenius norm is consistent.

Exercise 7.7 (Consistency of sum norm?)

Show that the sum norm is consistent.

Exercise 7.8 (Consistency of max norm?)

Show that the max norm is not consistent by considering $\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$.

Exercise 7.9 (Consistency of modified max norm)

(a) Show that the norm

$$\|\mathbf{A}\| := \sqrt{mn} \|\mathbf{A}\|_M, \quad \mathbf{A} \in \mathbb{C}^{m \times n}$$

is a consistent matrix norm.

(b) Show that the constant \sqrt{mn} can be replaced by m and by n .

For a consistent matrix norm on $\mathbb{C}^{n \times n}$ we have the inequality

$$\|\mathbf{A}^k\| \leq \|\mathbf{A}\|^k \text{ for } k \in \mathbb{N}. \quad (7.9)$$

When working with norms one often has to bound the vector norm of a matrix times a vector by the norm of the matrix times the norm of the vector. This leads to the following definition.

Definition 7.10 (Subordinate matrix norms)

Suppose $m, n \in \mathbb{N}$ are given, let $\|\cdot\|$ on \mathbb{C}^m and $\|\cdot\|_\beta$ on \mathbb{C}^n be vector norms, and let $\|\cdot\|$ be a matrix norm on $\mathbb{C}^{m \times n}$. We say that the matrix norm $\|\cdot\|$ is **subordinate** to the vector norms $\|\cdot\|$ and $\|\cdot\|_\beta$ if $\|\mathbf{Ax}\| \leq \|\mathbf{A}\| \|\mathbf{x}\|_\beta$ for all $\mathbf{A} \in \mathbb{C}^{m \times n}$ and all $\mathbf{x} \in \mathbb{C}^n$.

By Lemma 6.23 we have $\|\mathbf{Ax}\|_2 \leq \|\mathbf{A}\|_F \|\mathbf{x}\|_2$, for all $\mathbf{x} \in \mathbb{C}^n$. Thus the Frobenius norm is subordinate to the Euclidian vector norm.

Exercise 7.11 (What is the sum norm subordinate to?)

Show that the sum norm is subordinate to the l_1 -norm.

Exercise 7.12 (What is the max norm subordinate to?)

(a) Show that the max norm is subordinate to the ∞ and 1 norm, i.e., $\|\mathbf{Ax}\|_\infty \leq \|\mathbf{A}\|_M \|\mathbf{x}\|_1$ holds for all $\mathbf{A} \in \mathbb{C}^{m \times n}$ and all $\mathbf{x} \in \mathbb{C}^n$.

(b) Show that if $\|\mathbf{A}\|_M = |a_{kl}|$, then $\|\mathbf{Ae}_l\|_\infty = \|\mathbf{A}\|_M \|\mathbf{e}_l\|_1$.

(c) Show that $\|\mathbf{A}\|_M = \max_{\mathbf{x} \neq \mathbf{0}} \frac{\|\mathbf{Ax}\|_\infty}{\|\mathbf{x}\|_1}$.

7.2.2 Operator norms

Corresponding to vector norms on \mathbb{C}^n and \mathbb{C}^m there is an induced matrix norm on $\mathbb{C}^{m \times n}$ which we call the **operator norm**. It is possible to consider one vector norm on \mathbb{C}^m and another vector norm on \mathbb{C}^n , but we treat only the case of one vector norm defined on \mathbb{C}^n for all $n \in \mathbb{N}$ ¹⁵.

¹⁵In the case of one vector norm $\|\cdot\|$ on \mathbb{C}^m and another vector norm $\|\cdot\|_\beta$ on \mathbb{C}^n we would define $\|\mathbf{A}\| := \max_{\mathbf{x} \neq \mathbf{0}} \frac{\|\mathbf{Ax}\|}{\|\mathbf{x}\|_\beta}$.

Definition 7.13 (Operator norm)

Let $\|\cdot\|$ be a vector norm defined on \mathbb{C}^n for all $n \in \mathbb{N}$. For given $m, n \in \mathbb{N}$ and $\mathbf{A} \in \mathbb{C}^{m \times n}$ we define

$$\|\mathbf{A}\| := \max_{\mathbf{x} \neq 0} \frac{\|\mathbf{Ax}\|}{\|\mathbf{x}\|}. \quad (7.10)$$

We call this the **operator norm** corresponding to the vector norm $\|\cdot\|$.

With a risk of confusion we use the same symbol for the operator norm and the corresponding vector norm. Before we show that the operator norm is a matrix norm we make some observations.

1. It is enough to take the max over subsets of \mathbb{C}^n . For example

$$\|\mathbf{A}\| = \max_{\|\mathbf{x}\|=1} \|\mathbf{Ax}\|. \quad (7.11)$$

The set

$$\mathcal{S} := \{\mathbf{x} \in \mathbb{C}^n : \|\mathbf{x}\| = 1\} \quad (7.12)$$

is the unit sphere in \mathbb{C}^n with respect to the vector norm $\|\cdot\|$. It is enough to take the max over this unit sphere since

$$\max_{\mathbf{x} \neq 0} \frac{\|\mathbf{Ax}\|}{\|\mathbf{x}\|} = \max_{\mathbf{x} \neq 0} \left\| \mathbf{A} \left(\frac{\mathbf{x}}{\|\mathbf{x}\|} \right) \right\| = \max_{\|\mathbf{y}\|=1} \|\mathbf{Ay}\|.$$

2. The operator norm is subordinate to the corresponding vector norm. Thus,

$$\|\mathbf{Ax}\| \leq \|\mathbf{A}\| \|\mathbf{x}\| \text{ for all } \mathbf{A} \in \mathbb{C}^{m \times n} \text{ and } \mathbf{x} \in \mathbb{C}^n. \quad (7.13)$$

3. We can use max instead of sup in (7.10). This follows by the following compactness argument. The unit sphere \mathcal{S} given by (7.12) is bounded. It is also finite dimensional and closed, and hence compact. Moreover, since the vector norm $\|\cdot\| : \mathcal{S} \rightarrow \mathbb{R}$ is a continuous function, it follows that the function $f : \mathcal{S} \rightarrow \mathbb{R}$ given by $f(\mathbf{x}) = \|\mathbf{Ax}\|$ is continuous. But then f attains its max and min and we have

$$\|\mathbf{A}\| = \|\mathbf{Ax}^*\| \text{ for some } \mathbf{x}^* \in \mathcal{S}. \quad (7.14)$$

Lemma 7.14 (The operator norm is a matrix norm)

For any vector norm the operator norm given by (7.10) is a consistent matrix norm. Moreover, $\|\mathbf{I}\| = 1$.

Proof. We use (7.11). In 2. and 3. below we take the max over the unit sphere \mathcal{S} given by (7.12).

1. Nonnegativity is obvious. If $\|\mathbf{A}\| = 0$ then $\|\mathbf{A}\mathbf{y}\| = 0$ for each $\mathbf{y} \in \mathbb{C}^n$. In particular, each column $\mathbf{A}\mathbf{e}_j$ in \mathbf{A} is zero. Hence $\mathbf{A} = 0$.
2. $\|c\mathbf{A}\| = \max_{\mathbf{x}} \|c\mathbf{A}\mathbf{x}\| = \max_{\mathbf{x}} |c| \|\mathbf{A}\mathbf{x}\| = |c| \|\mathbf{A}\|$.
3. $\|\mathbf{A} + \mathbf{B}\| = \max_{\mathbf{x}} \|(\mathbf{A} + \mathbf{B})\mathbf{x}\| \leq \max_{\mathbf{x}} \|\mathbf{A}\mathbf{x}\| + \max_{\mathbf{x}} \|\mathbf{B}\mathbf{x}\| = \|\mathbf{A}\| + \|\mathbf{B}\|$.
4.
$$\begin{aligned} \|\mathbf{AB}\| &= \max_{\mathbf{x} \neq 0} \frac{\|\mathbf{ABx}\|}{\|\mathbf{x}\|} = \max_{\mathbf{Bx} \neq 0} \frac{\|\mathbf{ABx}\|}{\|\mathbf{x}\|} = \max_{\mathbf{Bx} \neq 0} \frac{\|\mathbf{Ax}\|}{\|\mathbf{Bx}\|} \frac{\|\mathbf{Bx}\|}{\|\mathbf{x}\|} \\ &\leq \max_{\mathbf{y} \neq 0} \frac{\|\mathbf{Ay}\|}{\|\mathbf{y}\|} \max_{\mathbf{x} \neq 0} \frac{\|\mathbf{Bx}\|}{\|\mathbf{x}\|} = \|\mathbf{A}\| \|\mathbf{B}\|. \end{aligned}$$

That $\|\mathbf{I}\| = 1$ for any operator norm follows immediately from the definition. \square

Since $\|\mathbf{I}\|_F = \sqrt{n}$, we see that the Frobenius norm is not an operator norm for $n > 1$.

7.2.3 The operator p -norms

Recall that the p or ℓ_p vector norms (7.1) are given by

$$\|\mathbf{x}\|_p := \left(\sum_{j=1}^n |x_j|^p \right)^{1/p}, \quad p \geq 1, \quad \|\mathbf{x}\|_\infty := \max_{1 \leq j \leq n} |x_j|.$$

The operator norms $\|\cdot\|_p$ defined from these p -vector norms are used quite frequently for $p = 1, 2, \infty$. We define for any $1 \leq p \leq \infty$

$$\|\mathbf{A}\|_p := \max_{\mathbf{x} \neq 0} \frac{\|\mathbf{Ax}\|_p}{\|\mathbf{x}\|_p} = \max_{\|\mathbf{y}\|_p=1} \|\mathbf{Ay}\|_p. \quad (7.15)$$

For $p = 1, 2, \infty$ we have explicit expressions for these norms.

Theorem 7.15 (one-two-inf-norms)

For $\mathbf{A} \in \mathbb{C}^{m \times n}$ we have

$$\begin{aligned} \|\mathbf{A}\|_1 &:= \max_{1 \leq j \leq n} \|\mathbf{A}\mathbf{e}_j\|_1 = \max_{1 \leq j \leq n} \sum_{k=1}^m |a_{k,j}|, && (\text{max column sum}) \\ \|\mathbf{A}\|_2 &:= \sigma_1, && (\text{largest singular value of } \mathbf{A}) \quad (7.16) \\ \|\mathbf{A}\|_\infty &= \max_{1 \leq k \leq m} \|\mathbf{e}_k^T \mathbf{A}\|_1 = \max_{1 \leq k \leq m} \sum_{j=1}^n |a_{k,j}|. && (\text{max row sum}) \end{aligned}$$

The **two-norm** $\|\mathbf{A}\|_2$ is also called the **spectral norm** of \mathbf{A} .

Proof. The result for $p = 2$ follows from the singular value decomposition $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}$, or $\Sigma = \mathbf{U}^*\mathbf{A}\mathbf{V}$, where \mathbf{U} and \mathbf{V} are square and unitary, and Σ is a diagonal matrix with nonnegative diagonal elements $\sigma_1 \geq \dots \geq \sigma_n \geq 0$. Moreover,

$\mathbf{A}\mathbf{v}_1 = \sigma_1 \mathbf{u}_1$, where \mathbf{v}_1 and \mathbf{u}_1 are the first columns in \mathbf{V} and \mathbf{U} , respectively (cf. (6.10)). It follows that $\|\mathbf{A}\mathbf{v}_1\|_2 = \|\sigma_1 \mathbf{u}_1\|_2 = \sigma_1$. The result will follow if we can show that $\|\mathbf{Ax}\|_2 \leq \sigma_1$ for all $\mathbf{x} \in \mathbb{C}^n$ with $\|\mathbf{x}\|_2 = 1$. Let $\mathbf{x} = \mathbf{V}\mathbf{c}$. Then $\|\mathbf{V}\mathbf{c}\|_2 = \|\mathbf{c}\|_2 = 1$ so that

$$\|\mathbf{Ax}\|_2^2 = \|\mathbf{AVc}\|_2^2 = \|\mathbf{U}^*\mathbf{AVc}\|_2^2 = \|\Sigma\mathbf{c}\|_2^2 = \sum_{j=1}^n \sigma_j^2 |c_j|^2 \leq \sigma_1^2 \sum_{j=1}^n |c_j|^2 = \sigma_1^2.$$

For $p = 1, \infty$ we proceed as follows:

- (a) We derive a constant K_p such that $\|\mathbf{Ax}\|_p \leq K_p$ for any $\mathbf{x} \in \mathbb{C}^n$ with $\|\mathbf{x}\|_p = 1$.
- (b) We give an extremal vector $\mathbf{y}^* \in \mathbb{C}^n$ with $\|\mathbf{y}^*\|_p = 1$ so that $\|\mathbf{Ay}^*\|_p = K_p$.

It then follows from (7.15) that $\|\mathbf{A}\|_p = \|\mathbf{Ay}^*\|_p = K_p$.

1-norm: Define K_1 , c and \mathbf{y}^* by $K_1 := \|\mathbf{Ae}_c\|_1 = \max_{1 \leq j \leq n} \|\mathbf{Ae}_j\|_1$ and $\mathbf{y}^* := \mathbf{e}_c$, a unit vector. Then $\|\mathbf{y}^*\|_1 = 1$ and we obtain

(a)

$$\|\mathbf{Ax}\|_1 = \sum_{k=1}^m \left| \sum_{j=1}^n a_{kj} x_j \right| \leq \sum_{k=1}^m \sum_{j=1}^n |a_{kj}| |x_j| = \sum_{j=1}^n \left(\sum_{k=1}^m |a_{kj}| \right) |x_j| \leq K_1.$$

(b) $\|\mathbf{Ay}^*\|_1 = K_1$.

∞ -norm: Define K_∞ , r and \mathbf{y}^* by $K_\infty := \|\mathbf{e}_r^T \mathbf{A}\|_1 = \max_{1 \leq k \leq m} \|\mathbf{e}_k^T \mathbf{A}\|_1$ and $\mathbf{y}^* := [e^{-i\theta_1}, \dots, e^{-i\theta_n}]^T$, where $a_{rj} = |a_{rj}| e^{i\theta_j}$ for $j = 1, \dots, n$.

- (a) $\|\mathbf{Ax}\|_\infty = \max_{1 \leq k \leq m} \left| \sum_{j=1}^n a_{kj} x_j \right| \leq \max_{1 \leq k \leq m} \sum_{j=1}^n |a_{kj}| |x_j| \leq K_\infty$.
- (b) $\|\mathbf{Ay}^*\|_\infty = \max_{1 \leq k \leq m} \left| \sum_{j=1}^n a_{kj} e^{-i\theta_j} \right| = K_\infty$.

The last equality is correct because $\left| \sum_{j=1}^n a_{kj} e^{-i\theta_j} \right| \leq \sum_{j=1}^n |a_{kj}| \leq K_\infty$ with equality for $k = r$.

□

Example 7.16 (Comparing one-two-inf-norms)

The largest singular value of the matrix $\mathbf{A} := \frac{1}{15} \begin{bmatrix} 14 & 4 & 16 \\ 2 & 22 & 13 \end{bmatrix}$, is $\sigma_1 = 2$ (cf. Example 6.14). We find

$$\|\mathbf{A}\|_1 = \frac{29}{15}, \quad \|\mathbf{A}\|_2 = 2, \quad \|\mathbf{A}\|_\infty = \frac{37}{15}, \quad \|\mathbf{A}\|_F = \sqrt{5}.$$

We observe that the values of these norms do not differ by much.

In some cases the spectral norm is equal to an eigenvalue of the matrix.

Theorem 7.17 (Spectral norm)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ has singular values $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$ and eigenvalues $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n|$. Then

$$\|\mathbf{A}\|_2 = \sigma_1 \text{ and } \|\mathbf{A}^{-1}\|_2 = \frac{1}{\sigma_n}, \quad (7.17)$$

$$\|\mathbf{A}\|_2 = \lambda_1 \text{ and } \|\mathbf{A}^{-1}\|_2 = \frac{1}{\lambda_n}, \quad \text{if } \mathbf{A} \text{ is Hermitian positive definite,} \quad (7.18)$$

$$\|\mathbf{A}\|_2 = |\lambda_1| \text{ and } \|\mathbf{A}^{-1}\|_2 = \frac{1}{|\lambda_n|}, \quad \text{if } \mathbf{A} \text{ is normal.} \quad (7.19)$$

For the norms of \mathbf{A}^{-1} we assume of course that \mathbf{A} is nonsingular.

Proof. Since $1/\sigma_n$ is the largest singular value of \mathbf{A}^{-1} , (7.17) follows. By Theorem 6.9 the singular values of a Hermitian positive definite matrix (normal matrix) are equal to the eigenvalues (absolute value of the eigenvalues). This implies (7.18) and (7.19). \square

The following result is sometimes useful.

Theorem 7.18 (Spectral norm bound)

For any $\mathbf{A} \in \mathbb{C}^{m \times n}$ we have $\|\mathbf{A}\|_2^2 \leq \|\mathbf{A}\|_1 \|\mathbf{A}\|_\infty$.

Proof. Let (σ^2, \mathbf{v}) be an eigenpair for $\mathbf{A}^* \mathbf{A}$ corresponding to the largest singular value σ of \mathbf{A} . Then

$$\|\mathbf{A}\|_2^2 \|\mathbf{v}\|_1 = \sigma^2 \|\mathbf{v}\|_1 = \|\sigma^2 \mathbf{v}\|_1 = \|\mathbf{A}^* \mathbf{A} \mathbf{v}\|_1 \leq \|\mathbf{A}^*\|_1 \|\mathbf{A}\|_1 \|\mathbf{v}\|_1.$$

Observing that $\|\mathbf{A}^*\|_1 = \|\mathbf{A}\|_\infty$ by Theorem 7.15 and canceling $\|\mathbf{v}\|_1$ proves the result. \square

Exercise 7.19 (Spectral norm)

Let $m, n \in \mathbb{N}$ and $\mathbf{A} \in \mathbb{C}^{m \times n}$. Show that

$$\|\mathbf{A}\|_2 = \max_{\|\mathbf{x}\|_2 = \|\mathbf{y}\|_2 = 1} |\mathbf{y}^* \mathbf{A} \mathbf{x}|.$$

Exercise 7.20 (Spectral norm of the inverse)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is nonsingular. Show that $\|\mathbf{A}\|_2 \geq \sigma_n$ for all $\mathbf{x} \in \mathbb{C}^n$ with $\|\mathbf{x}\|_2 = 1$. Use this and (7.17) to show that

$$\|\mathbf{A}^{-1}\|_2 = \max_{\mathbf{x} \neq 0} \frac{\|\mathbf{x}\|_2}{\|\mathbf{A}\mathbf{x}\|_2}.$$

Exercise 7.21 (p -norm example)*Let*

$$\mathbf{A} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}.$$

*Compute $\|\mathbf{A}\|_p$ and $\|\mathbf{A}^{-1}\|_p$ for $p = 1, 2, \infty$.***7.2.4 Unitary invariant matrix norms****Definition 7.22 (Unitary invariant norm)***A matrix norm $\|\cdot\|$ on $\mathbb{C}^{m \times n}$ is called **unitary invariant** if $\|\mathbf{U}\mathbf{A}\mathbf{V}\| = \|\mathbf{A}\|$ for any $\mathbf{A} \in \mathbb{C}^{m \times n}$ and any unitary matrices $\mathbf{U} \in \mathbb{C}^{m \times m}$ and $\mathbf{V} \in \mathbb{C}^{n \times n}$.*

When a unitary invariant matrix norm is used, the size of a perturbation is not increased by a unitary transformation. Thus if \mathbf{U} and \mathbf{V} are unitary then $\mathbf{U}(\mathbf{A} + \mathbf{E})\mathbf{V} = \mathbf{UAV} + \mathbf{F}$, where $\|\mathbf{F}\| = \|\mathbf{E}\|$.

It follows from Lemma 6.23 that the Frobenius norm is unitary invariant. We show here that this also holds for the spectral norm.

Theorem 7.23 (Unitary invariant norms)

The Frobenius norm and the spectral norm are unitary invariant. Moreover $\|\mathbf{A}^\|_F = \|\mathbf{A}\|_F$ and $\|\mathbf{A}^*\|_2 = \|\mathbf{A}\|_2$.*

Proof. The results for the Frobenius norm follow from Lemma 6.23. Suppose $\mathbf{A} \in \mathbb{C}^{m \times n}$ and let $\mathbf{U} \in \mathbb{C}^{m \times m}$ and $\mathbf{V} \in \mathbb{C}^{n \times n}$ be unitary. Since the 2-vector norm is unitary invariant we obtain

$$\|\mathbf{U}\mathbf{A}\|_2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{U}\mathbf{A}\mathbf{x}\|_2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{A}\mathbf{x}\|_2 = \|\mathbf{A}\|_2.$$

Now \mathbf{A} and \mathbf{A}^* have the same nonzero singular values, and it follows from Theorem 7.15 that $\|\mathbf{A}^*\|_2 = \|\mathbf{A}\|_2$. Moreover \mathbf{V}^* is unitary. Using these facts we find

$$\|\mathbf{A}\mathbf{V}\|_2 = \|(\mathbf{A}\mathbf{V})^*\|_2 = \|\mathbf{V}^*\mathbf{A}^*\|_2 = \|\mathbf{A}^*\|_2 = \|\mathbf{A}\|_2.$$

□

It can be shown that the spectral norm is the only unitary invariant operator norm, see [15] p. 308.

Exercise 7.24 (Unitary invariance of the spectral norm)

Show that $\|\mathbf{V}\mathbf{A}\|_2 = \|\mathbf{A}\|_2$ holds even for a rectangular \mathbf{V} as long as $\mathbf{V}^\mathbf{V} = \mathbf{I}$.*

Exercise 7.25 ($\|\mathbf{AU}\|_2$ rectangular \mathbf{A})

Find $\mathbf{A} \in \mathbb{R}^{2 \times 2}$ and $\mathbf{U} \in \mathbb{R}^{2 \times 1}$ with $\mathbf{U}^T\mathbf{U} = \mathbf{I}$ such that $\|\mathbf{AU}\|_2 < \|\mathbf{A}\|_2$. Thus, in general, $\|\mathbf{AU}\|_2 = \|\mathbf{A}\|_2$ does not hold for a rectangular \mathbf{U} even if $\mathbf{U}^\mathbf{U} = \mathbf{I}$.*

Exercise 7.26 (p -norm of diagonal matrix)

Show that $\|\mathbf{A}\|_p = \rho(\mathbf{A}) := \max |\lambda_i|$ (the largest eigenvalue of \mathbf{A}), $1 \leq p \leq \infty$, when \mathbf{A} is a diagonal matrix.

Exercise 7.27 (spectral norm of a column vector)

A vector $\mathbf{a} \in \mathbb{C}^m$ can also be considered as a matrix $\mathbf{A} \in \mathbb{C}^{m,1}$.

- (a) Show that the spectral matrix norm (2-norm) of \mathbf{A} equals the Euclidean vector norm of \mathbf{a} .
- (b) Show that $\|\mathbf{A}\|_p = \|\mathbf{a}\|_p$ for $1 \leq p \leq \infty$.

7.2.5 Absolute and monotone norms

A vector norm on \mathbb{C}^n is an **absolute norm** if $\|\mathbf{x}\| = \|\mathbf{|x|}\|$ for all $\mathbf{x} \in \mathbb{C}^n$. Here $\mathbf{|x|} := [|x_1|, \dots, |x_n|]^T$, the absolute values of the components of \mathbf{x} . Clearly the vector p norms are absolute norms. We state without proof (see Theorem 5.5.10 of [15]) that a vector norm on \mathbb{C}^n is an absolute norm if and only if it is a **monotone norm**, i.e.,

$$|x_i| \leq |y_i|, \quad i = 1, \dots, n \implies \|\mathbf{x}\| \leq \|\mathbf{y}\|, \text{ for all } \mathbf{x}, \mathbf{y} \in \mathbb{C}^n.$$

Absolute and monotone matrix norms are defined as for vector norms.

Exercise 7.28 (Norm of absolute value matrix)

If $\mathbf{A} \in \mathbb{C}^{m \times n}$ has elements a_{ij} , let $|\mathbf{A}| \in \mathbb{R}^{m \times n}$ be the matrix with elements $|a_{ij}|$.

- (a) Compute $|\mathbf{A}|$ if $\mathbf{A} = \begin{bmatrix} 1+i & -2 \\ 1 & 1-i \end{bmatrix}$, $i = \sqrt{-1}$.
- (b) Show that for any $\mathbf{A} \in \mathbb{C}^{m \times n}$ $\|\mathbf{A}\|_F = \||\mathbf{A}|\|_F$, $\|\mathbf{A}\|_p = \||\mathbf{A}|\|_p$ for $p = 1, \infty$.
- (c) Show that for any $\mathbf{A} \in \mathbb{C}^{m \times n}$ $\|\mathbf{A}\|_2 \leq \||\mathbf{A}|\|_2$.
- (d) Find a real symmetric 2×2 matrix \mathbf{A} such that $\|\mathbf{A}\|_2 < \||\mathbf{A}|\|_2$.

The study of matrix norms will be continued in Chapter 11.

7.3 The Condition Number with Respect to Inversion

Consider the system of two linear equations

$$\begin{array}{rcl} x_1 & + & x_2 = 20 \\ x_1 & + & (1 - 10^{-16})x_2 = 20 - 10^{-15} \end{array}$$

whose exact solution is $x_1 = x_2 = 10$. If we replace the second equation by

$$x_1 + (1 + 10^{-16})x_2 = 20 - 10^{-15},$$

the exact solution changes to $x_1 = 30$, $x_2 = -10$. Here a small change in one of the coefficients, from $1 - 10^{-16}$ to $1 + 10^{-16}$, changed the exact solution by a large amount.

A mathematical problem in which the solution is very sensitive to changes in the data is called **ill-conditioned**. Such problems can be difficult to solve on a computer.

In this section we consider what effect a small change (perturbation) in the data \mathbf{A}, \mathbf{b} has on the solution \mathbf{x} of a linear system $\mathbf{Ax} = \mathbf{b}$. Suppose \mathbf{y} solves $(\mathbf{A} + \mathbf{E})\mathbf{y} = \mathbf{b} + \mathbf{e}$ where \mathbf{E} is a (small) $n \times n$ matrix and \mathbf{e} a (small) vector. How large can $\mathbf{y} - \mathbf{x}$ be? To measure this we use vector and matrix norms. In this section $\|\cdot\|$ will denote a vector norm on \mathbb{C}^n and also a matrix norm on $\mathbb{C}^{n \times n}$ which for any $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n \times n}$ and any $\mathbf{x} \in \mathbb{C}^n$ satisfy

$$\|\mathbf{AB}\| \leq \|\mathbf{A}\| \|\mathbf{B}\| \text{ and } \|\mathbf{Ax}\| \leq \|\mathbf{A}\| \|\mathbf{x}\|.$$

This holds if the matrix norm is the operator norm corresponding to the given vector norm, but is also satisfied for the Frobenius matrix norm and the Euclidian vector norm. This follows from Lemma 6.23.

Suppose \mathbf{x} and \mathbf{y} are vectors in \mathbb{C}^n that we want to compare. The difference $\|\mathbf{y} - \mathbf{x}\|$ measures the **absolute error** in \mathbf{y} as an approximation to \mathbf{x} , while $\|\mathbf{y} - \mathbf{x}\|/\|\mathbf{x}\|$ and $\|\mathbf{y} - \mathbf{x}\|/\|\mathbf{y}\|$ are measures for the **relative error**.

We consider first a perturbation in the right-hand side \mathbf{b} .

Theorem 7.29 (Perturbation in the right-hand side)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is nonsingular, $\mathbf{b}, \mathbf{e} \in \mathbb{C}^n$, $\mathbf{b} \neq \mathbf{0}$ and $\mathbf{Ax} = \mathbf{b}$, $\mathbf{Ay} = \mathbf{b} + \mathbf{e}$. Then

$$\frac{1}{K(\mathbf{A})} \frac{\|\mathbf{e}\|}{\|\mathbf{b}\|} \leq \frac{\|\mathbf{y} - \mathbf{x}\|}{\|\mathbf{x}\|} \leq K(\mathbf{A}) \frac{\|\mathbf{e}\|}{\|\mathbf{b}\|}, \quad K(\mathbf{A}) := \|\mathbf{A}\| \|\mathbf{A}^{-1}\|. \quad (7.20)$$

Proof. Subtracting $\mathbf{Ax} = \mathbf{b}$ from $\mathbf{Ay} = \mathbf{b} + \mathbf{e}$ we have $\mathbf{A}(\mathbf{y} - \mathbf{x}) = \mathbf{e}$ or $\mathbf{y} - \mathbf{x} = \mathbf{A}^{-1}\mathbf{e}$. Combining $\|\mathbf{y} - \mathbf{x}\| = \|\mathbf{A}^{-1}\mathbf{e}\| \leq \|\mathbf{A}^{-1}\| \|\mathbf{e}\|$ and $\|\mathbf{b}\| = \|\mathbf{Ax}\| \leq \|\mathbf{A}\| \|\mathbf{x}\|$ we obtain the upper bound in (7.20). Combining $\|\mathbf{e}\| \leq \|\mathbf{A}\| \|\mathbf{y} - \mathbf{x}\|$ and $\|\mathbf{x}\| \leq \|\mathbf{A}^{-1}\| \|\mathbf{b}\|$ we obtain the lower bound. \square

Consider (7.20). $\|\mathbf{e}\|/\|\mathbf{b}\|$ is a measure of the size of the perturbation \mathbf{e} relative to the size of \mathbf{b} . The upper bound says that $\|\mathbf{y} - \mathbf{x}\|/\|\mathbf{x}\|$ in the worst case can be $K(\mathbf{A}) = \|\mathbf{A}\| \|\mathbf{A}^{-1}\|$ times as large as $\|\mathbf{e}\|/\|\mathbf{b}\|$. $K(\mathbf{A})$ is called the **condition number with respect to inversion of a matrix**, or just the condition number, if it is clear from the context that we are talking about solving

linear systems or inverting a matrix. The condition number depends on the matrix \mathbf{A} and on the norm used. If $K(\mathbf{A})$ is large, \mathbf{A} is called **ill-conditioned** (with respect to inversion). If $K(\mathbf{A})$ is small, \mathbf{A} is called **well-conditioned** (with respect to inversion). We always have $K(\mathbf{A}) \geq 1$. For since $\|\mathbf{x}\| = \|\mathbf{I}\mathbf{x}\| \leq \|\mathbf{I}\| \|\mathbf{x}\|$ for any \mathbf{x} we have $\|\mathbf{I}\| \geq 1$ and therefore $\|\mathbf{A}\| \|\mathbf{A}^{-1}\| \geq \|\mathbf{A}\mathbf{A}^{-1}\| = \|\mathbf{I}\| \geq 1$.

Since all matrix norms are equivalent, the dependence of $K(\mathbf{A})$ on the norm chosen is less important than the dependence on \mathbf{A} . Sometimes one chooses the spectral norm when discussing properties of the condition number, and the ℓ_1 , ℓ_∞ , or Frobenius norm when one wishes to compute it or estimate it.

The following explicit expressions for the 2-norm condition number follow from Theorem 7.17.

Theorem 7.30 (Spectral condition number)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is nonsingular with singular values $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n > 0$ and eigenvalues $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n| > 0$. Then $K_2(\mathbf{A}) := \|\mathbf{A}\|_2 \|\mathbf{A}^{-1}\|_2 = \sigma_1/\sigma_n$. Moreover,

$$K_2(\mathbf{A}) = \begin{cases} \lambda_1/\lambda_n, & \text{if } \mathbf{A} \text{ is Hermitian positive definite,} \\ |\lambda_1|/|\lambda_n|, & \text{if } \mathbf{A} \text{ is normal.} \end{cases} \quad (7.21)$$

It follows that \mathbf{A} is ill-conditioned with respect to inversion if and only if σ_1/σ_n is large, or λ_1/λ_n is large when \mathbf{A} is Hermitian positive definite.

Suppose we have computed an approximate solution \mathbf{y} to $\mathbf{Ax} = \mathbf{b}$. The vector $\mathbf{r}(\mathbf{y}) := \mathbf{Ay} - \mathbf{b}$ is called the **residual vector**, or just the residual. We can bound $\mathbf{x} - \mathbf{y}$ in terms of \mathbf{r} .

Theorem 7.31 (Perturbation and residual)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$, $\mathbf{b} \in \mathbb{C}^n$, \mathbf{A} is nonsingular and $\mathbf{b} \neq \mathbf{0}$. Let $\mathbf{r}(\mathbf{y}) = \mathbf{Ay} - \mathbf{b}$ for any $\mathbf{y} \in \mathbb{C}^n$. If $\mathbf{Ax} = \mathbf{b}$ then

$$\frac{1}{K(\mathbf{A})} \frac{\|\mathbf{r}(\mathbf{y})\|}{\|\mathbf{b}\|} \leq \frac{\|\mathbf{y} - \mathbf{x}\|}{\|\mathbf{x}\|} \leq K(\mathbf{A}) \frac{\|\mathbf{r}(\mathbf{y})\|}{\|\mathbf{b}\|}. \quad (7.22)$$

Proof. We simply take $\mathbf{e} = \mathbf{r}(\mathbf{y})$ in Theorem 7.29. \square

If \mathbf{A} is well-conditioned, (7.22) says that $\|\mathbf{y} - \mathbf{x}\|/\|\mathbf{x}\| \approx \|\mathbf{r}(\mathbf{y})\|/\|\mathbf{b}\|$. In other words, the accuracy in \mathbf{y} is about the same order of magnitude as the residual as long as $\|\mathbf{b}\| \approx 1$. If \mathbf{A} is ill-conditioned, anything can happen. We can for example have an accurate solution even if the residual is large.

Consider next the effect of a perturbation in the coefficient matrix. Suppose $\mathbf{A}, \mathbf{E} \in \mathbb{C}^{n \times n}$ with \mathbf{A} nonsingular. We like to compare the solution \mathbf{x} and \mathbf{y} of the systems $\mathbf{Ax} = \mathbf{b}$ and $(\mathbf{A} + \mathbf{E})\mathbf{y} = \mathbf{b}$. We expect $\mathbf{A} + \mathbf{E}$ to be nonsingular if the elements of \mathbf{E} are sufficiently small and we need to address this question. Consider first the case where $\mathbf{A} = \mathbf{I}$.

Theorem 7.32 (Nonsingularity of perturbation of identity)

Suppose $\mathbf{B} \in \mathbb{C}^{n \times n}$ and $\|\mathbf{B}\| < 1$ for some consistent matrix norm on $\mathbb{C}^{n \times n}$. Then $\mathbf{I} - \mathbf{B}$ is nonsingular and

$$\frac{1}{1 + \|\mathbf{B}\|} \leq \|(\mathbf{I} - \mathbf{B})^{-1}\| \leq \frac{1}{1 - \|\mathbf{B}\|}. \quad (7.23)$$

Proof. Suppose $\mathbf{I} - \mathbf{B}$ is singular. Then $(\mathbf{I} - \mathbf{B})\mathbf{x} = \mathbf{0}$ for some nonzero $\mathbf{x} \in \mathbb{C}^n$, and $\mathbf{x} = \mathbf{Bx}$ so that $\|\mathbf{x}\| = \|\mathbf{Bx}\| \leq \|\mathbf{B}\|\|\mathbf{x}\|$. But then $\|\mathbf{B}\| \geq 1$. It follows that $\mathbf{I} - \mathbf{B}$ is nonsingular if $\|\mathbf{B}\| < 1$. Next, since

$$\begin{aligned} \|\mathbf{I}\| &= \|(\mathbf{I} - \mathbf{B})(\mathbf{I} - \mathbf{B})^{-1}\| \leq \|\mathbf{I} - \mathbf{B}\| \|(\mathbf{I} - \mathbf{B})^{-1}\| \\ &\leq (\|\mathbf{I}\| + \|\mathbf{B}\|) \|(\mathbf{I} - \mathbf{B})^{-1}\|, \end{aligned}$$

and since $\|\mathbf{I}\| \geq 1$, we obtain the lower bound in (7.23):

$$\frac{1}{1 + \|\mathbf{B}\|} \leq \frac{\|\mathbf{I}\|}{\|\mathbf{I}\| + \|\mathbf{B}\|} \leq \|(\mathbf{I} - \mathbf{B})^{-1}\|. \quad (7.24)$$

Taking norms and using the inverse triangle inequality in

$$\mathbf{I} = (\mathbf{I} - \mathbf{B})(\mathbf{I} - \mathbf{B})^{-1} = (\mathbf{I} - \mathbf{B})^{-1} - \mathbf{B}(\mathbf{I} - \mathbf{B})^{-1}$$

implies

$$\|\mathbf{I}\| \geq \|(\mathbf{I} - \mathbf{B})^{-1}\| - \|\mathbf{B}(\mathbf{I} - \mathbf{B})^{-1}\| \geq (1 - \|\mathbf{B}\|) \|(\mathbf{I} - \mathbf{B})^{-1}\|.$$

If the matrix norm is an operator norm then $\|\mathbf{I}\| = 1$ and the upper bound follows. We show in Section 11.4 that the upper bound also holds for the Frobenius norm, and more generally for any consistent matrix norm on $\mathbb{C}^{n \times n}$. \square

Theorem 7.33 (Nonsingularity of perturbation)

Suppose $\mathbf{A}, \mathbf{E} \in \mathbb{C}^{n \times n}$, $\mathbf{b} \in \mathbb{C}^n$ with \mathbf{A} nonsingular and $\mathbf{b} \neq \mathbf{0}$. If $r := \|\mathbf{A}^{-1}\mathbf{E}\| < 1$ for some matrix norm consistent on $\mathbb{C}^{n \times n}$ then $\mathbf{A} + \mathbf{E}$ is nonsingular. If $\mathbf{Ax} = \mathbf{b}$ and $(\mathbf{A} + \mathbf{E})\mathbf{y} = \mathbf{b}$ then

$$\frac{\|\mathbf{y} - \mathbf{x}\|}{\|\mathbf{y}\|} \leq \|\mathbf{A}^{-1}\mathbf{E}\| \leq K(\mathbf{A}) \frac{\|\mathbf{E}\|}{\|\mathbf{A}\|}, \quad (7.25)$$

$$\frac{\|\mathbf{y} - \mathbf{x}\|}{\|\mathbf{x}\|} \leq 2K(\mathbf{A}) \frac{\|\mathbf{E}\|}{\|\mathbf{A}\|}. \quad (7.26)$$

In (7.26) we have assumed that $r \leq 1/2$.

Proof. Since $r < 1$ Theorem 7.32 implies that the matrix $\mathbf{I} - \mathbf{B} := \mathbf{I} + \mathbf{A}^{-1}\mathbf{E}$ is nonsingular and then $\mathbf{A} + \mathbf{E} = \mathbf{A}(\mathbf{I} + \mathbf{A}^{-1}\mathbf{E})$ is nonsingular. Subtracting $(\mathbf{A} + \mathbf{E})\mathbf{y} = \mathbf{b}$ from $\mathbf{Ax} = \mathbf{b}$ gives $\mathbf{A}(\mathbf{x} - \mathbf{y}) = \mathbf{E}\mathbf{y}$ or $\mathbf{x} - \mathbf{y} = \mathbf{A}^{-1}\mathbf{E}\mathbf{y}$. Taking norms and dividing by $\|\mathbf{y}\|$ proves (7.25). Solving $\mathbf{x} - \mathbf{y} = \mathbf{A}^{-1}\mathbf{E}\mathbf{y}$ for \mathbf{y} we obtain $\mathbf{y} = (\mathbf{I} + \mathbf{A}^{-1}\mathbf{E})^{-1}\mathbf{x}$. By (7.23)

$$\|\mathbf{y}\| \leq \|(\mathbf{I} + \mathbf{A}^{-1}\mathbf{E})^{-1}\| \|\mathbf{x}\| \leq \frac{\|\mathbf{x}\|}{1 - \|\mathbf{A}^{-1}\mathbf{E}\|} \leq 2\|\mathbf{x}\|.$$

But then (7.26) follows from (7.25). \square

In Theorem 7.33 we gave bounds for the relative error in \mathbf{x} as an approximation to \mathbf{y} and the relative error in \mathbf{y} as an approximation to \mathbf{x} . $\|\mathbf{E}\|/\|\mathbf{A}\|$ is a measure for the size of the perturbation \mathbf{E} in \mathbf{A} relative to the size of \mathbf{A} . The condition number again plays a crucial role. $\|\mathbf{y} - \mathbf{x}\|/\|\mathbf{y}\|$ can be as large as $K(\mathbf{A})$ times $\|\mathbf{E}\|/\|\mathbf{A}\|$. It can be shown that the upper bound can be attained for any \mathbf{A} and any \mathbf{b} . In deriving the upper bound we used the inequality $\|\mathbf{A}^{-1}\mathbf{E}\mathbf{y}\| \leq \|\mathbf{A}^{-1}\| \|\mathbf{E}\| \|\mathbf{y}\|$. For a more or less random perturbation \mathbf{E} this is not a severe overestimate for $\|\mathbf{A}^{-1}\mathbf{E}\mathbf{y}\|$. In the situation where \mathbf{E} is due to round-off errors (7.25) can give a fairly realistic estimate for $\|\mathbf{y} - \mathbf{x}\|/\|\mathbf{y}\|$.

We end this section with a perturbation result for the inverse matrix. Again the condition number plays an important role.

Theorem 7.34 (Perturbation of inverse matrix)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is nonsingular and let $\|\cdot\|$ be a consistent matrix norm on $\mathbb{C}^{n \times n}$. If $\mathbf{E} \in \mathbb{C}^{n \times n}$ is so small that $r := \|\mathbf{A}^{-1}\mathbf{E}\| < 1$ then $\mathbf{A} + \mathbf{E}$ is nonsingular and

$$\|(\mathbf{A} + \mathbf{E})^{-1}\| \leq \frac{\|\mathbf{A}^{-1}\|}{1 - r}. \quad (7.27)$$

If $r < 1/2$ then

$$\frac{\|(\mathbf{A} + \mathbf{E})^{-1} - \mathbf{A}^{-1}\|}{\|\mathbf{A}^{-1}\|} \leq 2K(\mathbf{A}) \frac{\|\mathbf{E}\|}{\|\mathbf{A}\|}. \quad (7.28)$$

Proof. We showed in Theorem 7.33 that $\mathbf{A} + \mathbf{E}$ is nonsingular and since $(\mathbf{A} + \mathbf{E})^{-1} = (\mathbf{I} + \mathbf{A}^{-1}\mathbf{E})^{-1}\mathbf{A}^{-1}$ we obtain

$$\|(\mathbf{A} + \mathbf{E})^{-1}\| \leq \|(\mathbf{I} + \mathbf{A}^{-1}\mathbf{E})^{-1}\| \|\mathbf{A}^{-1}\| \leq \frac{\|\mathbf{A}^{-1}\|}{1 - \|\mathbf{A}^{-1}\mathbf{E}\|}$$

and (7.27) follows. Since

$$(\mathbf{A} + \mathbf{E})^{-1} - \mathbf{A}^{-1} = (\mathbf{I} - \mathbf{A}^{-1}(\mathbf{A} + \mathbf{E}))(\mathbf{A} + \mathbf{E})^{-1} = -\mathbf{A}^{-1}\mathbf{E}(\mathbf{A} + \mathbf{E})^{-1}$$

we obtain by (7.27)

$$\|(\mathbf{A} + \mathbf{E})^{-1} - \mathbf{A}^{-1}\| \leq \|\mathbf{A}^{-1}\| \|\mathbf{E}\| \|(\mathbf{A} + \mathbf{E})^{-1}\| \leq K(\mathbf{A}) \frac{\|\mathbf{E}\| \|\mathbf{A}^{-1}\|}{\|\mathbf{A}\|} \frac{\|\mathbf{A}^{-1}\|}{1-r}.$$

Dividing by $\|\mathbf{A}^{-1}\|$ and setting $r < 1/2$ proves (7.28). \square

Exercise 7.35 (Sharpness of perturbation bounds)

The upper and lower bounds for $\|\mathbf{y} - \mathbf{x}\|/\|\mathbf{x}\|$ given by (7.20) can be attained for any matrix \mathbf{A} , but only for special choices of \mathbf{b} . Suppose $\mathbf{y}_\mathbf{A}$ and $\mathbf{y}_{\mathbf{A}^{-1}}$ are vectors with $\|\mathbf{y}_\mathbf{A}\| = \|\mathbf{y}_{\mathbf{A}^{-1}}\| = 1$ and $\|\mathbf{A}\| = \|\mathbf{A}\mathbf{y}_\mathbf{A}\|$ and $\|\mathbf{A}^{-1}\| = \|\mathbf{A}^{-1}\mathbf{y}_{\mathbf{A}^{-1}}\|$.

- (a) Show that the upper bound in (7.20) is attained if $\mathbf{b} = \mathbf{A}\mathbf{y}_\mathbf{A}$ and $\mathbf{e} = \mathbf{y}_{\mathbf{A}^{-1}}$.
- (b) Show that the lower bound is attained if $\mathbf{b} = \mathbf{y}_{\mathbf{A}^{-1}}$ and $\mathbf{e} = \mathbf{A}\mathbf{y}_\mathbf{A}$.

Exercise 7.36 (Condition number of 2. derivative matrix)

In this exercise we will show that for $m \geq 1$

$$\frac{4}{\pi^2}(m+1)^2 - 2/3 < \text{cond}_p(\mathbf{T}) \leq \frac{1}{2}(m+1)^2, \quad p = 1, 2, \infty, \quad (7.29)$$

where $\mathbf{T} := \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{m \times m}$ and $\text{cond}_p(\mathbf{T}) := \|\mathbf{T}\|_p \|\mathbf{T}^{-1}\|_p$ is the p -norm condition number of \mathbf{T} . The p matrix norm is given by (7.15). You will need the explicit inverse of \mathbf{T} given by (1.27) and the eigenvalues given in Lemma 1.31. As usual we define $h := 1/(m+1)$.

- a) Show that for $m \geq 3$

$$\text{cond}_1(\mathbf{T}) = \text{cond}_\infty(\mathbf{T}) = \frac{1}{2} \begin{cases} h^{-2}, & m \text{ odd}, \\ h^{-2} - 1, & m \text{ even}. \end{cases} \quad (7.30)$$

and that $\text{cond}_1(\mathbf{T}) = \text{cond}_\infty(\mathbf{T}) = 3$ for $m = 2$.

- b) Show that for $p = 2$ and $m \geq 1$ we have

$$\text{cond}_2(\mathbf{T}) = \cot^2\left(\frac{\pi h}{2}\right) = 1/\tan^2\left(\frac{\pi h}{2}\right).$$

- c) Show the bounds

$$\frac{4}{\pi^2}h^{-2} - \frac{2}{3} < \text{cond}_2(\mathbf{T}) < \frac{4}{\pi^2}h^{-2}. \quad (7.31)$$

Hint: For the upper bound use the inequality $\tan x > x$ valid for $0 < x < \pi/2$. For the lower bound we use (without proof) the inequality $\cot^2 x > \frac{1}{x^2} - \frac{2}{3}$ for $x > 0$.

- d) Show (7.29).

7.4 Proof that the p -Norms are Norms

We want to show

Theorem 7.37 (The p vector norms are norms)

Let for $1 \leq p \leq \infty$ and $\mathbf{x} \in \mathbb{C}^n$

$$\|\mathbf{x}\|_p := \left(\sum_{j=1}^n |x_j|^p \right)^{1/p}, \quad \|\mathbf{x}\|_\infty := \max_{1 \leq j \leq n} |x_j|.$$

Then for all $1 \leq p \leq \infty$, $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$ and all $a \in \mathbb{C}$

1. $\|\mathbf{x}\|_p \geq 0$ with equality if and only if $\mathbf{x} = \mathbf{0}$. (positivity)
2. $\|a\mathbf{x}\|_p = |a| \|\mathbf{x}\|_p$. (homogeneity)
3. $\|\mathbf{x} + \mathbf{y}\|_p \leq \|\mathbf{x}\|_p + \|\mathbf{y}\|_p$. (subadditivity)

Positivity and homogeneity follows immediately. To show the subadditivity we need some elementary properties of convex functions.

Definition 7.38 (Convex function)

Let $I \subset \mathbb{R}$ be an interval. A function $f : I \rightarrow \mathbb{R}$ is convex if

$$f((1 - \lambda)x_1 + \lambda x_2) \leq (1 - \lambda)f(x_1) + \lambda f(x_2) \quad (7.32)$$

for all $x_1, x_2 \in I$ with $x_1 < x_2$ and all $\lambda \in [0, 1]$. The sum $\sum_{j=1}^n \lambda_j x_j$ is called a **convex combination** of x_1, \dots, x_n if $\lambda_j \geq 0$ for $j = 1, \dots, n$ and $\sum_{j=1}^n \lambda_j = 1$.

The convexity condition is illustrated in Figure 7.1.

Lemma 7.39 (A sufficient condition for convexity)

If $f \in C^2[a, b]$ and $f''(x) \geq 0$ for $x \in [a, b]$ then f is convex.

Proof. We recall the formula for linear interpolation with remainder, (cf a book on numerical methods) For any $a \leq x_1 \leq x \leq x_2 \leq b$ there is a $c \in [x_1, x_2]$ such that

$$\begin{aligned} f(x) &= \frac{x_2 - x}{x_2 - x_1} f(x_1) + \frac{x - x_1}{x_2 - x_1} f(x_2) + (x - x_1)(x - x_2) f''(c)/2 \\ &= (1 - \lambda)f(x_1) + \lambda f(x_2) + (x_2 - x_1)^2 \lambda(\lambda - 1) f''(c)/2, \quad \lambda := \frac{x - x_1}{x_2 - x_1}. \end{aligned}$$

Since $\lambda \in [0, 1]$ the remainder term is not positive. Moreover,

$$x = \frac{x_2 - x}{x_2 - x_1} x_1 + \frac{x - x_1}{x_2 - x_1} x_2 = (1 - \lambda)x_1 + \lambda x_2$$

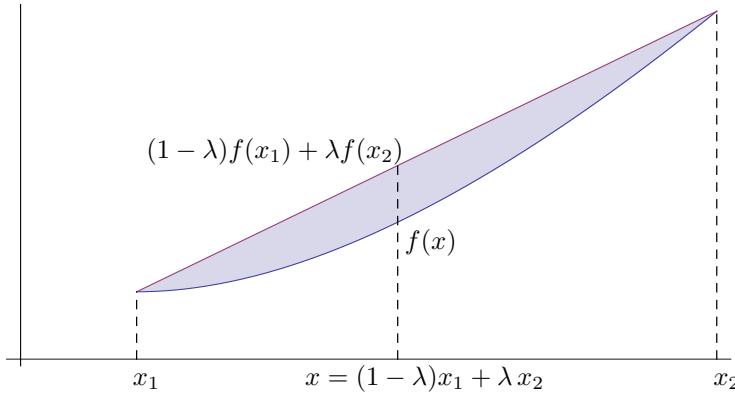


Figure 7.1: A convex function.

so that (7.32) holds, and f is convex. \square

The following inequality is elementary, but can be used to prove many non-trivial inequalities.

Theorem 7.40 (Jensen's inequality)

Suppose $I \in \mathbb{R}$ is an interval and $f : I \rightarrow \mathbb{R}$ is convex. Then for all $n \in \mathbb{N}$, all $\lambda_1, \dots, \lambda_n$ with $\lambda_j \geq 0$ for $j = 1, \dots, n$ and $\sum_{j=1}^n \lambda_j = 1$, and all $z_1, \dots, z_n \in I$

$$f\left(\sum_{j=1}^n \lambda_j z_j\right) \leq \sum_{j=1}^n \lambda_j f(z_j).$$

Proof. We use induction on n . The result is trivial for $n = 1$. Let $n \geq 2$, assume the inequality holds for $n - 1$, and let λ_j, z_j for $j = 1, \dots, n$ be given as in the theorem. Since $n \geq 2$ we have $\lambda_i < 1$ for at least one i so assume without loss of generality that $\lambda_1 < 1$, and define $u := \sum_{j=2}^n \frac{\lambda_j}{1-\lambda_1} z_j$. Since $\sum_{j=2}^n \lambda_j = 1 - \lambda_1$ this is a convex combination of $n - 1$ terms and the induction hypothesis implies that $f(u) \leq \sum_{j=2}^n \frac{\lambda_j}{1-\lambda_1} f(z_j)$. But then by the convexity of f

$$f\left(\sum_{j=1}^n \lambda_j z_j\right) = f(\lambda_1 z_1 + (1 - \lambda_1) u) \leq \lambda_1 f(z_1) + (1 - \lambda_1) f(u) \leq \sum_{j=1}^n \lambda_j f(z_j)$$

and the inequality holds for n . \square

Corollary 7.41 (Weighted geometric/arithmetic mean inequality)

Suppose $\sum_{j=1}^n \lambda_j a_j$ is a convex combination of nonnegative numbers a_1, \dots, a_n . Then

$$a_1^{\lambda_1} a_2^{\lambda_2} \cdots a_n^{\lambda_n} \leq \sum_{j=1}^n \lambda_j a_j, \quad (7.33)$$

where $0^0 := 0$.

Proof. The result is trivial if one or more of the a_j 's are zero so assume $a_j > 0$ for all j . Consider the function $f : (0, \infty) \rightarrow \mathbb{R}$ given by $f(x) = -\log x$. Since $f''(x) = 1/x^2 > 0$ for $x \in (0, \infty)$, this function is convex. By Jensen's inequality

$$-\log \left(\sum_{j=1}^n \lambda_j a_j \right) \leq -\sum_{j=1}^n \lambda_j \log(a_j) = -\log \left(a_1^{\lambda_1} \cdots a_n^{\lambda_n} \right)$$

or $\log \left(a_1^{\lambda_1} \cdots a_n^{\lambda_n} \right) \leq \log \left(\sum_{j=1}^n \lambda_j a_j \right)$. The inequality follows since $\exp(\log x) = x$ for $x > 0$ and the exponential function is monotone increasing. \square

Taking $\lambda_j = \frac{1}{n}$ for all j in (7.33) we obtain the classical **geometric/arithmetic mean inequality**

$$(a_1 a_2 \cdots a_n)^{\frac{1}{n}} \leq \frac{1}{n} \sum_{j=1}^n a_j. \quad (7.34)$$

Corollary 7.42 (Hölder's inequality)

For $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$ and $1 \leq p \leq \infty$

$$\sum_{j=1}^n |x_j y_j| \leq \|\mathbf{x}\|_p \|\mathbf{y}\|_q, \text{ where } \frac{1}{p} + \frac{1}{q} = 1.$$

Proof. We leave the proof for $p = 1$ and $p = \infty$ as an exercise so assume $1 < p < \infty$. For any $a, b \geq 0$ the weighted arithmetic/geometric mean inequality implies that

$$a^{\frac{1}{p}} b^{\frac{1}{q}} \leq \frac{1}{p} a + \frac{1}{q} b, \text{ where } \frac{1}{p} + \frac{1}{q} = 1. \quad (7.35)$$

If $\mathbf{x} = \mathbf{0}$ or $\mathbf{y} = \mathbf{0}$ there is nothing to prove so assume that both \mathbf{x} and \mathbf{y} are nonzero. Using 7.35 on each term we obtain

$$\frac{1}{\|\mathbf{x}\|_p \|\mathbf{y}\|_q} \sum_{j=1}^n |x_j y_j| = \sum_{j=1}^n \left(\frac{|x_j|^p}{\|\mathbf{x}\|_p^p} \right)^{\frac{1}{p}} \left(\frac{|y_j|^q}{\|\mathbf{y}\|_q^q} \right)^{\frac{1}{q}} \leq \sum_{j=1}^n \left(\frac{1}{p} \frac{|x_j|^p}{\|\mathbf{x}\|_p^p} + \frac{1}{q} \frac{|y_j|^q}{\|\mathbf{y}\|_q^q} \right) = 1$$

and the proof of the inequality is complete. \square

Corollary 7.43 (Minkowski's inequality)

For $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$ and $1 \leq p \leq \infty$

$$\|\mathbf{x} + \mathbf{y}\|_p \leq \|\mathbf{x}\|_p + \|\mathbf{y}\|_p.$$

Proof. We leave the proof for $p = 1$ and $p = \infty$ as an exercise so assume $1 < p < \infty$. We write

$$\|\mathbf{x} + \mathbf{y}\|_p^p = \sum_{j=1}^n |x_j + y_j|^p \leq \sum_{j=1}^n |x_j| |x_j + y_j|^{p-1} + \sum_{j=1}^n |y_j| |x_j + y_j|^{p-1}.$$

We apply Hölder's inequality with exponent p and q to each sum. In view of the relation $(p-1)q = p$ the result is

$$\|\mathbf{x} + \mathbf{y}\|_p^p \leq \|\mathbf{x}\|_p \|\mathbf{x} + \mathbf{y}\|_p^{p/q} + \|\mathbf{y}\|_p \|\mathbf{x} + \mathbf{y}\|_p^{p/q} = (\|\mathbf{x}\|_p + \|\mathbf{y}\|_p) \|\mathbf{x} + \mathbf{y}\|_p^{p-1},$$

and canceling the common factor, the inequality follows. \square

It is possible to characterize the p -norms that are derived from an inner product. We start with the following identity.

Theorem 7.44 (Parallelogram identity)

For all \mathbf{x}, \mathbf{y} in a real or complex inner product space

$$\|\mathbf{x} + \mathbf{y}\|^2 + \|\mathbf{x} - \mathbf{y}\|^2 = 2\|\mathbf{x}\|^2 + 2\|\mathbf{y}\|^2. \quad (7.36)$$

Proof. We set $a = \pm 1$ in (4.5) and add the two equations. \square

Theorem 7.45 (When is a norm an inner product norm?)

To a given norm on a real or complex vector space \mathcal{V} there exists an inner product on \mathcal{V} such that $\langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|^2$ if and only if the parallelogram identity (7.36) holds for all $\mathbf{x}, \mathbf{y} \in \mathcal{V}$.

Proof. If $\langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|^2$ then Theorem 7.44 shows that the parallelogram identity holds. For the converse we prove the real case and leave the complex case as an exercise. Suppose (7.36) holds for all \mathbf{x}, \mathbf{y} in the real vector space \mathcal{V} . We show that

$$\langle \mathbf{x}, \mathbf{y} \rangle := \frac{1}{4} (\|\mathbf{x} + \mathbf{y}\|^2 - \|\mathbf{x} - \mathbf{y}\|^2), \quad \mathbf{x}, \mathbf{y} \in \mathcal{V} \quad (7.37)$$

defines an inner product on \mathcal{V} . Clearly 1. and 2. in Definition 4.1 hold. The hard part is to show 3. We need to show that

$$\langle \mathbf{x}, \mathbf{z} \rangle + \langle \mathbf{y}, \mathbf{z} \rangle = \langle \mathbf{x} + \mathbf{y}, \mathbf{z} \rangle, \quad \mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{V}, \quad (7.38)$$

$$\langle a\mathbf{x}, \mathbf{y} \rangle = a\langle \mathbf{x}, \mathbf{y} \rangle, \quad a \in \mathbb{R}, \quad \mathbf{x}, \mathbf{y} \in \mathcal{V}. \quad (7.39)$$

Now

$$\begin{aligned} 4\langle \mathbf{x}, \mathbf{z} \rangle + 4\langle \mathbf{y}, \mathbf{z} \rangle &\stackrel{(7.37)}{=} \|\mathbf{x} + \mathbf{z}\|^2 - \|\mathbf{x} - \mathbf{z}\|^2 + \|\mathbf{y} + \mathbf{z}\|^2 - \|\mathbf{y} - \mathbf{z}\|^2 \\ &= \left\| \left(\mathbf{z} + \frac{\mathbf{x} + \mathbf{y}}{2} \right) + \frac{\mathbf{x} - \mathbf{y}}{2} \right\|^2 - \left\| \left(\mathbf{z} - \frac{\mathbf{x} + \mathbf{y}}{2} \right) + \frac{\mathbf{y} - \mathbf{x}}{2} \right\|^2 \\ &\quad + \left\| \left(\mathbf{z} + \frac{\mathbf{x} + \mathbf{y}}{2} \right) - \frac{\mathbf{x} - \mathbf{y}}{2} \right\|^2 - \left\| \left(\mathbf{z} - \frac{\mathbf{x} + \mathbf{y}}{2} \right) - \frac{\mathbf{y} - \mathbf{x}}{2} \right\|^2 \\ &\stackrel{(7.36)}{=} 2\left\| \mathbf{z} + \frac{\mathbf{x} + \mathbf{y}}{2} \right\|^2 + 2\left\| \frac{\mathbf{x} - \mathbf{y}}{2} \right\|^2 - 2\left\| \mathbf{z} - \frac{\mathbf{x} + \mathbf{y}}{2} \right\|^2 - 2\left\| \frac{\mathbf{y} - \mathbf{x}}{2} \right\|^2 \\ &\stackrel{(7.37)}{=} 8\left\langle \frac{\mathbf{x} + \mathbf{y}}{2}, \mathbf{z} \right\rangle, \end{aligned}$$

or

$$\langle \mathbf{x}, \mathbf{z} \rangle + \langle \mathbf{y}, \mathbf{z} \rangle = 2\left\langle \frac{\mathbf{x} + \mathbf{y}}{2}, \mathbf{z} \right\rangle, \quad \mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{V}.$$

In particular, since $\mathbf{y} = \mathbf{0}$ implies $\langle \mathbf{y}, \mathbf{z} \rangle = 0$ we obtain $\langle \mathbf{x}, \mathbf{z} \rangle = 2\langle \frac{\mathbf{x}}{2}, \mathbf{z} \rangle$ for all $\mathbf{x}, \mathbf{z} \in \mathcal{V}$. This means that $2\langle \frac{\mathbf{x} + \mathbf{y}}{2}, \mathbf{z} \rangle = \langle \mathbf{x} + \mathbf{y}, \mathbf{z} \rangle$ for all $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{V}$ and (7.38) follows.

We first show (7.39) when $a = n$ is a positive integer. By induction

$$\langle n\mathbf{x}, \mathbf{y} \rangle = \langle (n-1)\mathbf{x} + \mathbf{x}, \mathbf{y} \rangle \stackrel{(7.38)}{=} \langle (n-1)\mathbf{x}, \mathbf{y} \rangle + \langle \mathbf{x}, \mathbf{y} \rangle = n\langle \mathbf{x}, \mathbf{y} \rangle. \quad (7.40)$$

If $m, n \in \mathbb{N}$ then

$$m^2\left\langle \frac{n}{m}\mathbf{x}, \mathbf{y} \right\rangle \stackrel{(7.40)}{=} m\langle n\mathbf{x}, \mathbf{y} \rangle \stackrel{(7.40)}{=} mn\langle \mathbf{x}, \mathbf{y} \rangle,$$

implying that (7.39) holds for positive rational numbers

$$\left\langle \frac{n}{m}\mathbf{x}, \mathbf{y} \right\rangle = \frac{n}{m}\langle \mathbf{x}, \mathbf{y} \rangle.$$

Now if $a > 0$ there is a sequence $\{a_n\}$ of positive rational numbers converging to a . For each n

$$a_n\langle \mathbf{x}, \mathbf{y} \rangle = \langle a_n\mathbf{x}, \mathbf{y} \rangle \stackrel{(7.37)}{=} \frac{1}{4}(\|a_n\mathbf{x} + \mathbf{y}\|^2 - \|a_n\mathbf{x} - \mathbf{y}\|^2).$$

Taking limits and using continuity of norms we obtain $a\langle \mathbf{x}, \mathbf{y} \rangle = \langle a\mathbf{x}, \mathbf{y} \rangle$. This also holds for $a = 0$. Finally, if $a < 0$ then $(-a) > 0$ and from what we just showed

$$(-a)\langle \mathbf{x}, \mathbf{y} \rangle = \langle (-a)\mathbf{x}, \mathbf{y} \rangle \stackrel{(7.37)}{=} \frac{1}{4}(\|-a\mathbf{x} + \mathbf{y}\|^2 - \|-a\mathbf{x} - \mathbf{y}\|^2) = -\langle a\mathbf{x}, \mathbf{y} \rangle,$$

so (7.39) also holds for negative a . \square

Corollary 7.46 (Are the p -norms inner product norms?)

For the p vector norms on $\mathcal{V} = \mathbb{R}^n$ or $\mathcal{V} = \mathbb{C}^n$, $1 \leq p \leq \infty$, $n \geq 2$, there is an inner product on \mathcal{V} such that $\langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|_p^2$ for all $\mathbf{x} \in \mathcal{V}$ if and only if $p = 2$.

Proof. For $p = 2$ the p -norm is the Euclidian norm which corresponds to the standard inner product. If $p \neq 2$ then the parallelogram identity (7.36) does not hold for say $\mathbf{x} := \mathbf{e}_1$ and $\mathbf{y} := \mathbf{e}_2$. \square

Exercise 7.47 (When is a complex norm an inner product norm?)

Given a vector norm in a complex vector space \mathcal{V} , and suppose (7.36) holds for all \mathbf{x}, \mathbf{y} . Show that

$$\langle \mathbf{x}, \mathbf{y} \rangle := \frac{1}{4} (\|\mathbf{x} + \mathbf{y}\|^2 - \|\mathbf{x} - \mathbf{y}\|^2 + i\|\mathbf{x} + i\mathbf{y}\|^2 - i\|\mathbf{x} - i\mathbf{y}\|^2), \quad (7.41)$$

defines an inner product on \mathcal{V} , where $i = \sqrt{-1}$. The identity (7.41) is called the polarization identity.¹⁶

Exercise 7.48 (p norm for $p = 1$ and $p = \infty$)

Show that $\|\cdot\|_p$ is a vector norm in \mathbb{R}^n for $p = 1, p = \infty$.

Exercise 7.49 (The p - norm unit sphere)

The set

$$S_p = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|_p = 1\}$$

is called the unit sphere in \mathbb{R}^n with respect to p . Draw S_p for $p = 1, 2, \infty$ for $n = 2$.

Exercise 7.50 (Sharpness of p -norm inequality)

For $p \geq 1$, and any $\mathbf{x} \in \mathbb{C}^n$ we have $\|\mathbf{x}\|_\infty \leq \|\mathbf{x}\|_p \leq n^{1/p} \|\mathbf{x}\|_\infty$ (cf. (7.5)).

Produce a vector \mathbf{x}_l such that $\|\mathbf{x}_l\|_\infty = \|\mathbf{x}_l\|_p$ and another vector \mathbf{x}_u such that $\|\mathbf{x}_u\|_p = n^{1/p} \|\mathbf{x}_u\|_\infty$. Thus, these inequalities are sharp.

Exercise 7.51 (p -norm inequalities for arbitrary p)

If $1 \leq q \leq p \leq \infty$ then

$$\|\mathbf{x}\|_p \leq \|\mathbf{x}\|_q \leq n^{1/q-1/p} \|\mathbf{x}\|_p, \quad \mathbf{x} \in \mathbb{C}^n.$$

Hint: For the rightmost inequality use Jensen's inequality Cf. Theorem 7.40 with $f(z) = z^{p/q}$ and $z_i = |x_i|^q$. For the left inequality consider first $y_i = x_i/\|\mathbf{x}\|_\infty$, $i = 1, 2, \dots, n$.

¹⁶Hint: We have $\langle \mathbf{x}, \mathbf{y} \rangle = s(\mathbf{x}, \mathbf{y}) + is(\mathbf{x}, i\mathbf{y})$, where $s(\mathbf{x}, \mathbf{y}) := \frac{1}{4} (\|\mathbf{x} + \mathbf{y}\|^2 - \|\mathbf{x} - \mathbf{y}\|^2)$.

7.5 Review Questions

7.5.1]

- What is a consistent matrix norm?
- what is a subordinate matrix norm?
- is an operator norm consistent?
- why is the Frobenius norm not an operator norm?
- what is the spectral norm of a matrix?
- how do we compute $\|A\|_\infty$?
- what is the spectral condition number of a symmetric positive definite matrix?

7.5.2 Why is $\|A\|_2 \leq \|A\|_F$ for any matrix A ?

7.5.3 What is the spectral norm of the inverse of a normal matrix?

Chapter 8

Least Squares

Consider the linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ of m equations in n unknowns. It is over-determined, if $m > n$, square, if $m = n$, and underdetermined, if $m < n$. In either case the system can only be solved approximately if $\mathbf{b} \notin \text{span}(\mathbf{A})$. One way to solve $\mathbf{A}\mathbf{x} = \mathbf{b}$ approximately is to select a vector norm $\|\cdot\|$, say a p -norm, and look for $\mathbf{x} \in \mathbb{C}^n$ which minimizes $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|$. The use of the one and ∞ norm can be formulated as linear programming problems, while the Euclidian norm leads to a linear system and has applications in statistics. Only this norm is considered here.

Definition 8.1 (Least squares problem (LSQ))

Suppose $m, n \in \mathbb{N}$, $\mathbf{A} \in \mathbb{C}^{m \times n}$ and $\mathbf{b} \in \mathbb{C}^m$. To find $\mathbf{x} \in \mathbb{C}^n$ that minimizes $E : \mathbb{C}^n \rightarrow \mathbb{R}$ given by

$$E(\mathbf{x}) := \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2,$$

is called the **least squares problem**. A minimizer \mathbf{x} is called a **least squares solution**.

Since the square root function is monotone, minimizing $E(\mathbf{x})$ or $\sqrt{E(\mathbf{x})}$ is equivalent.

Example 8.2 (Average)

Consider an overdetermined linear system of 3 equations in one unknown

$$\begin{aligned} x_1 &= 1 \\ x_1 &= 1, \quad \mathbf{A} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \quad \mathbf{x} = [x_1], \quad \mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}. \end{aligned}$$

To solve this as a least squares problem we find

$$\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 = (x_1 - 1)^2 + (x_1 - 1)^2 + (x_1 - 2)^2 = 3x_1^2 - 8x_1 + 6.$$

Setting the first derivative with respect to x_1 equal to zero we obtain $6x_1 - 8 = 0$ or $x_1 = 4/3$, the average of b_1, b_2, b_3 . The second derivative is positive and $x_1 = 4/3$ is a global minimum.

We will show below the following results valid for any $m, n \in \mathbb{N}$, $\mathbf{A} \in \mathbb{C}^{m \times n}$ and $\mathbf{b} \in \mathbb{C}^n$.

Theorem 8.3 (Existence)

The least squares problem always has a solution.

Theorem 8.4 (Uniqueness)

The solution of the least squares problem is unique if and only if \mathbf{A} has linearly independent columns.

Theorem 8.5 (Characterization)

$\mathbf{x} \in \mathbb{C}^n$ is a solution of the least squares problem if and only if $\mathbf{A}^ \mathbf{A}\mathbf{x} = \mathbf{A}^*\mathbf{b}$.*

The linear system $\mathbf{A}^* \mathbf{A}\mathbf{x} = \mathbf{A}^*\mathbf{b}$ is known as the **normal equations**. By Lemma 3.8 the coefficient matrix $\mathbf{A}^* \mathbf{A}$ is symmetric and positive semidefinite, and it is positive definite if and only if \mathbf{A} has linearly independent columns. This is the same condition which guarantees that the least squares problem has a unique solution.

8.1 Examples

Example 8.6 (Linear regression)

We want to fit a straight line $p(t) = x_1 + x_2 t$ to $m \geq 2$ given data $(t_k, y_k) \in \mathbb{R}^2$, $k = 1, \dots, m$. This is part of the linear regression process in statistics. We obtain the linear system

$$\mathbf{A}\mathbf{x} = \begin{bmatrix} p(t_1) \\ \vdots \\ p(t_m) \end{bmatrix} = \begin{bmatrix} 1 & t_1 \\ \vdots & \vdots \\ 1 & t_m \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} = \mathbf{b}.$$

This is square for $m = 2$ and overdetermined for $m > 2$. The matrix \mathbf{A} has linearly independent columns if and only if the set $\{t_1, \dots, t_m\}$ of sites contains at least two distinct elements. For if say $t_i \neq t_j$ then

$$c_1 \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} + c_2 \begin{bmatrix} t_1 \\ \vdots \\ t_m \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \implies \begin{bmatrix} 1 & t_i \\ 1 & t_j \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \implies c_1 = c_2 = 0.$$

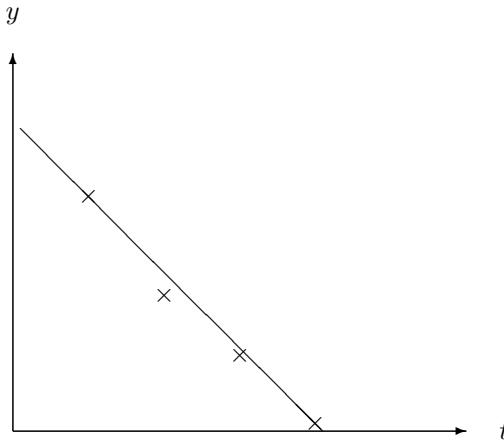


Figure 8.1: A least squares fit to data.

Conversely, if $t_1 = \dots = t_m$ then the columns of \mathbf{A} are not linearly independent. The normal equations are

$$\begin{aligned} \mathbf{A}^T \mathbf{A} \mathbf{x} &= \begin{bmatrix} 1 & \cdots & 1 \\ t_1 & \cdots & t_m \end{bmatrix} \begin{bmatrix} 1 & t_1 \\ \vdots & \\ 1 & t_m \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} m & \sum t_k \\ \sum t_k & \sum t_k^2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \\ &= \begin{bmatrix} 1 & \cdots & 1 \\ t_1 & \cdots & t_m \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} \sum y_k \\ \sum t_k y_k \end{bmatrix} = \mathbf{A}^T \mathbf{b}, \end{aligned}$$

where k ranges from 1 to m in the sums. By what we showed the coefficient matrix is positive semidefinite and positive definite if we at least two distinct cities. If $m = 2$ and $t_1 \neq t_2$ then both systems $\mathbf{A}\mathbf{x} = \mathbf{b}$ and $\mathbf{A}^T \mathbf{A}\mathbf{x} = \mathbf{A}^T \mathbf{b}$ are square, and p is the linear interpolant to the data. Indeed, p is linear and $p(t_k) = y_k$, $k = 1, 2$.

With the data

t	1.0	2.0	3.0	4.0
y	3.1	1.8	1.0	0.1

the normal equations become $\begin{bmatrix} 4 & 10 \\ 10 & 30 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 6 \\ 10.1 \end{bmatrix}$. The data and the least squares polynomial $p(t) = x_1 + x_2 t = 3.95 - 0.98t$ are shown in Figure 8.1.

Example 8.7 (Input/output model)

Suppose we have a simple input/output model. To every input $\mathbf{u} \in \mathbb{R}^n$ we obtain an output $y \in \mathbb{R}$. Assuming we have a linear relation

$$y = \mathbf{u}^T \mathbf{x} = \sum_{i=1}^n u_i x_i,$$

between \mathbf{u} and y , how can we determine \mathbf{x} ?

Performing $m \geq n$ experiments we obtain a table of values

\mathbf{u}	\mathbf{u}_1	\mathbf{u}_2	\dots	\mathbf{u}_m	.
y	y_1	y_2	\dots	y_m	

We would like to find \mathbf{x} such that

$$\mathbf{A}\mathbf{x} = \begin{bmatrix} \mathbf{u}_1^T \\ \mathbf{u}_2^T \\ \vdots \\ \mathbf{u}_m^T \end{bmatrix} \mathbf{x} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} = \mathbf{b}.$$

We can estimate \mathbf{x} by solving the least squares problem $\min \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$.

8.1.1 Curve Fitting

Given

- size: $1 \leq n \leq m$,
- sites: $\mathcal{S} := \{t_1, t_2, \dots, t_m\} \subset [a, b]$,
- y -values: $\mathbf{y} = [y_1, y_2, \dots, y_m]^T \in \mathbb{R}^m$,
- functions: $\phi_j : [a, b] \rightarrow \mathbb{R}$, $j = 1, \dots, n$.

Find a function (curve fit) $p : [a, b] \rightarrow \mathbb{R}$ given by $p := \sum_{j=1}^n x_j \phi_j$ such that $p(t_k) \approx y_k$ for $k = 1, \dots, m$.

The curve fitting problem can be solved by least squares from the following linear system:

$$\mathbf{A}\mathbf{x} = \begin{bmatrix} p(t_1) \\ \vdots \\ p(t_m) \end{bmatrix} = \begin{bmatrix} \phi_1(t_1) & \cdots & \phi_n(t_1) \\ \vdots & & \vdots \\ \phi_1(t_m) & \cdots & \phi_n(t_m) \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} =: \mathbf{b}. \quad (8.1)$$

Then we find $\mathbf{x} \in \mathbb{R}^n$ as a solution of the corresponding least squares problem given by

$$E(\mathbf{x}) := \|\mathbf{Ax} - \mathbf{b}\|_2^2 = \sum_{k=1}^m \left(\sum_{j=1}^n x_j \phi_j(t_k) - y_k \right)^2. \quad (8.2)$$

Typical examples of functions ϕ_j are polynomials, trigonometric functions, exponential functions, or splines.

In (8.2) one can also include **weights** $w_k > 0$ for $k = 1, \dots, m$ and minimize

$$E(\mathbf{x}) := \sum_{k=1}^m w_k \left(\sum_{j=1}^n x_j \phi_j(t_k) - y_k \right)^2.$$

If y_k is an accurate observation, we can choose a large weight w_k . This will force $p(t_k) - y_k$ to be small. Similarly, a small w_k will allow $p(t_k) - y_k$ to be large. If an estimate for the standard deviation δy_k in y_k is known for each k , we can choose $w_k = 1/(\delta y_k)^2$, $k = 1, 2, \dots, m$. For simplicity we will assume in the following that $w_k = 1$ for all k .

Lemma 8.8 (Curve fitting)

Let \mathbf{A} be given by (8.1). The matrix $\mathbf{A}^T \mathbf{A}$ is symmetric positive definite if and only if $\{\phi_1, \dots, \phi_n\}$ is linearly independent on \mathcal{S} , i.e.,

$$p(t_k) := \sum_{j=1}^n x_j \phi_j(t_k) = 0, \quad k = 1, \dots, m \Rightarrow x_1 = \dots = x_n = 0. \quad (8.3)$$

Proof. By Lemma 3.8 $\mathbf{A}^T \mathbf{A}$ is positive definite if and only if \mathbf{A} has linearly independent columns. Since $(\mathbf{Ax})_k = \sum_{j=1}^n x_j \phi_j(t_k)$, $k = 1, \dots, m$ this is equivalent to (8.3). \square

Example 8.9 (Ill conditioning and the Hilbert matrix)

The normal equations can be extremely ill-conditioned. Consider the curve fitting problem using the polynomials $\phi_j(t) := t^{j-1}$, for $j = 1, \dots, n$ and equidistant sites $t_k = (k-1)/(m-1)$ for $k = 1, \dots, m$. The normal equations are $\mathbf{B}_n \mathbf{x} = \mathbf{c}_n$, where for $n = 3$

$$\mathbf{B}_3 \mathbf{x} := \begin{bmatrix} m & \sum t_k & \sum t_k^2 \\ \sum t_k & \sum t_k^2 & \sum t_k^3 \\ \sum t_k^2 & \sum t_k^3 & \sum t_k^4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} \sum y_k \\ \sum t_k y_k \\ \sum t_k^2 y_k \end{bmatrix}.$$

\mathbf{B}_n is symmetric positive definite if at least n of the t 's are distinct. However \mathbf{B}_n is extremely ill-conditioned even for moderate n . Indeed, $\frac{1}{m} \mathbf{B}_n \approx \mathbf{H}_n$, where

$\mathbf{H}_n \in \mathbb{R}^{n \times n}$ is the **Hilbert Matrix** with i, j element $1/(i + j - 1)$. Thus for $n = 3$

$$\mathbf{H}_3 = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \end{bmatrix}.$$

The elements of $\frac{1}{m}\mathbf{B}_n$ are Riemann sums approximations to the elements of \mathbf{H}_n . In fact, if $\mathbf{B}_n = [b_{i,j}]_{i,j=1}^n$ then

$$\frac{1}{m}b_{i,j} = \frac{1}{m} \sum_{k=1}^m t_k^{i+j-2} = \frac{1}{m} \sum_{k=1}^m \left(\frac{k-1}{m-1} \right)^{i+j-2} \approx \int_0^1 x^{i+j-2} dx = \frac{1}{i+j-1} = h_{i,j}.$$

The elements of \mathbf{H}_n^{-1} are determined in Exercise 0.35. We find $K_1(\mathbf{H}_6) := \|\mathbf{H}_6\|_1 \|\mathbf{H}_6^{-1}\|_1 \approx 3 \cdot 10^7$. It appears that $\frac{1}{m}\mathbf{B}_n$ and hence \mathbf{B}_n is ill-conditioned for moderate n at least if m is large. The cure for this problem is to use a different basis for polynomials. Orthogonal polynomials are an excellent choice. Another possibility is to use the shifted power basis $(t - \tilde{t})^{j-1}$, $j = 1, \dots, n$, for a suitable \tilde{t} .

Exercise 8.10 (Fitting a circle to points)

In this problem we derive an algorithm to fit a circle $(t - c_1)^2 + (y - c_2)^2 = r^2$ to $m \geq 3$ given points $(t_i, y_i)_{i=1}^m$ in the (t, y) -plane. We obtain the overdetermined system

$$(t_i - c_1)^2 + (y_i - c_2)^2 = r^2, \quad i = 1, \dots, m, \quad (8.4)$$

of m equations in the three unknowns c_1, c_2 and r . This system is nonlinear, but it can be solved from the linear system

$$t_i x_1 + y_i x_2 + x_3 = t_i^2 + y_i^2, \quad i = 1, \dots, m, \quad (8.5)$$

and then setting $c_1 = x_1/2$, $c_2 = x_2/2$ and $r^2 = c_1^2 + c_2^2 + x_3$.

- a) Derive (8.5) from (8.4). Explain how we can find c_1, c_2, r once $[x_1, x_2, x_3]$ is determined.
- b) Formulate (8.5) as a linear least squares problem for suitable \mathbf{A} and \mathbf{b} .
- c) Does the matrix \mathbf{A} in b) have linearly independent columns?
- d) Use (8.5) to find the circle passing through the three points $(1, 4), (3, 2), (1, 0)$.

8.2 Geometric Least Squares theory

The least squares problem can be studied as a quadratic minimization problem. In the real case we have

$$E(\mathbf{x}) := \|\mathbf{Ax} - \mathbf{b}\|_2^2 = (\mathbf{Ax} - \mathbf{b})^T (\mathbf{Ax} - \mathbf{b}) = \mathbf{x}^T \mathbf{A}^T \mathbf{Ax} - 2\mathbf{x}^T \mathbf{A}^T \mathbf{b} + \mathbf{b}^T \mathbf{b}.$$

Minimization of a quadratic function like $E(\mathbf{x})$ will be considered in Chapter 12. Here we consider a geometric approach based on orthogonal sums of subspaces.

8.2.1 Sum of subspaces and orthogonal projections

Suppose \mathcal{S} and \mathcal{T} are subspaces of \mathbb{R}^n or \mathbb{C}^n endowed with the usual inner product $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{y}^* \mathbf{x}$.¹⁷ The following subsets are subspaces of \mathbb{R}^n or \mathbb{C}^n .

- **Sum:** $\mathcal{S} + \mathcal{T} := \{\mathbf{s} + \mathbf{t} : \mathbf{s} \in \mathcal{S} \text{ and } \mathbf{t} \in \mathcal{T}\}$,
- **direct sum** $\mathcal{S} \oplus \mathcal{T}$: a sum where $\mathcal{S} \cap \mathcal{T} = \{\mathbf{0}\}$,
- **orthogonal sum** $\mathcal{S} \overset{\perp}{\oplus} \mathcal{T}$: a sum where $\langle \mathbf{s}, \mathbf{t} \rangle = 0$ for all $\mathbf{s} \in \mathcal{S}$ and $\mathbf{t} \in \mathcal{T}$.

We note that

- Every $\mathbf{x} \in \mathcal{S} \oplus \mathcal{T}$ can be decomposed uniquely in the form $\mathbf{x} = \mathbf{s} + \mathbf{t}$, where $\mathbf{s} \in \mathcal{S}$ and $\mathbf{t} \in \mathcal{T}$. For if $\mathbf{x} = \mathbf{s}_1 + \mathbf{t}_1 = \mathbf{s}_2 + \mathbf{t}_2$ for $\mathbf{s}_1, \mathbf{s}_2 \in \mathcal{S}$ and $\mathbf{t}_1, \mathbf{t}_2 \in \mathcal{T}$, then $\mathbf{s}_1 - \mathbf{s}_2 = \mathbf{t}_2 - \mathbf{t}_1$ and it follows that $\mathbf{s}_1 - \mathbf{s}_2$ and $\mathbf{t}_2 - \mathbf{t}_1$ belong to both \mathcal{S} and \mathcal{T} and hence to $\mathcal{S} \cap \mathcal{T}$. But then $\mathbf{s}_1 - \mathbf{s}_2 = \mathbf{t}_2 - \mathbf{t}_1 = \mathbf{0}$ so $\mathbf{s}_1 = \mathbf{s}_2$ and $\mathbf{t}_2 = \mathbf{t}_1$.
- An orthogonal sum is a direct sum. For if $\mathbf{b} \in \mathcal{S} \cap \mathcal{T}$ then \mathbf{b} is orthogonal to itself, $\langle \mathbf{b}, \mathbf{b} \rangle = 0$, which implies that $\mathbf{b} = \mathbf{0}$.
- Thus, every $\mathbf{b} \in \mathcal{S} \overset{\perp}{\oplus} \mathcal{T}$ can be written uniquely as $\mathbf{b} = \mathbf{b}_1 + \mathbf{b}_2$, where $\mathbf{b}_1 \in \mathcal{S}$ and $\mathbf{b}_2 \in \mathcal{T}$. The vectors \mathbf{b}_1 and \mathbf{b}_2 are called the **orthogonal projections** of \mathbf{b} into \mathcal{S} and \mathcal{T} . For any $\mathbf{s} \in \mathcal{S}$ we have $\langle \mathbf{b} - \mathbf{b}_1, \mathbf{s} \rangle = \langle \mathbf{b}_2, \mathbf{s} \rangle = 0$, see Figure 8.2.

Using orthogonal sums and projections we can prove the existence, uniqueness and characterization theorems for least squares problems. For $\mathbf{A} \in \mathbb{C}^{m \times n}$ we consider the column space of \mathbf{A} and the null space of \mathbf{A}^*

$$\mathcal{S} := \text{span}(\mathbf{A}), \quad \mathcal{T} := \ker(\mathbf{A}^*).$$

¹⁷other inner products could be used as well

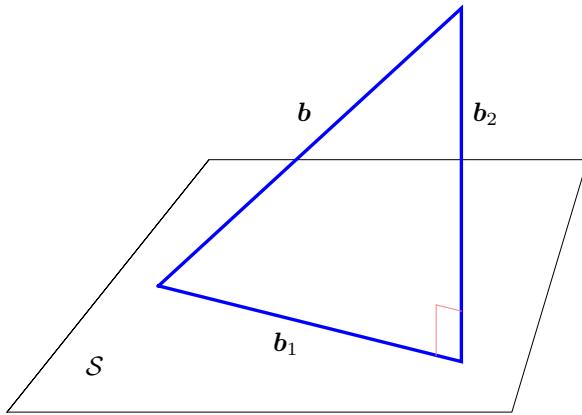


Figure 8.2: The orthogonal projection of \mathbf{b} into \mathcal{S} .

These are subspaces and the sum is an orthogonal sum $\mathcal{S} + \mathcal{T} = \mathcal{S} \overset{\perp}{\oplus} \mathcal{T}$. For if $\mathbf{s} \in \text{span}(\mathbf{A})$ then $\mathbf{s} = \mathbf{Ac}$ for some $\mathbf{c} \in \mathbb{R}^n$ and if $\mathbf{t} \in \ker(\mathbf{A}^*)$ then $\langle \mathbf{t}, \mathbf{s} \rangle = \mathbf{s}^* \mathbf{t} = \mathbf{c}^* \mathbf{A}^* \mathbf{t} = 0$. Moreover, (cf, Theorem 8.13) $\mathcal{S} + \mathcal{T}$ equals all of \mathbb{C}^m

$$\mathbb{C}^m = \text{span}(\mathbf{A}) \overset{\perp}{\oplus} \ker(\mathbf{A}^*).$$

It follows that any $\mathbf{b} \in \mathbb{C}^m$ can be decomposed uniquely as $\mathbf{b} = \mathbf{b}_1 + \mathbf{b}_2$, where \mathbf{b}_1 is the orthogonal projection of \mathbf{b} into $\text{span}(\mathbf{A})$ and \mathbf{b}_2 is the orthogonal projection of \mathbf{b} into $\ker(\mathbf{A}^*)$.

Proof of Theorem 8.3

Suppose $\mathbf{x} \in \mathbb{C}^n$. Clearly $\mathbf{Ax} - \mathbf{b}_1 \in \text{span}(\mathbf{A})$ since it is a subspace and $\mathbf{b}_2 \in \ker(\mathbf{A}^*)$. But then $\langle \mathbf{Ax} - \mathbf{b}_1, \mathbf{b}_2 \rangle = 0$ and by Pythagoras

$$\|\mathbf{Ax} - \mathbf{b}\|_2^2 = \|(\mathbf{Ax} - \mathbf{b}_1) - \mathbf{b}_2\|_2^2 = \|\mathbf{Ax} - \mathbf{b}_1\|_2^2 + \|\mathbf{b}_2\|_2^2 \geq \|\mathbf{b}_2\|_2^2$$

with equality if and only if $\mathbf{Ax} = \mathbf{b}_1$. It follows that the set of all least squares solutions is

$$\{\mathbf{x} \in \mathbb{C}^n : \mathbf{Ax} = \mathbf{b}_1\}. \quad (8.6)$$

This set is nonempty since $\mathbf{b}_1 \in \text{span}(\mathbf{A})$.

Proof of Theorem 8.4

The set (8.6) contains exactly one element if and only if \mathbf{A} has linearly independent columns.

Proof of Theorem 8.5

If \mathbf{x} solves the least squares problem then $\mathbf{Ax} - \mathbf{b}_1 = \mathbf{0}$ and it follows that $\mathbf{A}^*(\mathbf{Ax} -$

$\mathbf{b}_1) = \mathbf{0}$ showing that the normal equations hold. Conversely, if $\mathbf{A}^* \mathbf{A}x = \mathbf{A}^* \mathbf{b}$ then $\mathbf{A}^* \mathbf{b}_2 = \mathbf{0}$ implies that $\mathbf{A}^*(\mathbf{A}x - \mathbf{b}_1) = \mathbf{0}$. But then $\mathbf{A}x - \mathbf{b}_1 \in \text{span}(\mathbf{A}) \cap \ker(\mathbf{A}^*)$ showing that $\mathbf{A}x - \mathbf{b}_1 = \mathbf{0}$, and x is a least squares solution.

8.3 Numerical Solution

We assume that $m \geq n$, $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$. Numerical methods can be based on normal equations, QR factorization, or Singular Value Factorization. We discuss each of these approaches in turn. Another possibility is to use an iterative method like the conjugate gradient method (cf. Exercise 12.18).

8.3.1 Normal equations

We assume that $\text{rank}(\mathbf{A}) = n$, i.e., \mathbf{A} has linearly independent columns. The coefficient matrix $\mathbf{B} := \mathbf{A}^* \mathbf{A}$ in the normal equations is symmetric positive definite, and we can solve these equations using the Cholesky factorization of \mathbf{B} . Consider forming the normal equations. We can use either a column oriented (inner product)- or a row oriented (outer product) approach.

$$\begin{aligned} 1. \text{ inner product: } (\mathbf{A}^* \mathbf{A})_{i,j} &= \sum_{k=1}^m \bar{a}_{k,i} a_{k,j}, \quad i, j = 1, \dots, n, \\ (\mathbf{A}^* \mathbf{b})_i &= \sum_{k=1}^m \bar{a}_{k,i} b_k, \quad i = 1, \dots, n, \end{aligned}$$

$$2. \text{ outer product: } \mathbf{A}^* \mathbf{A} = \sum_{k=1}^m \begin{bmatrix} \bar{a}_{k,1} \\ \vdots \\ \bar{a}_{k,n} \end{bmatrix} [a_{k1} \ \cdots \ a_{kn}], \quad \mathbf{A}^* \mathbf{b} = \sum_{k=1}^m \begin{bmatrix} \bar{a}_{k,1} \\ \vdots \\ \bar{a}_{k,n} \end{bmatrix} b_k.$$

The outer product form is suitable for large problems since it uses only one pass through the data importing one row of \mathbf{A} at a time from some separate storage.

Consider the number of operations to find the least squares solution for real data. We need $2m$ arithmetic operations for each inner product. Since \mathbf{B} is symmetric we only need to compute $n(n+1)/2$ such inner products. It follows that \mathbf{B} can be computed in approximately mn^2 arithmetic operations. In conclusion the number of operations are mn^2 to find \mathbf{B} , $2mn$ to find $\mathbf{c} := \mathbf{A}^* \mathbf{b}$, $n^3/3$ to find \mathbf{L} , n^2 to solve $\mathbf{L}^T \mathbf{y} = \mathbf{c}$ and n^2 to solve $\mathbf{L} \mathbf{x} = \mathbf{y}$. If $m \approx n$ it takes $\frac{4}{3}n^3 = 2G_n$ arithmetic operations. If m is much bigger than n the number of operations is approximately mn^2 , the work to compute \mathbf{B} .

Conditioning of \mathbf{A} can be a problem with the normal equation approach. We have

Theorem 8.11 (Spectral condition number of $\mathbf{A}^* \mathbf{A}$)

Suppose $1 \leq n \leq m$ and that $\mathbf{A} \in \mathbb{C}^{m \times n}$ has linearly independent columns. Then

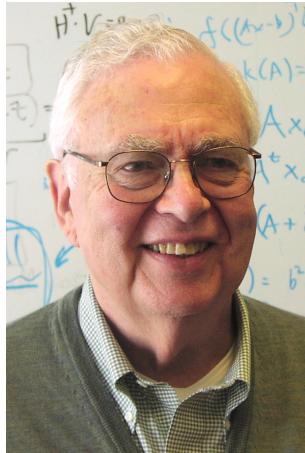
$$K_2(\mathbf{A}^* \mathbf{A}) := \|\mathbf{A}^* \mathbf{A}\|_2 \|(\mathbf{A}^* \mathbf{A})^{-1}\|_2 = \frac{\lambda_1}{\lambda_n} = \frac{\sigma_1^2}{\sigma_n^2} = K_2(\mathbf{A})^2, \quad (8.7)$$

where $\lambda_1 \geq \dots \geq \lambda_n > 0$ are the eigenvalues of $\mathbf{A}^* \mathbf{A}$, and $\sigma_1 \geq \dots \geq \sigma_n > 0$ are the singular values of \mathbf{A} .

Proof. Since $\mathbf{A}^* \mathbf{A}$ is Hermitian it follows from Theorem 7.30 that $K_2(\mathbf{A}) = \frac{\sigma_1}{\sigma_n}$ and $K_2(\mathbf{A}^* \mathbf{A}) = \frac{\lambda_1}{\lambda_n}$. But $\lambda_i = \sigma_i^2$ by Theorem 6.5 and the proof is complete. \square

It follows from Theorem 8.11 that the 2-norm condition number of $\mathbf{B} := \mathbf{A}^* \mathbf{A}$ is the square of the condition number of \mathbf{A} and therefore can be quite large even if \mathbf{A} is only mildly illconditioned. Another difficulty which can be encountered is that the computed $\mathbf{A}^* \mathbf{A}$ might not be positive definite. See Problem 8.36 for an example.

8.3.2 QR factorization



Gene Golub, 1932-2007. He pioneered use of the QR factorization to solve least square problems.

The QR factorization can be used to solve the least squares problem. We assume that $\text{rank}(\mathbf{A}) = n$, i.e., \mathbf{A} has linearly independent columns. Suppose $\mathbf{A} = \mathbf{Q}_1 \mathbf{R}_1$ is a QR factorization of \mathbf{A} . Since $\mathbf{Q}_1 \in \mathbb{C}^{m \times n}$ has orthonormal columns we find

$$\mathbf{A}^* \mathbf{A} = \mathbf{R}_1^* \mathbf{Q}_1^* \mathbf{Q}_1 \mathbf{R}_1 = \mathbf{R}_1^* \mathbf{R}_1, \quad \mathbf{A}^* \mathbf{b} = \mathbf{R}_1^* \mathbf{Q}_1^* \mathbf{b}.$$

Since \mathbf{A} has rank n the matrix \mathbf{R}_1^* is nonsingular and can be canceled. Thus

$$\mathbf{A}^* \mathbf{A} \mathbf{x} = \mathbf{A}^* \mathbf{b} \implies \mathbf{R}_1 \mathbf{x} = \mathbf{c}_1, \quad \mathbf{c}_1 := \mathbf{Q}_1^* \mathbf{b}.$$

We can use Householder transformations or Givens rotations to find \mathbf{R}_1 and \mathbf{c}_1 . Consider using the Householder triangulation algorithm Algorithm 4.21. We find $\mathbf{R} = \mathbf{Q}^* \mathbf{A}$ and $\mathbf{c} = \mathbf{Q}^* \mathbf{b}$, where $\mathbf{A} = \mathbf{QR}$ is the QR decomposition of \mathbf{A} . The matrices \mathbf{R}_1 and \mathbf{c}_1 are located in the first n rows of \mathbf{R} and \mathbf{c} . Using also Algorithm 2.7 we have the following method to solve the full rank least squares problem.

1. $[\mathbf{R}, \mathbf{c}] = \text{housetriang}(\mathbf{A}, \mathbf{b})$.
2. $\mathbf{x} = \text{rbacksolve}(\mathbf{R}(1:n, 1:n), \mathbf{c}(1:n), n)$.

Example 8.12 (Solution using QR factorization)

Consider the least squares problem with

$$\mathbf{A} = \begin{bmatrix} 1 & 3 & 1 \\ 1 & 3 & 7 \\ 1 & -1 & -4 \\ 1 & -1 & 2 \end{bmatrix} \quad \text{and } \mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}.$$

This is the matrix in Example 4.23. The least squares solution \mathbf{x} is found by solving the system

$$\begin{bmatrix} 2 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

and we find $\mathbf{x} = [1, 0, 0]^*$.

Using Householder triangulation is a useful alternative to normal equations for solving full rank least squares problems. It can even be extended to rank deficient problems, see [3]. The 2 norm condition number for the system $\mathbf{R}_1 \mathbf{x} = \mathbf{c}_1$ is $K_2(\mathbf{R}_1) = K_2(\mathbf{Q}_1 \mathbf{R}_1) = K_2(\mathbf{A})$, and as discussed in the previous section this is the square root of $K_2(\mathbf{A}^* \mathbf{A})$, the condition number for the normal equations. Thus if \mathbf{A} is mildly ill-conditioned the normal equations can be quite ill-conditioned and solving the normal equations can give inaccurate results. On the other hand Algorithm 4.21 is quite stable.

But using Householder transformations requires more work. The leading term in the number of arithmetic operations in Algorithm 4.21 is approximately $2mn^2 - 2n^3/3$, (cf. (4.14) while the number of arithmetic operations needed to form the normal equations, taking advantage of symmetry is approximately mn^2 . Thus for m much larger than n using Householder triangulation requires twice as many arithmetic operations as the approach based on the normal equations. Also, Householder triangulation have problems taking advantage of the structure in sparse problems.

8.3.3 Least squares and singular value decomposition

Consider a singular value decomposition of \mathbf{A} and the corresponding singular value factorization:

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^* = [\mathbf{U}_1, \mathbf{U}_2] \begin{bmatrix} \Sigma_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^* \\ \mathbf{V}_2^* \end{bmatrix} = \mathbf{U}_1\Sigma_1\mathbf{V}_1^*, \quad \Sigma_1 = \text{diag}(\sigma_1, \dots, \sigma_r),$$

where \mathbf{A} has rank r so that $\sigma_1 \geq \dots \geq \sigma_r > 0$ and $\mathbf{U}_1 = [\mathbf{u}_1, \dots, \mathbf{u}_r]$, $\mathbf{V}_1 = [\mathbf{v}_1, \dots, \mathbf{v}_r]$. Moreover, $\mathbf{U}^*\mathbf{U} = \mathbf{I}$ and $\mathbf{V}^*\mathbf{V} = \mathbf{I}$. We recall (cf. Theorem 6.15)

- the set of columns of \mathbf{U}_1 is an orthonormal basis for the column space $\text{span}(\mathbf{A})$,
- the set of columns of \mathbf{U}_2 is an orthonormal basis for the null space $\ker(\mathbf{A}^*)$,
- the set of columns of \mathbf{V}_2 is an orthonormal basis for the null space $\ker(\mathbf{A})$,

Theorem 8.13 (Orthogonal projection and least squares solution)

Let $\mathbf{A} \in \mathbb{C}^{m \times n}$, $\mathbf{b} \in \mathbb{C}^m$, $\mathcal{S} := \text{span}(\mathbf{A})$ and $\mathcal{T} := \ker(\mathbf{A}^*)$. Then

1. $\mathbb{C}^m = \mathcal{S} \dot{+} \mathcal{T}$ is an **orthogonal decomposition** of \mathbb{C}^m with respect to the usual inner product $\langle \mathbf{s}, \mathbf{t} \rangle = \mathbf{t}^* \mathbf{s}$.
2. If $\mathbf{A} = \mathbf{U}_1\Sigma_1\mathbf{V}_1^*$ is a singular value factorization of \mathbf{A} then the orthogonal projection \mathbf{b}_1 of \mathbf{b} into \mathcal{S} is

$$\mathbf{b}_1 = \mathbf{U}_1\mathbf{U}_1^*\mathbf{b} = \mathbf{A}\mathbf{A}^\dagger\mathbf{b}, \quad \text{where } \mathbf{A}^\dagger := \mathbf{V}_1\Sigma_1^{-1}\mathbf{U}_1^* \in \mathbb{C}^{n \times m}. \quad (8.8)$$

Proof. By block multiplication $\mathbf{b} = \mathbf{U}\mathbf{U}^*\mathbf{b} = [\mathbf{U}_1, \mathbf{U}_2] \begin{bmatrix} \mathbf{U}_1^* \\ \mathbf{U}_2^* \end{bmatrix} \mathbf{b} = \mathbf{b}_1 + \mathbf{b}_2$, where $\mathbf{b}_1 = \mathbf{U}_1\mathbf{U}_1^*\mathbf{b} \in \mathcal{S}$ and $\mathbf{b}_2 = \mathbf{U}_2\mathbf{U}_2^*\mathbf{b} \in \mathcal{T}$. Since $\mathbf{U}_2^*\mathbf{U}_1 = \mathbf{0}$ it follows that $\langle \mathbf{b}_1, \mathbf{b}_2 \rangle = \mathbf{b}_2^*\mathbf{b}_1 = 0$ implying Part 1. Moreover, \mathbf{b}_1 is the orthogonal projection into \mathcal{S} . Since $\mathbf{V}_1^*\mathbf{V}_1 = \mathbf{I}$ we find

$$\mathbf{A}\mathbf{A}^\dagger\mathbf{b} = (\mathbf{U}_1\Sigma_1\mathbf{V}_1^*)(\mathbf{V}_1\Sigma_1^{-1}\mathbf{U}_1^*)\mathbf{b} = \mathbf{U}_1\mathbf{U}_1^*\mathbf{b} = \mathbf{b}_1$$

and Part 2 follows. \square

Corollary 8.14 (LSQ characterization using SVD)

$\mathbf{x} \in \mathbb{C}^n$ solves the least squares problem $\min_x \|\mathbf{Ax} - \mathbf{b}\|_2^2$ if and only if $\mathbf{x} = \mathbf{A}^\dagger\mathbf{b} + \mathbf{z}$, where \mathbf{A}^\dagger is given by (8.8) and $\mathbf{z} \in \ker(\mathbf{A})$.

Proof. Let \mathbf{x} be a least squares solution, i.e., $\mathbf{Ax} = \mathbf{b}_1$. If $\mathbf{z} := \mathbf{x} - \mathbf{A}^\dagger \mathbf{b}$ then $\mathbf{Az} = \mathbf{Ax} - \mathbf{AA}^\dagger \mathbf{b} = \mathbf{b}_1 - \mathbf{b}_1 = \mathbf{0}$ and $\mathbf{x} = \mathbf{A}^\dagger \mathbf{b} + \mathbf{z}$. If $\mathbf{x} = \mathbf{A}^\dagger \mathbf{b} + \mathbf{z}$ with $\mathbf{Az} = \mathbf{0}$ then

$$\mathbf{A}^* \mathbf{Ax} = \mathbf{A}^* \mathbf{A}(\mathbf{A}^\dagger \mathbf{b} + \mathbf{z}) = \mathbf{A}^*(\mathbf{AA}^\dagger \mathbf{b} + \mathbf{Az}) = \mathbf{A}^* \mathbf{b}_1 = \mathbf{A}^* \mathbf{b}.$$

The last equality follows since $\mathbf{b} = \mathbf{b}_1 + \mathbf{b}_2$ and $\mathbf{b}_2 \in \ker(\mathbf{A}^*)$. Thus \mathbf{x} satisfies the normal equations and by Theorem 8.5 is a least squares solution. \square

Example 8.15 (Projections and least squares solutions)

We have the singular value factorization (cf. Example 6.12)

$$\mathbf{A} := \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} [2] \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \end{bmatrix}.$$

We then find

$$\mathbf{A}^\dagger = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} = \frac{1}{4} \mathbf{A}^T.$$

Thus,

$$\mathbf{b}_1 = \mathbf{U}_1 \mathbf{U}_1^* \mathbf{b} = \frac{1}{2} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} [1 \ 1 \ 0] \mathbf{b} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \mathbf{A} \mathbf{A}^\dagger \mathbf{b} = \begin{bmatrix} (b_1 + b_2)/2 \\ (b_1 + b_2)/2 \\ 0 \end{bmatrix}.$$

Moreover, the set of all least squares solutions is

$$\{\mathbf{x} \in \mathbb{R}^2 : \mathbf{Ax} = \mathbf{b}_1\} = \{x_1, x_2 \in \mathbb{R} : x_1 + x_2 = \frac{b_1 + b_2}{2}\}. \quad (8.9)$$

Since $\ker(\mathbf{A}) = \{[\begin{smallmatrix} z \\ -z \end{smallmatrix}] : z \in \mathbb{R}\}$ we obtain the general least squares solution

$$\mathbf{x} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} + \begin{bmatrix} z \\ -z \end{bmatrix} = \begin{bmatrix} \frac{b_1+b_2}{4} + z \\ \frac{b_1+b_2}{4} - z \end{bmatrix}.$$

When $\text{rank}(\mathbf{A})$ is less than the number of columns of \mathbf{A} then $\ker(\mathbf{A}) \neq \{\mathbf{0}\}$, and we have a choice of \mathbf{z} . One possible choice is $\mathbf{z} = \mathbf{0}$ giving the solution $\mathbf{A}^\dagger \mathbf{b}$.

Theorem 8.16 (Minimal solution)

The least squares solution with minimal Euclidian norm is $\mathbf{x} = \mathbf{A}^\dagger \mathbf{b}$ corresponding to $\mathbf{z} = \mathbf{0}$.

Proof. Suppose $\mathbf{x} = \mathbf{A}^\dagger \mathbf{b} + \mathbf{z}$, with $\mathbf{z} \in \ker(\mathbf{A})$. Recall that if the right singular vectors of \mathbf{A} are partitioned as $[\mathbf{v}_1, \dots, \mathbf{v}_r, \mathbf{v}_{r+1}, \dots, \mathbf{v}_n] = [\mathbf{V}_1, \mathbf{V}_2]$, then \mathbf{V}_2 is a basis for $\ker(\mathbf{A})$. Moreover, $\mathbf{V}_2^* \mathbf{V}_1 = \mathbf{0}$ since \mathbf{V} has orthonormal columns. If $\mathbf{A}^\dagger = \mathbf{V}_1 \Sigma_1^{-1} \mathbf{U}_1^*$ and $\mathbf{z} \in \ker(\mathbf{A})$ then $\mathbf{z} = \mathbf{V}_2 \mathbf{y}$ for some $\mathbf{y} \in \mathbb{C}^{n-r}$ and we obtain

$$\mathbf{z}^* \mathbf{A}^\dagger \mathbf{b} = \mathbf{y}^* \mathbf{V}_2^* \mathbf{V}_1 \Sigma_1^{-1} \mathbf{U}_1^* \mathbf{b} = \mathbf{0}.$$

Thus \mathbf{z} and $\mathbf{A}^\dagger \mathbf{b}$ are orthogonal so that by Pythagoras $\|\mathbf{x}\|_2^2 = \|\mathbf{A}^\dagger \mathbf{b} + \mathbf{z}\|_2^2 = \|\mathbf{A}^\dagger \mathbf{b}\|_2^2 + \|\mathbf{z}\|_2^2 \geq \|\mathbf{A}^\dagger \mathbf{b}\|_2^2$ with equality for $\mathbf{z} = \mathbf{0}$. \square

Using MATLAB a least squares solution can be found using $\mathbf{x} = \mathbf{A} \backslash \mathbf{b}$ if \mathbf{A} has full rank. For rank deficient problems the function $\mathbf{x} = \text{lscov}(\mathbf{A}, \mathbf{b})$ finds a least squares solution with a maximal number of zeros in \mathbf{x} .

Example 8.17 (Rank deficient least squares solution)

Consider the least squares problem with $\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$ and $\mathbf{b} = [1, 1]^T$. The singular value factorization, \mathbf{A}^\dagger and $\mathbf{A}^\dagger \mathbf{b}$ are given by

$$\mathbf{A} := \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \end{bmatrix}, \quad \mathbf{A}^\dagger = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \end{bmatrix} = \frac{1}{4} \mathbf{A}, \quad \mathbf{A}^\dagger \mathbf{b} = \begin{bmatrix} 1/2 \\ 1/2 \end{bmatrix}.$$

Using Corollary 8.14 we find the general solution $[1/2, 1/2] + [a, -a]$ for any $a \in \mathbb{C}$. `lscov` gives the solution $[1, 0]^T$ corresponding to $a = 1/2$, while the minimal norm solution is $[1/2, 1/2]$ obtained for $a = 0$.

8.3.4 The generalized inverse

Consider the matrix $\mathbf{A}^\dagger := \mathbf{V}_1 \Sigma_1^{-1} \mathbf{U}_1^*$ in (8.8). If \mathbf{A} is square and nonsingular then $\mathbf{A}^\dagger \mathbf{A} = \mathbf{A} \mathbf{A}^\dagger = \mathbf{I}$ and \mathbf{A}^\dagger is the usual inverse of \mathbf{A} . Thus \mathbf{A}^\dagger is a possible generalization of the usual inverse. The matrix \mathbf{A}^\dagger satisfies Properties (1)-(4) in Exercise 8.18 and these properties define \mathbf{A}^\dagger uniquely (cf. Exercise 8.19). The unique matrix \mathbf{B} satisfying Properties (1)-(4) in Exercise 8.18 is called the **generalized inverse** or **pseudo inverse** of \mathbf{A} and denoted \mathbf{A}^\dagger . It follows that $\mathbf{A}^\dagger := \mathbf{V}_1 \Sigma_1^{-1} \mathbf{U}_1^*$ for any singular value factorization $\mathbf{U}_1 \Sigma_1 \mathbf{V}_1^*$ of \mathbf{A} . We show in Exercise 8.21 that if \mathbf{A} has linearly independent columns then

$$\mathbf{A}^\dagger = (\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}^*. \tag{8.10}$$

For further properties and examples see the exercises.

Exercise 8.18 (The generalized inverse)

Show that $\mathbf{B} := \mathbf{V}_1 \Sigma_1^{-1} \mathbf{U}_1^*$ satisfies (1) $\mathbf{ABA} = \mathbf{A}$, (2) $\mathbf{BAB} = \mathbf{B}$, (3) $(\mathbf{BA})^* = \mathbf{BA}$, and (4) $(\mathbf{AB})^* = \mathbf{AB}$.

Exercise 8.19 (Uniqueness of generalized inverse)

Given $\mathbf{A} \in \mathbb{C}^{m \times n}$, and suppose $\mathbf{B}, \mathbf{C} \in \mathbb{C}^{n \times m}$ satisfy

$$\begin{array}{lll} \mathbf{ABA} = \mathbf{A} & (1) & \mathbf{ACA} = \mathbf{A}, \\ \mathbf{BAB} = \mathbf{B} & (2) & \mathbf{CAC} = \mathbf{C}, \\ (\mathbf{AB})^* = \mathbf{AB} & (3) & (\mathbf{AC})^* = \mathbf{AC}, \\ (\mathbf{BA})^* = \mathbf{BA} & (4) & (\mathbf{CA})^* = \mathbf{CA}. \end{array}$$

Verify the following proof that $\mathbf{B} = \mathbf{C}$.

$$\begin{aligned} \mathbf{B} &= (\mathbf{BA})\mathbf{B} = (\mathbf{A}^*)\mathbf{B}^*\mathbf{B} = (\mathbf{A}^*\mathbf{C}^*)\mathbf{A}^*\mathbf{B}^*\mathbf{B} = \mathbf{CA}(\mathbf{A}^*\mathbf{B}^*)\mathbf{B} \\ &= \mathbf{CA}(\mathbf{BAB}) = (\mathbf{C})\mathbf{AB} = \mathbf{C}(\mathbf{AC})\mathbf{AB} = \mathbf{CC}^*\mathbf{A}^*(\mathbf{AB}) \\ &= \mathbf{CC}^*(\mathbf{A}^*\mathbf{B}^*\mathbf{A}^*) = \mathbf{C}(\mathbf{C}^*\mathbf{A}^*) = \mathbf{CAC} = \mathbf{C}. \end{aligned}$$

Exercise 8.20 (Verify that a matrix is a generalized inverse)

Show that the matrices $\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 0 \end{bmatrix}$ and $\mathbf{B} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$ satisfy the axioms in Exercise 8.18. Thus we can conclude that $\mathbf{B} = \mathbf{A}^\dagger$ without computing the singular value decomposition of \mathbf{A} .

Exercise 8.21 (Linearly independent columns and generalized inverse)

Suppose $\mathbf{A} \in \mathbb{C}^{m \times n}$ has linearly independent columns. Show that $\mathbf{A}^*\mathbf{A}$ is nonsingular and $\mathbf{A}^\dagger = (\mathbf{A}^*\mathbf{A})^{-1}\mathbf{A}^*$. If \mathbf{A} has linearly independent rows, then show that $\mathbf{A}\mathbf{A}^*$ is nonsingular and $\mathbf{A}^\dagger = \mathbf{A}^*(\mathbf{A}\mathbf{A}^*)^{-1}$.

Exercise 8.22 (The generalized inverse of a vector)

Show that $\mathbf{u}^\dagger = (\mathbf{u}^*\mathbf{u})^{-1}\mathbf{u}^*$ if $\mathbf{u} \in \mathbb{C}^{n,1}$ is nonzero.

Exercise 8.23 (The generalized inverse of an outer product)

If $\mathbf{A} = \mathbf{u}\mathbf{v}^*$ where $\mathbf{u} \in \mathbb{C}^m$, $\mathbf{v} \in \mathbb{C}^n$ are nonzero, show that

$$\mathbf{A}^\dagger = \frac{1}{\alpha} \mathbf{A}^*, \quad \alpha = \|\mathbf{u}\|_2^2 \|\mathbf{v}\|_2^2.$$

Exercise 8.24 (The generalized inverse of a diagonal matrix)

Show that $\text{diag}(\lambda_1, \dots, \lambda_n)^\dagger = \text{diag}(\lambda_1^\dagger, \dots, \lambda_n^\dagger)$ where

$$\lambda_i^\dagger = \begin{cases} 1/\lambda_i, & \lambda_i \neq 0 \\ 0 & \lambda_i = 0. \end{cases}$$

Exercise 8.25 (Properties of the generalized inverse)

Suppose $\mathbf{A} \in \mathbb{C}^{m \times n}$. Show that

- a) $(\mathbf{A}^*)^\dagger = (\mathbf{A}^\dagger)^*$.
- b) $(\mathbf{A}^\dagger)^\dagger = \mathbf{A}$.
- c) $(\alpha \mathbf{A})^\dagger = \frac{1}{\alpha} \mathbf{A}^\dagger$, $\alpha \neq 0$.

Exercise 8.26 (The generalized inverse of a product)

Suppose $k, m, n \in \mathbb{N}$, $\mathbf{A} \in \mathbb{C}^{m \times n}$, $\mathbf{B} \in \mathbb{C}^{n \times k}$. Suppose \mathbf{A} has linearly independent columns and \mathbf{B} has linearly independent rows.

- a) Show that $(\mathbf{AB})^\dagger = \mathbf{B}^\dagger \mathbf{A}^\dagger$. Hint: Let $\mathbf{E} = \mathbf{AF}$, $\mathbf{F} = \mathbf{B}^\dagger \mathbf{A}^\dagger$. Show by using $\mathbf{A}^\dagger \mathbf{A} = \mathbf{BB}^\dagger = \mathbf{I}$ that \mathbf{F} is the generalized inverse of \mathbf{E} .
- b) Find $\mathbf{A} \in \mathbb{R}^{1,2}$, $\mathbf{B} \in \mathbb{R}^{2,1}$ such that $(\mathbf{AB})^\dagger \neq \mathbf{B}^\dagger \mathbf{A}^\dagger$.

Exercise 8.27 (The generalized inverse of the conjugate transpose)

Show that $\mathbf{A}^* = \mathbf{A}^\dagger$ if and only if all singular values of \mathbf{A} are either zero or one.

Exercise 8.28 (Linearly independent columns)

Show that if \mathbf{A} has rank n then $\mathbf{A}(\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}^* \mathbf{b}$ is the projection of \mathbf{b} into $\text{span}(\mathbf{A})$. (Cf. Exercise 8.21.)

Exercise 8.29 (Analysis of the general linear system)

Consider the linear system $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{A} \in \mathbb{C}^{n \times n}$ has rank $r > 0$ and $\mathbf{b} \in \mathbb{C}^n$.

Let

$$\mathbf{U}^* \mathbf{A} \mathbf{V} = \begin{bmatrix} \Sigma_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$$

represent the singular value decomposition of \mathbf{A} .

- a) Let $\mathbf{c} = [c_1, \dots, c_n]^T = \mathbf{U}^* \mathbf{b}$ and $\mathbf{y} = [y_1, \dots, y_n]^T = \mathbf{V}^* \mathbf{x}$. Show that $\mathbf{Ax} = \mathbf{b}$ if and only if

$$\begin{bmatrix} \Sigma_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{y} = \mathbf{c}.$$

- b) Show that $\mathbf{Ax} = \mathbf{b}$ has a solution \mathbf{x} if and only if $c_{r+1} = \dots = c_n = 0$.
- c) Deduce that a linear system $\mathbf{Ax} = \mathbf{b}$ has either no solution, one solution or infinitely many solutions.

Exercise 8.30 (Fredholm's alternative)

For any $\mathbf{A} \in \mathbb{C}^{m \times n}$, $\mathbf{b} \in \mathbb{C}^n$ show that one and only one of the following systems has a solution

$$(1) \quad \mathbf{Ax} = \mathbf{b}, \quad (2) \quad \mathbf{A}^* \mathbf{y} = \mathbf{0}, \quad \mathbf{y}^* \mathbf{b} \neq 0.$$

In other words either $\mathbf{b} \in \text{span}(\mathbf{A})$, or we can find $\mathbf{y} \in \ker(\mathbf{A}^*)$ such that $\mathbf{y}^* \mathbf{b} \neq 0$. This is called **Fredholm's alternative**.

8.4 Perturbation Theory for Least Squares

In this section we consider what effect small changes in the data \mathbf{A}, \mathbf{b} have on the solution \mathbf{x} of the least squares problem $\min \|\mathbf{Ax} - \mathbf{b}\|_2$.

If \mathbf{A} has linearly independent columns then we can write the least squares solution \mathbf{x} (the solution of $\mathbf{A}^* \mathbf{Ax} = \mathbf{A}^* \mathbf{b}$) as

$$\mathbf{x} = \mathbf{A}^\dagger \mathbf{b} = \mathbf{A}^\dagger \mathbf{b}_1, \quad \mathbf{A}^\dagger := (\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}^*,$$

where \mathbf{b}_1 is the orthogonal projection of \mathbf{b} into the column space $\text{span}(\mathbf{A})$.

8.4.1 Perturbing the right hand side

Let us now consider the effect of a perturbation in \mathbf{b} on \mathbf{x} .

Theorem 8.31 (Perturbing the right hand side)

Suppose $\mathbf{A} \in \mathbb{C}^{m \times n}$ has linearly independent columns, and let $\mathbf{b}, \mathbf{e} \in \mathbb{C}^m$. Let $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$ be the solutions of $\min \|\mathbf{Ax} - \mathbf{b}\|_2$ and $\min \|\mathbf{Ay} - \mathbf{b} - \mathbf{e}\|_2$. Finally, let $\mathbf{b}_1, \mathbf{e}_1$ be the orthogonal projections of \mathbf{b} and \mathbf{e} into $\text{span}(\mathbf{A})$. If $\mathbf{b}_1 \neq \mathbf{0}$, we have for any operator norm

$$\frac{1}{K(\mathbf{A})} \frac{\|\mathbf{e}_1\|}{\|\mathbf{b}_1\|} \leq \frac{\|\mathbf{y} - \mathbf{x}\|}{\|\mathbf{x}\|} \leq K(\mathbf{A}) \frac{\|\mathbf{e}_1\|}{\|\mathbf{b}_1\|}, \quad K(\mathbf{A}) = \|\mathbf{A}\| \|\mathbf{A}^\dagger\|. \quad (8.11)$$

Proof. Subtracting $\mathbf{x} = \mathbf{A}^\dagger \mathbf{b}_1$ from $\mathbf{y} = \mathbf{A}^\dagger \mathbf{b}_1 + \mathbf{A}^\dagger \mathbf{e}_1$ we have $\mathbf{y} - \mathbf{x} = \mathbf{A}^\dagger \mathbf{e}_1$. Thus $\|\mathbf{y} - \mathbf{x}\| = \|\mathbf{A}^\dagger \mathbf{e}_1\| \leq \|\mathbf{A}^\dagger\| \|\mathbf{e}_1\|$. Moreover, $\|\mathbf{b}_1\| = \|\mathbf{Ax}\| \leq \|\mathbf{A}\| \|\mathbf{x}\|$. Therefore $\|\mathbf{y} - \mathbf{x}\| / \|\mathbf{x}\| \leq \|\mathbf{A}\| \|\mathbf{A}^\dagger\| \|\mathbf{e}_1\| / \|\mathbf{b}_1\|$ proving the rightmost inequality. From $\mathbf{A}(\mathbf{x} - \mathbf{y}) = \mathbf{e}_1$ and $\mathbf{x} = \mathbf{A}^\dagger \mathbf{b}_1$ we obtain the leftmost inequality. \square

(8.11) is analogous to the bound (7.20) for linear systems. We see that the number $K(\mathbf{A}) = \|\mathbf{A}\| \|\mathbf{A}^\dagger\|$ generalizes the condition number $\|\mathbf{A}\| \|\mathbf{A}^{-1}\|$ for a square matrix. The main difference between (8.11) and (7.20) is however that $\|\mathbf{e}\| / \|\mathbf{b}\|$ in (7.20) has been replaced by $\|\mathbf{e}_1\| / \|\mathbf{b}_1\|$, the orthogonal projections of \mathbf{e} and \mathbf{b} into $\text{span}(\mathbf{A})$. If \mathbf{b} lies almost entirely in $\ker(\mathbf{A}^*)$, i.e. $\|\mathbf{b}\| / \|\mathbf{b}_1\|$ is large,

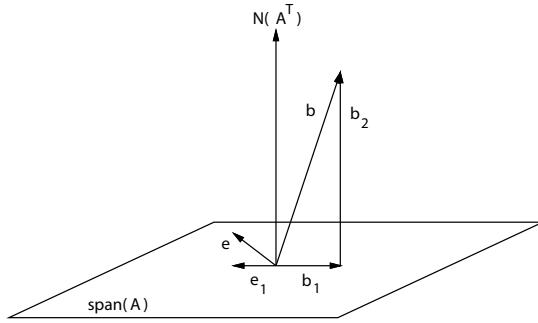


Figure 8.3: Graphical interpretation of the bounds in Theorem 8.31.

then $\|e_1\|/\|b_1\|$ can be much larger than $\|e\|/\|b\|$. This is illustrated in Figure 8.3. If b is almost orthogonal to $\text{span}(A)$, $\|e_1\|/\|b_1\|$ will normally be much larger than $\|e\|/\|b\|$.

Example 8.32 (Perturbing the right hand side)

Suppose

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad b = \begin{bmatrix} 10^{-4} \\ 0 \\ 1 \end{bmatrix}, \quad e = \begin{bmatrix} 10^{-6} \\ 0 \\ 0 \end{bmatrix}.$$

For this example we can compute $K(A)$ by finding A^\dagger explicitly. Indeed,

$$A^T A = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}, \quad (A^T A)^{-1} = \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix}, \quad A^\dagger = (A^T A)^{-1} A^T = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

Thus $K_\infty(A) = \|A\|_\infty \|A^\dagger\|_\infty = 2 \cdot 2 = 4$ is quite small.

Consider now the projections b_1 and e_1 . We find $A A^\dagger = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$. Hence

$$b_1 = A A^\dagger b = [10^{-4}, 0, 0]^T, \quad \text{and} \quad e_1 = A A^\dagger e = [10^{-6}, 0, 0]^T.$$

Thus $\|e_1\|_\infty/\|b_1\|_\infty = 10^{-2}$ and (8.11) takes the form

$$\frac{1}{4} 10^{-2} \leq \frac{\|\mathbf{y} - \mathbf{x}\|_\infty}{\|\mathbf{x}\|_\infty} \leq 4 \cdot 10^{-2}. \quad (8.12)$$

To verify the bounds we compute the solutions as $\mathbf{x} = A^\dagger b = [10^{-4}, 0]^T$ and $\mathbf{y} = A^\dagger(b + e) = [10^{-4} + 10^{-6}, 0]^T$. Hence

$$\frac{\|\mathbf{y} - \mathbf{x}\|_\infty}{\|\mathbf{x}\|_\infty} = \frac{10^{-6}}{10^{-4}} = 10^{-2},$$

in agreement with (8.12)

Exercise 8.33 (Condition number)

Let

$$\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 1 & 1 \\ 1 & 1 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}.$$

- a) Determine the projections \mathbf{b}_1 and \mathbf{b}_2 of \mathbf{b} on $\text{span}(\mathbf{A})$ and $\ker(\mathbf{A}^T)$.
- b) Compute $K(\mathbf{A}) = \|\mathbf{A}\|_2 \|\mathbf{A}^\dagger\|_2$.

For each \mathbf{A} we can find \mathbf{b} and \mathbf{e} so that we have equality in the upper bound in (8.11). The lower bound is best possible in a similar way.

Exercise 8.34 (Equality in perturbation bound)

- a) Let $\mathbf{A} \in \mathbb{C}^{m \times n}$. Show that we have equality to the right in (8.11) if $\mathbf{b} = \mathbf{A}\mathbf{y}_A$, $\mathbf{e}_1 = \mathbf{y}_{A^\dagger}$ where $\|\mathbf{A}\mathbf{y}_A\| = \|\mathbf{A}\|$, $\|\mathbf{A}^\dagger\mathbf{y}_{A^\dagger}\| = \|\mathbf{A}^\dagger\|$.
- b) Show that we have equality to the left if we switch \mathbf{b} and \mathbf{e} in a).
- c) Let \mathbf{A} be as in Example 8.32. Find extremal \mathbf{b} and \mathbf{e} when the l_∞ norm is used.

8.4.2 Perturbing the matrix

The analysis of the effects of a perturbation \mathbf{E} in \mathbf{A} is quite difficult. The following result is stated without proof, see [22, p. 51]. For other estimates see [3] and [30].

Theorem 8.35 (Perturbing the matrix)

Suppose $\mathbf{A}, \mathbf{E} \in \mathbb{C}^{m \times n}$, $m > n$, where \mathbf{A} has linearly independent columns and $\alpha := 1 - \|\mathbf{E}\|_2 \|\mathbf{A}^\dagger\|_2 > 0$. Then $\mathbf{A} + \mathbf{E}$ has linearly independent columns. Let $\mathbf{b} = \mathbf{b}_1 + \mathbf{b}_2 \in \mathbb{C}^m$ where \mathbf{b}_1 and \mathbf{b}_2 are the orthogonal projections into $\text{span}(\mathbf{A})$ and $\ker(\mathbf{A}^*)$ respectively. Suppose $\mathbf{b}_1 \neq \mathbf{0}$. Let \mathbf{x} and \mathbf{y} be the solutions of $\min \|\mathbf{Ax} - \mathbf{b}\|_2$ and $\min \|(\mathbf{A} + \mathbf{E})\mathbf{y} - \mathbf{b}\|_2$. Then

$$\rho = \frac{\|\mathbf{x} - \mathbf{y}\|_2}{\|\mathbf{x}\|_2} \leq \frac{1}{\alpha} K(1 + \beta K) \frac{\|\mathbf{E}\|_2}{\|\mathbf{A}\|_2}, \quad \beta = \frac{\|\mathbf{b}_2\|_2}{\|\mathbf{b}_1\|_2}, \quad K = \|\mathbf{A}\|_2 \|\mathbf{A}^\dagger\|_2. \quad (8.13)$$

(8.13) says that the relative error in \mathbf{y} as an approximation to \mathbf{x} can be at most $K(1 + \beta K)/\alpha$ times as large as the size $\|\mathbf{E}\|_2/\|\mathbf{A}\|_2$ of the relative perturbation in \mathbf{A} . β will be small if \mathbf{b} lies almost entirely in $\text{span}(\mathbf{A})$, and we have approximately $\rho \leq \frac{1}{\alpha} K \|\mathbf{E}\|_2 / \|\mathbf{A}\|_2$. This corresponds to the estimate (7.26) for

linear systems. If β is not small, the term $\frac{1}{\alpha}K^2\beta\|\mathbf{E}\|_2/\|\mathbf{A}\|_2$ will dominate. In other words, the condition number is roughly $K(\mathbf{A})$ if β is small and $K(\mathbf{A})^2\beta$ if β is not small. Note that β is large if \mathbf{b} is almost orthogonal to $\text{span}(\mathbf{A})$ and that $\mathbf{b}_2 = \mathbf{b} - \mathbf{Ax}$ is the residual of \mathbf{x} .

Exercise 8.36 (Problem using normal equations)

Consider the least squares problems where

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1+\epsilon \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 2 \\ 3 \\ 2 \end{bmatrix}, \quad \epsilon \in \mathbb{R}.$$

- a) Find the normal equations and the exact least squares solution.
- b) Suppose ϵ is small and we replace the $(2, 2)$ entry $3+2\epsilon+\epsilon^2$ in $\mathbf{A}^T \mathbf{A}$ by $3+2\epsilon$. (This will be done in a computer if $\epsilon < \sqrt{u}$, u being the round-off unit). For example, if $u = 10^{-16}$ then $\sqrt{u} = 10^{-8}$. Solve $\mathbf{A}^T \mathbf{Ax} = \mathbf{A}^T \mathbf{b}$ for \mathbf{x} and compare with the \mathbf{x} found in a). (We will get a much more accurate result using the QR factorization or the singular value decomposition on this problem).

8.5 Perturbation Theory for Singular Values

In this section we consider what effect a small change in the matrix \mathbf{A} has on the singular values.

8.5.1 The Minmax Theorem for Singular Values and the Hoffman-Wielandt Theorem

We have a minmax and maxmin characterization for singular values.

Theorem 8.37 (The Courant-Fischer Theorem for Singular Values) Suppose $\mathbf{A} \in \mathbb{C}^{m,n}$ has singular values $\sigma_1, \sigma_2, \dots, \sigma_n$ ordered so that $\sigma_1 \geq \dots \geq \sigma_n$. Then for $k = 1, \dots, n$

$$\sigma_k = \min_{\dim(\mathcal{S})=n-k+1} \max_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} \frac{\|\mathbf{Ax}\|_2}{\|\mathbf{x}\|_2} = \max_{\dim(\mathcal{S})=k} \min_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} \frac{\|\mathbf{Ax}\|_2}{\|\mathbf{x}\|_2}. \quad (8.14)$$

Proof. Since

$$\frac{\|\mathbf{Ax}\|_2^2}{\|\mathbf{x}\|_2^2} = \frac{\langle \mathbf{Ax}, \mathbf{Ax} \rangle}{\langle \mathbf{x}, \mathbf{x} \rangle} = \frac{\langle \mathbf{x}, \mathbf{A}^* \mathbf{Ax} \rangle}{\langle \mathbf{x}, \mathbf{x} \rangle}$$

denotes the Rayleigh quotient $R_{\mathbf{A}^* \mathbf{A}}(\mathbf{x})$ of $\mathbf{A}^* \mathbf{A}$, and since the singular values of \mathbf{A} are the nonnegative square roots of the eigenvalues of $\mathbf{A}^* \mathbf{A}$, the results follow from the Courant-Fischer Theorem for eigenvalues, see Theorem 5.44. \square

By taking $k = 1$ and $k = n$ in (8.14) we obtain for any $\mathbf{A} \in \mathbb{C}^{m,n}$

$$\sigma_1 = \max_{\substack{\mathbf{x} \in \mathbb{C}^n \\ \mathbf{x} \neq \mathbf{0}}} \frac{\|\mathbf{Ax}\|_2}{\|\mathbf{x}\|_2}, \quad \sigma_n = \min_{\substack{\mathbf{x} \in \mathbb{C}^n \\ \mathbf{x} \neq \mathbf{0}}} \frac{\|\mathbf{Ax}\|_2}{\|\mathbf{x}\|_2}. \quad (8.15)$$

This follows since the only subspace of \mathbb{C}^n of dimension n is \mathbb{C}^n itself.

The Hoffman-Wielandt Theorem, see Theorem 5.47, for eigenvalues of Hermitian matrices can be written

$$\sum_{j=1}^n |\mu_j - \lambda_j|^2 \leq \|\mathbf{A} - \mathbf{B}\|_F^2 := \sum_{i=1}^n \sum_{j=1}^n |a_{ij} - b_{ij}|^2, \quad (8.16)$$

where $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n,n}$ are both Hermitian matrices with eigenvalues $\lambda_1 \geq \dots \geq \lambda_n$ and $\mu_1 \geq \dots \geq \mu_n$, respectively.

For singular values we have a similar result.

Theorem 8.38 (Hoffman-Wielandt Theorem for singular values) *For any $m, n \in \mathbb{N}$ and $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{m,n}$ we have*

$$\sum_{j=1}^n |\beta_j - \alpha_j|^2 \leq \|\mathbf{A} - \mathbf{B}\|_F^2. \quad (8.17)$$

where $\alpha_1 \geq \dots \geq \alpha_n$ and $\beta_1 \geq \dots \geq \beta_n$ are the singular values of \mathbf{A} and \mathbf{B} , respectively.

Proof. We apply the Hoffman-Wielandt Theorem for eigenvalues to the Hermitian matrices

$$\mathbf{C} := \begin{bmatrix} \mathbf{0} & \mathbf{A} \\ \mathbf{A}^* & \mathbf{0} \end{bmatrix} \text{ and } \mathbf{D} := \begin{bmatrix} \mathbf{0} & \mathbf{B} \\ \mathbf{B}^* & \mathbf{0} \end{bmatrix} \in \mathbb{C}^{m+n, m+n}.$$

If \mathbf{C} and \mathbf{D} has eigenvalues $\lambda_1 \geq \dots \geq \lambda_{m+n}$ and $\mu_1 \geq \dots \geq \mu_{m+n}$, respectively then

$$\sum_{j=1}^{m+n} |\lambda_j - \mu_j|^2 \leq \|\mathbf{C} - \mathbf{D}\|_F^2. \quad (8.18)$$

Suppose \mathbf{A} has rank r and SVD $\mathbf{U}\Sigma\mathbf{V}^*$. We use (6.10) and determine the eigen-

pairs of \mathbf{C} as follows.

$$\begin{aligned} \begin{bmatrix} \mathbf{0} & \mathbf{A} \\ \mathbf{A}^* & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_i \end{bmatrix} &= \begin{bmatrix} \mathbf{Av}_i \\ \mathbf{A}^*\mathbf{u}_i \end{bmatrix} = \begin{bmatrix} \alpha_i \mathbf{u}_i \\ \alpha_i \mathbf{v}_i \end{bmatrix} = \alpha_i \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_i \end{bmatrix}, \quad i = 1, \dots, r, \\ \begin{bmatrix} \mathbf{0} & \mathbf{A} \\ \mathbf{A}^* & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u}_i \\ -\mathbf{v}_i \end{bmatrix} &= \begin{bmatrix} -\mathbf{Av}_i \\ \mathbf{A}^*\mathbf{u}_i \end{bmatrix} = \begin{bmatrix} -\alpha_i \mathbf{u}_i \\ \alpha_i \mathbf{v}_i \end{bmatrix} = -\alpha_i \begin{bmatrix} \mathbf{u}_i \\ -\mathbf{v}_i \end{bmatrix}, \quad i = 1, \dots, r, \\ \begin{bmatrix} \mathbf{0} & \mathbf{A} \\ \mathbf{A}^* & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u}_i \\ \mathbf{0} \end{bmatrix} &= \begin{bmatrix} \mathbf{0} \\ \mathbf{A}^*\mathbf{u}_i \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} = 0 \begin{bmatrix} \mathbf{u}_i \\ \mathbf{0} \end{bmatrix}, \quad i = r+1, \dots, m, \\ \begin{bmatrix} \mathbf{0} & \mathbf{A} \\ \mathbf{A}^* & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{v}_i \end{bmatrix} &= \begin{bmatrix} \mathbf{Av}_i \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} = 0 \begin{bmatrix} \mathbf{0} \\ \mathbf{v}_i \end{bmatrix}, \quad i = r+1, \dots, n. \end{aligned}$$

Thus \mathbf{C} has the $2r$ eigenvalues $\alpha_1, -\alpha_1, \dots, \alpha_r, -\alpha_r$ and $m+n-2r$ additional zero eigenvalues. Similarly, if \mathbf{B} has rank s then \mathbf{D} has the $2s$ eigenvalues $\beta_1, -\beta_1, \dots, \beta_s, -\beta_s$ and $m+n-2s$ additional zero eigenvalues. Let

$$t := \max(r, s).$$

Then

$$\begin{aligned} \lambda_1 \geq \dots \geq \lambda_{m+n} &= \alpha_1 \geq \dots \geq \alpha_t \geq 0 = \dots = 0 \geq -\alpha_t \geq \dots \geq -\alpha_1, \\ \mu_1 \geq \dots \geq \mu_{m+n} &= \beta_1 \geq \dots \geq \beta_t \geq 0 = \dots = 0 \geq -\beta_t \geq \dots \geq -\beta_1. \end{aligned}$$

We find

$$\sum_{j=1}^{m+n} |\lambda_j - \mu_j|^2 = \sum_{i=1}^t |\alpha_i - \beta_i|^2 + \sum_{i=1}^t |-\alpha_i + \beta_i|^2 = 2 \sum_{i=1}^t |\alpha_i - \beta_i|^2$$

and

$$\|\mathbf{C} - \mathbf{D}\|_F^2 = \left\| \begin{bmatrix} \mathbf{0} & \mathbf{A} - \mathbf{B} \\ \mathbf{A}^* - \mathbf{B}^* & \mathbf{0} \end{bmatrix} \right\|_F^2 = \|\mathbf{B} - \mathbf{A}\|_F^2 + \|(\mathbf{B} - \mathbf{A})^*\|_F^2 = 2\|\mathbf{B} - \mathbf{A}\|_F^2.$$

But then (8.18) implies $\sum_{i=1}^t |\alpha_i - \beta_i|^2 \leq \|\mathbf{B} - \mathbf{A}\|_F^2$. Since $t \leq n$ and $\alpha_i = \beta_i = 0$ for $i = t+1, \dots, n$ we obtain (8.17). \square

The Hoffman-Wielandt Theorem for singular values, Theorem 8.38 shows that the singular values of a matrix are well conditioned. Changing the Frobenius norm of a matrix by small amount only changes the singular values by a small amount.

Using the 2-norm we have a similar result.

Theorem 8.39 (Perturbation of singular values)

Let $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{m \times n}$ be rectangular matrices with singular values $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_n$ and $\beta_1 \geq \beta_2 \geq \dots \geq \beta_n$. Then

$$|\alpha_j - \beta_j| \leq \|\mathbf{A} - \mathbf{B}\|_2, \text{ for } j = 1, 2, \dots, n. \quad (8.19)$$

Proof. Fix j and let \mathcal{S} be the $n - j + 1$ dimensional subspace for which the minimum in Theorem 8.37 is obtained for \mathbf{A} . Then

$$\alpha_j = \max_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} \frac{\|(\mathbf{B} + (\mathbf{A} - \mathbf{B}))\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \leq \max_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} \frac{\|\mathbf{Bx}\|_2}{\|\mathbf{x}\|_2} + \max_{\substack{\mathbf{x} \in \mathcal{S} \\ \mathbf{x} \neq \mathbf{0}}} \frac{\|(\mathbf{A} - \mathbf{B})\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \leq \beta_j + \|\mathbf{A} - \mathbf{B}\|_2.$$

By symmetry we obtain $\beta_j \leq \alpha_j + \|\mathbf{A} - \mathbf{B}\|_2$ and the proof is complete. \square

The following result is an analogue of Theorem 7.34.

Theorem 8.40 (Generalized inverse when perturbing the matrix)

Let $\mathbf{A}, \mathbf{E} \in \mathbb{R}^{m \times n}$ have singular values $\alpha_1 \geq \dots \geq \alpha_n$ and $\epsilon_1 \geq \dots \geq \epsilon_n$. If $\|\mathbf{A}^\dagger\|_2 \|\mathbf{E}\|_2 < 1$ then

1. $\text{rank}(\mathbf{A} + \mathbf{E}) \geq \text{rank}(\mathbf{A})$,
2. $\|(\mathbf{A} + \mathbf{E})^\dagger\|_2 \leq \frac{\|\mathbf{A}^\dagger\|_2}{1 - \|\mathbf{A}^\dagger\|_2 \|\mathbf{E}\|_2} = \frac{1}{\alpha_r - \epsilon_1}$,

where r is the rank of \mathbf{A} .

Proof. Suppose \mathbf{A} has rank r and let $\mathbf{B} := \mathbf{A} + \mathbf{E}$ have singular values $\beta_1 \geq \dots \geq \beta_n$. In terms of singular values the inequality $\|\mathbf{A}^\dagger\|_2 \|\mathbf{E}\|_2 < 1$ can be written $\epsilon_1/\alpha_r < 1$ or $\alpha_r > \epsilon_1$. By Theorem 8.39 we have $\alpha_r - \beta_r \leq \epsilon_1$, which implies $\beta_r \geq \alpha_r - \epsilon_1 > 0$, and this shows that $\text{rank}(\mathbf{A} + \mathbf{E}) > r$. To prove 2., the inequality $\beta_r \geq \alpha_r - \epsilon_1$ implies that

$$\|(\mathbf{A} + \mathbf{E})^\dagger\|_2 \leq \frac{1}{\beta_r} \leq \frac{1}{\alpha_r - \epsilon_1} = \frac{1/\alpha_r}{1 - \epsilon_1/\alpha_r} = \frac{\|\mathbf{A}^\dagger\|_2}{1 - \|\mathbf{A}^\dagger\|_2 \|\mathbf{E}\|_2}.$$

\square

8.6 Review Questions

8.6.1 Do the normal equations always have a solution?

8.6.2 When is the least squares solution unique?

8.6.3 Express the general least squares solution in terms of the generalized inverse.

8.6.4 Consider perturbing the right-hand side in a linear equation and a least squares problem. What is the main difference in the perturbation inequalities?

8.6.5 Why does one often prefer using QR factorization instead of normal equations for solving least squares problems.

8.6.6 What is an orthogonal sum?

8.6.7 How is an orthogonal projection defined?

Part III

Kronecker Products and Fourier Transforms

Chapter 9

The Kronecker Product



Leopold Kronecker, 1823-1891 (left), Siméon Denis Poisson, 1781-1840 (right).

Matrices arising from 2D and 3D problems sometimes have a Kronecker product structure. Identifying a Kronecker structure can be very rewarding since it simplifies the study of such matrices.

9.0.1 The 2D Poisson problem

Let $\Omega := (0, 1)^2 = \{(x, y) : 0 < x, y < 1\}$ be the open unit square with boundary $\partial\Omega$. Consider the problem

$$-\Delta u := -\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} = f \text{ on } \Omega, \quad (9.1)$$

$$u := 0 \text{ on } \partial\Omega.$$

Here the function f is given and continuous on Ω , and we seek a function $u = u(x, y)$ such that (9.1) holds and which is zero on $\partial\Omega$.

Let m be a positive integer. We solve the problem numerically by finding approximations $v_{j,k} \approx u(jh, kh)$ on a grid of points given by

$$\bar{\Omega}_h := \{(jh, kh) : j, k = 0, 1, \dots, m+1\}, \quad \text{where } h = 1/(m+1).$$

The points $\Omega_h := \{(jh, kh) : j, k = 1, \dots, m\}$ are the interior points, while $\bar{\Omega}_h \setminus \Omega_h$ are the boundary points. The solution is zero at the boundary points. Using the difference approximation from Chapter 1 for the second derivative we obtain the following approximations for the partial derivatives

$$\frac{\partial^2 u(jh, kh)}{\partial x^2} \approx \frac{v_{j-1,k} - 2v_{j,k} + v_{j+1,k}}{h^2}, \quad \frac{\partial^2 u(jh, kh)}{\partial y^2} \approx \frac{v_{j,k-1} - 2v_{j,k} + v_{j,k+1}}{h^2}.$$

Inserting this in (9.1) we get the following discrete analog of (9.1)

$$\begin{aligned} -\Delta_h v_{j,k} &= f_{j,k}, & (jh, kh) \in \Omega_h, \\ v_{j,k} &= 0, & (jh, kh) \in \partial\Omega_h, \end{aligned} \tag{9.2}$$

where $f_{j,k} := f(jh, kh)$ and

$$-\Delta_h v_{j,k} := \frac{-v_{j-1,k} + 2v_{j,k} - v_{j+1,k}}{h^2} + \frac{-v_{j,k-1} + 2v_{j,k} - v_{j,k+1}}{h^2}. \tag{9.3}$$

Multiplying both sides of (9.2) by h^2 we obtain

$$\begin{aligned} 4v_{j,k} - v_{j-1,k} - v_{j+1,k} - v_{j,k-1} - v_{j,k+1} &= h^2 f_{j,k}, & (jh, kh) \in \Omega_h, \\ v_{0,k} = v_{m+1,k} = v_{j,0} = v_{j,m+1} &= 0, & j, k = 0, 1, \dots, m+1. \end{aligned} \tag{9.4}$$

The equations in (9.4) define a set of linear equations for the unknowns $\mathbf{V} := [v_{jk}] \in \mathbb{R}^{m \times m}$.

Observe that (9.4) can be written as a matrix equation in the form

$$\mathbf{T}\mathbf{V} + \mathbf{V}\mathbf{T} = h^2 \mathbf{F} \quad \text{with } h = 1/(m+1), \tag{9.5}$$

where $\mathbf{T} = \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{m \times m}$ is the second derivative matrix given by (1.23) and $\mathbf{F} = (f_{jk}) = (f(jh, kh)) \in \mathbb{R}^{m \times m}$. Indeed, the (j, k) element in $\mathbf{T}\mathbf{V} + \mathbf{V}\mathbf{T}$ is given by

$$\sum_{i=1}^m \mathbf{T}_{j,i} v_{i,k} + \sum_{i=1}^m v_{j,i} \mathbf{T}_{i,k},$$

and this is precisely the left hand side of (9.4).

To write (9.4) in standard form $\mathbf{Ax} = \mathbf{b}$ we need to order the unknowns $v_{j,k}$ in some way. The following operation of **vectorization** of a matrix gives one possible ordering.

1,1	1,2	1,3
2,1	2,2	2,3
3,1	3,2	3,3

1,3	2,3	3,3
1,2	2,2	3,2
1,1	2,1	3,1

7	8	9
4	5	6
1	2	3

$v_{j,k}$ in V -matrix $v_{j,k}$ in grid x_i in grid

Figure 9.1: Numbering of grid points

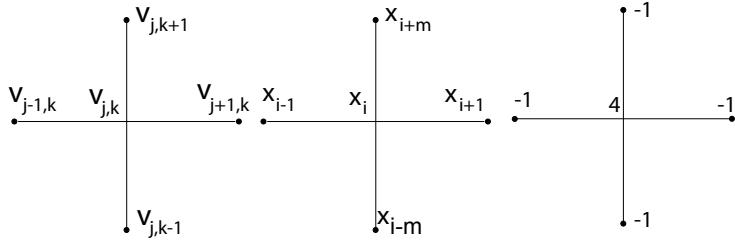


Figure 9.2: The 5-point stencil

Definition 9.1 (vec operation)For any $\mathbf{B} \in \mathbb{R}^{m \times n}$ we define the vector

$$\text{vec}(\mathbf{B}) := [b_{11}, \dots, b_{m1}, b_{12}, \dots, b_{m2}, \dots, b_{1n}, \dots, b_{mn}]^T \in \mathbb{R}^{mn}$$

by stacking the columns of \mathbf{B} on top of each other.

Let $n = m^2$ and $\mathbf{x} := \text{vec}(\mathbf{V}) \in \mathbb{R}^n$. Note that forming \mathbf{x} by stacking the columns of \mathbf{V} on top of each other means an ordering of the grid points which for $m = 3$ is illustrated in Figure 9.1. We call this the **natural ordering**. The elements in (9.4) form a 5-point stencil, as shown in Figure 9.2.

To find the matrix \mathbf{A} we note that for values of j, k where the 5-point stencil does not touch the boundary, (9.4) takes the form

$$4x_i - x_{i-1} - x_{i+1} - x_{i-m} - x_{i+m} = b_i,$$

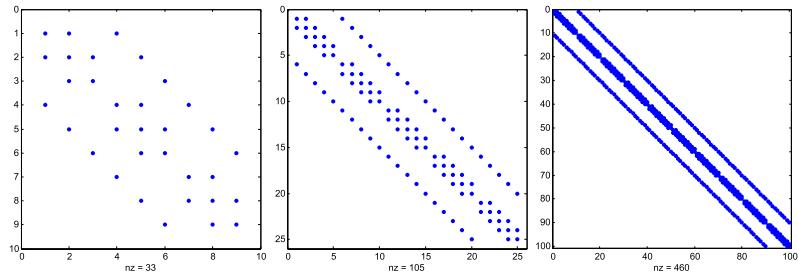


Figure 9.3: Band structure of the 2D test matrix, $n = 9$, $n = 25$, $n = 100$

where $x_i = v_{jk}$ and $b_i = h^2 f_{jk}$. This must be modified close to the boundary. We obtain the linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \quad \mathbf{A} \in \mathbb{R}^{n \times n}, \quad \mathbf{b} \in \mathbb{R}^n, \quad n = m^2, \quad (9.6)$$

where $\mathbf{x} = \text{vec}(\mathbf{V})$, $\mathbf{b} = h^2 \text{vec}(\mathbf{F})$ with $\mathbf{F} = (f_{jk}) \in \mathbb{R}^{m \times m}$, and \mathbf{A} is the **Poisson matrix** given by

$$\begin{aligned} a_{ii} &= 4, & i &= 1, \dots, n, \\ a_{i+1,i} &= a_{i,i+1} = -1, & i &= 1, \dots, n-1, \quad i \neq m, 2m, \dots, (m-1)m, \\ a_{i+m,i} &= a_{i,i+m} = -1, & i &= 1, \dots, n-m, \\ a_{ij} &= 0, & & \text{otherwise.} \end{aligned} \quad (9.7)$$

For $m = 3$ we have the following matrix

$$\mathbf{A} = \left[\begin{array}{ccccccccc} 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 4 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 4 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 \end{array} \right].$$

Exercise 9.2 (4 × 4 Poisson matrix)

Write down the Poisson matrix for $m = 2$ and show that it is strictly diagonally dominant.

9.0.2 The test matrices

The (2-dimensional) Poisson matrix is a special case of the matrix $\mathbf{T}_2 = [a_{ij}] \in \mathbb{R}^{n \times n}$ with elements

$$\begin{aligned} a_{ii} &= 2d, \quad i = 1, \dots, n, \\ a_{i,i+1} = a_{i+1,i} &= a, \quad i = 1, \dots, n-1, \quad i \neq m, 2m, \dots, (m-1)m, \\ a_{i,i+m} = a_{i+m,i} &= a, \quad i = 1, \dots, n-m, \\ a_{ij} &= 0, \quad \text{otherwise}, \end{aligned} \quad (9.8)$$

and where a, d are real numbers. We will refer to this matrix as simply the **2D test matrix**. For $m = 3$ the 2D test matrix looks as follows

$$\mathbf{T}_2 = \left[\begin{array}{ccc|ccc|ccc} 2d & a & 0 & a & 0 & 0 & 0 & 0 & 0 \\ a & 2d & a & 0 & a & 0 & 0 & 0 & 0 \\ 0 & a & 2d & 0 & 0 & a & 0 & 0 & 0 \\ \hline a & 0 & 0 & 2d & a & 0 & a & 0 & 0 \\ 0 & a & 0 & a & 2d & a & 0 & a & 0 \\ 0 & 0 & a & 0 & a & 2d & 0 & 0 & a \\ \hline 0 & 0 & 0 & a & 0 & 0 & 2d & a & 0 \\ 0 & 0 & 0 & 0 & a & 0 & a & 2d & a \\ 0 & 0 & 0 & 0 & 0 & a & 0 & a & 2d \end{array} \right]. \quad (9.9)$$

The partition into 3×3 sub matrices shows that \mathbf{T}_2 is block tridiagonal.

Properties of \mathbf{T}_2 can be derived from properties of $\mathbf{T}_1 = \text{tridiagonal}(a, d, a)$ by using properties of the Kronecker product.

9.1 The Kronecker Product

Definition 9.3 (Kronecker product)

For any positive integers p, q, r, s we define the **Kronecker product** of two matrices $\mathbf{A} \in \mathbb{R}^{p \times q}$ and $\mathbf{B} \in \mathbb{R}^{r \times s}$ as a matrix $\mathbf{C} \in \mathbb{R}^{pr \times qs}$ given in block form as

$$\mathbf{C} = \left[\begin{array}{cccc} \mathbf{Ab}_{1,1} & \mathbf{Ab}_{1,2} & \cdots & \mathbf{Ab}_{1,s} \\ \mathbf{Ab}_{2,1} & \mathbf{Ab}_{2,2} & \cdots & \mathbf{Ab}_{2,s} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{Ab}_{r,1} & \mathbf{Ab}_{r,2} & \cdots & \mathbf{Ab}_{r,s} \end{array} \right].$$

We denote the Kronecker product of \mathbf{A} and \mathbf{B} by $\mathbf{C} = \mathbf{A} \otimes \mathbf{B}$.

This definition of the Kronecker product is known more precisely as the **left Kronecker product**. In the literature one often finds the **right Kronecker product** which in our notation is given by $\mathbf{B} \otimes \mathbf{A}$.

The Kronecker product $\mathbf{u} \otimes \mathbf{v} = [\mathbf{u}^T v_1, \dots, \mathbf{u}^T v_r]^T$ of two column vectors $\mathbf{u} \in \mathbb{R}^p$ and $\mathbf{v} \in \mathbb{R}^r$ is a column vector of length $p \cdot r$.

As examples of Kronecker products which are relevant for our discussion, if

$$\mathbf{T}_1 = \begin{bmatrix} d & a & 0 \\ a & d & a \\ 0 & a & d \end{bmatrix} \quad \text{and} \quad \mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

then

$$\mathbf{T}_1 \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{T}_1 = \begin{bmatrix} \mathbf{T}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{T}_1 \end{bmatrix} + \begin{bmatrix} d\mathbf{I} & a\mathbf{I} & \mathbf{0} \\ a\mathbf{I} & d\mathbf{I} & a\mathbf{I} \\ \mathbf{0} & a\mathbf{I} & d\mathbf{I} \end{bmatrix} = \mathbf{T}_2$$

given by (9.9). The same equation holds for any integer $m \geq 2$

$$\mathbf{T}_1 \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{T}_1 = \mathbf{T}_2, \quad \mathbf{T}_1, \mathbf{I} \in \mathbb{R}^{m \times m}, \quad \mathbf{T}_2 \in \mathbb{R}^{(m^2) \times (m^2)}. \quad (9.10)$$

The sum of two Kronecker products involving the identity matrix is worthy of a special name.

Definition 9.4 (Kronecker sum)

For positive integers r, s, k , let $\mathbf{A} \in \mathbb{R}^{r \times r}$, $\mathbf{B} \in \mathbb{R}^{s \times s}$, and \mathbf{I}_k be the identity matrix of order k . The sum $\mathbf{A} \otimes \mathbf{I}_s + \mathbf{I}_r \otimes \mathbf{B}$ is known as the **Kronecker sum** of \mathbf{A} and \mathbf{B} .

In other words, the 2D test matrix \mathbf{T}_2 is the Kronecker sum involving the 1D test matrix \mathbf{T}_1 .

The following simple arithmetic rules hold for Kronecker products. For scalars λ, μ and matrices $\mathbf{A}, \mathbf{A}_1, \mathbf{A}_2, \mathbf{B}, \mathbf{B}_1, \mathbf{B}_2, \mathbf{C}$ of dimensions such that the operations are defined, we have

$$\begin{aligned} (\lambda \mathbf{A}) \otimes (\mu \mathbf{B}) &= \lambda \mu (\mathbf{A} \otimes \mathbf{B}), \\ (\mathbf{A}_1 + \mathbf{A}_2) \otimes \mathbf{B} &= \mathbf{A}_1 \otimes \mathbf{B} + \mathbf{A}_2 \otimes \mathbf{B}, \\ \mathbf{A} \otimes (\mathbf{B}_1 + \mathbf{B}_2) &= \mathbf{A} \otimes \mathbf{B}_1 + \mathbf{A} \otimes \mathbf{B}_2, \\ (\mathbf{A} \otimes \mathbf{B}) \otimes \mathbf{C} &= \mathbf{A} \otimes (\mathbf{B} \otimes \mathbf{C}). \end{aligned} \quad (9.11)$$

Note however that in general we have $\mathbf{A} \otimes \mathbf{B} \neq \mathbf{B} \otimes \mathbf{A}$, but it can be shown that there are permutation matrices \mathbf{P}, \mathbf{Q} such that $\mathbf{B} \otimes \mathbf{A} = \mathbf{P}(\mathbf{A} \otimes \mathbf{B})\mathbf{Q}$, see [16].

Exercise 9.5 (Properties of Kronecker products)

Prove (9.11).

The following **mixed product rule** is an essential tool for dealing with Kronecker products and sums.

Lemma 9.6 (Mixed product rule)

Suppose $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ are rectangular matrices with dimensions so that the products \mathbf{AC} and \mathbf{BD} are defined. Then the product $(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D})$ is defined and

$$(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) = (\mathbf{AC}) \otimes (\mathbf{BD}). \quad (9.12)$$

Proof. If $\mathbf{B} \in \mathbb{R}^{r,t}$ and $\mathbf{D} \in \mathbb{R}^{t,s}$ for some integers r, s, t , then

$$(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) = \begin{bmatrix} \mathbf{Ab}_{1,1} & \cdots & \mathbf{Ab}_{1,t} \\ \vdots & & \vdots \\ \mathbf{Ab}_{r,1} & \cdots & \mathbf{Ab}_{r,t} \end{bmatrix} \begin{bmatrix} \mathbf{Cd}_{1,1} & \cdots & \mathbf{Cd}_{1,s} \\ \vdots & & \vdots \\ \mathbf{Cd}_{t,1} & \cdots & \mathbf{Cd}_{t,s} \end{bmatrix}.$$

Thus for all i, j

$$((\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}))_{i,j} = \mathbf{AC} \sum_{k=1}^t b_{i,k} d_{k,j} = (\mathbf{AC})(\mathbf{BD})_{i,j} = ((\mathbf{AC}) \otimes (\mathbf{BD}))_{i,j}.$$

□

Using the mixed product rule we obtain the following properties of Kronecker products and sums.

Theorem 9.7 (Properties of Kronecker products)

Suppose for $r, s \in \mathbb{N}$ that $\mathbf{A} \in \mathbb{R}^{r,r}$ and $\mathbf{B} \in \mathbb{R}^{s,s}$ are square matrices with eigenpairs $(\lambda_i, \mathbf{u}_i)$ $i = 1, \dots, r$ and (μ_j, \mathbf{v}_j) , $j = 1, \dots, s$. Moreover, let $\mathbf{F}, \mathbf{V} \in \mathbb{R}^{r \times s}$. Then

1. $(\mathbf{A} \otimes \mathbf{B})^T = \mathbf{A}^T \otimes \mathbf{B}^T$, (this also holds for rectangular matrices).
2. If \mathbf{A} and \mathbf{B} are nonsingular then $\mathbf{A} \otimes \mathbf{B}$ is nonsingular. with $(\mathbf{A} \otimes \mathbf{B})^{-1} = \mathbf{A}^{-1} \otimes \mathbf{B}^{-1}$.
3. If \mathbf{A} and \mathbf{B} are symmetric then $\mathbf{A} \otimes \mathbf{B}$ and $\mathbf{A} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{B}$ are symmetric.
4. $(\mathbf{A} \otimes \mathbf{B})(\mathbf{u}_i \otimes \mathbf{v}_j) = \lambda_i \mu_j (\mathbf{u}_i \otimes \mathbf{v}_j)$, $i = 1, \dots, r$, $j = 1, \dots, s$,
5. $(\mathbf{A} \otimes \mathbf{I}_s + \mathbf{I}_r \otimes \mathbf{B})(\mathbf{u}_i \otimes \mathbf{v}_j) = (\lambda_i + \mu_j)(\mathbf{u}_i \otimes \mathbf{v}_j)$, $i = 1, \dots, r$, $j = 1, \dots, s$,
6. If one of \mathbf{A} , \mathbf{B} is symmetric positive definite and the other is symmetric positive semidefinite then $\mathbf{A} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{B}$ is symmetric positive definite.
7. $\mathbf{AVB}^T = \mathbf{F} \Leftrightarrow (\mathbf{A} \otimes \mathbf{B}) \text{vec}(\mathbf{V}) = \text{vec}(\mathbf{F})$,
8. $\mathbf{AV} + \mathbf{VB}^T = \mathbf{F} \Leftrightarrow (\mathbf{A} \otimes \mathbf{I}_s + \mathbf{I}_r \otimes \mathbf{B}) \text{vec}(\mathbf{V}) = \text{vec}(\mathbf{F})$.

Before giving the simple proofs of this theorem we present some comments.

1. The transpose (or the inverse) of an ordinary matrix product equals the transpose (or the inverse) of the matrices in reverse order. For Kronecker products the order is kept.
2. The eigenvalues of the Kronecker product (or sum) are the product (or sum) of the eigenvalues of the factors. The eigenvectors are the Kronecker products of the eigenvectors of the factors. In particular, the eigenvalues of the test matrix \mathbf{T}_2 are sums of eigenvalues of \mathbf{T}_1 . We will find these eigenvalues in the next section.
3. Since we already know that $\mathbf{T} = \text{tridiag}(-1, 2, -1)$ is positive definite the 2D Poisson matrix $\mathbf{A} = \mathbf{T} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{T}$ is also positive definite.
4. The system $\mathbf{AVB}^T = \mathbf{F}$ in part 7 can be solved by first finding \mathbf{W} from $\mathbf{AW} = \mathbf{F}$, and then finding \mathbf{V} from $\mathbf{BV}^T = \mathbf{W}^T$. This is preferable to solving the much larger linear system $(\mathbf{A} \otimes \mathbf{B}) \text{vec}(\mathbf{V}) = \text{vec}(\mathbf{F})$.
5. A fast way to solve the 2D Poisson problem in the form $\mathbf{TV} + \mathbf{VT} = \mathbf{F}$ will be considered in the next chapter.

Proof.

1. Exercise.
2. By the mixed product rule $(\mathbf{A} \otimes \mathbf{B})(\mathbf{A}^{-1} \otimes \mathbf{B}^{-1}) = (\mathbf{AA}^{-1}) \otimes (\mathbf{BB}^{-1}) = \mathbf{I}_r \otimes \mathbf{I}_s = \mathbf{I}_{rs}$. Thus $(\mathbf{A} \otimes \mathbf{B})$ is nonsingular with the indicated inverse.
3. By 1, $(\mathbf{A} \otimes \mathbf{B})^T = \mathbf{A}^T \otimes \mathbf{B}^T = \mathbf{A} \otimes \mathbf{B}$. Moreover, since then $\mathbf{A} \otimes \mathbf{I}$ and $\mathbf{I} \otimes \mathbf{B}$ are symmetric, their sum is symmetric.
4. $(\mathbf{A} \otimes \mathbf{B})(\mathbf{u}_i \otimes \mathbf{v}_j) = (\mathbf{Au}_i) \otimes (\mathbf{Bv}_j) = (\lambda_i \mathbf{u}_i) \otimes (\mu_j \mathbf{v}_j) = (\lambda_i \mu_j)(\mathbf{u}_i \otimes \mathbf{v}_j)$, for all i, j , where we used the mixed product rule.
5. $(\mathbf{A} \otimes \mathbf{I}_s)(\mathbf{u}_i \otimes \mathbf{v}_j) = \lambda_i(\mathbf{u}_i \otimes \mathbf{v}_j)$, and $(\mathbf{I}_r \otimes \mathbf{B})(\mathbf{u}_i \otimes \mathbf{v}_j) = \mu_j(\mathbf{u}_i \otimes \mathbf{v}_j)$. The result now follows by summing these relations.
6. By 1, $\mathbf{A} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{B}$ is symmetric. Moreover, the eigenvalues $\lambda_i + \mu_j$ are positive since for all i, j , both λ_i and μ_j are nonnegative and one of them is positive. It follows that $\mathbf{A} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{B}$ is symmetric positive definite.

7. We partition \mathbf{V} , \mathbf{F} , and \mathbf{B}^T by columns as $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_s]$, $\mathbf{F} = [\mathbf{f}_1, \dots, \mathbf{f}_s]$ and $\mathbf{B}^T = [\mathbf{b}_1, \dots, \mathbf{b}_s]$. Then we have

$$\begin{aligned} & (\mathbf{A} \otimes \mathbf{B}) \text{vec}(\mathbf{V}) = \text{vec}(\mathbf{F}) \\ & \Leftrightarrow \begin{bmatrix} \mathbf{A}\mathbf{b}_{11} & \cdots & \mathbf{A}\mathbf{b}_{1s} \\ \vdots & & \vdots \\ \mathbf{A}\mathbf{b}_{s1} & \cdots & \mathbf{A}\mathbf{b}_{ss} \end{bmatrix} \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_s \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_s \end{bmatrix} \\ & \Leftrightarrow \mathbf{A} \left[\sum_j b_{1j} \mathbf{v}_j, \dots, \sum_j b_{sj} \mathbf{v}_j \right] = [\mathbf{f}_1, \dots, \mathbf{f}_s] \\ & \Leftrightarrow \mathbf{A}[\mathbf{V}\mathbf{b}_1, \dots, \mathbf{V}\mathbf{b}_s] = \mathbf{F} \quad \Leftrightarrow \quad \mathbf{AVB}^T = \mathbf{F}. \end{aligned}$$

8. This follows immediately from (7) as follows

$$\begin{aligned} & (\mathbf{A} \otimes \mathbf{I}_s + \mathbf{I}_r \otimes \mathbf{B}) \text{vec}(\mathbf{V}) = \text{vec}(\mathbf{F}) \\ & \Leftrightarrow (\mathbf{AVI}_s^T + \mathbf{I}_r \mathbf{VB}^T) = \mathbf{F} \quad \Leftrightarrow \quad \mathbf{AV} + \mathbf{VB}^T = \mathbf{F}. \end{aligned}$$

□

For more on Kronecker products see [16].

9.2 Properties of the 2D Test Matrices

Using Theorem 9.7 we can derive properties of the 2D test matrix \mathbf{T}_2 from those of \mathbf{T}_1 . We need to determine the eigenpairs of \mathbf{T}_1 .

Theorem 9.8 (Eigenpairs of 2D test matrix)

For fixed $m \geq 2$ let \mathbf{T}_2 be the matrix given by (9.8) and let $h = 1/(m+1)$.

1. We have $\mathbf{T}_2 \mathbf{x}_{j,k} = \lambda_{j,k} \mathbf{x}_{j,k}$ for $j, k = 1, \dots, m$, where

$$\mathbf{x}_{j,k} = \mathbf{s}_j \otimes \mathbf{s}_k, \tag{9.13}$$

$$\mathbf{s}_j = [\sin(j\pi h), \sin(2j\pi h), \dots, \sin(mj\pi h)]^T, \tag{9.14}$$

$$\lambda_{j,k} = 2d + 2a \cos(j\pi h) + 2a \cos(k\pi h). \tag{9.15}$$

2. The eigenvectors are orthogonal

$$\mathbf{x}_{j,k}^T \mathbf{x}_{p,q} = \frac{1}{4h^2} \delta_{j,p} \delta_{k,q}, \quad j, k, p, q = 1, \dots, m. \tag{9.16}$$

3. \mathbf{T}_2 is symmetric positive definite if $d > 0$ and $d \geq 2|a|$.

Proof. By Theorem 9.7 the eigenvalues of $\mathbf{T}_2 = \mathbf{T}_1 \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{T}_1$ are sums of eigenvalues of \mathbf{T}_1 and the eigenvectors are Kronecker products of the eigenvectors of \mathbf{T}_1 . Part 1 now follows from Lemma 1.31. Using the transpose rule, the mixed product rule and (1.37) we find for $j, k, p, q = 1, \dots, m$

$$(\mathbf{s}_j \otimes \mathbf{s}_k)^T (\mathbf{s}_p \otimes \mathbf{s}_q) = (\mathbf{s}_j^T \otimes \mathbf{s}_k^T)(\mathbf{s}_p \otimes \mathbf{s}_q) = (\mathbf{s}_j^T \mathbf{s}_p) \otimes (\mathbf{s}_k^T \mathbf{s}_q) = \frac{1}{4h^2} \delta_{j,p} \delta_{k,q}$$

and part 2 follows. Since \mathbf{T}_2 is symmetric, part 3 will follow if the eigenvalues are positive. But this is true if $d > 0$ and $d \geq 2|a|$. Thus \mathbf{T}_2 is positive definite. \square

Exercise 9.9 (2. derivative matrix is positive definite)

Write down the eigenvalues of $\mathbf{T} = \text{tridiag}(-1, 2, -1)$ using Lemma 1.31 and conclude that \mathbf{T} is symmetric positive definite.

Exercise 9.10 (1D test matrix is positive definite?)

Use Lemma 1.31 to show that the matrix $\mathbf{T}_1 := \text{tridiag}(a, d, a) \in \mathbb{R}^{n \times n}$ is symmetric positive definite if $d > 0$ and $d \geq 2|a|$.

Exercise 9.11 (Eigenvalues for 2D test matrix of order 4)

For $m = 2$ the matrix (9.8) is given by

$$\mathbf{A} = \begin{bmatrix} 2d & a & a & 0 \\ a & 2d & 0 & a \\ a & 0 & 2d & a \\ 0 & a & a & 2d \end{bmatrix}.$$

Show that $\lambda = 2a + 2d$ is an eigenvalue corresponding to the eigenvector $\mathbf{x} = [1, 1, 1, 1]^T$. Verify that apart from a scaling of the eigenvector this agrees with (9.15) and (9.14) for $j = k = 1$ and $m = 2$.

Exercise 9.12 (Nine point scheme for Poisson problem)

Consider the following 9 point difference approximation to the Poisson problem $-\Delta u = f$, $u = 0$ on the boundary of the unit square (cf. (9.1))

$$\begin{aligned} (a) \quad -(\square_h v)_{j,k} &= (\mu f)_{j,k} & j, k = 1, \dots, m \\ (b) \quad 0 &= v_{0,k} = v_{m+1,k} = v_{j,0} = v_{j,m+1}, & j, k = 0, 1, \dots, m+1, \\ (c) \quad -(\square_h v)_{j,k} &= [20v_{j,k} - 4v_{j-1,k} - 4v_{j,k-1} - 4v_{j+1,k} - 4v_{j,k+1} \\ &\quad - v_{j-1,k-1} - v_{j+1,k-1} - v_{j-1,k+1} - v_{j+1,k+1}] / (6h^2), \\ (d) \quad (\mu f)_{j,k} &= [8f_{j,k} + f_{j-1,k} + f_{j,k-1} + f_{j+1,k} + f_{j,k+1}] / 12. \end{aligned} \tag{9.17}$$

- a) Write down the 4-by-4 system we obtain for $m = 2$.
- b) Find $v_{j,k}$ for $j, k = 1, 2$, if $f(x, y) = 2\pi^2 \sin(\pi x) \sin(\pi y)$ and $m = 2$. Answer: $v_{j,k} = 5\pi^2/66$.

It can be shown that (9.17) defines an $O(h^4)$ approximation to (9.1).

Exercise 9.13 (Matrix equation for nine point scheme)

Consider the nine point difference approximation to (9.1) given by (9.17) in Problem 9.12.

- a) Show that (9.17) is equivalent to the matrix equation

$$\mathbf{T}\mathbf{V} + \mathbf{V}\mathbf{T} - \frac{1}{6}\mathbf{T}\mathbf{V}\mathbf{T} = h^2\mu\mathbf{F}. \quad (9.18)$$

Here $\mu\mathbf{F}$ has elements $(\mu f)_{j,k}$ given by (9.17d) and $\mathbf{T} = \text{tridiag}(-1, 2, -1)$.

- b) Show that the standard form of the matrix equation (9.18) is $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A} = \mathbf{T} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{T} - \frac{1}{6}\mathbf{T} \otimes \mathbf{T}$, $\mathbf{x} = \text{vec}(\mathbf{V})$, and $\mathbf{b} = h^2 \text{vec}(\mu\mathbf{F})$.

Exercise 9.14 (Biharmonic equation)

Consider the biharmonic equation

$$\begin{aligned} \Delta^2 u(s, t) &:= \Delta(\Delta u(s, t)) = f(s, t) & (s, t) \in \Omega, \\ u(s, t) &= 0, \quad \Delta u(s, t) = 0 & (s, t) \in \partial\Omega. \end{aligned} \quad (9.19)$$

Here Ω is the open unit square. The condition $\Delta u = 0$ is called the Navier boundary condition. Moreover, $\Delta^2 u = u_{xxxx} + 2u_{xxyy} + u_{yyyy}$.

- a) Let $v = -\Delta u$. Show that (9.19) can be written as a system

$$\begin{aligned} -\Delta v(s, t) &= f(s, t) & (s, t) \in \Omega \\ -\Delta u(s, t) &= v(s, t) & (s, t) \in \Omega \\ u(s, t) &= 0 & (s, t) \in \partial\Omega. \end{aligned} \quad (9.20)$$

- b) Discretizing, using (9.3), with $\mathbf{T} = \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{m \times m}$, $h = 1/(m+1)$, and $\mathbf{F} = (f(jh, kh))_{j,k=1}^m$ we get two matrix equations

$$\mathbf{T}\mathbf{V} + \mathbf{V}\mathbf{T} = h^2\mathbf{F}, \quad \mathbf{T}\mathbf{U} + \mathbf{U}\mathbf{T} = h^2\mathbf{V}.$$

Show that

$$(\mathbf{T} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{T}) \text{vec}(\mathbf{V}) = h^2 \text{vec}(\mathbf{F}), \quad (\mathbf{T} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{T}) \text{vec}(\mathbf{U}) = h^2 \text{vec}(\mathbf{V}).$$

and hence $\mathbf{A} = (\mathbf{T} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{T})^2$ is the matrix for the standard form of the discrete biharmonic equation.

- c) Show that with $n = m^2$ the vector form and standard form of the systems in
 b) can be written

$$\mathbf{T}^2 \mathbf{U} + 2\mathbf{T}\mathbf{U}\mathbf{T} + \mathbf{U}\mathbf{T}^2 = h^4 \mathbf{F} \quad \text{and} \quad \mathbf{A}\mathbf{x} = \mathbf{b}, \quad (9.21)$$

where $\mathbf{A} = \mathbf{T}^2 \otimes \mathbf{I} + 2\mathbf{T} \otimes \mathbf{T} + \mathbf{I} \otimes \mathbf{T}^2 \in \mathbb{R}^{n \times n}$, $\mathbf{x} = \text{vec}(\mathbf{U})$, and $\mathbf{b} = h^4 \text{vec}(\mathbf{F})$.

- d) Determine the eigenvalues and eigenvectors of the matrix \mathbf{A} in c) and show that it is symmetric positive definite. Also determine the bandwidth of \mathbf{A} .
- e) Suppose we want to solve the standard form equation $\mathbf{A}\mathbf{x} = \mathbf{b}$. We have two representations for the matrix \mathbf{A} , the product one in b) and the one in c). Which one would you prefer for the basis of an algorithm? Why?

9.3 Review Questions

9.3.1 Consider the Poisson matrix.

- Write this matrix as a Kronecker sum,
- how are its eigenvalues and eigenvectors related to the second derivative matrix?
- is it symmetric? positive definite?

9.3.2 What are the eigenpairs of $\mathbf{T}_1 := \text{tridiagonal}(a, d, a)$?

9.3.3 What are the inverse and transpose of a Kronecker product?

- 9.3.4**
- give an economical general way to solve the linear system $(\mathbf{A} \otimes \mathbf{B}) \text{vec}(\mathbf{V}) = \text{vec}(\mathbf{F})$?
 - Same for $(\mathbf{A} \otimes \mathbf{I}_s + \mathbf{I}_r \otimes \mathbf{B}) \text{vec}(\mathbf{V}) = \text{vec}(\mathbf{F})$.

Chapter 10

Fast Direct Solution of a Large Linear System

10.1 Algorithms for a Banded Positive Definite System

In this chapter we present a fast method for solving $\mathbf{A}\mathbf{x} = \mathbf{b}$, where \mathbf{A} is the Poisson matrix (9.7). Thus, for $n = 9$

$$\begin{aligned}\mathbf{A} &= \left[\begin{array}{ccc|ccc|ccc} 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 4 & 0 & 0 & -1 & 0 & 0 & 0 \\ \hline -1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\ \hline 0 & 0 & 0 & -1 & 0 & 0 & 4 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 \end{array} \right] \\ &= \left[\begin{array}{ccc} \mathbf{T} + 2\mathbf{I} & -\mathbf{I} & \mathbf{0} \\ -\mathbf{I} & \mathbf{T} + 2\mathbf{I} & -\mathbf{I} \\ \mathbf{0} & -\mathbf{I} & \mathbf{T} + 2\mathbf{I} \end{array} \right],\end{aligned}$$

where $\mathbf{T} = \text{tridiag}(-1, 2, -1)$. For the matrix \mathbf{A} we know by now that

1. It is symmetric positive definite.
2. It is banded.
3. It is block-tridiagonal.

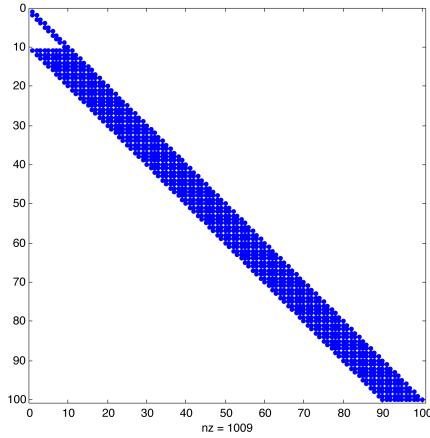


Figure 10.1: Fill-inn in the Cholesky factor of the Poisson matrix ($n = 100$).

4. We know the eigenvalues and eigenvectors of \mathbf{A} .
5. The eigenvectors are orthogonal.

10.1.1 Cholesky factorization

Since \mathbf{A} is symmetric positive definite we can use the Cholesky factorization $\mathbf{A} = \mathbf{L}\mathbf{L}^T$, with \mathbf{L} lower triangular, to solve $\mathbf{A}\mathbf{x} = \mathbf{b}$. Since \mathbf{A} and \mathbf{L} has the same bandwidth $d = \sqrt{n}$ the complexity of this factorization is $O(nd^2) = O(n^2)$. We need to store \mathbf{A} , and this can be done in sparse form.

The nonzero elements in \mathbf{L} are shown in Figure 10.1 for $n = 100$. Note that most of the zeros between the diagonals in \mathbf{A} have become nonzero in \mathbf{L} . This is known as **fill-inn**.

10.1.2 Block LU factorization of a block tridiagonal matrix

The Poisson matrix has a block tridiagonal structure. Consider finding the block LU factorization of a block tridiagonal matrix. We are looking for a factorization of the form

$$\begin{bmatrix} D_1 & C_1 & & \\ A_1 & D_2 & C_2 & \\ & \ddots & \ddots & \ddots & \\ & & A_{m-2} & D_{m-1} & C_{m-1} \\ & & & A_{m-1} & D_m \end{bmatrix} = \begin{bmatrix} I & & & \\ L_1 & I & & \\ & \ddots & \ddots & \\ & & L_{m-1} & I \end{bmatrix} \begin{bmatrix} U_1 & C_1 & & \\ & \ddots & \ddots & \\ & & U_{m-1} & C_{m-1} \\ & & & U_m \end{bmatrix}. \quad (10.1)$$

Here D_1, \dots, D_m and U_1, \dots, U_m are square matrices while $A_1, \dots, A_{m-1}, L_1, \dots, L_{m-1}$ and C_1, \dots, C_{m-1} can be rectangular.

Using block multiplication the formulas (1.16) generalize to

$$\mathbf{U}_1 = \mathbf{D}_1, \quad \mathbf{L}_k = \mathbf{A}_k \mathbf{U}_k^{-1}, \quad \mathbf{U}_{k+1} = \mathbf{D}_{k+1} - \mathbf{L}_k \mathbf{C}_k, \quad k = 1, 2, \dots, m-1. \quad (10.2)$$

To solve the system $\mathbf{Ax} = \mathbf{b}$ we partition \mathbf{b} conformally with \mathbf{A} in the form $\mathbf{b}^T = [\mathbf{b}_1^T, \dots, \mathbf{b}_m^T]$. The formulas for solving $\mathbf{Ly} = \mathbf{b}$ and $\mathbf{Ux} = \mathbf{y}$ are as follows:

$$\begin{aligned} \mathbf{y}_1 &= \mathbf{b}_1, & \mathbf{y}_k &= \mathbf{b}_k - \mathbf{L}_{k-1} \mathbf{y}_{k-1}, & k &= 2, 3, \dots, m, \\ \mathbf{x}_m &= \mathbf{U}_m^{-1} \mathbf{y}_m, & \mathbf{x}_k &= \mathbf{U}_k^{-1} (\mathbf{y}_k - \mathbf{C}_k \mathbf{x}_{k+1}), & k &= m-1, \dots, 2, 1. \end{aligned} \quad (10.3)$$

The solution is then $\mathbf{x}^T = [\mathbf{x}_1^T, \dots, \mathbf{x}_m^T]$. To find \mathbf{L}_k in (10.2) we solve the linear systems $\mathbf{L}_k \mathbf{U}_k = \mathbf{A}_k$. Similarly we need to solve a linear system to find \mathbf{x}_k in (10.3).

The number of arithmetic operations using block factorizations is $O(n^2)$, asymptotically the same as for Cholesky factorization. However we only need to store the $m \times m$ blocks and using matrix operations can be an advantage.

10.1.3 Other methods

Other methods include

- Iterative methods, (we study this in Chapters 11 and 12),
- multigrid. See [9],
- fast solvers based on diagonalization and the fast Fourier transform. See Sections 10.2, 10.3.

10.2 A Fast Poisson Solver based on Diagonalization

The algorithm we now derive will only require $O(n^{3/2})$ arithmetic operations and we only need to work with matrices of order m . Using the fast Fourier transform the number of arithmetic operations can be reduced further to $O(n \log n)$.

To start we recall that $\mathbf{Ax} = \mathbf{b}$ can be written as a matrix equation in the form (cf. (9.5))

$$\mathbf{TV} + \mathbf{VT} = h^2 \mathbf{F} \quad \text{with} \quad h = 1/(m+1),$$

where $\mathbf{T} = \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{m \times m}$ is the second derivative matrix, $\mathbf{V} = (v_{jk}) \in \mathbb{R}^{m \times m}$ are the unknowns, and $\mathbf{F} = (f_{jk}) = (f(jh, kh)) \in \mathbb{R}^{m \times m}$ contains function values.

Recall that the eigenpairs of \mathbf{T} are given by

$$\begin{aligned}\mathbf{T}\mathbf{s}_j &= \lambda_j \mathbf{s}_j, \quad j = 1, \dots, m, \\ \mathbf{s}_j &= [\sin(j\pi h), \sin(2j\pi h), \dots, \sin(mj\pi h)]^T, \\ \lambda_j &= 2 - 2 \cos(j\pi h) = 4 \sin^2(j\pi h/2), \quad h = 1/(m+1), \\ \mathbf{s}_j^T \mathbf{s}_k &= \delta_{jk}/(2h) \text{ for all } j, k.\end{aligned}$$

Let

$$\mathbf{S} := [\mathbf{s}_1, \dots, \mathbf{s}_m] = [\sin(jk\pi h)]_{j,k=1}^m \in \mathbb{R}^{m \times m}, \quad \mathbf{D} = \text{diag}(\lambda_1, \dots, \lambda_m). \quad (10.4)$$

Then

$$\mathbf{T}\mathbf{S} = [\mathbf{T}\mathbf{s}_1, \dots, \mathbf{T}\mathbf{s}_m] = [\lambda_1 \mathbf{s}_1, \dots, \lambda_m \mathbf{s}_m] = \mathbf{S}\mathbf{D}, \quad \mathbf{S}^2 = \mathbf{S}^T \mathbf{S} = \frac{1}{2h} \mathbf{I}.$$

Define $\mathbf{X} \in \mathbb{R}^{m \times m}$ by $\mathbf{V} = \mathbf{S}\mathbf{X}\mathbf{S}$, where \mathbf{V} is the solution of $\mathbf{T}\mathbf{V} + \mathbf{V}\mathbf{T} = h^2 \mathbf{F}$. Then

$$\begin{aligned}\mathbf{T}\mathbf{V} + \mathbf{V}\mathbf{T} &= h^2 \mathbf{F} \\ \xrightleftharpoons[\mathbf{V}=\mathbf{S}\mathbf{X}\mathbf{S}]{\mathbf{S}^T\mathbf{S}} \mathbf{T}\mathbf{S}\mathbf{X}\mathbf{S} + \mathbf{S}\mathbf{X}\mathbf{T}\mathbf{S} &= h^2 \mathbf{F} \\ \xrightleftharpoons[\mathbf{S}^2=\mathbf{S}\mathbf{D}]{\mathbf{S}^2\mathbf{X}\mathbf{S}^2 + \mathbf{S}^2\mathbf{X}\mathbf{S}\mathbf{T}\mathbf{S}} \mathbf{S}\mathbf{T}\mathbf{S}\mathbf{X}\mathbf{S}^2 + \mathbf{S}^2\mathbf{X}\mathbf{T}\mathbf{S}\mathbf{S} &= h^2 \mathbf{S}\mathbf{F}\mathbf{S} = h^2 \mathbf{G} \\ \xrightleftharpoons[\mathbf{S}^2=\mathbf{I}/(2h)]{\mathbf{S}^2\mathbf{D}\mathbf{X}\mathbf{S}^2 + \mathbf{S}^2\mathbf{X}\mathbf{S}^2\mathbf{D}} \mathbf{S}^2\mathbf{D}\mathbf{X}\mathbf{S}^2 + \mathbf{S}^2\mathbf{X}\mathbf{S}^2\mathbf{D} &= h^2 \mathbf{G} \\ \mathbf{D}\mathbf{X} + \mathbf{X}\mathbf{D} &= 4h^4 \mathbf{G}.\end{aligned}$$

Since \mathbf{D} is diagonal, the equation $\mathbf{D}\mathbf{X} + \mathbf{X}\mathbf{D} = 4h^4 \mathbf{G}$, is easy to solve. For the j, k element we find

$$(\mathbf{D}\mathbf{X} + \mathbf{X}\mathbf{D})_{j,k} = \sum_{\ell=1}^m d_{j,\ell} x_{\ell,k} + \sum_{\ell=1}^m x_{j,\ell} d_{\ell,k} = \lambda_j x_{j,k} + \lambda_k x_{j,k}$$

so that for all j, k

$$x_{jk} = 4h^4 g_{jk}/(\lambda_j + \lambda_k) = h^4 g_{jk}/(\sigma_j + \sigma_k), \quad \sigma_j := \lambda_j/4 = \sin^2(j\pi h/2).$$

Thus to find \mathbf{V} we compute

1. $\mathbf{G} = \mathbf{S}\mathbf{F}\mathbf{S}$,
2. $x_{j,k} = h^4 g_{jk}/(\sigma_j + \sigma_k), \quad j, k = 1, \dots, m,$
3. $\mathbf{V} = \mathbf{S}\mathbf{X}\mathbf{S}$.

We can compute mX , \mathbf{S} and the σ 's without using loops. Using outer products, element by element division, and raising a matrix element by element to a power we find

$$\mathbf{X} = h^4 \mathbf{G}/\mathbf{M}, \text{ where } \mathbf{M} := \begin{bmatrix} \sigma_1 \\ \vdots \\ \sigma_m \end{bmatrix} [1, \dots, 1] + \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} [\sigma_1, \dots, \sigma_m],$$

$$\mathbf{S} = \sin(\pi h \begin{bmatrix} \frac{1}{2} \\ \vdots \\ \frac{1}{m} \end{bmatrix} [1 \ 2 \ \dots \ m]), \quad \boldsymbol{\sigma} = \sin\left(\frac{\pi h}{2} \begin{bmatrix} \frac{1}{2} \\ \vdots \\ \frac{1}{m} \end{bmatrix}\right) \wedge 2.$$

We now get the following algorithm to solve numerically the Poisson problem $-\Delta u = f$ on $\Omega = (0, 1)^2$ and $u = 0$ on $\partial\Omega$ using the 5-point scheme, i.e., let $m \in \mathbb{N}$, $h = 1/(m+1)$, and $\mathbf{F} = (f(jh, kh)) \in \mathbb{R}^{m \times m}$. We compute $\mathbf{V} \in \mathbb{R}^{(m+2) \times (m+2)}$ using diagonalization of $\mathbf{T} = \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{m \times m}$.

Algorithm 10.1 (Fast Poisson solver)

```

1 function V=fastpoisson(F)
2 %function V=fastpoisson(F)
3 m=length(F); h=1/(m+1); hv=pi*h*(1:m)';
4 sigma=sin(hv/2).^2;
5 S=sin(hv*(1:m));
6 G=S*F*S;
7 X=h.^4*G./(sigma*ones(1,m)+ ones(m,1)*sigma)';
8 V=zeros(m+2,m+2);
9 V(2:m+1,2:m+1)=S*X*S;

```

The formulas are fully vectorized. Since the 6th line in Algorithm 10.1 only requires $O(m^2)$ arithmetic operations the complexity of this algorithm is for large m determined by the 4 m -by- m matrix multiplications and is given by $O(4 \times 2m^3) = O(8n^{3/2})$.¹⁸ The method is very fast and will be used as a preconditioner for a more complicated problem in Chapter 12. In 2012 it took about 0.2 seconds on a laptop to find the 10^6 unknowns $v_{j,k}$ on a 1000×1000 grid.

10.3 A Fast Poisson Solver based on the discrete sine and Fourier transforms

In Algorithm 10.1 we need to compute the product of the sine matrix $\mathbf{S} \in \mathbb{R}^{m \times m}$ given by (10.4) and a matrix $\mathbf{A} \in \mathbb{R}^{m \times m}$. Since the matrices are m -by- m this will normally require $O(m^3)$ operations. In this section we show that it is possible to calculate the products \mathbf{SA} and \mathbf{AS} in $O(m^2 \log_2 m)$ operations.

¹⁸It is possible to compute \mathbf{V} using only two matrix multiplications and hence reduce the complexity to $O(4n^{3/2})$. This is detailed in Problem 10.8.

We need to discuss certain transforms known as the **discrete sine transform**, the **discrete Fourier transform** and the **fast Fourier transform**. In addition we have the **discrete cosine transform** which will not be discussed here. These transforms are of independent interest. They have applications to signal processing and image analysis, and are often used when one is dealing with discrete samples of data on a computer.

10.3.1 The discrete sine transform (DST)

Given $\mathbf{v} = [v_1, \dots, v_m]^T \in \mathbb{R}^m$ we say that the vector $\mathbf{w} = [w_1, \dots, w_m]^T$ given by

$$w_j = \sum_{k=1}^m \sin\left(\frac{jk\pi}{m+1}\right) v_k, \quad j = 1, \dots, m$$

is the **discrete sine transform** (DST) of \mathbf{v} . In matrix form we can write the DST as the matrix times vector $\mathbf{w} = \mathbf{S}\mathbf{v}$, where \mathbf{S} is the sine matrix given by (10.4). We can then identify the matrix $\mathbf{B} = \mathbf{S}\mathbf{A}$ as the DST of $\mathbf{A} \in \mathbb{R}^{m,n}$, i.e. as the DST of the columns of \mathbf{A} . The product $\mathbf{B} = \mathbf{A}\mathbf{S}$ can also be interpreted as a DST. Indeed, since \mathbf{S} is symmetric we have $\mathbf{B} = (\mathbf{S}\mathbf{A}^T)^T$ which means that \mathbf{B} is the transpose of the DST of the rows of \mathbf{A} . It follows that we can compute the unknowns \mathbf{V} in Algorithm 10.1 by carrying out discrete sine transforms on 4 m -by- m matrices in addition to the computation of \mathbf{X} .

10.3.2 The discrete Fourier transform (DFT)



Jean Baptiste Joseph Fourier, 1768 - 1830.

The fast computation of the DST is based on its relation to the discrete Fourier transform (DFT) and the fact that the DFT can be computed by a technique known as the fast Fourier transform (FFT). To define the DFT let for $N \in \mathbb{N}$

$$\omega_N = \exp^{-2\pi i/N} = \cos(2\pi/N) - i \sin(2\pi/N), \quad (10.5)$$

where $i = \sqrt{-1}$ is the imaginary unit. Given $\mathbf{y} = [y_1, \dots, y_N]^T \in \mathbb{R}^N$ we say that

$\mathbf{z} = [z_1, \dots, z_N]^T$ given by

$$\mathbf{z} = \mathbf{F}_N \mathbf{y}, \quad z_{j+1} = \sum_{k=0}^{N-1} \omega_N^{jk} y_{k+1}, \quad j = 0, \dots, N-1$$

is the **discrete Fourier transform** (DFT) of \mathbf{y} . We can write this as a matrix times vector product $\mathbf{z} = \mathbf{F}_N \mathbf{y}$, where the **Fourier matrix** $\mathbf{F}_N \in \mathbb{C}^{N \times N}$ has elements ω_N^{jk} , $j, k = 0, 1, \dots, N-1$. For a matrix we say that $\mathbf{B} = \mathbf{F}_N \mathbf{A}$ is the DFT of \mathbf{A} .

As an example, since

$$\omega_4 = \exp^{-2\pi i/4} = \cos(\pi/2) - i \sin(\pi/2) = -i$$

we find $\omega_4^2 = (-i)^2 = -1$, $\omega_4^3 = (-i)(-1) = i$, $\omega_4^4 = (-1)^2 = 1$, $\omega_4^6 = i^2 = -1$, $\omega_4^9 = i^3 = -i$, and so

$$\mathbf{F}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega_4 & \omega_4^2 & \omega_4^3 \\ 1 & \omega_4^2 & \omega_4^4 & \omega_4^6 \\ 1 & \omega_4^3 & \omega_4^6 & \omega_4^9 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{bmatrix}. \quad (10.6)$$

The following lemma shows how the discrete sine transform of order m can be computed from the discrete Fourier transform of order $2m+2$. We recall that for any complex number w

$$\sin w = \frac{e^{iw} - e^{-iw}}{2i}.$$

Lemma 10.2 (Sine transform as Fourier transform)

Given a positive integer m and a vector $\mathbf{x} \in \mathbb{R}^m$. Component k of \mathbf{Sx} is equal to $i/2$ times component $k+1$ of $\mathbf{F}_{2m+2}\mathbf{z}$ where

$$\mathbf{z}^T = [0, \mathbf{x}^T, 0, -\mathbf{x}_B^T] \in \mathbb{R}^{2m+2}, \quad \mathbf{x}_B^T := [x_m, \dots, x_2, x_1].$$

In symbols

$$(\mathbf{Sx})_k = \frac{i}{2} (\mathbf{F}_{2m+2}\mathbf{z})_{k+1}, \quad k = 1, \dots, m.$$

Proof. Let $\omega = \omega_{2m+2} = e^{-2\pi i/(2m+2)} = e^{-\pi i/(m+1)}$. We note that

$$\omega^{jk} = e^{-\pi ijk/(m+1)}, \quad \omega^{(2m+2-j)k} = e^{-2\pi i} e^{\pi ijk/(m+1)} = e^{\pi ijk/(m+1)}.$$

Component $k + 1$ of $\mathbf{F}_{2m+2}\mathbf{z}$ is then given by

$$\begin{aligned} (\mathbf{F}_{2m+2}\mathbf{z})_{k+1} &= \sum_{j=0}^{2m-1} \omega^{jk} z_{j+1} = \sum_{j=1}^m x_j \omega^{jk} - \sum_{j=1}^m x_j \omega^{(2m+2-j)k} \\ &= \sum_{j=1}^m x_j (e^{-\pi i j k / (m+1)} - e^{\pi i j k / (m+1)}) \\ &= -2i \sum_{j=1}^m x_j \sin\left(\frac{jk\pi}{m+1}\right) = -2i(\mathbf{S}_m \mathbf{x})_k. \end{aligned}$$

Dividing both sides by $-2i$ and noting $-1/(2i) = -i/(2i^2) = i/2$, proves the lemma. \square

It follows that we can compute the DST of length m by extracting m components from the DFT of length $N = 2m + 2$.

10.3.3 The fast Fourier transform (FFT)

From a linear algebra viewpoint the fast Fourier transform is a quick way to compute the matrix- vector product $\mathbf{F}_N \mathbf{y}$. Suppose N is even. The key to the FFT is a connection between \mathbf{F}_N and $\mathbf{F}_{N/2}$ which makes it possible to compute the FFT of order N as two FFT's of order $N/2$. By repeating this process we can reduce the number of arithmetic operations to compute a DFT from $O(N^2)$ to $O(N \log_2 N)$.

Suppose N is even. The connection between \mathbf{F}_N and $\mathbf{F}_{N/2}$ involves a permutation matrix $\mathbf{P}_N \in \mathbb{R}^{N \times N}$ given by

$$\mathbf{P}_N = [\mathbf{e}_1, \mathbf{e}_3, \dots, \mathbf{e}_{N-1}, \mathbf{e}_2, \mathbf{e}_4, \dots, \mathbf{e}_N],$$

where the $\mathbf{e}_k = (\delta_{j,k})$ are unit vectors. If \mathbf{A} is a matrix with N columns $[\mathbf{a}_1, \dots, \mathbf{a}_N]$ then

$$\mathbf{A}\mathbf{P}_N = [\mathbf{a}_1, \mathbf{a}_3, \dots, \mathbf{a}_{N-1}, \mathbf{a}_2, \mathbf{a}_4, \dots, \mathbf{a}_N],$$

i.e. post multiplying \mathbf{A} by \mathbf{P}_N permutes the columns of \mathbf{A} so that all the odd-indexed columns are followed by all the even-indexed columns. For example we have from (10.6)

$$\mathbf{P}_4 = [\mathbf{e}_1 \ \mathbf{e}_3 \ \mathbf{e}_2 \ \mathbf{e}_4] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{F}_4 \mathbf{P}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & -i & i \\ 1 & 1 & -1 & -1 \\ 1 & -1 & i & -i \end{bmatrix},$$

where we have indicated a certain block structure of $\mathbf{F}_4 \mathbf{P}_4$. These blocks can be related to the 2-by-2 matrix \mathbf{F}_2 . We define the diagonal scaling matrix \mathbf{D}_2 by

$$\mathbf{D}_2 = \text{diag}(1, \omega_4) = \begin{bmatrix} 1 & 0 \\ 0 & -i \end{bmatrix}.$$

Since $\omega_2 = \exp^{-2\pi i/2} = -1$ we find

$$\mathbf{F}_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad \mathbf{D}_2 \mathbf{F}_2 = \begin{bmatrix} 1 & 1 \\ -i & i \end{bmatrix},$$

and we see that

$$\mathbf{F}_4 \mathbf{P}_4 = \left[\begin{array}{c|c} \mathbf{F}_2 & \mathbf{D}_2 \mathbf{F}_2 \\ \hline \mathbf{F}_2 & -\mathbf{D}_2 \mathbf{F}_2 \end{array} \right].$$

This result holds in general.

Theorem 10.3 (Fast Fourier transform)

If $N = 2m$ is even then

$$\mathbf{F}_{2m} \mathbf{P}_{2m} = \left[\begin{array}{c|c} \mathbf{F}_m & \mathbf{D}_m \mathbf{F}_m \\ \hline \mathbf{F}_m & -\mathbf{D}_m \mathbf{F}_m \end{array} \right], \quad (10.7)$$

where

$$\mathbf{D}_m = \text{diag}(1, \omega_N, \omega_N^2, \dots, \omega_N^{m-1}). \quad (10.8)$$

Proof. Fix integers p, q with $1 \leq p, q \leq m$ and set $j := p - 1$ and $k := q - 1$. Since

$$\omega_m^m = 1, \quad \omega_{2m}^{2k} = \omega_m^k, \quad \omega_{2m}^m = -1, \quad (\mathbf{F}_m)_{p,q} = \omega_m^{jk}, \quad (\mathbf{D}_m \mathbf{F}_m)_{p,q} = \omega_{2m}^j \omega_m^{jk},$$

we find by considering elements in the four sub-blocks in turn

$$\begin{aligned} (\mathbf{F}_{2m} \mathbf{P}_{2m})_{p,q} &= \omega_{2m}^{j(2k)} &= \omega_m^{jk}, \\ (\mathbf{F}_{2m} \mathbf{P}_{2m})_{p+m,q} &= \omega_{2m}^{(j+m)(2k)} &= \omega_m^{(j+m)k} &= \omega_m^{jk}, \\ (\mathbf{F}_{2m} \mathbf{P}_{2m})_{p,q+m} &= \omega_{2m}^{j(2k+1)} &= \omega_{2m}^j \omega_m^{jk}, \\ (\mathbf{F}_{2m} \mathbf{P}_{2m})_{p+m,q+m} &= \omega_{2m}^{(j+m)(2k+1)} &= \omega_{2m}^{j+m} \omega_m^{(j+m)k} &= -\omega_{2m}^j \omega_m^{jk}. \end{aligned}$$

It follows that the four m -by- m blocks of $\mathbf{F}_{2m} \mathbf{P}_{2m}$ have the required structure.

□

Using Theorem 10.3 we can carry out the DFT as a block multiplication. Let $\mathbf{y} \in \mathbb{R}^{2m}$ and set $\mathbf{w} = \mathbf{P}_{2m}^T \mathbf{y} = [\mathbf{w}_1^T, \mathbf{w}_2^T]^T$, where

$$\mathbf{w}_1^T = [y_1, y_3, \dots, y_{2m-1}], \quad \mathbf{w}_2^T = [y_2, y_4, \dots, y_{2m}].$$

Then

$$\begin{aligned}\mathbf{F}_{2m}\mathbf{y} &= \mathbf{F}_{2m}\mathbf{P}_{2m}\mathbf{P}_{2m}^T\mathbf{y} = \mathbf{F}_{2m}\mathbf{P}_{2m}\mathbf{w} \\ &= \left[\begin{array}{c|c} \mathbf{F}_m & \mathbf{D}_m\mathbf{F}_m \\ \hline \mathbf{F}_m & -\mathbf{D}_m\mathbf{F}_m \end{array} \right] \left[\begin{array}{c} \mathbf{w}_1 \\ \mathbf{w}_2 \end{array} \right] = \left[\begin{array}{c} \mathbf{q}_1 + \mathbf{q}_2 \\ \mathbf{q}_1 - \mathbf{q}_2 \end{array} \right],\end{aligned}$$

where

$$\mathbf{q}_1 = \mathbf{F}_m\mathbf{w}_1, \quad \text{and} \quad \mathbf{q}_2 = \mathbf{D}_m(\mathbf{F}_m\mathbf{w}_2).$$

In order to compute $\mathbf{F}_{2m}\mathbf{y}$ we need to compute $\mathbf{F}_m\mathbf{w}_1$ and $\mathbf{F}_m\mathbf{w}_2$. Thus, by combining two FFT's of order m we obtain an FFT of order $2m$. If $n = 2^k$ then this process can be applied recursively as in the following Matlab function:

Algorithm 10.4 (Recursive FFT)

```

1 function z=fftrec(y)
2 %function z=fftrec(y)
3 y=y(:);
4 n=length(y);
5 if n==1
6     z=y;
7 else
8     q1=fftrec(y(1:2:n-1))
9     q2=exp(-2*pi*i/n).^(0:n/2-1).*fftrec(y(2:2:n))
10    z=[q1+q2; q1-q2];
11 end

```

Statement 3 is included so that the input $\mathbf{y} \in \mathbb{R}^n$ can be either a row or column vector, while the output \mathbf{z} is a column vector.

Such a recursive version of FFT is useful for testing purposes, but is much too slow for large problems. A challenge for FFT code writers is to develop nonrecursive versions and also to handle efficiently the case where N is not a power of two. We refer to [34] for further details.

The complexity of the FFT is given by $\gamma N \log_2 N$ for some constant γ independent of N . To show this for the special case when N is a power of two let x_k be the complexity (the number of arithmetic operations) when $N = 2^k$. Since we need two FFT's of order $N/2 = 2^{k-1}$ and a multiplication with the diagonal matrix $\mathbf{D}_{N/2}$, it is reasonable to assume that $x_k = 2x_{k-1} + \gamma 2^k$ for some constant γ independent of k . Since $x_0 = 0$ we obtain by induction on k that $x_k = \gamma k 2^k$. Indeed, this holds for $k = 0$ and if $x_{k-1} = \gamma(k-1)2^{k-1}$ then $x_k = 2x_{k-1} + \gamma 2^k = 2\gamma(k-1)2^{k-1} + \gamma 2^k = \gamma k 2^k$. Reasonable implementations of FFT typically have $\gamma \approx 5$, see [34].

The efficiency improvement using the FFT to compute the DFT is spectacular for large N . The direct multiplication $\mathbf{F}_N\mathbf{y}$ requires $O(8n^2)$ arithmetic

operations since complex arithmetic is involved. Assuming that the FFT uses $5N \log_2 N$ arithmetic operations we find for $N = 2^{20} \approx 10^6$ the ratio

$$\frac{8N^2}{5N \log_2 N} \approx 84000.$$

Thus if the FFT takes one second of computing time and the computing time is proportional to the number of arithmetic operations then the direct multiplication would take something like 84000 seconds or 23 hours.

10.3.4 A poisson solver based on the FFT

We now have all the ingredients to compute the matrix products \mathbf{SA} and \mathbf{AS} using FFT's of order $2m + 2$ where m is the order of \mathbf{S} and \mathbf{A} . This can then be used for quick computation of the exact solution \mathbf{V} of the discrete Poisson problem in Algorithm 10.1. We first compute $\mathbf{H} = \mathbf{SF}$ using Lemma 10.2 and m FFT's, one for each of the m columns of \mathbf{F} . We then compute $\mathbf{G} = \mathbf{HS}$ by m FFT's, one for each of the rows of \mathbf{H} . After \mathbf{X} is determined we compute $\mathbf{Z} = \mathbf{SX}$ and $\mathbf{V} = \mathbf{ZS}$ by another $2m$ FFT's. In total the work amounts to $4m$ FFT's of order $2m + 2$. Since one FFT requires $O(\gamma(2m + 2) \log_2(2m + 2))$ arithmetic operations the $4m$ FFT's amount to

$$8\gamma m(m + 1) \log_2(2m + 2) \approx 8\gamma m^2 \log_2 m = 4\gamma n \log_2 n,$$

where $n = m^2$ is the size of the linear system $\mathbf{Ax} = \mathbf{b}$ we would be solving if Cholesky factorization was used. This should be compared to the $O(8n^{3/2})$ arithmetic operations used in Algorithm 10.1 requiring 4 straightforward matrix multiplications with \mathbf{S} . What is faster will depend heavily on the programming of the FFT and the size of the problem. We refer to [34] for other efficient ways to implement the DST.

Exercise 10.5 (Fourier matrix)

Show that the Fourier matrix \mathbf{F}_4 is symmetric, but not Hermitian.

Exercise 10.6 (Sine transform as Fourier transform)

Verify Lemma 10.2 directly when $m = 1$.

Exercise 10.7 (Explicit solution of the discrete Poisson equation)

Show that the exact solution of the discrete Poisson equation (9.4) can be written $\mathbf{V} = (v_{i,j})_{i,j=1}^m$, where

$$v_{ij} = \frac{1}{(m+1)^4} \sum_{p=1}^m \sum_{r=1}^m \sum_{k=1}^m \sum_{l=1}^m \frac{\sin\left(\frac{ip\pi}{m+1}\right) \sin\left(\frac{jr\pi}{m+1}\right) \sin\left(\frac{kp\pi}{m+1}\right) \sin\left(\frac{lr\pi}{m+1}\right)}{\left[\sin\left(\frac{p\pi}{2(m+1)}\right)\right]^2 + \left[\sin\left(\frac{r\pi}{2(m+1)}\right)\right]^2} f_{p,r}.$$

Exercise 10.8 (Improved version of Algorithm 10.1)

Algorithm 10.1 involves multiplying a matrix by \mathbf{S} four times. In this problem we show that it is enough to multiply by \mathbf{S} two times. We achieve this by diagonalizing only the second \mathbf{T} in $\mathbf{TV} + \mathbf{VT} = h^2 \mathbf{F}$. Let $\mathbf{D} = \text{diag}(\lambda_1, \dots, \lambda_m)$, where $\lambda_j = 4 \sin^2(j\pi h/2)$, $j = 1, \dots, m$.

(a) Show that

$$\mathbf{TX} + \mathbf{XD} = \mathbf{C}, \text{ where } \mathbf{X} = \mathbf{VS}, \text{ and } \mathbf{C} = h^2 \mathbf{FS}.$$

(b) Show that

$$(\mathbf{T} + \lambda_j \mathbf{I}) \mathbf{x}_j = \mathbf{c}_j \quad j = 1, \dots, m, \quad (10.9)$$

where $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_m]$ and $\mathbf{C} = [\mathbf{c}_1, \dots, \mathbf{c}_m]$. Thus we can find \mathbf{X} by solving m linear systems, one for each of the columns of \mathbf{X} . Recall that a tridiagonal $m \times m$ system can be solved by Algorithms 1.8 and 1.9 in $8m - 7$ arithmetic operations. Give an algorithm to find \mathbf{X} which only requires $O(\delta m^2)$ arithmetic operations for some constant δ independent of m .

(c) Describe a method to compute \mathbf{V} which only requires $O(4m^3) = O(4n^{3/2})$ arithmetic operations.

(d) Describe a method based on the fast Fourier transform which requires $O(2\gamma n \log_2 n)$ where γ is the same constant as mentioned at the end of the last section.

Exercise 10.9 (Fast solution of 9 point scheme)

Consider the equation

$$\mathbf{TV} + \mathbf{VT} - \frac{1}{6} \mathbf{TVT} = h^2 \mu \mathbf{F},$$

that was derived in Exercise 9.13 for the 9-point scheme. Define the matrix \mathbf{X} by $\mathbf{V} = \mathbf{SX} \mathbf{S} = (x_{j,k})$ where \mathbf{V} is the solution of (9.18). Show that

$$\mathbf{DX} + \mathbf{XD} - \frac{1}{6} \mathbf{DXD} = 4h^4 \mathbf{G}, \text{ where } \mathbf{G} = \mathbf{S} \mu \mathbf{FS},$$

where $\mathbf{D} = \text{diag}(\lambda_1, \dots, \lambda_m)$, with $\lambda_j = 4 \sin^2(j\pi h/2)$, $j = 1, \dots, m$, and that

$$x_{j,k} = \frac{h^4 g_{j,k}}{\sigma_j + \sigma_k - \frac{2}{3} \sigma_j \sigma_k}, \text{ where } \sigma_j = \sin^2((j\pi h)/2) \text{ for } j, k = 1, 2, \dots, m.$$

Show that $\sigma_j + \sigma_k - \frac{2}{3} \sigma_j \sigma_k > 0$ for $j, k = 1, 2, \dots, m$. Conclude that the matrix \mathbf{A} in Exercise 9.13 b) is symmetric positive definite and that (9.17) always has a solution \mathbf{V} .

Exercise 10.10 (Algorithm for fast solution of 9 point scheme)

Derive an algorithm for solving (9.17) which for large m requires essentially the same number of operations as in Algorithm 10.1. (We assume that $\mu\mathbf{F}$ already has been formed).

Exercise 10.11 (Fast solution of biharmonic equation)

For the biharmonic problem we derived in Exercise 9.14 the equation

$$\mathbf{T}^2\mathbf{U} + 2\mathbf{T}\mathbf{U}\mathbf{T} + \mathbf{U}\mathbf{T}^2 = h^4\mathbf{F}.$$

Define the matrix $\mathbf{X} = (x_{j,k})$ by $\mathbf{U} = \mathbf{S}\mathbf{X}\mathbf{S}$ where \mathbf{U} is the solution of (9.21). Show that

$$\mathbf{D}^2\mathbf{X} + 2\mathbf{D}\mathbf{X}\mathbf{D} + \mathbf{X}\mathbf{D}^2 = 4h^6\mathbf{G}, \text{ where } \mathbf{G} = \mathbf{S}\mathbf{F}\mathbf{S},$$

and that

$$x_{j,k} = \frac{h^6 g_{j,k}}{4(\sigma_j + \sigma_k)^2}, \text{ where } \sigma_j = \sin^2((j\pi h)/2) \text{ for } j, k = 1, 2, \dots, m.$$

Exercise 10.12 (Algorithm for fast solution of biharmonic equation)

Use Exercise 10.11 to derive an algorithm

```
function U=simplefastbiharmonic(F)
```

which requires only $O(\delta n^{3/2})$ operations to find \mathbf{U} in Problem 9.14. Here δ is some constant independent of n .

Exercise 10.13 (Check algorithm for fast solution of biharmonic equation)

In Exercise 10.12 compute the solution \mathbf{U} corresponding to $\mathbf{F} = \text{ones}(m, m)$. For some small m 's check that you get the same solution obtained by solving the standard form $\mathbf{Ax} = \mathbf{b}$ in (9.21). You can use $\mathbf{x} = \mathbf{A}\backslash\mathbf{b}$ for solving $\mathbf{Ax} = \mathbf{b}$. Use $\mathbf{F}(:)$ to vectorize a matrix and $\text{reshape}(\mathbf{x}, m, m)$ to turn a vector $\mathbf{x} \in \mathbb{R}^{m^2}$ into an $m \times m$ matrix. Use the Matlab command `surf(U)` for plotting U for, say, $m = 50$. Compare the result with Exercise 10.12 by plotting the difference between both matrices.

Exercise 10.14 (Fast solution of biharmonic equation using 9 point rule)

Repeat Exercises 9.14, 10.12 and 10.13 using the nine point rule (9.17) to solve the system (9.20).

10.4 Review Questions

10.4.1 Consider the Poisson matrix.

- What is the bandwidth of its Cholesky factor?
- approximately how many arithmetic operations does it take to find the Cholesky factor?
- same question for block LU,
- same question for the fast Poisson solver with and without FFT.

10.4.2 What is the discrete sine transform and discrete Fourier transform of a vector?

Part IV

Iterative Methods for Large Linear Systems

Chapter 11

The Classical Iterative Methods

Gaussian elimination and Cholesky factorization are **direct methods**. In absence of rounding errors they find the exact solution using a finite number of arithmetic operations. In an **iterative method** we start with an approximation \mathbf{x}_0 to the exact solution \mathbf{x} and then compute a sequence $\{\mathbf{x}_k\}$ such that hopefully $\mathbf{x}_k \rightarrow \mathbf{x}$. Iterative methods are mainly used for large sparse systems, i.e., where many of the elements in the coefficient matrix are zero. The main advantages of iterative methods are reduced storage requirements and ease of implementation. In an iterative method the main work in each iteration is a matrix times vector multiplication, an operation which often does not need storing the matrix, not even in sparse form.

In this chapter we consider the classical iterative methods of Richardson, Jacobi, Gauss-Seidel and an accelerated version of Gauss-Seidel's method called successive overrelaxation (SOR). David Young developed in his thesis a beautiful theory describing the convergence rate of SOR, see [37].

We give the main points of this theory specialized to the discrete Poisson matrix. With a careful choice of an acceleration parameter the amount of work using SOR on the discrete Poisson problem is the same as for the fast Poisson solver without FFT (cf. Algorithm 10.1). Moreover, SOR is not restricted to constant coefficient methods on a rectangle. However, to obtain fast convergence using SOR it is necessary to have a good estimate for an acceleration parameter.

For convergence we need to study convergence of powers of matrices.

11.1 Classical Iterative Methods; Component Form

We start with an example showing how a linear system can be solved using an iterative method.

Example 11.1 (Iterative methods on a special 2×2 matrix)

Solving for the diagonal elements the linear system $\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ can be written in component form as $y = (z + 1)/2$ and $z = (y + 1)/2$. Starting with y_0, z_0 we generate two sequences $\{y_k\}$ and $\{z_k\}$ using the difference equations $y_{k+1} = (z_k + 1)/2$ and $z_{k+1} = (y_k + 1)/2$. This is known as Jacobi's method. If $y_0 = z_0 = 0$ then we find $y_1 = z_1 = 1/2$ and in general $y_k = z_k = 1 - 2^{-k}$ for $k = 0, 1, 2, 3, \dots$. The iteration converges to the exact solution $[1, 1]^T$, and the error is halved in each iteration.

We can improve the convergence rate by using the most current approximation in each iteration. This leads to Gauss-Seidel's method: $y_{k+1} = (z_k + 1)/2$ and $z_{k+1} = (y_{k+1} + 1)/2$. If $y_0 = z_0 = 0$ then we find $y_1 = 1/2$, $z_1 = 3/4$, $y_2 = 7/8$, $z_2 = 15/16$, and in general $y_k = 1 - 2 \cdot 4^{-k}$ and $z_k = 1 - 4^{-k}$ for $k = 1, 2, 3, \dots$. The error is now reduced by a factor 4 in each iteration.

Consider the general case. Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is nonsingular and $\mathbf{b} \in \mathbb{C}^n$. Suppose we know an approximation $\mathbf{x}_k = [\mathbf{x}_k(1), \dots, \mathbf{x}_k(n)]^T$ to the exact solution \mathbf{x} of $\mathbf{Ax} = \mathbf{b}$.



Lewis Fry Richardson, 1881-1953 (left), Carl Gustav Jacob Jacobi, 1804-1851 (right).



Philipp Ludwig von Seidel, 1821-1896 (left), David M. Young Jr., 1923-2008 (right)

We need to assume that the rows are ordered so that \mathbf{A} has nonzero diagonal elements. Solving the i th equation of $\mathbf{Ax} = \mathbf{b}$ for $\mathbf{x}(i)$, we obtain a **fixed-point form** of $\mathbf{Ax} = \mathbf{b}$

$$\mathbf{x}(i) = \left(-\sum_{j=1}^{i-1} a_{ij} \mathbf{x}(j) - \sum_{j=i+1}^n a_{ij} \mathbf{x}(j) + b_i \right) / a_{ii}, \quad i = 1, 2, \dots, n. \quad (11.1)$$

1. In **Jacobi's method (J method)** we substitute \mathbf{x}_k into the right hand side of (11.1) and compute a new approximation by

$$\mathbf{x}_{k+1}(i) = \left(-\sum_{j=1}^{i-1} a_{ij} \mathbf{x}_k(j) - \sum_{j=i+1}^n a_{ij} \mathbf{x}_k(j) + b_i \right) / a_{ii}, \text{ for } i = 1, 2, \dots, n. \quad (11.2)$$

2. **Gauss-Seidel's method (GS method)** is a modification of Jacobi's method, where we use the new $\mathbf{x}_{k+1}(i)$ immediately after it has been computed.

$$\mathbf{x}_{k+1}(i) = \left(-\sum_{j=1}^{i-1} a_{ij} \mathbf{x}_{k+1}(j) - \sum_{j=i+1}^n a_{ij} \mathbf{x}_k(j) + b_i \right) / a_{ii}, \text{ for } i = 1, 2, \dots, n. \quad (11.3)$$

3. The **Successive overrelaxation method (SOR method)** is obtained by introducing an acceleration parameter $0 < \omega < 2$ in the GS method. We write $\mathbf{x}(i) = \omega \mathbf{x}(i) + (1 - \omega) \mathbf{x}(i)$ and this leads to the method

$$\mathbf{x}_{k+1}(i) = \omega \left(-\sum_{j=1}^{i-1} a_{ij} \mathbf{x}_{k+1}(j) - \sum_{j=i+1}^n a_{ij} \mathbf{x}_k(j) + b_i \right) / a_{ii} + (1 - \omega) \mathbf{x}_k(i). \quad (11.4)$$

The SOR method reduces to the Gauss-Seidel method for $\omega = 1$. Denoting the right hand side of (11.3) by \mathbf{x}_{k+1}^{gs} we can write (11.4) as $\mathbf{x}_{k+1} = \omega \mathbf{x}_{k+1}^{gs} +$

$(1-\omega)\mathbf{x}_k$, and we see that \mathbf{x}_{k+1} is located on the straight line passing through the two points \mathbf{x}_{k+1}^{gs} and \mathbf{x}_k . The restriction $0 < \omega < 2$ is necessary for convergence (cf. Theorem 11.14). Normally, the best results are obtained for the relaxation parameter ω in the range $1 \leq \omega < 2$ and then \mathbf{x}_{k+1} is computed by linear extrapolation, i.e., it is not located between \mathbf{x}_{k+1}^{gs} and \mathbf{x}_k .

4. We mention also briefly the symmetric successive overrelaxation method **SSOR**. One iteration in SSOR consists of two SOR sweeps. A forward SOR sweep (11.4), computing an approximation denoted $\mathbf{x}_{k+1/2}$ instead of \mathbf{x}_{k+1} , is followed by a backward SOR sweep computing

$$\mathbf{x}_{k+1}(i) = \omega \left(-\sum_{j=1}^{i-1} a_{ij} \mathbf{x}_{k+1/2}(j) - \sum_{j=i+1}^n a_{ij} \mathbf{x}_{k+1}(j) + b_i \right) / a_{ii} + (1-\omega) \mathbf{x}_{k+1/2}(i) \quad (11.5)$$

in the order $i = n, n-1, \dots, 1$. The method is slower and more complicated than the SOR method. Its main use is as a symmetric preconditioner. For if \mathbf{A} is symmetric then SSOR combines the two SOR steps in such a way that the resulting iteration matrix is similar to a symmetric matrix. We will not discuss this method any further here and refer to Section 12.6 for an alternative example of a preconditioner.

We will refer to the J, GS and SOR methods as the **classical (iteration) methods**.

11.1.1 The discrete Poisson system

Consider the classical methods applied to the discrete Poisson matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ given by (9.7). Let $n = m^2$ and set $h = 1/(m+1)$. In component form the linear system $\mathbf{Ax} = \mathbf{b}$ can be written (cf. (9.4))

$$4\mathbf{v}(i,j) - \mathbf{v}(i-1,j) - \mathbf{v}(i+1,j) - \mathbf{v}(i,j-1) - \mathbf{v}(i,j+1) = h^2 f_{i,j}, \quad i, j = 1, \dots, m,$$

with homogenous boundary conditions also given in (9.4). Solving for $\mathbf{v}(i,j)$ we obtain the **fixed point form**

$$\mathbf{v}(i,j) = (\mathbf{v}(i-1,j) + \mathbf{v}(i+1,j) + \mathbf{v}(i,j-1) + \mathbf{v}(i,j+1) + e_{i,j}) / 4, \quad e_{i,j} := f_{i,j} / (m+1)^2. \quad (11.6)$$

The J, GS , and SOR methods take the form

$$\begin{aligned}
 J : \mathbf{v}_{k+1}(i, j) &= (\mathbf{v}_k(i-1, j) + \mathbf{v}_k(i, j-1) + \mathbf{v}_k(i+1, j) + \mathbf{v}_k(i, j+1) \\
 &\quad + \mathbf{e}(i, j))/4 \\
 GS : \mathbf{v}_{k+1}(i, j) &= (\mathbf{v}_{k+1}(i-1, j) + \mathbf{v}_{k+1}(i, j-1) + \mathbf{v}_k(i+1, j) + \mathbf{v}_k(i, j+1) \\
 &\quad + \mathbf{e}(i, j))/4 \\
 SOR : \mathbf{v}_{k+1}(i, j) &= \omega(\mathbf{v}_{k+1}(i-1, j) + \mathbf{v}_{k+1}(i, j-1) + \mathbf{v}_k(i+1, j) + \mathbf{v}_k(i, j+1) \\
 &\quad + \mathbf{e}(i, j))/4 + (1 - \omega)\mathbf{v}_k(i, j).
 \end{aligned} \tag{11.7}$$

We note that for GS and SOR we have used the **natural ordering**, i.e., $(i_1, j_1) < (i_2, j_2)$ if and only if $j_1 \leq j_2$ and $i_1 < i_2$ if $j_1 = j_2$. For the J method any ordering can be used.

In Algorithm 11.2 we give a Matlab program to test the convergence of Jacobi's method on the discrete Poisson problem. We carry out Jacobi iterations on the linear system (11.6) with $\mathbf{F} = (f_{ij}) \in \mathbb{R}^{m \times m}$, starting with $\mathbf{V}_0 = \mathbf{0} \in \mathbb{R}^{(m+2) \times (m+2)}$. The output is the number of iterations k , to obtain $\|\mathbf{V}^{(k)} - \mathbf{U}\|_M := \max_{i,j} |v_{ij} - u_{ij}| < tol$. Here $[u_{ij}] \in \mathbb{R}^{(m+2) \times (m+2)}$ is the "exact" solution of (11.6) computed using the fast Poisson solver in Algorithm 10.1. We set $k = K + 1$ if convergence is not obtained in K iterations. In Table 11.1 we show the output $k = k_n$ from this algorithm using $\mathbf{F} = \text{ones}(m, m)$ for $m = 10, 50$, $K = 10^4$, and $tol = 10^{-8}$. We also show the number of iterations for Gauss-Seidel and SOR with a value of ω known as the optimal acceleration parameter $\omega^* := 2/(1 + \sin(\frac{\pi}{m+1}))$. We will derive this value later.

Algorithm 11.2 (Jacobi)

```

1 function k=jdp(F,K,tol)
2 m=length(F); U=fastpoisson(F); V=zeros(m+2,m+2); E=F/(m+1)^2;
3 for k=1:K
4   V(2:m+1,2:m+1)=(V(1:m,2:m+1)+V(3:m+2,2:m+1)...
5     +V(2:m+1,1:m)+V(2:m+1,3:m+2)+E)/4;
6   if max(max(abs(V-U)))<tol, return
7   end
8 end
9 k=K+1;

```

For the GS and SOR methods we have used Algorithm 11.3. This is the analog of Algorithm 11.2 using SOR instead of J to solve the discrete Poisson problem. w is an acceleration parameter with $0 < w < 2$. For $w = 1$ we obtain Gauss-Seidel's method.

	k_{100}	k_{2500}	$k_{10\ 000}$	$k_{40\ 000}$	$k_{160\ 000}$
J	385	8386			
GS	194	4194			
SOR	35	164	324	645	1286

Table 11.1: The number of iterations k_n to solve the discrete Poisson problem with n unknowns using the methods of Jacobi, Gauss-Seidel, and SOR (see text) with a tolerance 10^{-8} .

Algorithm 11.3 (SOR)

```

1 function k=sordp(F,K,w,tol)
2 m=length(F); U=fastpoisson(F); V=zeros(m+2,m+2); E=F/(m+1)^2;
3 for k=1:K
4   for j=2:m+1
5     for i=2:m+1
6       V(i,j)=w*(V(i-1,j)+V(i+1,j)+V(i,j-1)...
7           +V(i,j+1)+E(i-1,j-1))/4+(1-w)*V(i,j);
8     end
9   end
10  if max(max(abs(V-U)))<tol, return
11  end
12 end
13 k=K+1;

```

We make several remarks about these programs and the results in Table 11.1.

1. The rate (speed) of convergence is quite different for the four methods. The J and GS methods converge, but rather slowly. The J method needs about twice as many iterations as the GS method. The improvement using the SOR method with optimal ω is spectacular.
2. We show in Section 11.3.4 that the number of iterations k_n for a size n problem is $k_n = O(n)$ for the J and GS method and $k_n = O(\sqrt{n})$ for SOR with optimal ω . The choice of tol will only influence the constants multiplying n or \sqrt{n} .
3. From (11.7) it follows that each iteration requires $O(n)$ arithmetic operations. Thus the number of arithmetic operations to achieve a given tolerance is $O(k_n \times n)$. Therefore the number of arithmetic operations for the J and GS method is $O(n^2)$, while it is only $O(n^{3/2})$ for the SOR method with optimal ω . Asymptotically, for J and GS this is the same as using banded Cholesky, while SOR competes with the fast method (without FFT).

4. We do not need to store the coefficient matrix so the storage requirements for these methods on the discrete Poisson problem is $O(n)$, asymptotically the same as for the fast methods.
5. Jacobi's method has the advantage that it can be easily parallelized.

11.2 Classical Iterative Methods; Matrix Form

To study convergence we need matrix formulations of the classical methods.

11.2.1 Fixed-point form

In general we can construct an iterative method by choosing a nonsingular matrix \mathbf{M} and write $\mathbf{A}\mathbf{x} = \mathbf{b}$ in the equivalent form

$$\mathbf{M}\mathbf{x} = (\mathbf{M} - \mathbf{A})\mathbf{x} + \mathbf{b}. \quad (11.8)$$

The matrix \mathbf{M} is known as a **splitting matrix**.

The corresponding iterative method is given by

$$\mathbf{M}\mathbf{x}_{k+1} = (\mathbf{M} - \mathbf{A})\mathbf{x}_k + \mathbf{b} \quad (11.9)$$

or

$$\mathbf{x}_{k+1} := \mathbf{G}\mathbf{x}_k + \mathbf{c}, \quad \mathbf{G} = \mathbf{I} - \mathbf{M}^{-1}\mathbf{A}, \quad , \mathbf{c} = \mathbf{M}^{-1}\mathbf{b}. \quad (11.10)$$

This is known as a **fixed-point iteration**. Starting with \mathbf{x}_0 this defines a sequence $\{\mathbf{x}_k\}$ of vectors in \mathbb{C}^n . For a general $\mathbf{G} \in \mathbb{C}^{n \times n}$ and $\mathbf{c} \in \mathbb{C}^n$ a solution of $\mathbf{x} = \mathbf{G}\mathbf{x} + \mathbf{c}$ is called a **fixed-point**. The fixed-point is unique if $\mathbf{I} - \mathbf{G}$ is nonsingular.

If $\lim_{k \rightarrow \infty} \mathbf{x}_k = \mathbf{x}$ for some $\mathbf{x} \in \mathbb{C}^n$ then \mathbf{x} is a fixed point since

$$\mathbf{x} = \lim_{k \rightarrow \infty} \mathbf{x}_{k+1} = \lim_{k \rightarrow \infty} (\mathbf{G}\mathbf{x}_k + \mathbf{c}) = \mathbf{G} \lim_{k \rightarrow \infty} \mathbf{x}_k + \mathbf{c} = \mathbf{G}\mathbf{x} + \mathbf{c}.$$

The matrix \mathbf{M} can also be interpreted as a **preconditioning matrix**. We first write $\mathbf{A}\mathbf{x} = \mathbf{b}$ in the equivalent form $\mathbf{M}^{-1}\mathbf{A}\mathbf{x} = \mathbf{M}^{-1}\mathbf{b}$ or $\mathbf{x} = \mathbf{x} - \mathbf{M}^{-1}\mathbf{A}\mathbf{x} + \mathbf{M}^{-1}\mathbf{b}$. This again leads to the iterative method (11.10), and \mathbf{M} is chosen to reduce the condition number of \mathbf{A} .

11.2.2 The splitting matrices for the classical methods

Different choices of \mathbf{M} in (11.9) lead to different iterative methods. We now derive \mathbf{M} for the classical methods. For GS and SOR it is convenient to write \mathbf{A} as a

sum of three matrices, $\mathbf{A} = \mathbf{D} - \mathbf{A}_L - \mathbf{A}_R$, where $-\mathbf{A}_L$, \mathbf{D} , and $-\mathbf{A}_R$ are the lower, diagonal, and upper part of \mathbf{A} , respectively. Thus $\mathbf{D} := \text{diag}(a_{11}, \dots, a_{nn})$,

$$\mathbf{A}_L := \begin{bmatrix} 0 & & & \\ -a_{2,1} & 0 & & \\ \vdots & \ddots & \ddots & \\ -a_{n,1} & \cdots & -a_{n,n-1} & 0 \end{bmatrix}, \quad \mathbf{A}_R := \begin{bmatrix} 0 & -a_{1,2} & \cdots & -a_{1,n} \\ & \ddots & \ddots & \vdots \\ & & 0 & -a_{n-1,n} \\ & & & 0 \end{bmatrix}. \quad (11.11)$$

Theorem 11.4 (Splitting matrices for J, GS and SOR)

The splitting matrices \mathbf{M}_J , \mathbf{M}_1 and \mathbf{M}_ω for the J, GS and SOR methods are given by

$$\mathbf{M}_J = \mathbf{D}, \quad \mathbf{M}_1 = \mathbf{D} - \mathbf{A}_L, \quad \mathbf{M}_\omega = \omega^{-1}\mathbf{D} - \mathbf{A}_L. \quad (11.12)$$

Proof. To find \mathbf{M} we write the methods in the form (11.9) where the coefficient of \mathbf{b} is equal to one. Moving a_{ii} to the left hand side of the Jacobi iteration (11.2) we obtain the matrix form $\mathbf{D}\mathbf{x}_{k+1} = (\mathbf{D} - \mathbf{A})\mathbf{x}_k + \mathbf{b}$ showing that $\mathbf{M}_J = \mathbf{D}$.

For the SOR method a matrix form is

$$\mathbf{D}\mathbf{x}_{k+1} = \omega(\mathbf{A}_L\mathbf{x}_{k+1} + \mathbf{A}_R\mathbf{x}_k + \mathbf{b}) + (1 - \omega)\mathbf{D}\mathbf{x}_k. \quad (11.13)$$

Dividing both sides by ω and moving $\mathbf{A}_L\mathbf{x}_{k+1}$ to the left hand side this takes the form $(\omega^{-1}\mathbf{D} - \mathbf{A}_L)\mathbf{x}_{k+1} = \mathbf{A}_R\mathbf{x}_k + \mathbf{b} + (\omega^{-1} - 1)\mathbf{D}\mathbf{x}_k$ showing that $\mathbf{M}_\omega = \omega^{-1}\mathbf{D} - \mathbf{A}_L$. We obtain \mathbf{M}_1 by letting $\omega = 1$ in \mathbf{M}_ω . \square

Example 11.5 (Splitting matrices)

For the system

$$\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

we find

$$\mathbf{A}_L = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \quad \mathbf{A}_R = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix},$$

and

$$\mathbf{M}_J = \mathbf{D} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \quad \mathbf{M}_\omega = \omega^{-1}\mathbf{D} - \mathbf{A}_L = \begin{bmatrix} 2\omega^{-1} & 0 \\ -1 & 2\omega^{-1} \end{bmatrix}.$$

The iteration matrix $\mathbf{G}_\omega = \mathbf{I} - \mathbf{M}_\omega^{-1}\mathbf{A}$ is given by

$$\mathbf{G}_\omega = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} \omega/2 & 0 \\ \omega^2/4 & \omega/2 \end{bmatrix} \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} = \begin{bmatrix} 1 - \omega & \omega/2 \\ \omega(1 - \omega)/2 & 1 - \omega + \omega^2/4 \end{bmatrix}. \quad (11.14)$$

For the J and GS method we have

$$\mathbf{G}_J = \mathbf{I} - \mathbf{D}^{-1}\mathbf{A} = \begin{bmatrix} 0 & 1/2 \\ 1/2 & 0 \end{bmatrix}, \quad \mathbf{G}_1 = \begin{bmatrix} 0 & 1/2 \\ 0 & 1/4 \end{bmatrix}. \quad (11.15)$$

We could have derived these matrices directly from the component form of the iteration. For example, for the GS method we have the component form

$$\mathbf{x}_{k+1}(1) = \frac{1}{2}\mathbf{x}_k(2) + \frac{1}{2}, \quad \mathbf{x}_{k+1}(2) = \frac{1}{2}\mathbf{x}_{k+1}(1) + \frac{1}{2}.$$

Substituting the value of $\mathbf{x}_{k+1}(1)$ from the first equation into the second equation we find

$$\mathbf{x}_{k+1}(2) = \frac{1}{2}\left(\frac{1}{2}\mathbf{x}_k(2) + \frac{1}{2}\right) + \frac{1}{2} = \frac{1}{4}\mathbf{x}_k(2) + \frac{3}{4}.$$

Thus

$$\mathbf{x}_{k+1} = \begin{bmatrix} \mathbf{x}_{k+1}(1) \\ \mathbf{x}_{k+1}(2) \end{bmatrix} = \begin{bmatrix} 0 & 1/2 \\ 0 & 1/4 \end{bmatrix} \begin{bmatrix} \mathbf{x}_k(1) \\ \mathbf{x}_k(2) \end{bmatrix} + \begin{bmatrix} 1/2 \\ 3/4 \end{bmatrix} = \mathbf{G}_1\mathbf{x}_k + \mathbf{c}.$$

11.3 Convergence

For Newton's method the choice of starting value is important. This is not the case for methods of the form $\mathbf{x}_{k+1} := \mathbf{G}\mathbf{x}_k + \mathbf{c}$.

Definition 11.6 (Convergence of $\mathbf{x}_{k+1} := \mathbf{G}\mathbf{x}_k + \mathbf{c}$)

We say that the iterative method $\mathbf{x}_{k+1} := \mathbf{G}\mathbf{x}_k + \mathbf{c}$ converges if the sequence $\{\mathbf{x}_k\}$ converges for any starting vector \mathbf{x}_0 .

We have the following necessary and sufficient condition for convergence:

Theorem 11.7 (Convergence of an iterative method)

The iterative method $\mathbf{x}_{k+1} := \mathbf{G}\mathbf{x}_k + \mathbf{c}$ converges if and only if $\lim_{k \rightarrow \infty} \mathbf{G}^k = \mathbf{0}$.

Proof. We subtract $\mathbf{x} = \mathbf{G}\mathbf{x} + \mathbf{c}$ from $\mathbf{x}_{k+1} = \mathbf{G}\mathbf{x}_k + \mathbf{c}$. The vector \mathbf{c} cancels and we obtain $\mathbf{x}_{k+1} - \mathbf{x} = \mathbf{G}(\mathbf{x}_k - \mathbf{x})$. By induction on k

$$\mathbf{x}_k - \mathbf{x} = \mathbf{G}^k(\mathbf{x}_0 - \mathbf{x}), \quad k = 0, 1, 2, \dots \quad (11.16)$$

Clearly $\mathbf{x}_k - \mathbf{x} \rightarrow \mathbf{0}$ if $\mathbf{G}^k \rightarrow \mathbf{0}$. The converse follows by choosing $\mathbf{x}_0 - \mathbf{x} = \mathbf{e}_j$, the j th unit vector for $j = 1, \dots, n$. \square

Theorem 11.8 (Sufficient condition for convergence)

If $\|\mathbf{G}\| < 1$ for some consistent matrix norm on $\mathbb{C}^{n \times n}$, then the iteration $\mathbf{x}_{k+1} = \mathbf{G}\mathbf{x}_k + \mathbf{c}$ converges.

Proof. We have

$$\|\mathbf{x}_k - \mathbf{x}\| = \|\mathbf{G}^k(\mathbf{x}_0 - \mathbf{x})\| \leq \|\mathbf{G}^k\| \|\mathbf{x}_0 - \mathbf{x}\| \leq \|\mathbf{G}\|^k \|\mathbf{x}_0 - \mathbf{x}\| \rightarrow 0, \quad k \rightarrow \infty.$$

□

A necessary and sufficient condition for convergence involves the eigenvalues of \mathbf{G} . We define the **spectral radius** of a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ as the maximum absolute value of its eigenvalues.

$$\rho(\mathbf{A}) := \max_{\lambda \in \sigma(\mathbf{A})} |\lambda|. \quad (11.17)$$

Theorem 11.9 (When does an iterative method converge?)

Suppose $\mathbf{G} \in \mathbb{C}^{n \times n}$ and $\mathbf{c} \in \mathbb{C}^n$. The iteration $\mathbf{x}_{k+1} = \mathbf{G}\mathbf{x}_k + \mathbf{c}$ converges if and only if $\rho(\mathbf{G}) < 1$.

We will prove this theorem using Theorem 11.27 in Section 11.4.

11.3.1 Richardson's method.

The **Richardson's method (R method)** is defined by

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha(\mathbf{b} - \mathbf{A}\mathbf{x}_k). \quad (11.18)$$

Here we pick an acceleration parameter α and compute a new approximation by adding a multiple of the residual vector $\mathbf{r}_k := \mathbf{b} - \mathbf{A}\mathbf{x}_k$. Note that we do not need the assumption of nonzero diagonal elements. Richardson considered this method in 1910.

We will assume that α is real. If all eigenvalues of \mathbf{A} have positive real parts then the R method converges provided α is positive and sufficiently small. We show this result for positive eigenvalues and leave the more general case to Exercise 11.13.

Theorem 11.10 (Convergence of Richardson's method)

If \mathbf{A} has positive eigenvalues $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n > 0$ then the R method given by $\mathbf{x}_{k+1} = (\mathbf{I} - \alpha\mathbf{A})\mathbf{x}_k + \mathbf{b}$ converges if and only if $0 < \alpha < 2/\lambda_1$. Moreover,

$$\rho(\mathbf{I} - \alpha\mathbf{A}) > \rho(\mathbf{I} - \alpha^*\mathbf{A}) = \frac{\kappa - 1}{\kappa + 1}, \quad \alpha \in \mathbb{R} \setminus \{\alpha^*\}, \quad \kappa := \frac{\lambda_1}{\lambda_n}, \quad \alpha^* := \frac{2}{\lambda_1 + \lambda_n}. \quad (11.19)$$

Proof. The eigenvalues of $\mathbf{I} - \alpha\mathbf{A}$ are $1 - \alpha\lambda_j$, $j = 1, \dots, n$. We have

$$\rho_\alpha = \rho(\mathbf{I} - \alpha\mathbf{A}) := \max_j |1 - \alpha\lambda_j| = \begin{cases} 1 - \alpha\lambda_1, & \text{if } \alpha \leq 0 \\ 1 - \alpha\lambda_n, & \text{if } 0 < \alpha \leq \alpha^* \\ \alpha\lambda_1 - 1, & \text{if } \alpha > \alpha^*, \end{cases}$$

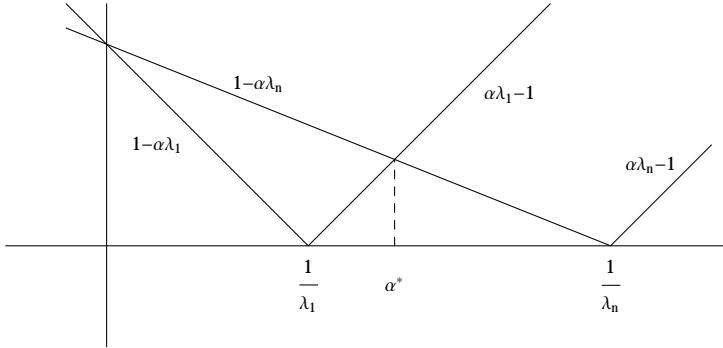


Figure 11.1: The functions $\alpha \rightarrow |1 - \alpha\lambda_1|$ and $\alpha \rightarrow |1 - \alpha\lambda_n|$.

see Figure 11.1. Clearly $1 - \alpha\lambda_n = \alpha\lambda_1 - 1$ for $\alpha = \alpha^*$ and

$$\rho_{\alpha^*} = \alpha^*\lambda_1 - 1 = \frac{\lambda_1 - \lambda_n}{\lambda_1 + \lambda_n} = \frac{\kappa - 1}{\kappa + 1} < 1.$$

We have $\rho_\alpha < 1$ if and only if $\alpha > 0$ and $\alpha\lambda_1 - 1 < 1$ showing convergence if and only if $0 < \alpha < 2/\lambda_1$ and $\rho_\alpha > \rho_{\alpha^*}$ for $\alpha \leq 0$ and $\alpha \geq 2/\lambda_1$. Finally, if $0 < \alpha < \alpha^*$ then $\rho_\alpha = 1 - \alpha\lambda_n > 1 - \alpha^*\lambda_n = \rho_{\alpha^*}$ and if $\alpha^* < \alpha < 2/\lambda_1$ then $\rho_\alpha = \alpha\lambda_1 - 1 > \alpha^*\lambda_1 - 1 = \rho_{\alpha^*}$. \square

For a symmetric positive definite matrix we obtain

Corollary 11.11 (Rate of convergence for the R method)

Suppose \mathbf{A} is symmetric positive definite with largest and smallest eigenvalue λ_{max} and λ_{min} , respectively. Richardson's method $\mathbf{x}_{k+1} = (\mathbf{I} - \alpha\mathbf{A})\mathbf{x}_k + \mathbf{b}$ converges if and only if $0 < \alpha < 2/\lambda_{max}$. With $\alpha^* := \frac{2}{\lambda_{max} + \lambda_{min}}$ we have the error estimate

$$\|\mathbf{x}_k - \mathbf{x}\|_2 \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^k \|\mathbf{x}_0 - \mathbf{x}\|_2, \quad k = 0, 1, 2, \dots \quad (11.20)$$

where $\kappa := \lambda_{max}/\lambda_{min}$ is the spectral condition number of \mathbf{A} .

Proof. The spectral norm $\|\cdot\|_2$ is consistent and therefore $\|\mathbf{x}_k - \mathbf{x}\|_2 \leq \|\mathbf{I} - \alpha^*\mathbf{A}\|_2^k \|\mathbf{x}_0 - \mathbf{x}\|_2$. But for a symmetric positive definite matrix the spectral norm is equal to the spectral radius and the result follows from (11.19). \square

Exercise 11.12 (Richardson and Jacobi)

Show that if $a_{ii} = d \neq 0$ for all i then Richardson's method with $\alpha := 1/d$ is the same as Jacobi's method.

Exercise 11.13 (R-method when eigenvalues have positive real part) Suppose all eigenvalues λ_j of \mathbf{A} have positive real parts u_j for $j = 1, \dots, n$ and that α is real. Show that the R method converges if and only if $0 < \alpha < \min_j(2u_j/|\lambda_j|^2)$.

11.3.2 Convergence of SOR

The condition $\omega \in (0, 2)$ is necessary for convergence of the SOR method.

Theorem 11.14 (Necessary condition for convergence of SOR)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is nonsingular with nonzero diagonal elements. If the SOR method applied to \mathbf{A} converges then $\omega \in (0, 2)$.

Proof. We have (cf. (11.13)) $\mathbf{D}\mathbf{x}_{k+1} = \omega(\mathbf{A}_L\mathbf{x}_{k+1} + \mathbf{A}_R\mathbf{x}_k + \mathbf{b}) + (1 - \omega)\mathbf{D}\mathbf{x}_k$ or $\mathbf{x}_{k+1} = \omega(\mathbf{L}\mathbf{x}_{k+1} + \mathbf{R}\mathbf{x}_k + \mathbf{D}^{-1}\mathbf{b}) + (1 - \omega)\mathbf{x}_k$, where $\mathbf{L} := \mathbf{D}^{-1}\mathbf{A}_L$ and $\mathbf{R} := \mathbf{D}^{-1}\mathbf{A}_R$. Thus $(\mathbf{I} - \omega\mathbf{L})\mathbf{x}_{k+1} = (\omega\mathbf{R} + (1 - \omega)\mathbf{I})\mathbf{x}_k + \mathbf{D}^{-1}\mathbf{b}$ so the following form of the iteration matrix is obtained

$$\mathbf{G}_\omega = (\mathbf{I} - \omega\mathbf{L})^{-1}(\omega\mathbf{R} + (1 - \omega)\mathbf{I}). \quad (11.21)$$

We next compute the determinant of \mathbf{G}_ω . Since $\mathbf{I} - \omega\mathbf{L}$ is lower triangular with ones on the diagonal, the same holds for the inverse by Lemma 1.35, and therefore the determinant of this matrix is equal to one. The matrix $\omega\mathbf{R} + (1 - \omega)\mathbf{I}$ is upper triangular with $1 - \omega$ on the diagonal and therefore its determinant equals $(1 - \omega)^n$. It follows that $\det(\mathbf{G}_\omega) = (1 - \omega)^n$. Since the determinant of a matrix equals the product of its eigenvalues we must have $|\lambda| \geq |1 - \omega|$ for at least one eigenvalue λ of \mathbf{G}_ω and we conclude that $\rho(\mathbf{G}_\omega) \geq |\omega - 1|$. But then $\rho(\mathbf{G}_\omega) \geq 1$ if ω is not in the interval $(0, 2)$ and by Theorem 11.9 SOR diverges. \square

The SOR method always converges for a symmetric positive definite matrix.

Theorem 11.15 (SOR on positive definite matrix)

SOR converges for a symmetric positive definite matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ if and only if $0 < \omega < 2$. In particular, Gauss-Seidel's method converges for a symmetric positive definite matrix.

Proof. By Theorem 11.14 convergence implies $0 < \omega < 2$. Suppose now $0 < \omega < 2$ and let (λ, \mathbf{x}) be an eigenpair for \mathbf{G}_ω . Note that λ and \mathbf{x} can be complex. We need to show that $|\lambda| < 1$. The following identity will be shown below:

$$\omega^{-1}(2 - \omega)|1 - \lambda|^2 \mathbf{x}^* \mathbf{D} \mathbf{x} = (1 - |\lambda|^2) \mathbf{x}^* \mathbf{A} \mathbf{x}, \quad (11.22)$$

where $\mathbf{D} := \text{diag}(a_{11}, \dots, a_{nn})$. Now $\mathbf{x}^* \mathbf{A} \mathbf{x}$ and $\mathbf{x}^* \mathbf{D} \mathbf{x}$ are positive for all nonzero $\mathbf{x} \in \mathbb{C}^n$ since a positive definite matrix has positive diagonal elements $a_{ii} =$

$e_i^T \mathbf{A} e_i > 0$. It follows that the left hand side of (11.22) is nonnegative and then the right hand side must be nonnegative as well. This implies $|\lambda| \leq 1$. It remains to show that we cannot have $\lambda = 1$. By (11.12) the eigenpair equation $\mathbf{G}_\omega \mathbf{x} = \lambda \mathbf{x}$ can be written $\mathbf{x} - (\omega^{-1} \mathbf{D} - \mathbf{A}_L)^{-1} \mathbf{A} \mathbf{x} = \lambda \mathbf{x}$ or

$$\mathbf{A} \mathbf{x} = (\omega^{-1} \mathbf{D} - \mathbf{A}_L) \mathbf{y}, \quad \mathbf{y} := (1 - \lambda) \mathbf{x}. \quad (11.23)$$

Now $\mathbf{A} \mathbf{x} \neq \mathbf{0}$ implies that $\lambda \neq 1$.

To prove equation (11.22) consider the matrix $\mathbf{E} := \omega^{-1} \mathbf{D} + \mathbf{A}_R - \mathbf{D}$. Since $\mathbf{A}_R - \mathbf{D} = -\mathbf{A}_L - \mathbf{A}$ we find $\mathbf{E} \mathbf{y} = (\omega^{-1} \mathbf{D} - \mathbf{A}_L - \mathbf{A}) \mathbf{y} \stackrel{(11.23)}{=} \mathbf{A} \mathbf{x} - \mathbf{A} \mathbf{y} = \lambda \mathbf{A} \mathbf{x}$. Observe that $(\omega^{-1} \mathbf{D} - \mathbf{A}_L)^* = \omega^{-1} \mathbf{D} - \mathbf{A}_R$ so that by (11.23)

$$\begin{aligned} (\mathbf{A} \mathbf{x})^* \mathbf{y} + \mathbf{y}^* (\lambda \mathbf{A} \mathbf{x}) &= \mathbf{y}^* (\omega^{-1} \mathbf{D} - \mathbf{A}_R) \mathbf{y} + \mathbf{y}^* \mathbf{E} \mathbf{y} = \mathbf{y}^* (2\omega^{-1} - 1) \mathbf{D} \mathbf{y} \\ &= \omega^{-1} (2 - \omega) |1 - \lambda|^2 \mathbf{x}^* \mathbf{D} \mathbf{x}. \end{aligned}$$

Since $(\mathbf{A} \mathbf{x})^* = \mathbf{x}^* \mathbf{A}$, $\mathbf{y} := (1 - \lambda) \mathbf{x}$ and $\mathbf{y}^* = (1 - \bar{\lambda}) \mathbf{x}^*$ this also equals

$$(\mathbf{A} \mathbf{x})^* \mathbf{y} + \mathbf{y}^* (\lambda \mathbf{A} \mathbf{x}) = (1 - \lambda) \mathbf{x}^* \mathbf{A} \mathbf{x} + \lambda (1 - \bar{\lambda}) \mathbf{x}^* \mathbf{A} \mathbf{x} = (1 - |\lambda|^2) \mathbf{x}^* \mathbf{A} \mathbf{x},$$

and (11.22) follows. \square

Exercise 11.16 (Example: GS converges, J diverges)

Show (by finding its eigenvalues) that the matrix

$$\begin{bmatrix} 1 & a & a \\ a & 1 & a \\ a & a & 1 \end{bmatrix}$$

is symmetric positive definite for $-1/2 < a < 1$. Thus, GS converges for these values of a . Show that the J method does not converge for $1/2 < a < 1$.

Exercise 11.17 (Divergence example for J and GS)

Show that both Jacobi's method and Gauss-Seidel's method diverge for $\mathbf{A} = [\begin{smallmatrix} 1 & 2 \\ 3 & 4 \end{smallmatrix}]$.

Exercise 11.18 (Strictly diagonally dominance; The J method)

Show that the J method converges if $|a_{ii}| > \sum_{j \neq i} |a_{ij}|$ for $i = 1, \dots, n$.

Exercise 11.19 (Strictly diagonally dominance; The GS method)

Consider the GS method. Suppose $r := \max_i r_i < 1$, where $r_i = \sum_{j \neq i} \frac{|a_{ij}|}{|a_{ii}|}$. Show using induction on i that $|\epsilon_{k+1}(j)| \leq r \|\epsilon_k\|_\infty$ for $j = 1, \dots, i$. Conclude that Gauss-Seidel's method is convergent when \mathbf{A} is strictly diagonally dominant.

11.3.3 Convergence of the classical methods for the discrete Poisson matrix

We know the eigenvalues of the discrete Poisson matrix \mathbf{A} given by (9.7) and we can use this to estimate the number of iterations necessary to achieve a given accuracy for the various methods.

Recall that by (9.15) the eigenvalues $\lambda_{j,k}$ of \mathbf{A} are

$$\lambda_{j,k} = 4 - 2 \cos(j\pi h) - 2 \cos(k\pi h), \quad j, k = 1, \dots, m, h = 1/(m+1).$$

It follows that the largest and smallest eigenvalue of \mathbf{A} , and the spectral condition number κ of \mathbf{A} , are given by

$$\lambda_{max} = 8 \cos^2 w, \quad \lambda_{min} = 8 \sin^2 w, \quad \kappa := \frac{\cos^2 w}{\sin^2 w}, \quad w := \frac{\pi}{2(m+1)}. \quad (11.24)$$

Consider first the J method. The matrix $\mathbf{G}_J = \mathbf{I} - \mathbf{D}^{-1}\mathbf{A} = \mathbf{I} - \mathbf{A}/4$ has eigenvalues

$$\mu_{j,k} = 1 - \frac{1}{4}\lambda_{j,k} = \frac{1}{2} \cos(j\pi h) + \frac{1}{2} \cos(k\pi h), \quad j, k = 1, \dots, m. \quad (11.25)$$

It follows that $\rho(\mathbf{G}_J) = \cos(\pi h) < 1$. Since \mathbf{G}_J is symmetric it is normal, and the spectral norm is equal to the spectral radius (cf. Theorem 7.17). We obtain

$$\|\mathbf{x}_k - \mathbf{x}\|_2 \leq \|\mathbf{G}_J\|_2^k \|\mathbf{x}_0 - \mathbf{x}\|_2 = \cos^k(\pi h) \|\mathbf{x}_0 - \mathbf{x}\|_2, \quad k = 0, 1, 2, \dots \quad (11.26)$$

The R method given by $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha \mathbf{r}_k$ with $\alpha = 2/(\lambda_{max} + \lambda_{min}) = 1/4$ is the same as the J-method so (11.26) holds in this case as well. This also follows from Corollary 11.11 with κ given by (11.24).

For the SOR method it is possible to explicitly determine $\rho(\mathbf{G}_\omega)$ for any $\omega \in (0, 2)$. The following result will be shown in Section 11.5.

Theorem 11.20 (The spectral radius of SOR matrix)

Consider the SOR iteration (11.7), with the natural ordering. The spectral radius of \mathbf{G}_ω is

$$\rho(\mathbf{G}_\omega) = \begin{cases} \frac{1}{4} \left(\omega\beta + \sqrt{(\omega\beta)^2 - 4(\omega-1)} \right)^2, & \text{for } 0 < \omega \leq \omega^*, \\ \omega - 1, & \text{for } \omega^* < \omega < 2, \end{cases} \quad (11.27)$$

where $\beta := \rho(\mathbf{G}_J) = \cos(\pi h)$ and

$$\omega^* := \frac{2}{1 + \sqrt{1 - \beta^2}} > 1. \quad (11.28)$$

Moreover,

$$\rho(\mathbf{G}_\omega) > \rho(\mathbf{G}_{\omega^*}) \text{ for } \omega \in (0, 2) \setminus \{\omega^*\}. \quad (11.29)$$

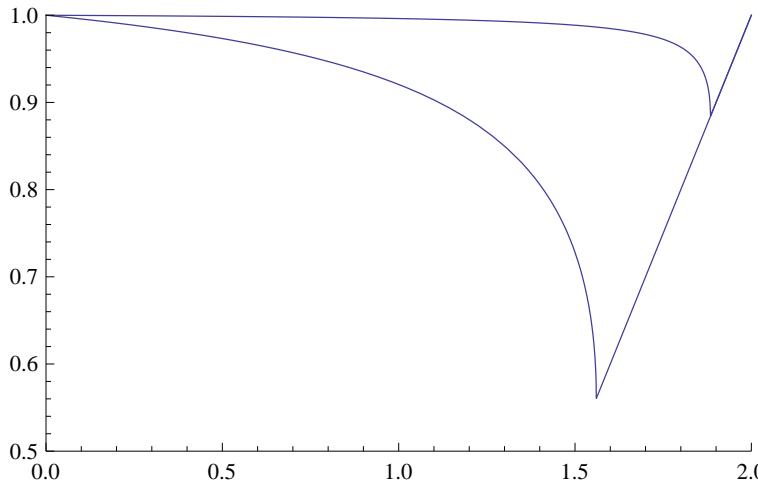


Figure 11.2: $\rho(\mathbf{G}_\omega)$ with $\omega \in [0, 2]$ for $n = 100$, (lower curve) and $n = 2500$ (upper curve).

A plot of $\rho(\mathbf{G}_\omega)$ as a function of $\omega \in (0, 2)$ is shown in Figure 11.2 for $n = 100$ (lower curve) and $n = 2500$ (upper curve). As ω increases the spectral radius of \mathbf{G}_ω decreases monotonically to the minimum ω^* . Then it increases linearly to the value one for $\omega = 2$. We call ω^* the **optimal relaxation parameter**.

For the discrete Poisson problem we have $\beta = \cos(\pi h)$ and it follows from (11.27),(11.28) that

$$\omega^* = \frac{2}{1 + \sin(\pi h)}, \quad \rho(\mathbf{G}_{\omega^*}) = \omega^* - 1 = \frac{1 - \sin(\pi h)}{1 + \sin(\pi h)}, \quad h = \frac{1}{m+1}. \quad (11.30)$$

Letting $\omega = 1$ in (11.27) we find $\rho(\mathbf{G}_1) = \beta^2 = \rho(\mathbf{G}_J)^2 = \cos^2(\pi h)$ for the GS method. Thus, for the discrete Poisson problem the J method needs twice as many iterations as the GS method for a given accuracy.

The values of $\rho(\mathbf{G}_J)$, $\rho(\mathbf{G}_1)$, and $\rho(\mathbf{G}_{\omega^*}) = \omega^* - 1$ are shown in Table 11.2 for $n = 100$ and $n = 2500$. We also show the smallest integer k_n such that $\rho(\mathbf{G})^{k_n} \leq 10^{-8}$. This is an estimate for the number of iteration needed to obtain an accuracy of 10^{-8} . These values are comparable to the exact values given in Table 11.1.

11.3.4 Number of iterations

Consider next the **rate of convergence** of the iteration $\mathbf{x}_{k+1} = \mathbf{G}\mathbf{x}_k + \mathbf{c}$. We like to know how fast the iterative method converges. Suppose $\| \cdot \|$ is a matrix

	n=100	n=2500	k_{100}	k_{2500}
J	0.959493	0.998103	446	9703
GS	0.920627	0.99621	223	4852
SOR	0.56039	0.88402	32	150

Table 11.2: Spectral radia for \mathbf{G}_J , \mathbf{G}_1 , \mathbf{G}_{ω^*} and the smallest integer k_n such that $\rho(\mathbf{G})^{k_n} \leq 10^{-8}$.

norm that is subordinate to a vector norm also denoted by $\| \cdot \|$. Recall that $\mathbf{x}_k - \mathbf{x} = \mathbf{G}^k(\mathbf{x}_0 - \mathbf{x})$. For k sufficiently large

$$\|\mathbf{x}_k - \mathbf{x}\| \leq \|\mathbf{G}^k\| \|\mathbf{x}_0 - \mathbf{x}\| \approx \rho(\mathbf{G})^k \|\mathbf{x}_0 - \mathbf{x}\|.$$

For the last formula we apply Theorem 11.30 which says that $\lim_{k \rightarrow \infty} \|\mathbf{G}^k\|^{1/k} = \rho(\mathbf{G})$. For Jacobi's method and the spectral norm we have $\|\mathbf{G}_J^k\|_2 = \rho(\mathbf{G}_J)^k$ (cf. (11.26)).

For fast convergence we should use a \mathbf{G} with small spectral radius.

Lemma 11.22 (Number of iterations)

Suppose $\rho(\mathbf{G}) = 1 - \eta$ for some $0 < \eta < 1$, $\| \cdot \|$ a consistent matrix norm on $\mathbb{C}^{n \times n}$, and let $s \in \mathbb{N}$. Then

$$\tilde{k} := \frac{s \log(10)}{\eta} \quad (11.31)$$

is an estimate for the smallest number of iterations k so that $\rho(\mathbf{G})^k \leq 10^{-s}$.

Proof. The estimate \tilde{k} is an approximate solution of the equation $\rho(\mathbf{G})^k = 10^{-s}$. Thus, since $-\log(1 - \eta) \approx \eta$ when η is small

$$k = -\frac{s \log(10)}{\log(1 - \eta)} \approx \frac{s \log(10)}{\eta} = \tilde{k}.$$

□

The following estimates are obtained. They agree with those we found numerically in Section 11.1.1.

- R and J: $\rho(\mathbf{G}_J) = \cos(\pi h) = 1 - \eta$, $\eta = 1 - \cos(\pi h) = \frac{1}{2}\pi^2 h^2 + O(h^4) = \frac{\pi^2}{2}/n + O(n^{-2})$. Thus

$$\tilde{k}_n = \frac{2 \log(10)s}{\pi^2} n + O(n^{-1}) = O(n).$$

- GS: $\rho(\mathbf{G}_1) = \cos^2(\pi h) = 1 - \eta$, $\eta = 1 - \cos^2(\pi h) = \sin^2 \pi h = \pi^2 h^2 + O(h^4) = \pi^2/n + O(n^{-2})$. Thus

$$\tilde{k}_n = \frac{\log(10)s}{\pi^2} n + O(n^{-1}) = O(n).$$

- SOR: $\rho(\mathbf{G}_{\omega^*}) = \frac{1-\sin(\pi h)}{1+\sin(\pi h)} = 1 - 2\pi h + O(h^2)$. Thus,

$$\tilde{k}_n = \frac{\log(10)s}{2\pi} \sqrt{n} + O(n^{-1/2}) = O(\sqrt{n}).$$

Exercise 11.23 (Convergence example for fix point iteration)

Consider for $a \in \mathbb{C}$

$$\mathbf{x} := \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 & a \\ a & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1-a \\ 1-a \end{bmatrix} =: \mathbf{G}\mathbf{x} + \mathbf{c}.$$

Starting with $\mathbf{x}_0 = \mathbf{0}$ show by induction

$$\mathbf{x}_k(1) = \mathbf{x}_k(2) = 1 - a^k, \quad k \geq 0,$$

and conclude that the iteration converges to the fixed-point $\mathbf{x} = [1, 1]^T$ for $|a| < 1$ and diverges for $|a| > 1$. Show that $\rho(\mathbf{G}) = 1 - \eta$ with $\eta = 1 - |a|$. Compute the estimate (11.31) for the rate of convergence for $a = 0.9$ and $s = 16$ and compare with the true number of iterations determined from $|a|^k \leq 10^{-16}$.

Exercise 11.24 (Estimate in Lemma 11.22 can be exact)

Consider the iteration in Example 11.5. Show that $\rho(\mathbf{G}_J) = 1/2$. Then show that $\mathbf{x}_k(1) = \mathbf{x}_k(2) = 1 - 2^{-k}$ for $k \geq 0$. Thus the estimate in Lemma 11.22 is exact in this case.

We note that

1. The convergence depends on the behavior of the powers \mathbf{G}^k as k increases. The matrix \mathbf{M} should be chosen so that all elements in \mathbf{G}^k converge quickly to zero and such that the linear system (11.9) is easy to solve for \mathbf{x}_{k+1} . These are conflicting demands. \mathbf{M} should be an approximation to \mathbf{A} to obtain a \mathbf{G} with small elements, but then (11.9) might not be easy to solve for \mathbf{x}_{k+1} .
2. The convergence $\lim_{k \rightarrow \infty} \|\mathbf{G}^k\|^{1/k} = \rho(\mathbf{G})$ can be quite slow (cf. Exercise 11.25).

Exercise 11.25 (Slow spectral radius convergence)

The convergence $\lim_{k \rightarrow \infty} \|\mathbf{A}^k\|^{1/k} = \rho(\mathbf{A})$ can be quite slow. Consider

$$\mathbf{A} := \begin{bmatrix} \lambda & a & 0 & \cdots & 0 & 0 \\ 0 & \lambda & a & \cdots & 0 & 0 \\ 0 & 0 & \lambda & \cdots & 0 & 0 \\ \vdots & & & & \vdots & \\ 0 & 0 & 0 & \cdots & \lambda & a \\ 0 & 0 & 0 & \cdots & 0 & \lambda \end{bmatrix} \in \mathbb{R}^{n \times n}.$$

If $|\lambda| = \rho(\mathbf{A}) < 1$ then $\lim_{k \rightarrow \infty} \mathbf{A}^k = \mathbf{0}$ for any $a \in \mathbb{R}$. We show below that the $(1, n)$ element of \mathbf{A}^k is given by $f(k) := \binom{k}{n-1} a^{n-1} \lambda^{k-n+1}$ for $k \geq n-1$.

- (a) Pick an n , e.g. $n = 5$, and make a plot of $f(k)$ for $\lambda = 0.9$, $a = 10$, and $n-1 \leq k \leq 200$. Your program should also compute $\max_k f(k)$. Use your program to determine how large k must be before $f(k) < 10^{-8}$.
- (b) We can determine the elements of \mathbf{A}^k explicitly for any k . Let $\mathbf{E} := (\mathbf{A} - \lambda \mathbf{I})/a$. Show by induction that $\mathbf{E}^k = \begin{bmatrix} \mathbf{0} & \mathbf{I}_{n-k} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$ for $1 \leq k \leq n-1$ and that $\mathbf{E}^n = \mathbf{0}$.
- (c) We have $\mathbf{A}^k = (a\mathbf{E} + \lambda \mathbf{I})^k = \sum_{j=0}^{\min\{k, n-1\}} \binom{k}{j} a^j \lambda^{k-j} \mathbf{E}^j$ and conclude that the $(1, n)$ element is given by $f(k)$ for $k \geq n-1$.

11.3.5 Stopping the iteration

In Algorithms 11.2 and 11.3 we had access to the exact solution and could stop the iteration when the error was sufficiently small in the infinity norm. The decision when to stop is obviously more complicated when the exact solution is not known. One possibility is to choose a vector norm, keep track of $\|\mathbf{x}_{k+1} - \mathbf{x}_k\|$, and stop when this number is sufficiently small. The following result indicates that $\|\mathbf{x}_k - \mathbf{x}\|$ can be quite large if $\|\mathbf{G}\|$ is close to one.

Lemma 11.26 (Be careful when stopping)

Suppose $\|\mathbf{G}\| < 1$ for some consistent matrix norm on $\mathbb{C}^{n \times n}$ which is subordinate to a vector norm also denoted by $\|\cdot\|$. If $\mathbf{x}_k = \mathbf{G}\mathbf{x}_{k-1} + \mathbf{c}$ and $\mathbf{x} = \mathbf{G}\mathbf{x} + \mathbf{c}$. Then

$$\|\mathbf{x}_k - \mathbf{x}_{k-1}\| \geq \frac{1 - \|\mathbf{G}\|}{\|\mathbf{G}\|} \|\mathbf{x}_k - \mathbf{x}\|, \quad k \geq 1. \quad (11.32)$$

Proof. We find

$$\begin{aligned} \|\mathbf{x}_k - \mathbf{x}\| &= \|\mathbf{G}(\mathbf{x}_{k-1} - \mathbf{x})\| \leq \|\mathbf{G}\| \|\mathbf{x}_{k-1} - \mathbf{x}\| \\ &= \|\mathbf{G}\| \|\mathbf{x}_{k-1} - \mathbf{x}_k + \mathbf{x}_k - \mathbf{x}\| \leq \|\mathbf{G}\| (\|\mathbf{x}_{k-1} - \mathbf{x}_k\| + \|\mathbf{x}_k - \mathbf{x}\|). \end{aligned}$$

Thus $(1 - \|\mathbf{G}\|) \|\mathbf{x}_k - \mathbf{x}\| \leq \|\mathbf{G}\| \|\mathbf{x}_{k-1} - \mathbf{x}_k\|$ which implies (11.32). \square

Another possibility is to stop when the residual vector $\mathbf{r}_k := \mathbf{b} - \mathbf{A}\mathbf{x}_k$ is sufficiently small in some norm. To use the residual vector for stopping it is convenient to write the iterative method (11.10) in an alternative form. If \mathbf{M} is the splitting matrix of the method then by (11.9) we have $\mathbf{M}\mathbf{x}_{k+1} = \mathbf{M}\mathbf{x}_k - \mathbf{A}\mathbf{x}_k + \mathbf{b}$. This leads to

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{M}^{-1}\mathbf{r}_k, \quad \mathbf{r}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k. \quad (11.33)$$

Testing on \mathbf{r}_k works fine if \mathbf{A} is well conditioned, but Theorem 7.31 shows that the relative error in the solution can be much larger than the relative error in \mathbf{r}_k if \mathbf{A} is ill-conditioned.

11.4 Powers of a matrix

Let $\mathbf{A} \in \mathbb{C}^{n \times n}$ be a square matrix. In this section we consider the special matrix sequence $\{\mathbf{A}^k\}$ of powers of \mathbf{A} . We want to know when this sequence converges to the zero matrix. Such a sequence occurs in iterative methods (cf. (11.16)), in Markov processes in statistics, in the converge of geometric series of matrices (Neumann series cf. Section 11.4.2) and in many other applications.

11.4.1 The spectral radius

In this section we show the following theorem.

Theorem 11.27 (When is $\lim_{k \rightarrow \infty} \mathbf{A}^k = \mathbf{0}$?)

For any $\mathbf{A} \in \mathbb{C}^{n \times n}$ we have

$$\lim_{k \rightarrow \infty} \mathbf{A}^k = \mathbf{0} \iff \rho(\mathbf{A}) < 1,$$

where $\rho(\mathbf{A})$ is the spectral radius of \mathbf{A} given by (11.17).

Clearly $\rho(\mathbf{A}) < 1$ is a necessary condition for $\lim_{k \rightarrow \infty} \mathbf{A}^k = \mathbf{0}$. For if (λ, \mathbf{x}) is an eigenpair of \mathbf{A} with $|\lambda| \geq 1$ and $\|\mathbf{x}\|_2 = 1$ then $\mathbf{A}^k \mathbf{x} = \lambda^k \mathbf{x}$, and this implies $\|\mathbf{A}^k\|_2 \geq \|\mathbf{A}^k \mathbf{x}\|_2 = \|\lambda^k \mathbf{x}\|_2 = |\lambda|^k$, and it follows that \mathbf{A}^k does not tend to zero.

The sufficiency condition is harder to show. We construct a consistent matrix norm on $\mathbb{C}^{n \times n}$ such that $\|\mathbf{A}\| < 1$ and then use Theorems 11.7 and 11.8.

We start with

Theorem 11.28 (Any consistent norm majorizes the spectral radius)

For any matrix norm $\|\cdot\|$ that is consistent on $\mathbb{C}^{n \times n}$ and any $\mathbf{A} \in \mathbb{C}^{n \times n}$ we have $\rho(\mathbf{A}) \leq \|\mathbf{A}\|$.

Proof. Let (λ, \mathbf{x}) be an eigenpair for \mathbf{A} and define $\mathbf{X} := [\mathbf{x}, \dots, \mathbf{x}] \in \mathbb{C}^{n \times n}$. Then $\lambda \mathbf{X} = \mathbf{A} \mathbf{X}$, which implies $|\lambda| \|\mathbf{X}\| = \|\lambda \mathbf{X}\| = \|\mathbf{A} \mathbf{X}\| \leq \|\mathbf{A}\| \|\mathbf{X}\|$. Since $\|\mathbf{X}\| \neq 0$ we obtain $|\lambda| \leq \|\mathbf{A}\|$. \square

The next theorem shows that if $\rho(\mathbf{A}) < 1$ then $\|\mathbf{A}\| < 1$ for some consistent matrix norm on $\mathbb{C}^{n \times n}$, thus completing the proof of Theorem 11.27.

Theorem 11.29 (The spectral radius can be approximated by a norm)

Let $\mathbf{A} \in \mathbb{C}^{n \times n}$ and $\epsilon > 0$ be given. There is a consistent matrix norm $\|\cdot\|$ on $\mathbb{C}^{n \times n}$ such that $\rho(\mathbf{A}) \leq \|\mathbf{A}\| \leq \rho(\mathbf{A}) + \epsilon$.

Proof. Let \mathbf{A} have eigenvalues $\lambda_1, \dots, \lambda_n$. By the Schur Triangulation Theorem 5.28 there is a unitary matrix \mathbf{U} and an upper triangular matrix $\mathbf{R} = [r_{ij}]$ such that $\mathbf{U}^* \mathbf{A} \mathbf{U} = \mathbf{R}$. For $t > 0$ we define $\mathbf{D}_t := \text{diag}(t, t^2, \dots, t^n) \in \mathbb{R}^{n \times n}$, and note that the (i, j) element in $\mathbf{D}_t \mathbf{R} \mathbf{D}_t^{-1}$ is given by $t^{i-j} r_{ij}$ for all i, j . For $n = 3$

$$\mathbf{D}_t \mathbf{R} \mathbf{D}_t^{-1} = \begin{bmatrix} \lambda_1 & t^{-1} r_{12} & t^{-2} r_{13} \\ 0 & \lambda_2 & t^{-1} r_{23} \\ 0 & 0 & \lambda_3 \end{bmatrix}.$$

For each $\mathbf{B} \in \mathbb{C}^{n \times n}$ and $t > 0$ we use the one norm to define the matrix norm $\|\mathbf{B}\|_t := \|\mathbf{D}_t \mathbf{U}^* \mathbf{B} \mathbf{U} \mathbf{D}_t^{-1}\|_1$. We leave it as an exercise to show that $\|\cdot\|_t$ is a consistent matrix norm on $\mathbb{C}^{n \times n}$. We define $\|\mathbf{B}\| := \|\mathbf{B}\|_t$, where t is chosen so large that the sum of the absolute values of all off-diagonal elements in $\mathbf{D}_t \mathbf{R} \mathbf{D}_t^{-1}$ is less than ϵ . Then

$$\begin{aligned} \|\mathbf{A}\| &= \|\mathbf{D}_t \mathbf{U}^* \mathbf{A} \mathbf{U} \mathbf{D}_t^{-1}\|_1 = \|\mathbf{D}_t \mathbf{R} \mathbf{D}_t^{-1}\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |(\mathbf{D}_t \mathbf{R} \mathbf{D}_t^{-1})_{ij}| \\ &\leq \max_{1 \leq j \leq n} (|\lambda_j| + \epsilon) = \rho(\mathbf{A}) + \epsilon. \end{aligned}$$

□

A consistent matrix norm of a matrix can be much larger than the spectral radius. However the following result holds.

Theorem 11.30 (Spectral radius convergence)

For any consistent matrix norm $\|\cdot\|$ on $\mathbb{C}^{n \times n}$ and any $\mathbf{A} \in \mathbb{C}^{n \times n}$ we have

$$\lim_{k \rightarrow \infty} \|\mathbf{A}^k\|^{1/k} = \rho(\mathbf{A}). \quad (11.34)$$

Proof. If λ is an eigenvalue of \mathbf{A} then λ^k is an eigenvalue of \mathbf{A}^k for any $k \in \mathbb{N}$. By Theorem 11.28 we then obtain $\rho(\mathbf{A})^k = \rho(\mathbf{A}^k) \leq \|\mathbf{A}^k\|$ for any $k \in \mathbb{N}$ so that $\rho(\mathbf{A}) \leq \|\mathbf{A}^k\|^{1/k}$. Let $\epsilon > 0$ and consider the matrix $\mathbf{B} := (\rho(\mathbf{A}) + \epsilon)^{-1} \mathbf{A}$. Then $\rho(\mathbf{B}) = \rho(\mathbf{A}) / (\rho(\mathbf{A}) + \epsilon) < 1$ and $\|\mathbf{B}^k\| \rightarrow 0$ by Theorem 11.27 as $k \rightarrow \infty$. Choose $N \in \mathbb{N}$ such that $\|\mathbf{B}^k\| < 1$ for all $k \geq N$. Then for $k \geq N$

$$\|\mathbf{A}^k\| = \|(\rho(\mathbf{A}) + \epsilon)^k \mathbf{B}^k\| = (\rho(\mathbf{A}) + \epsilon)^k \|\mathbf{B}^k\| < (\rho(\mathbf{A}) + \epsilon)^k.$$

We have shown that $\rho(\mathbf{A}) \leq \|\mathbf{A}^k\|^{1/k} \leq \rho(\mathbf{A}) + \epsilon$ for $k \geq N$. Since ϵ is arbitrary the result follows. \square

Exercise 11.31 (A special norm)

Show that $\|\mathbf{B}\|_t := \|\mathbf{D}_t \mathbf{U}^* \mathbf{B} \mathbf{U} \mathbf{D}_t^{-1}\|_1$ defined in the proof of Theorem 11.29 is a consistent matrix norm on $\mathbb{C}^{n \times n}$.

11.4.2 Neumann series



Carl Neumann., 1832–1925. He studied potential theory. The Neumann boundary conditions are named after him.

Let \mathbf{B} be a square matrix. In this section we consider the **Neumann series** $\sum_{k=0}^{\infty} \mathbf{B}^k$ which is a matrix analogue of a geometric series of numbers.

Consider an infinite series $\sum_{k=0}^{\infty} \mathbf{A}_k$ of matrices in $\mathbb{C}^{n \times n}$. We say that the series converges if the sequence of partial sums $\{\mathbf{S}_m\}$ given by $\mathbf{S}_m = \sum_{k=0}^m \mathbf{A}_k$ converges. The series converges if and only if $\{\mathbf{S}_m\}$ is a Cauchy sequence, i.e. to each $\epsilon > 0$ there exists an integer N so that $\|\mathbf{S}_l - \mathbf{S}_m\| < \epsilon$ for all $l > m \geq N$.

Theorem 11.32 (Neumann series)

Suppose $\mathbf{B} \in \mathbb{C}^{n \times n}$. Then

1. The series $\sum_{k=0}^{\infty} \mathbf{B}^k$ converges if and only if $\rho(\mathbf{B}) < 1$.
2. If $\rho(\mathbf{B}) < 1$ then $(\mathbf{I} - \mathbf{B})$ is nonsingular and $(\mathbf{I} - \mathbf{B})^{-1} = \sum_{k=0}^{\infty} \mathbf{B}^k$.
3. If $\|\mathbf{B}\| < 1$ for some consistent matrix norm $\|\cdot\|$ on $\mathbb{C}^{n \times n}$ then

$$\|(\mathbf{I} - \mathbf{B})^{-1}\| \leq \frac{1}{1 - \|\mathbf{B}\|}. \quad (11.35)$$

Proof.

1. Suppose $\rho(\mathbf{B}) < 1$. We show that $\mathbf{S}_m := \sum_{k=0}^m \mathbf{B}^k$ is a Cauchy sequence and hence convergent. Let $\epsilon > 0$. By Theorem 11.29 there is a consistent matrix norm $\|\cdot\|$ on $\mathbb{C}^{n \times n}$ such that $\|\mathbf{B}\| < 1$. Then for $l > m$

$$\|\mathbf{S}_l - \mathbf{S}_m\| = \left\| \sum_{k=m+1}^l \mathbf{B}^k \right\| \leq \sum_{k=m+1}^l \|\mathbf{B}\|^k \leq \|\mathbf{B}\|^{m+1} \sum_{k=0}^{\infty} \|\mathbf{B}\|^k = \frac{\|\mathbf{B}\|^{m+1}}{1 - \|\mathbf{B}\|}.$$

But then $\{\mathbf{S}_m\}$ is a Cauchy sequence provided N is such that $\frac{\|\mathbf{B}\|^{N+1}}{1-\|\mathbf{B}\|} < \epsilon$.

Conversely, suppose (λ, \mathbf{x}) is an eigenpair for \mathbf{B} with $|\lambda| \geq 1$. We find $\mathbf{S}_m \mathbf{x} = \sum_{k=0}^m \mathbf{B}^k \mathbf{x} = (\sum_{k=0}^m \lambda^k) \mathbf{x}$. Since λ^k does not tend to zero the series $\sum_{k=0}^{\infty} \lambda^k$ is not convergent and therefore $\{\mathbf{S}_m \mathbf{x}\}$ and hence $\{\mathbf{S}_m\}$ does not converge.

2. We have

$$\left(\sum_{k=0}^m \mathbf{B}^k \right) (\mathbf{I} - \mathbf{B}) = \mathbf{I} + \mathbf{B} + \cdots + \mathbf{B}^m - (\mathbf{B} + \cdots + \mathbf{B}^{m+1}) = \mathbf{I} - \mathbf{B}^{m+1}. \quad (11.36)$$

Since $\rho(\mathbf{B}) < 1$ we conclude that $\mathbf{B}^{m+1} \rightarrow 0$ and hence taking limits in (11.36) we obtain $(\sum_{k=0}^{\infty} \mathbf{B}^k)(\mathbf{I} - \mathbf{B}) = \mathbf{I}$ which completes the proof of 2.

3. By 2: $\|(\mathbf{I} - \mathbf{B})^{-1}\| = \|\sum_{k=0}^{\infty} \mathbf{B}^k\| \leq \sum_{k=0}^{\infty} \|\mathbf{B}\|^k = \frac{1}{1-\|\mathbf{B}\|}$.

□

Exercise 11.33 (When is $\mathbf{A} + \mathbf{E}$ nonsingular?)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is nonsingular and $\mathbf{E} \in \mathbb{C}^{n \times n}$. Show that $\mathbf{A} + \mathbf{E}$ is nonsingular if and only if $\rho(\mathbf{A}^{-1} \mathbf{E}) < 1$.

11.5 The Optimal SOR Parameter ω

The following analysis is only carried out for the discrete Poisson matrix. It also holds for the averaging matrix given by (9.8). A more general theory is presented in [37]. We will compare the eigenpair equations for \mathbf{G}_J and \mathbf{G}_{ω} . It is convenient to write these equations using the matrix formulation $\mathbf{T}\mathbf{V} + \mathbf{V}\mathbf{T} = h^2\mathbf{F}$. If $\mathbf{G}_J \mathbf{v} = \mu \mathbf{v}$ is an eigenpair of \mathbf{G}_J then

$$\frac{1}{4}(v_{i-1,j} + v_{i,j-1} + v_{i+1,j} + v_{i,j+1}) = \mu v_{i,j}, \quad i, j = 1, \dots, m, \quad (11.37)$$

where $\mathbf{V} := \text{vec}(\mathbf{v}) \in \mathbb{R}^{m \times m}$ and $v_{i,j} = 0$ if $i \in \{0, m+1\}$ or $j \in \{0, m+1\}$.

Suppose (λ, \mathbf{w}) is an eigenpair for \mathbf{G}_{ω} . By (11.21) $(\mathbf{I} - \omega \mathbf{L})^{-1}(\omega \mathbf{R} + (1-\omega)\mathbf{I})\mathbf{w} = \lambda \mathbf{w}$ or

$$(\omega \mathbf{R} + \lambda \omega \mathbf{L})\mathbf{w} = (\lambda + \omega - 1)\mathbf{w}, \quad (11.38)$$

where $l_{i,i-m} = r_{i,i+m} = 1/4$ for all i , and all other elements in \mathbf{L} and \mathbf{R} are equal to zero. Let $\mathbf{w} = \text{vec}(\mathbf{W})$, where $\mathbf{W} \in \mathbb{C}^{m \times m}$. Then (11.38) can be written

$$\frac{\omega}{4}(\lambda w_{i-1,j} + \lambda w_{i,j-1} + w_{i+1,j} + w_{i,j+1}) = (\lambda + \omega - 1)w_{i,j}, \quad (11.39)$$

where $w_{i,j} = 0$ if $i \in \{0, m+1\}$ or $j \in \{0, m+1\}$.

Theorem 11.34 (The optimal ω)

Consider the SOR method applied to the discrete Poisson matrix (9.8), where we use the natural ordering. Moreover, assume $\omega \in (0, 2)$.

1. If $\lambda \neq 0$ is an eigenvalue of \mathbf{G}_ω then

$$\mu := \frac{\lambda + \omega - 1}{\omega \lambda^{1/2}} \quad (11.40)$$

is an eigenvalue of \mathbf{G}_J .

2. If μ is an eigenvalue of \mathbf{G}_J and λ satisfies the equation

$$\mu \omega \lambda^{1/2} = \lambda + \omega - 1 \quad (11.41)$$

then λ is an eigenvalue of \mathbf{G}_ω .

Proof. Suppose (λ, \mathbf{w}) is an eigenpair for \mathbf{G}_ω . We claim that (μ, \mathbf{v}) is an eigenpair for \mathbf{G}_J , where μ is given by (11.40) and $\mathbf{v} = \vec{\mathbf{V}}$ with $v_{i,j} := \lambda^{-(i+j)/2} w_{i,j}$. Indeed, replacing $w_{i,j}$ by $\lambda^{(i+j)/2} v_{i,j}$ in (11.39) and cancelling the common factor $\lambda^{(i+j)/2}$ we obtain

$$\frac{\omega}{4}(v_{i-1,j} + v_{i,j-1} + v_{i+1,j} + v_{i,j+1}) = \lambda^{-1/2}(\lambda + \omega - 1)v_{i,j}.$$

But then

$$\mathbf{G}_J \mathbf{v} = (\mathbf{L} + \mathbf{R}) \mathbf{v} = \frac{\lambda + \omega - 1}{\omega \lambda^{1/2}} \mathbf{v} = \mu \mathbf{v}.$$

For the converse let (μ, \mathbf{v}) be an eigenpair for \mathbf{G}_J and let λ be a solution of (11.41). We define as before $\mathbf{v} = \text{vec}(\mathbf{V})$, $\mathbf{W} = \text{vec}(\mathbf{W})$ with $w_{i,j} := \lambda^{(i+j)/2} v_{i,j}$. Inserting this in (11.37) and canceling $\lambda^{-(i+j)/2}$ we obtain

$$\frac{1}{4}(\lambda^{1/2} w_{i-1,j} + \lambda^{1/2} w_{i,j-1} + \lambda^{-1/2} w_{i+1,j} + \lambda^{-1/2} w_{i,j+1}) = \mu w_{i,j}.$$

Multiplying by $\omega \lambda^{1/2}$ we obtain

$$\frac{\omega}{4}(\lambda w_{i-1,j} + \lambda w_{i,j-1} + w_{i+1,j} + w_{i,j+1}) = \omega \mu \lambda^{1/2} w_{i,j},$$

Thus, if $\omega \mu \lambda^{1/2} = \lambda + \omega - 1$ then by (11.39) (λ, \mathbf{w}) is an eigenpair for \mathbf{G}_ω . \square

Proof of Theorem 11.20

Combining statement 1 and 2 in Theorem 11.34 we see that $\rho(\mathbf{G}_\omega) = |\lambda(\mu)|$, where $\lambda(\mu)$ is an eigenvalue of \mathbf{G}_ω satisfying (11.41) for some eigenvalue μ of \mathbf{G}_J . The eigenvalues of \mathbf{G}_J are $\frac{1}{2} \cos(j\pi h) + \frac{1}{2} \cos(k\pi h)$, $j, k = 1, \dots, m$, so μ is real and

both μ and $-\mu$ are eigenvalues. Thus, to compute $\rho(\mathbf{G}_\omega)$ it is enough to consider (11.41) for a positive eigenvalue μ of \mathbf{G}_J . Solving (11.41) for $\lambda = \lambda(\mu)$ gives

$$\lambda(\mu) := \frac{1}{4} \left(\omega\mu \pm \sqrt{(\omega\mu)^2 - 4(\omega-1)} \right)^2. \quad (11.42)$$

Both roots $\lambda(\mu)$ are eigenvalues of \mathbf{G}_ω . The discriminant

$$d(\omega) := (\omega\mu)^2 - 4(\omega-1).$$

is strictly decreasing on $(0, 2)$ since

$$d'(\omega) = 2(\omega\mu^2 - 2) < 2(\omega-2) < 0.$$

Moreover $d(0) = 4 > 0$ and $d(2) = 4\mu^2 - 4 < 0$. As a function of ω , $\lambda(\mu)$ changes from real to complex when $d(\omega) = 0$. The root in $(0, 2)$ is

$$\omega = \tilde{\omega}(\mu) := 2 \frac{1 - \sqrt{1 - \mu^2}}{\mu^2} = \frac{2}{1 + \sqrt{1 - \mu^2}}. \quad (11.43)$$

In the complex case we find

$$|\lambda(\mu)| = \frac{1}{4} \left((\omega\mu)^2 + 4(\omega-1) - (\omega\mu)^2 \right) = \omega - 1, \quad \tilde{\omega}(\mu) < \omega < 2.$$

In the real case both roots of (11.42) are positive and the larger one is

$$\lambda(\mu) = \frac{1}{4} \left(\omega\mu + \sqrt{(\omega\mu)^2 - 4(\omega-1)} \right)^2, \quad 0 < \omega \leq \tilde{\omega}(\mu). \quad (11.44)$$

Both $\lambda(\mu)$ and $\tilde{\omega}(\mu)$ are strictly increasing as functions of μ . It follows that $|\lambda(\mu)|$ is maximized for $\mu = \rho(\mathbf{G}_J) =: \beta$ and for this value of μ we obtain (11.27) for $0 < \omega \leq \tilde{\omega}(\beta) = \omega^*$.

Evidently $\rho(\mathbf{G}_\omega) = \omega - 1$ is strictly increasing in $\omega^* < \omega < 2$. Equation (11.29) will follow if we can show that $\rho(\mathbf{G}_\omega)$ is strictly decreasing in $0 < \omega < \omega^*$. By differentiation

$$\frac{d}{d\omega} \left(\omega\beta + \sqrt{(\omega\beta)^2 - 4(\omega-1)} \right) = \frac{\beta\sqrt{(\omega\beta)^2 - 4(\omega-1)} + \omega\beta^2 - 2}{\sqrt{(\omega\beta)^2 - 4(\omega-1)}}.$$

Since $\beta^2(\omega^2\beta^2 - 4\omega + 4) < (2 - \omega\beta^2)^2$ the numerator is negative and the strict decrease of $\rho(\mathbf{G}_\omega)$ in $0 < \omega < \omega^*$ follows.

11.6 Review Questions

11.6.1 Consider a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ with nonzero diagonal elements.

- Define the J and GS method in component form,
- Do they always converge?
- Give a necessary and sufficient condition that $\mathbf{A}^n \rightarrow \mathbf{0}$.
- Is there a matrix norm $\| \cdot \|$ consistent on $\mathbb{C}^{n \times n}$ such that $\|\mathbf{A}\| < \rho(\mathbf{A})$?

11.6.2 What is a Neumann series? when does it converge?

11.6.3 How do we define convergence of a fixed point iteration $\mathbf{x}_{k+1} = \mathbf{Gx}_k + \mathbf{c}$?
When does it converge?

11.6.4 Define Richardson's method.

Chapter 12

The Conjugate Gradient Method



Magnus Rudolph Hestenes, 1906-1991 (left), Eduard L. Stiefel, 1909-1978 (right).

The **conjugate gradient method** was published by Hestenes and Stiefel in 1952, [12] as a direct method for solving linear systems. Today its main use is as an iterative method for solving large sparse linear systems. On a test problem we show that it performs as well as the SOR method with optimal acceleration parameter, and we do not have to estimate any such parameter. However the conjugate gradient method is restricted to symmetric positive definite systems. We also consider a mathematical formulation of the **preconditioned conjugate gradient method**. It is used to speed up convergence of the conjugate gradient method. We only give one example of a possible preconditioner. See [1] for a more complete treatment of iterative methods and preconditioning.

The conjugate gradient method can also be used for minimization and is related to a method known as **steepest descent**. This method and the conjugate gradient method are both minimization methods and iterative methods for solving linear equations.

Throughout this chapter $\mathbf{A} \in \mathbb{R}^{n \times n}$ will be a symmetric positive definite matrix. We recall that \mathbf{A} has positive eigenvalues and that the spectral (2-norm) condition number of \mathbf{A} is given by $\kappa := \frac{\lambda_{\max}}{\lambda_{\min}}$, where λ_{\max} and λ_{\min} are the largest

and smallest eigenvalue of \mathbf{A} .

The analysis of the methods in this chapter is in terms of two inner products on \mathbb{R}^n , the usual inner product $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T \mathbf{y}$ with the associated Euclidian norm $\|\mathbf{x}\|_2 = \sqrt{\mathbf{x}^T \mathbf{x}}$, and the **\mathbf{A} -inner product** and the corresponding **\mathbf{A} -norm** given by

$$\langle \mathbf{x}, \mathbf{y} \rangle_{\mathbf{A}} := \mathbf{x}^T \mathbf{A} \mathbf{y}, \quad \|\mathbf{y}\|_{\mathbf{A}} := \sqrt{\mathbf{y}^T \mathbf{A} \mathbf{y}}, \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \quad (12.1)$$

We note that the \mathbf{A} -inner product is an inner product on \mathbb{R}^n . Indeed, for any $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbb{R}^n$

1. $\langle \mathbf{x}, \mathbf{x} \rangle_{\mathbf{A}} = \mathbf{x}^T \mathbf{A} \mathbf{x} \geq 0$ and $\langle \mathbf{x}, \mathbf{x} \rangle_{\mathbf{A}} = 0$ if and only if $\mathbf{x} = \mathbf{0}$, since \mathbf{A} is positive definite,
2. $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathbf{A}} := \mathbf{x}^T \mathbf{A} \mathbf{y} = (\mathbf{x}^T \mathbf{A} \mathbf{y})^T = \mathbf{y}^T \mathbf{A}^T \mathbf{x} = \mathbf{y}^T \mathbf{A} \mathbf{x} = \langle \mathbf{y}, \mathbf{x} \rangle_{\mathbf{A}}$ by symmetry of \mathbf{A} ,
3. $\langle \mathbf{x} + \mathbf{y}, \mathbf{z} \rangle_{\mathbf{A}} := \mathbf{x}^T \mathbf{A} \mathbf{z} + \mathbf{y}^T \mathbf{A} \mathbf{z} = \langle \mathbf{x}, \mathbf{z} \rangle_{\mathbf{A}} + \langle \mathbf{y}, \mathbf{z} \rangle_{\mathbf{A}}$, true for any \mathbf{A} .

By Theorem 4.3 the \mathbf{A} -norm is a vector norm on \mathbb{R}^n since it is an inner product norm.

Exercise 12.1 (\mathbf{A} -norm)

One can show that the \mathbf{A} -norm is a vector norm on \mathbb{R}^n without using the fact that it is an inner product norm. Show this with the help of the Cholesky factorization of \mathbf{A} .

12.1 Quadratic Minimization and Steepest Descent

We start by discussing some aspect of quadratic minimization and its relation to solving linear systems.

Consider for $\mathbf{A} \in \mathbb{R}^{n \times n}$ and $\mathbf{b} \in \mathbb{R}^n$ the quadratic function $Q : \mathbb{R}^n \rightarrow \mathbb{R}$ given by

$$Q(\mathbf{y}) := \frac{1}{2} \mathbf{y}^T \mathbf{A} \mathbf{y} - \mathbf{b}^T \mathbf{y}. \quad (12.2)$$

As an example, some level curves of

$$Q(x, y) := \frac{1}{2} \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = x^2 - xy + y^2 \quad (12.3)$$

are shown in Figure 12.1. The level curves are ellipses and the graph of Q is a paraboloid (cf. Exercise 12.2).

Exercise 12.2 (Paraboloid)

Let $\mathbf{A} = \mathbf{U} \mathbf{D} \mathbf{U}^T$ be the spectral decomposition of \mathbf{A} , i.e., \mathbf{U} is orthonormal and

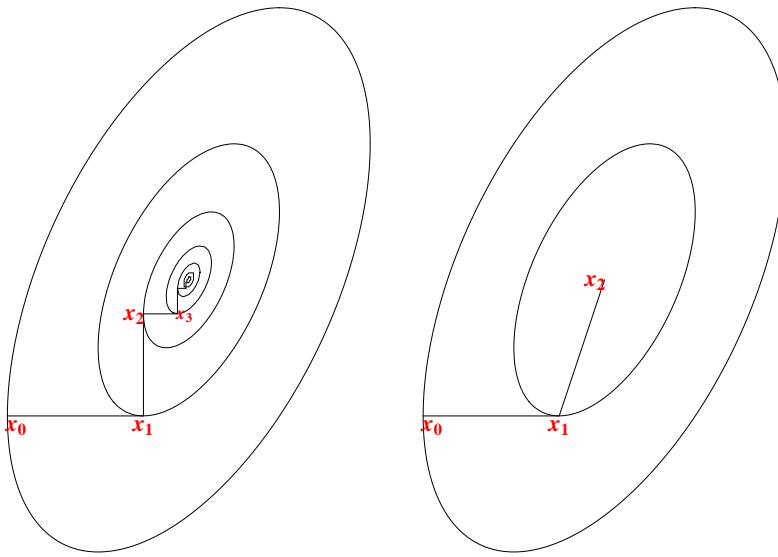


Figure 12.1: Level curves for $Q(x, y)$ given by (12.3). Also shown is a steepest descent iteration (left) and a conjugate gradient iteration (right) to find the minimum of Q . (cf Examples 12.4, 12.7)

$\mathbf{D} = \text{diag}(\lambda_1, \dots, \lambda_n)$ is diagonal. Define new variables $\mathbf{v} = [v_1, \dots, v_n]^T := \mathbf{U}^T \mathbf{y}$, and set $\mathbf{c} := \mathbf{U}^T \mathbf{b} = [c_1, \dots, c_n]^T$. Show that

$$Q(\mathbf{y}) = \frac{1}{2} \sum_{j=1}^n \lambda_j v_j^2 - \sum_{j=1}^n c_j v_j.$$

The following expansion will be used repeatedly. For $\mathbf{y}, \mathbf{h} \in \mathbb{R}^n$ and $\varepsilon \in \mathbb{R}$

$$Q(\mathbf{y} + \varepsilon \mathbf{h}) = Q(\mathbf{y}) - \varepsilon \mathbf{h}^T r(\mathbf{y}) + \frac{1}{2} \varepsilon^2 \mathbf{h}^T \mathbf{A} \mathbf{h}, \text{ where } r(\mathbf{y}) := \mathbf{b} - \mathbf{A} \mathbf{y}. \quad (12.4)$$

Minimizing a quadratic function is equivalent to solving a linear system.

Lemma 12.3 (Quadratic function)

A vector $\mathbf{x} \in \mathbb{R}^n$ minimizes Q given by (12.2) if and only if $\mathbf{A}\mathbf{x} = \mathbf{b}$. Moreover, the residual $r(\mathbf{y}) := \mathbf{b} - \mathbf{A}\mathbf{y}$ at any $\mathbf{y} \in \mathbb{R}^n$ is equal to the negative gradient, i.e., $r(\mathbf{y}) = -\nabla Q(\mathbf{y})$, where $\nabla := \left[\frac{\partial}{\partial y_1}, \dots, \frac{\partial}{\partial y_n} \right]^T$.

Proof. If $\mathbf{y} = \mathbf{x}$, $\varepsilon = 1$, and $\mathbf{A}\mathbf{x} = \mathbf{b}$ then (12.4) simplifies to $Q(\mathbf{x} + \mathbf{h}) = Q(\mathbf{x}) + \frac{1}{2} \mathbf{h}^T \mathbf{A} \mathbf{h}$, and since \mathbf{A} is symmetric positive definite $Q(\mathbf{x} + \mathbf{h}) > Q(\mathbf{x})$ for

all nonzero $\mathbf{h} \in \mathbb{R}^n$. It follows that \mathbf{x} is the unique minimum of Q . Conversely, if $\mathbf{Ax} \neq \mathbf{b}$ and $\mathbf{h} := \mathbf{r}(\mathbf{x})$, then by (12.4), $Q(\mathbf{x} + \varepsilon\mathbf{h}) - Q(\mathbf{x}) = -\varepsilon(\mathbf{h}^T \mathbf{r}(\mathbf{x}) - \frac{1}{2}\varepsilon\mathbf{h}^T \mathbf{A}\mathbf{h}) < 0$ for $\varepsilon > 0$ sufficiently small. Thus \mathbf{x} does not minimize Q . By (12.4) for $\mathbf{y} \in \mathbb{R}^n$

$$\begin{aligned}\frac{\partial}{\partial y_i} Q(\mathbf{y}) &:= \lim_{\varepsilon \rightarrow 0} \frac{1}{\varepsilon} (Q(\mathbf{y} + \varepsilon\mathbf{e}_i) - Q(\mathbf{y})) \\ &= \lim_{\varepsilon \rightarrow 0} \frac{1}{\varepsilon} \left(-\varepsilon\mathbf{e}_i^T \mathbf{r}(\mathbf{y}) + \frac{1}{2}\varepsilon^2 \mathbf{e}_i^T \mathbf{A} \mathbf{e}_i \right) = -\mathbf{e}_i^T \mathbf{r}(\mathbf{y}), \quad i = 1, \dots, n,\end{aligned}$$

showing that $\mathbf{r}(\mathbf{y}) = -\nabla Q(\mathbf{y})$. \square

A general class of minimization algorithms for Q and solution algorithms for a linear system is given as follows:

1. Choose $\mathbf{x}_0 \in \mathbb{R}^n$.

2. For $k = 0, 1, 2, \dots$

Choose a “search direction” \mathbf{p}_k ,
 Choose a “step length” α_k ,
 Compute $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$.

(12.5)

We would like to generate a sequence $\{\mathbf{x}_k\}$ that converges quickly to the minimum \mathbf{x} of Q .

For a fixed direction \mathbf{p}_k we say that α_k is **optimal** if $Q(\mathbf{x}_{k+1})$ is as small as possible, i.e.

$$Q(\mathbf{x}_{k+1}) = Q(\mathbf{x}_k + \alpha_k \mathbf{p}_k) = \min_{\alpha \in \mathbb{R}} Q(\mathbf{x}_k + \alpha \mathbf{p}_k).$$

By (12.4) we have $Q(\mathbf{x}_k + \alpha \mathbf{p}_k) = Q(\mathbf{x}_k) - \alpha \mathbf{p}_k^T \mathbf{r}_k + \frac{1}{2}\alpha^2 \mathbf{p}_k^T \mathbf{A} \mathbf{p}_k$, where $\mathbf{r}_k := \mathbf{b} - \mathbf{A}\mathbf{x}_k$. Since $\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k \geq 0$ we find a minimum α_k by solving $\frac{\partial}{\partial \alpha} Q(\mathbf{x}_k + \alpha \mathbf{p}_k) = 0$. It follows that the optimal α_k is uniquely given by

$$\alpha_k := \frac{\mathbf{p}_k^T \mathbf{r}_k}{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k}. \quad (12.6)$$

In the method of **Steepest descent**, also known as the **Gradient method** we choose $\mathbf{p}_k = \mathbf{r}_k$ the negative gradient, and the optimal α_k . Starting from \mathbf{x}_0 we compute for $k = 0, 1, 2, \dots$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \left(\frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k} \right) \mathbf{r}_k. \quad (12.7)$$

This is similar to Richardson's method (11.18), but in that method we used a constant step length. Computationally, a step in the steepest descent iteration can be organized as follows

$$\boxed{\begin{aligned}\mathbf{p}_k &= \mathbf{r}_k, \quad \mathbf{t}_k = \mathbf{A}\mathbf{p}_k, \\ \alpha_k &= (\mathbf{r}_k^T \mathbf{r}_k) / (\mathbf{p}_k^T \mathbf{t}_k), \\ \mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_k \mathbf{p}_k, \\ \mathbf{r}_{k+1} &= \mathbf{r}_k - \alpha_k \mathbf{t}_k.\end{aligned}} \quad (12.8)$$

Here, and in general, the following updating of the residual is used:

$$\mathbf{r}_{k+1} = \mathbf{b} - \mathbf{A}\mathbf{x}_{k+1} = \mathbf{b} - \mathbf{A}(\mathbf{x}_k + \alpha_k \mathbf{p}_k) = \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k. \quad (12.9)$$

In the steepest descent method the choice $\mathbf{p}_k = \mathbf{r}_k$ implies that the last two gradients are orthogonal. Indeed, by (12.9), $\mathbf{r}_{k+1}^T \mathbf{r}_k = (\mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k)^T \mathbf{p}_k = 0$ since $\alpha_k = \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{p}_k^T \mathbf{A}\mathbf{p}_k}$ and \mathbf{A} is symmetric.

Example 12.4 (Steepest descent iteration)

Suppose $Q(x, y)$ is given by (12.3). Starting with $\mathbf{x}_0 = [-1, -1/2]^T$ and $\mathbf{r}_0 = -\mathbf{A}\mathbf{x}_0 = [3/2, 0]^T$ we find

$$\begin{aligned}\mathbf{t}_0 &= 3 \begin{bmatrix} -1/2 \\ -1/2 \end{bmatrix}, \quad \alpha_0 = \frac{1}{2}, \quad \mathbf{x}_1 = -4^{-1} \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad \mathbf{r}_1 = 3 * 4^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \mathbf{t}_1 &= 3 * 4^{-1} \begin{bmatrix} -1 \\ 2 \end{bmatrix}, \quad \alpha_1 = \frac{1}{2}, \quad \mathbf{x}_2 = -4^{-1} \begin{bmatrix} 1 \\ 1/2 \end{bmatrix}, \quad \mathbf{r}_2 = 3 * 4^{-1} \begin{bmatrix} 1/2 \\ 0 \end{bmatrix},\end{aligned}$$

and in general for $k \geq 1$

$$\begin{aligned}\mathbf{t}_{2k-2} &= 3 * 4^{1-k} \begin{bmatrix} -1/2 \\ -1/2 \end{bmatrix}, \quad \mathbf{x}_{2k-1} = -4^{-k} \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad \mathbf{r}_{2k-1} = 3 * 4^{-k} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \mathbf{t}_{2k-1} &= 3 * 4^{-k} \begin{bmatrix} -1 \\ 2 \end{bmatrix}, \quad \mathbf{x}_{2k} = -4^{-k} \begin{bmatrix} 1 \\ 1/2 \end{bmatrix}, \quad \mathbf{r}_{2k} = 3 * 4^{-k} \begin{bmatrix} 1/2 \\ 0 \end{bmatrix}.\end{aligned}$$

Since $\alpha_k = 1/2$ is constant for all k the methods of Richardson, Jacobi and steepest descent are the same on this simple problem. See the left part of Figure 12.1. The rate of convergence is determined from $\|\mathbf{x}_{j+1}\|_2 / \|\mathbf{x}_j\| = \|\mathbf{r}_{j+1}\|_2 / \|\mathbf{r}_j\|_2 = 1/2$ for all j .

Exercise 12.5 (Steepest descent iteration)

Verify the numbers in Example 12.4.

12.2 The Conjugate Gradient Method

In the steepest descent method the last two gradients are orthogonal. In the conjugate gradient method all gradients are orthogonal¹⁹. We achieve this by using **A -orthogonal search directions** i.e., $\mathbf{p}_i^T \mathbf{A}\mathbf{p}_j = 0$ for all $i \neq j$.

¹⁹It is this property that has given the method its name.

12.2.1 Derivation of the method

As in the steepest descent method we choose a starting vector $\mathbf{x}_0 \in \mathbb{R}^n$. If $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0 = \mathbf{0}$ then \mathbf{x}_0 is the exact solution and we are finished, otherwise we initially make a steepest descent step. It follows that $\mathbf{r}_1^T \mathbf{r}_0 = 0$ and $\mathbf{p}_0 := \mathbf{r}_0$.

For the general case we define for $j \geq 0$

$$\mathbf{p}_j := \mathbf{r}_j - \sum_{i=0}^{j-1} \left(\frac{\mathbf{r}_j^T \mathbf{A} \mathbf{p}_i}{\mathbf{p}_i^T \mathbf{A} \mathbf{p}_i} \right) \mathbf{p}_i, \quad (12.10)$$

$$\mathbf{x}_{j+1} := \mathbf{x}_j + \alpha_j \mathbf{p}_j \quad \alpha_j := \frac{\mathbf{r}_j^T \mathbf{r}_j}{\mathbf{p}_j^T \mathbf{A} \mathbf{p}_j}, \quad (12.11)$$

$$\mathbf{r}_{j+1} = \mathbf{r}_j - \alpha_j \mathbf{A} \mathbf{p}_j. \quad (12.12)$$

We note that

1. \mathbf{p}_j is computed by the Gram-Schmidt orthogonalization process applied to the residuals $\mathbf{r}_0, \dots, \mathbf{r}_j$ using the \mathbf{A} -inner product. The search directions are therefore \mathbf{A} -orthogonal and nonzero as long as the residuals are linearly independent.
2. Equation (12.12) follows from (12.9).
3. It can be shown that the step length α_j is optimal for all j (cf. Exercise 12.10)).

Lemma 12.6 (The residuals are orthogonal)

Suppose that for some $k \geq 0$ that \mathbf{x}_j is well defined, $\mathbf{r}_j \neq 0$, and $\mathbf{r}_i^T \mathbf{r}_j = 0$ for $i, j = 0, 1, \dots, k$, $i \neq j$. Then \mathbf{x}_{k+1} is well defined and $\mathbf{r}_{k+1}^T \mathbf{r}_j = 0$ for $j = 0, 1, \dots, k$.

Proof. Since the residuals \mathbf{r}_j are orthogonal and nonzero for $j \leq k$, they are linearly independent, and it follows from the Gram-Schmidt Theorem 4.9 that \mathbf{p}_k is nonzero and $\mathbf{p}_i^T \mathbf{A} \mathbf{p}_k = 0$ for $i < k$. But then \mathbf{x}_{k+1} and \mathbf{r}_{k+1} are well defined. Now

$$\begin{aligned} \mathbf{r}_{k+1}^T \mathbf{r}_j &\stackrel{(12.12)}{=} (\mathbf{r}_k - \alpha_k \mathbf{A} \mathbf{p}_k)^T \mathbf{r}_j \\ &\stackrel{(12.10)}{=} \mathbf{r}_k^T \mathbf{r}_j - \alpha_k \mathbf{p}_k^T \mathbf{A} \left(\mathbf{p}_j + \sum_{i=0}^{j-1} \left(\frac{\mathbf{r}_j^T \mathbf{A} \mathbf{p}_i}{\mathbf{p}_i^T \mathbf{A} \mathbf{p}_i} \right) \mathbf{p}_i \right) \\ &\stackrel{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_i = 0}{=} \mathbf{r}_k^T \mathbf{r}_j - \alpha_k \mathbf{p}_k^T \mathbf{A} \mathbf{p}_j = 0, \quad j = 0, 1, \dots, k. \end{aligned}$$

That the final expression is equal to zero follows by orthogonality and \mathbf{A} -orthogonality for $j < k$ and by the definition of α_k for $j = k$. This completes the proof. \square

The conjugate gradient method is also a direct method. The residuals are orthogonal and therefore linearly independent if they are nonzero. Since $\dim \mathbb{R}^n = n$ the $n + 1$ residuals $\mathbf{r}_0, \dots, \mathbf{r}_n$ cannot all be nonzero and we must have $\mathbf{r}_k = 0$ for some $k \leq n$. Thus we find the exact solution in at most n iterations.

The expression (12.10) for \mathbf{p}_k can be greatly simplified. All terms except the last one vanish, since by orthogonality of the residuals

$$\mathbf{r}_j^T \mathbf{A} \mathbf{p}_i \stackrel{(12.12)}{=} \mathbf{r}_j^T \left(\frac{\mathbf{r}_i - \mathbf{r}_{i+1}}{\alpha_i} \right) = 0, \quad i = 0, 1, \dots, j-2.$$

With $j = k + 1$ (12.10) therefore takes the simple form $\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k$ and we find

$$\beta_k := -\frac{\mathbf{r}_{k+1}^T \mathbf{A} \mathbf{p}_k}{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k} \stackrel{(12.12)}{=} \frac{\mathbf{r}_{k+1}^T (\mathbf{r}_{k+1} - \mathbf{r}_k)}{\alpha_k \mathbf{p}_k^T \mathbf{A} \mathbf{p}_k} \stackrel{(12.11)}{=} \frac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k}. \quad (12.13)$$

To summarize, in the **conjugate gradient method** we start with $\mathbf{x}_0, \mathbf{p}_0 = \mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$ and then generate a sequence of vectors $\{\mathbf{x}_k\}$ as follows:

For $k = 0, 1, 2, \dots$

$$\mathbf{x}_{k+1} := \mathbf{x}_k + \alpha_k \mathbf{p}_k, \quad \alpha_k := \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k}, \quad (12.14)$$

$$\mathbf{r}_{k+1} := \mathbf{r}_k - \alpha_k \mathbf{A} \mathbf{p}_k, \quad (12.15)$$

$$\mathbf{p}_{k+1} := \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k, \quad \beta_k := \frac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k}. \quad (12.16)$$

The residuals and search directions are orthogonal and \mathbf{A} -orthogonal, respectively.

For computation we organize the iterations as follows for $k = 0, 1, 2, \dots$

$$\begin{aligned} \mathbf{t}_k &= \mathbf{A} \mathbf{p}_k, \\ \alpha_k &= (\mathbf{r}_k^T \mathbf{r}_k) / (\mathbf{p}_k^T \mathbf{t}_k), \\ \mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_k \mathbf{p}_k, \\ \mathbf{r}_{k+1} &= \mathbf{r}_k - \alpha_k \mathbf{t}_k, \\ \beta_k &= (\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}) / (\mathbf{r}_k^T \mathbf{r}_k), \\ \mathbf{p}_{k+1} &:= \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k. \end{aligned} \quad (12.17)$$

Note that (12.17) differs from (12.8) only in the computation of the search direction.

Example 12.7 (Conjugate gradient iteration)

Consider (12.17) applied to the positive definite linear system $\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$.

Starting as in Example 12.4 with $\mathbf{x}_0 = \begin{bmatrix} -1 \\ -1/2 \end{bmatrix}$ we find $\mathbf{p}_0 = \mathbf{r}_0 = \begin{bmatrix} 3/2 \\ 0 \end{bmatrix}$ and then

$$\mathbf{t}_0 = \begin{bmatrix} 3 \\ -3/2 \end{bmatrix}, \quad \alpha_0 = 1/2, \quad \mathbf{x}_1 = \begin{bmatrix} -1/4 \\ -1/2 \end{bmatrix}, \quad \mathbf{r}_1 = \begin{bmatrix} 0 \\ 3/4 \end{bmatrix}, \quad \beta_0 = 1/4, \quad \mathbf{p}_1 = \begin{bmatrix} 3/8 \\ 3/4 \end{bmatrix},$$

$$\mathbf{t}_1 = \begin{bmatrix} 0 \\ 9/8 \end{bmatrix}, \quad \alpha_1 = 2/3, \quad \mathbf{x}_2 = \mathbf{0}, \quad \mathbf{r}_2 = \mathbf{0}.$$

Thus \mathbf{x}_2 is the exact solution as illustrated in the right part of Figure 12.1.

Exercise 12.8 (Conjugate gradient iteration, II)

Do one iteration with the conjugate gradient method when $\mathbf{x}_0 = \mathbf{0}$. (Answer:
 $\mathbf{x}_1 = \left(\frac{\mathbf{b}^T \mathbf{b}}{\mathbf{b}^T \mathbf{A} \mathbf{b}} \right) \mathbf{b}$.)

Exercise 12.9 (Conjugate gradient iteration, III)

Do two conjugate gradient iterations for the system

$$\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 3 \end{bmatrix}$$

starting with $\mathbf{x}_0 = \mathbf{0}$.

Exercise 12.10 (The cg step length is optimal)

Show that the step length α_k in the conjugate gradient method is optimal²⁰.

Exercise 12.11 (Starting value in cg)

Show that the conjugate gradient method (12.17) for $\mathbf{Ax} = \mathbf{b}$ starting with \mathbf{x}_0 is the same as applying the method to the system $\mathbf{Ay} = \mathbf{r}_0 := \mathbf{b} - \mathbf{Ax}_0$ starting with $\mathbf{y}_0 = \mathbf{0}$.²¹

12.2.2 The conjugate gradient algorithm

In this section we give numerical examples and discuss implementation.

The formulas in (12.17) form a basis for an algorithm.

²⁰Hint: use induction on k to show that $\mathbf{p}_k = \mathbf{r}_k + \sum_{j=0}^{k-1} a_{k,j} \mathbf{r}_j$ for some constants $a_{k,j}$.

²¹Hint: The conjugate gradient method for $\mathbf{Ay} = \mathbf{r}_0$ can be written $\mathbf{y}_{k+1} := \mathbf{y}_k + \gamma_k \mathbf{q}_k$, $\gamma_k := \frac{\mathbf{s}_k^T \mathbf{s}_k}{\mathbf{q}_k^T \mathbf{A} \mathbf{q}_k}$, $\mathbf{s}_{k+1} := \mathbf{s}_k - \gamma_k \mathbf{A} \mathbf{q}_k$, $\mathbf{q}_{k+1} := \mathbf{s}_{k+1} + \delta_k \mathbf{q}_k$, $\delta_k := \frac{\mathbf{s}_{k+1}^T \mathbf{s}_{k+1}}{\mathbf{s}_k^T \mathbf{s}_k}$. Show that $\mathbf{y}_k = \mathbf{x}_k - \mathbf{x}_0$, $\mathbf{s}_k = \mathbf{r}_k$, and $\mathbf{q}_k = \mathbf{p}_k$, for $k = 0, 1, 2, \dots$.

Algorithm 12.12 (Conjugate gradient iteration)

The symmetric positive definite linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ is solved by the conjugate gradient method. \mathbf{x} is a starting vector for the iteration. The iteration is stopped when $\|\mathbf{r}_k\|_2/\|\mathbf{b}\|_2 \leq \text{tol}$ or $k > \text{itmax}$. K is the number of iterations used.

```

1 function [x,K]=cg(A,b,x,tol,itmax)
2 r=b-A*x; p=r; rho0=b'*b; rho=r'*r;
3 for k=0:itmax
4   if sqrt(rho/rho0)<= tol
5     K=k; return
6   end
7   t=A*p; a=rho/(p'*t);
8   x=x+a*p; r=r-a*t;
9   rhos=rho; rho=r'*r;
10  p=r+(rho/rhos)*p;
11 end
12 K=itmax+1;
```

The work involved in each iteration is

1. one matrix times vector ($\mathbf{t} = \mathbf{A}\mathbf{p}$),
2. two inner products ($(\mathbf{p}^T \mathbf{t}$ and $\mathbf{r}^T \mathbf{r}$),
3. three vector-plus-scalar-times-vector ($\mathbf{x} = \mathbf{x} + a\mathbf{p}$, $\mathbf{r} = \mathbf{r} - a\mathbf{t}$ and $\mathbf{p} = \mathbf{r} + (rho/rhos)\mathbf{p}$),

The dominating part is the computation of $\mathbf{t} = \mathbf{A}\mathbf{p}$.

12.2.3 Numerical example

We test the conjugate gradient method on two examples. For a similar test for the steepest descent method see Exercise 12.17. Consider the matrix given by the Kronecker sum $\mathbf{T}_2 := \mathbf{T}_1 \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{T}_1$, where $\mathbf{T}_1 := \text{tridiag}_m(a, d, a) \in \mathbb{R}^{m \times m}$ and $a, d \in \mathbb{R}$. We recall that this matrix is symmetric positive definite if $d > 0$ and $d \geq 2|a|$ (cf. Theorem 9.8). We set $h = 1/(m+1)$ and $\mathbf{f} = [1, \dots, 1]^T \in \mathbb{R}^n$.

We consider two problems.

1. $a = 1/9$, $d = 5/18$, the Averaging matrix.
2. $a = -1$, $d = 2$, the Poisson matrix.

12.2.4 Implementation issues

Note that for our test problems \mathbf{T}_2 only has $O(5n)$ nonzero elements. Therefore, taking advantage of the sparseness of \mathbf{T}_2 we can compute \mathbf{t} in Algorithm 12.12

n	2 500	10 000	40 000	1 000 000	4 000 000
K	19	18	18	16	15

Table 12.13: The number of iterations K for the averaging problem on a $\sqrt{n} \times \sqrt{n}$ grid for various n

n	2 500	10 000	40 000	160 000
K	94	188	370	735
K/\sqrt{n}	1.88	1.88	1.85	1.84

Table 12.15: The number of iterations K for the Poisson problem on a $\sqrt{n} \times \sqrt{n}$ grid for various n

in $O(n)$ arithmetic operations. With such an implementation the total number of arithmetic operations in one iteration is $O(n)$. We also note that it is not necessary to store the matrix \mathbf{T}_2 .

To use the Conjugate Gradient Algorithm on the test matrix for large n it is advantageous to use a matrix equation formulation. We define matrices $\mathbf{V}, \mathbf{R}, \mathbf{P}, \mathbf{B}, \mathbf{T} \in \mathbb{R}^{m \times m}$ by $\mathbf{x} = \text{vec}(\mathbf{V})$, $\mathbf{r} = \text{vec}(\mathbf{R})$, $\mathbf{p} = \text{vec}(\mathbf{P})$, $\mathbf{t} = \text{vec}(\mathbf{T})$, and $h^2 \mathbf{f} = \text{vec}(\mathbf{B})$. Then $\mathbf{T}_2 \mathbf{x} = h^2 \mathbf{f} \iff \mathbf{T}_1 \mathbf{V} + \mathbf{V} \mathbf{T}_1 = \mathbf{B}$, and $\mathbf{t} = \mathbf{T}_2 \mathbf{p} \iff \mathbf{T} = \mathbf{T}_1 \mathbf{P} + \mathbf{P} \mathbf{T}_1$.

This leads to the following algorithm for testing the conjugate gradient algorithm on the matrix

$$\mathbf{A} = \text{tridiag}_m(a, d, a) \otimes \mathbf{I}_m + \mathbf{I}_m \otimes \text{tridiag}_m(a, d, a) \in \mathbb{R}^{(m^2) \times (m^2)}.$$

Algorithm 12.14 (Testing conjugate gradient)

```

1 function [V,K]=cgtest(m,a,d,tol,itmax)
2 R=ones(m)/(m+1)^2; rho=sum(sum(R.*R)); rho0=rho; P=R;
3 V=zeros(m,m); T1=sparse(tridiagonal(a,d,a,m));
4 for k=1:itmax
5   if sqrt(rho/rho0)<= tol
6     K=k; return
7   end
8   T=T1*P+P*T1;
9   a=rho/sum(sum(P.*T)); V=V+a*P; R=R-a*T;
10  rhos=rho; rho=sum(sum(R.*R)); P=R+(rho/rhos)*P;
11 end
12 K=itmax+1;
```

For both the averaging- and Poisson matrix we use $tol = 10^{-8}$.

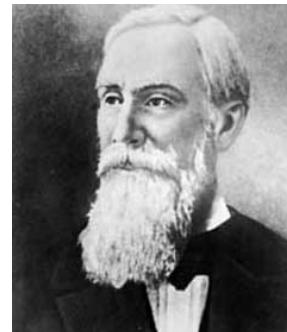
For the averaging matrix we obtain the values in Table 12.13.

The convergence is quite rapid. It appears that the number of iterations can be bounded independently of n , and therefore we solve the problem in $O(n)$ operations. This is the best we can do for a problem with n unknowns.

Consider next the Poisson problem. In Table 12.15 we list K , the required number of iterations, and K/\sqrt{n} .

The results show that K is much smaller than n and appears to be proportional to \sqrt{n} . This is the same speed as for SOR and we don't have to estimate any acceleration parameter.

12.3 Convergence



Leonid Vitaliyevich Kantorovich, 1912–1986 (left), Aleksey Nikolaevich Krylov, 1863–1945 (center), Pafnuty Lvovich Chebyshev, 1821–1894 (right)

12.3.1 The Main Theorem

Recall that the \mathbf{A} -norm of a vector $\mathbf{x} \in \mathbb{R}^n$ is given by $\|\mathbf{x}\|_{\mathbf{A}} := \sqrt{\mathbf{x}^T \mathbf{A} \mathbf{x}}$. The following theorem gives upper bounds for the \mathbf{A} -norm of the error in both steepest descent and conjugate gradients.

Theorem 12.16 (Error bound for steepest descent and conjugate gradients)

Suppose \mathbf{A} is symmetric positive definite. For the \mathbf{A} -norms of the errors in the steepest descent method (12.7) the following upper bounds hold

$$\frac{\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}}{\|\mathbf{x} - \mathbf{x}_0\|_{\mathbf{A}}} \leq \left(\frac{\kappa - 1}{\kappa + 1} \right)^k < e^{-\frac{2}{\kappa} k}, \quad , k > 0, \quad (12.18)$$

while for the conjugate gradient method we have

$$\frac{\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}}{\|\mathbf{x} - \mathbf{x}_0\|_{\mathbf{A}}} \leq 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k < 2e^{-\frac{2}{\sqrt{\kappa}}k}, \quad k \geq 0. \quad (12.19)$$

Here $\kappa = \text{cond}_2(\mathbf{A}) := \lambda_{\max}/\lambda_{\min}$ is the spectral condition number of \mathbf{A} , and λ_{\max} and λ_{\min} are the largest and smallest eigenvalue of \mathbf{A} , respectively.

Theorem 12.16 implies

1. Since $\frac{\kappa-1}{\kappa+1} < 1$ the steepest descent method always converges for a symmetric positive definite matrix. The convergence can be slow when $\frac{\kappa-1}{\kappa+1}$ is close to one, and this happens even for a moderately ill-conditioned \mathbf{A} .
2. The rate of convergence for the conjugate gradient method appears to be determined by the square root of the spectral condition number. This is much better than the estimate for the steepest descent method. Especially for problems with large condition numbers.
3. The proofs of the estimates in (12.18) and (12.19) are quite different. This is in spite of their similar appearance.

12.3.2 The number of iterations for the model problems

Consider the test matrix

$$\mathbf{T}_2 := \text{tridiag}_m(a, d, a) \otimes \mathbf{I}_m + \mathbf{I}_m \otimes \text{tridiag}_m(a, d, a) \in \mathbb{R}^{(m^2) \times (m^2)}.$$

The eigenvalues were given in (9.15) as

$$\lambda_{j,k} = 2d + 2a \cos(j\pi h) + 2a \cos(k\pi h), \quad j, k = 1, \dots, m. \quad (12.20)$$

For the averaging problem given by $d = 5/18$, $a = 1/9$, the largest and smallest eigenvalue of \mathbf{T}_2 are given by $\lambda_{\max} = \frac{5}{9} + \frac{4}{9} \cos(\pi h)$ and $\lambda_{\min} = \frac{5}{9} - \frac{4}{9} \cos(\pi h)$. Thus

$$\kappa_A = \frac{5 + 4 \cos(\pi h)}{5 - 4 \cos(\pi h)} \leq 9,$$

and the condition number is bounded independently of n . It follows from (12.19) that the number of iterations can be bounded independently of the size n of the problem, and this is in agreement with what we observed in Table 12.13.

For the Poisson problem we have by (11.24) the condition number

$$\kappa_P = \frac{\lambda_{\max}}{\lambda_{\min}} = \frac{\cos^2(\pi h/2)}{\sin^2(\pi h/2)} \text{ and } \sqrt{\kappa_P} = \frac{\cos(\pi h/2)}{\sin(\pi h/2)} \approx \frac{2}{\pi h} \approx \frac{2}{\pi} \sqrt{n}.$$

Thus, (see also Exercise 7.36) we solve the discrete Poisson problem in $O(n^{3/2})$ arithmetic operations using the conjugate gradient method. This is the same as for the SOR method and for the fast method without the FFT. In comparison the Cholesky Algorithm requires $O(n^2)$ arithmetic operations both for the averaging and the Poisson problem.

Exercise 12.17 (Program code for testing steepest descent)

Write a function $K=sdtest(m,a,d,tol,itmax)$ to test the Steepest descent method on the matrix \mathbf{T}_2 . Make the analogues of Table 12.13 and Table 12.15. For Table 12.15 it is enough to test for say $n = 100, 400, 1600, 2500$, and tabulate K/n instead of K/\sqrt{n} in the last row. Conclude that the upper bound (12.18) is realistic. Compare also with the number of iterations for the J and GS method in Table 11.1.

Exercise 12.18 (Using cg to solve normal equations)

Consider solving the linear system $\mathbf{A}^T \mathbf{A} \mathbf{x} = \mathbf{A}^T \mathbf{b}$ by using the conjugate gradient method. Here $\mathbf{A} \in \mathbb{R}^{m,n}$, $\mathbf{b} \in \mathbb{R}^m$ and $\mathbf{A}^T \mathbf{A}$ is positive definite²². Explain why only the following modifications in Algorithm 12.12 are necessary

1. $r=\mathbf{A}'(\mathbf{b}-\mathbf{A}^* \mathbf{x})$; $p=r$;
2. $a=rho/(t'*t)$;
3. $r=r-a*\mathbf{A}'*t$;

Note that the condition number of the normal equations is $\text{cond}_2(\mathbf{A})^2$, the square of the condition number of \mathbf{A} .

12.3.3 Krylov spaces and the best approximation property

For the convergence analysis of the conjugate gradient method certain subspaces of \mathbb{R}^n called **Krylov spaces** play a central role. In fact the iterates in the conjugate gradient method are best approximation of the solution from these subspaces using the \mathbf{A} -norm to measure the error.

The Krylov spaces are defined by $\mathbb{W}_0 = \{\mathbf{0}\}$ and

$$\mathbb{W}_k = \text{span}(\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \mathbf{A}^2\mathbf{r}_0, \dots, \mathbf{A}^{k-1}\mathbf{r}_0), \quad k = 1, 2, 3, \dots.$$

They are nested subspaces

$$\mathbb{W}_0 \subset \mathbb{W}_1 \subset \mathbb{W}_2 \subset \dots \subset \mathbb{W}_n \subset \mathbb{R}^n$$

with $\dim(\mathbb{W}_k) \leq k$ for all $k \geq 0$. Moreover, If $\mathbf{v} \in \mathbb{W}_k$ then $\mathbf{A}\mathbf{v} \in \mathbb{W}_{k+1}$.

²²This system known as the **normal equations** appears in linear least squares problems and was considered in this context in Chapter 8.

Lemma 12.19 (Krylov space)

For the iterates in the conjugate gradient method we have

$$\mathbf{x}_k - \mathbf{x}_0 \in \mathbb{W}_k, \quad \mathbf{r}_k, \mathbf{p}_k \in \mathbb{W}_{k+1}, \quad k = 0, 1, \dots, \quad (12.21)$$

and

$$\mathbf{r}_k^T \mathbf{w} = \mathbf{p}_k^T \mathbf{A} \mathbf{w} = 0, \quad \mathbf{w} \in \mathbb{W}_k. \quad (12.22)$$

Proof. (12.21) clearly holds for $k = 0$ since $\mathbf{p}_0 = \mathbf{r}_0$. Suppose it holds for some $k \geq 0$. Then $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A} \mathbf{p}_k \in \mathbb{W}_{k+2}$ and by $\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k \in \mathbb{W}_{k+2}$ and $\mathbf{x}_{k+1} - \mathbf{x}_0 \stackrel{(12.11)}{=} \mathbf{x}_k - \mathbf{x}_0 + \alpha_k \mathbf{p}_k \in \mathbb{W}_{k+1}$. Thus (12.21) follows by induction. Since any $\mathbf{w} \in \mathbb{W}_k$ is a linear combination of $\{\mathbf{r}_0, \mathbf{r}_1, \dots, \mathbf{r}_{k-1}\}$ and also $\{\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_{k-1}\}$, (12.22) follows. \square

Theorem 12.20 (Best approximation property)

Suppose $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{n \times n}$ is symmetric positive definite and $\{\mathbf{x}_k\}$ is generated by the conjugate gradient method (cf. (12.14)). Then

$$\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}} = \min_{\mathbf{w} \in \mathbb{W}_k} \|\mathbf{x} - \mathbf{x}_0 - \mathbf{w}\|_{\mathbf{A}}. \quad (12.23)$$

Proof. Fix k , let $\mathbf{w} \in \mathbb{W}_k$ and $\mathbf{u} := \mathbf{x}_k - \mathbf{x}_0 - \mathbf{w}$. By (12.21) $\mathbf{u} \in \mathbb{W}_k$ and then (12.22) implies that $\mathbf{r}_k^T \mathbf{u} = 0$. Since $(\mathbf{x} - \mathbf{x}_k)^T \mathbf{A} \mathbf{u} = \mathbf{r}_k^T \mathbf{u}$ we find

$$\begin{aligned} \|\mathbf{x} - \mathbf{x}_0 - \mathbf{w}\|_{\mathbf{A}}^2 &= (\mathbf{x} - \mathbf{x}_k + \mathbf{u})^T \mathbf{A} (\mathbf{x} - \mathbf{x}_k + \mathbf{u}) \\ &= (\mathbf{x} - \mathbf{x}_k) \mathbf{A} (\mathbf{x} - \mathbf{x}_k) + 2\mathbf{r}_k^T \mathbf{u} + \mathbf{u}^T \mathbf{A} \mathbf{u} \\ &= \|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}^2 + \|\mathbf{u}\|_{\mathbf{A}}^2 \geq \|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}^2. \end{aligned}$$

Taking square roots we obtain $\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}} \leq \|\mathbf{x} - \mathbf{x}_0 - \mathbf{w}\|_{\mathbf{A}}$ with equality for $\mathbf{u} = \mathbf{0}$ and the result follows. \square

If $\mathbf{x}_0 = \mathbf{0}$ then (12.23) says that \mathbf{x}_k is the element in \mathbb{W}_k that is closest to the solution \mathbf{x} in the \mathbf{A} -norm. More generally, if $\mathbf{x}_0 \neq \mathbf{0}$ then $\mathbf{x} - \mathbf{x}_k = (\mathbf{x} - \mathbf{x}_0) - (\mathbf{x}_k - \mathbf{x}_0)$ and $\mathbf{x}_k - \mathbf{x}_0$ is the element in \mathbb{W}_k that is closest to $\mathbf{x} - \mathbf{x}_0$ in the \mathbf{A} -norm. This is the orthogonal projection of $\mathbf{x} - \mathbf{x}_0$ into \mathbb{W}_k , see Figure 12.2.

Recall that to each polynomial $p(t) := \sum_{j=0}^m a_j t^m$ there corresponds a matrix polynomial $p(\mathbf{A}) := a_0 \mathbf{I} + a_1 \mathbf{A} + \dots + a_m \mathbf{A}^m$. Moreover, if $(\lambda_j, \mathbf{u}_j)$ are eigenpairs of \mathbf{A} then $(p(\lambda_j), \mathbf{u}_j)$ are eigenpairs of $p(\mathbf{A})$ for $j = 1, \dots, n$.

Lemma 12.21 (Krylov space and polynomials)

Suppose $\mathbf{A}\mathbf{x} = \mathbf{b}$ where $\mathbf{A} \in \mathbb{R}^{n \times n}$ is symmetric positive definite with orthonormal

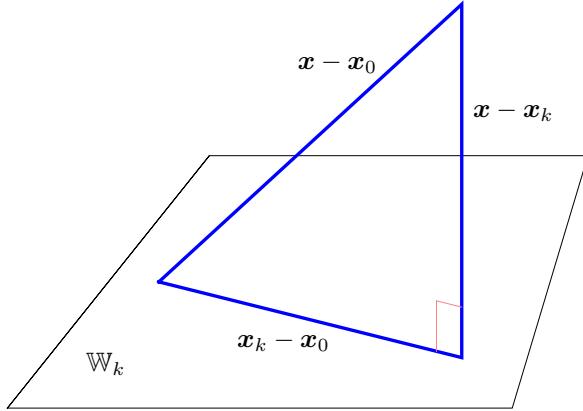


Figure 12.2: The orthogonal projection of $\mathbf{x} - \mathbf{x}_0$ into \mathbb{W}_k .

eigenpairs $(\lambda_j, \mathbf{u}_j)$, $j = 1, 2, \dots, n$, and let $\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$ for some $\mathbf{x}_0 \in \mathbb{R}^n$. To each $\mathbf{w} \in \mathbb{W}_k$ there corresponds a polynomial $P(t) := \sum_{j=0}^{k-1} a_j t^{k-1}$ such that $\mathbf{w} = P(\mathbf{A})\mathbf{r}_0$. Moreover, if $\mathbf{r}_0 = \sum_{j=1}^n \sigma_j \mathbf{u}_j$ then

$$\|\mathbf{x} - \mathbf{x}_0 - \mathbf{w}\|_{\mathbf{A}}^2 = \sum_{j=1}^n \frac{\sigma_j^2}{\lambda_j} Q(\lambda_j)^2, \quad Q(t) := 1 - tP(t). \quad (12.24)$$

Proof. If $\mathbf{w} \in \mathbb{W}_k$ then $\mathbf{w} = a_0 \mathbf{r}_0 + a_1 \mathbf{A}\mathbf{r}_0 + \dots + a_{k-1} \mathbf{A}^{k-1} \mathbf{r}_0$ for some scalars a_0, \dots, a_{k-1} . But then $\mathbf{w} = P(\mathbf{A})\mathbf{r}_0$. We find $\mathbf{x} - \mathbf{x}_0 - P(\mathbf{A})\mathbf{r}_0 = \mathbf{A}^{-1}(\mathbf{r}_0 - \mathbf{A}P(\mathbf{A}))\mathbf{r}_0 = \mathbf{A}^{-1}Q(\mathbf{A})\mathbf{r}_0$ and $\mathbf{A}(\mathbf{x} - \mathbf{x}_0 - P(\mathbf{A})\mathbf{r}_0) = Q(\mathbf{A})\mathbf{r}_0$. Therefore,

$$\|\mathbf{x} - \mathbf{x}_0 - P(\mathbf{A})\mathbf{r}_0\|_{\mathbf{A}}^2 = \mathbf{c}^T \mathbf{A}^{-1} \mathbf{c} \text{ where } \mathbf{c} = (\mathbf{I} - \mathbf{A}P(\mathbf{A}))\mathbf{r}_0 = Q(\mathbf{A})\mathbf{r}_0. \quad (12.25)$$

Using the eigenvector expansion for \mathbf{r}_0 we obtain

$$\mathbf{c} = \sum_{j=1}^n \sigma_j Q(\lambda_j) \mathbf{u}_j, \quad \mathbf{A}^{-1} \mathbf{c} = \sum_{i=1}^n \sigma_i \frac{Q(\lambda_i)}{\lambda_i} \mathbf{u}_i. \quad (12.26)$$

Now (12.24) follows by the orthonormality of the eigenvectors. \square

We will use the following theorem to estimate the rate of convergence.

Theorem 12.22 (cg and best polynomial approximation)

Suppose $[a, b]$ with $0 < a < b$ is an interval containing all the eigenvalues of \mathbf{A} .

Then in the conjugate gradient method

$$\frac{\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}}{\|\mathbf{x} - \mathbf{x}_0\|_{\mathbf{A}}} \leq \min_{\substack{Q \in \Pi_k \\ Q(0)=1}} \max_{a \leq x \leq b} |Q(x)|, \quad (12.27)$$

where Π_k denotes the class of univariate polynomials of degree $\leq k$ with real coefficients.

Proof. By (12.24) with $Q(t) = 1$ (corresponding to $P(\mathbf{A}) = \mathbf{0}$) we find $\|\mathbf{x} - \mathbf{x}_0\|_{\mathbf{A}}^2 = \sum_{j=1}^n \frac{\sigma_j^2}{\lambda_j}$. Therefore, by the best approximation property Theorem 12.20 and (12.24), for any $\mathbf{w} \in \mathbb{W}_k$

$$\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}^2 \leq \|\mathbf{x} - \mathbf{x}_0 - \mathbf{w}\|_{\mathbf{A}}^2 \leq \max_{a \leq x \leq b} |Q(x)|^2 \sum_{j=1}^n \frac{\sigma_j^2}{\lambda_j} = \max_{a \leq x \leq b} |Q(x)|^2 \|\mathbf{x} - \mathbf{x}_0\|_{\mathbf{A}}^2,$$

where $Q \in \Pi_k$ and $Q(0) = 1$. Minimizing over such polynomials Q and taking square roots the result follows. \square

In the next section we use properties of the Chebyshev polynomials to show that

$$\frac{\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}}{\|\mathbf{x} - \mathbf{x}_0\|_{\mathbf{A}}} \leq \min_{\substack{Q \in \Pi_k \\ Q(0)=1}} \max_{\lambda_{min} \leq x \leq \lambda_{max}} |Q(x)| = \frac{2}{a^{-k} + a^k}, \quad a := \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}, \quad (12.28)$$

where $\kappa = \lambda_{max}/\lambda_{min}$ is the spectral condition number of \mathbf{A} . Ignoring the second term in the denominator this implies the first inequality in (12.19).

Consider the second inequality in (12.19). The inequality

$$\frac{x-1}{x+1} < e^{-2/x} \quad \text{for } x > 1 \quad (12.29)$$

follows from the familiar series expansion of the exponential function. Indeed, with $y = 1/x$, using $2^k/k! = 2$, $k = 1, 2$, and $2^k/k! < 2$ for $k > 2$, we find

$$e^{2/x} = e^{2y} = \sum_{k=0}^{\infty} \frac{(2y)^k}{k!} < 1 + 2 \sum_{k=1}^{\infty} y^k = \frac{1+y}{1-y} = \frac{x+1}{x-1}$$

and (12.29) follows.

Exercise 12.23 (Krylov space and cg iterations)

Consider the linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ where

$$\mathbf{A} = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}, \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} 4 \\ 0 \\ 0 \end{bmatrix}.$$

a) Determine the vectors defining the Krylov spaces for $k \leq 3$ taking as initial approximation $\mathbf{x} = \mathbf{0}$. Answer: $[\mathbf{b}, \mathbf{Ab}, \mathbf{A}^2\mathbf{b}] = \begin{bmatrix} 4 & 8 & 20 \\ 0 & -4 & -16 \\ 0 & 0 & 4 \end{bmatrix}$.

b) Carry out three CG-iterations on $\mathbf{Ax} = \mathbf{b}$. Answer:

$$[\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3] = \begin{bmatrix} 0 & 2 & 8/3 & 3 \\ 0 & 0 & 4/3 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$[\mathbf{r}_0, \mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3] = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 4/3 & 0 \end{bmatrix},$$

$$[\mathbf{Ap}_0, \mathbf{Ap}_1, \mathbf{Ap}_2] = \begin{bmatrix} 8 & 0 & 0 \\ -4 & 3 & 0 \\ 0 & -2 & 16/9 \end{bmatrix},$$

$$[\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3] = \begin{bmatrix} 4 & 1 & 4/9 & 0 \\ 0 & 2 & 8/9 & 0 \\ 0 & 0 & 12/9 & 0 \end{bmatrix},$$

c) Verify that

- $\dim(\mathbb{W}_k) = k$ for $k = 0, 1, 2, 3$.
- \mathbf{x}_3 is the exact solution of $\mathbf{Ax} = \mathbf{b}$.
- $\mathbf{r}_0, \dots, \mathbf{r}_{k-1}$ is an orthogonal basis for \mathbb{W}_k for $k = 1, 2, 3$.
- $\mathbf{p}_0, \dots, \mathbf{p}_{k-1}$ is an \mathbf{A} -orthogonal basis for \mathbb{W}_k for $k = 1, 2, 3$.
- $\{|r_k \epsilon|\}$ is monotonically decreasing.
- $\{|\mathbf{x}_k - \mathbf{x}\epsilon|\}$ is monotonically decreasing.

12.4 Proof of the Convergence Estimates

12.4.1 Chebyshev polynomials

The proof of the estimate (12.28) for the error in the conjugate gradient method is based on an extremal property of the Chebyshev polynomials. Suppose $a < b$, $c \notin [a, b]$ and $k \in \mathbb{N}$. Consider the set \mathcal{S}_k of all polynomials Q of degree $\leq k$ such that $Q(c) = 1$. For any continuous function f on $[a, b]$ we define

$$\|f\|_\infty = \max_{a \leq x \leq b} |f(x)|.$$

We want to find a polynomial $Q^* \in \mathcal{S}_k$ such that

$$\|Q^*\|_\infty = \min_{Q \in \mathcal{S}_k} \|Q\|_\infty.$$

We will show that Q^* is uniquely given as a suitably shifted and normalized version of the **Chebyshev polynomial**. The Chebyshev polynomial T_n of degree n can be defined recursively by

$$T_{n+1}(t) = 2tT_n(t) - T_{n-1}(t), \quad n \geq 1, \quad t \in \mathbb{R},$$

starting with $T_0(t) = 1$ and $T_1(t) = t$. Thus $T_2(t) = 2t^2 - 1$, $T_3(t) = 4t^3 - 3t$ etc. In general T_n is a polynomial of degree n .

There are some convenient closed form expressions for T_n .

Lemma 12.24 (Closed forms of Chebyshev polynomials)

For $n \geq 0$

1. $T_n(t) = \cos(n \arccos t)$ for $t \in [-1, 1]$,
2. $T_n(t) = \frac{1}{2}[(t + \sqrt{t^2 - 1})^n + (t - \sqrt{t^2 - 1})^{-n}]$ for $|t| \geq 1$.

Proof. 1. With $P_n(t) = \cos(n \arccos t)$ we have $P_n(t) = \cos n\phi$, where $t = \cos \phi$. Therefore,

$$P_{n+1}(t) + P_{n-1}(t) = \cos((n+1)\phi) + \cos((n-1)\phi) = 2\cos\phi \cos n\phi = 2tP_n(t),$$

and it follows that P_n satisfies the same recurrence relation as T_n . Since $P_0 = T_0$ and $P_1 = T_1$ we have $P_n = T_n$ for all $n \geq 0$.

2. Fix t with $|t| \geq 1$ and let $x_n := T_n(t)$ for $n \geq 0$. The recurrence relation for the Chebyshev polynomials can then be written

$$x_{n+1} - 2tx_n + x_{n-1} = 0 \text{ for } n \geq 1, \text{ with } x_0 = 1, x_1 = t. \quad (12.30)$$

To solve this difference equation we insert $x_n = z^n$ into (12.30) and obtain $z^{n+1} - 2tz^n + z^{n-1} = 0$ or $z^2 - 2tz + 1 = 0$. The roots of this equation are

$$z_1 = t + \sqrt{t^2 - 1}, \quad z_2 = t - \sqrt{t^2 - 1} = (t + \sqrt{t^2 - 1})^{-1}.$$

Now z_1^n , z_2^n and more generally $c_1 z_1^n + c_2 z_2^n$ are solutions of (12.30) for any constants c_1 and c_2 . We find these constants from the initial conditions $x_0 = c_1 + c_2 = 1$ and $x_1 = c_1 z_1 + c_2 z_2 = t$. Since $z_1 + z_2 = 2t$ the solution is $c_1 = c_2 = \frac{1}{2}$. \square

We show that the unique solution to our minimization problem is

$$Q^*(x) = \frac{T_k(u(x))}{T_k(u(c))}, \quad u(x) = \frac{b+a-2x}{b-a}. \quad (12.31)$$

Clearly $Q^* \in \mathcal{S}_k$.

Theorem 12.25 (A minimal norm problem)

Suppose $a < b$, $c \notin [a, b]$ and $k \in \mathbb{N}$. If $Q \in S_k$ and $Q \neq Q^*$ then $\|Q\|_\infty > \|Q^*\|_\infty$.

Proof. Recall that a nonzero polynomial p of degree k can have at most k zeros. If $p(z) = p'(z) = 0$, we say that p has a double zero at z . Counting such a zero as two zeros it is still true that a nonzero polynomial of degree k has at most k zeros.

$|Q^*|$ takes on its maximum $1/|T_k(u(c))|$ at the $k + 1$ points μ_0, \dots, μ_k in $[a, b]$ such that $u(\mu_i) = \cos(i\pi/k)$ for $i = 0, 1, \dots, k$. Suppose $Q \in S_k$ and that $\|Q\|_\infty \leq \|Q^*\|_\infty$. We have to show that $Q \equiv Q^*$. Let $f \equiv Q - Q^*$. We show that f has at least k zeros in $[a, b]$. Since f is a polynomial of degree $\leq k$ and $f(c) = 0$, this means that $f \equiv 0$ or equivalently $Q \equiv Q^*$.

Consider $I_j = [\mu_{j-1}, \mu_j]$ for a fixed j . Let

$$\sigma_j = f(\mu_{j-1})f(\mu_j).$$

We have $\sigma_j \leq 0$. For if say $Q^*(\mu_j) > 0$ then

$$Q(\mu_j) \leq \|Q\|_\infty \leq \|Q^*\|_\infty = Q^*(\mu_j)$$

so that $f(\mu_j) \leq 0$. Moreover,

$$-Q(\mu_{j-1}) \leq \|Q\|_\infty \leq \|Q^*\|_\infty = -Q^*(\mu_{j-1}).$$

Thus $f(\mu_{j-1}) \geq 0$ and it follows that $\sigma_j \leq 0$. Similarly, $\sigma_j \leq 0$ if $Q^*(\mu_j) < 0$.

If $\sigma_j < 0$, f must have a zero in I_j since it is continuous. Suppose $\sigma_j = 0$. Then $f(\mu_{j-1}) = 0$ or $f(\mu_j) = 0$. If $f(\mu_j) = 0$ then $Q(\mu_j) = Q^*(\mu_j)$. But then μ_j is a maximum or minimum both for Q and Q^* . If $\mu_j \in (a, b)$ then $Q'(\mu_j) = Q^{*\prime}(\mu_j) = 0$. Thus $f(\mu_j) = f'(\mu_j) = 0$, and f has a double zero at μ_j . We can count this as one zero for I_j and one for I_{j+1} . If $\mu_j = b$, we still have a zero in I_j . Similarly, if $f(\mu_{j-1}) = 0$, a double zero of f at μ_{j-1} appears if $\mu_{j-1} \in (a, b)$. We count this as one zero for I_{j-1} and one for I_j .

In this way we associate one zero of f for each of the k intervals I_j , $j = 1, 2, \dots, k$. We conclude that f has at least k zeros in $[a, b]$. \square

Exercise 12.26 (Another explicit formula for the Chebyshev polynomial)

Show that

$$T_n(t) = \cosh(n \operatorname{arccosh} t) \text{ for } |t| \geq 1,$$

where $\operatorname{arccosh}$ is the inverse function of $\cosh x := (e^x + e^{-x})/2$.

Theorem 12.25 with a , and b , the smallest and largest eigenvalue of \mathbf{A} , and $c = 0$ implies that the minimizing polynomial in (12.28) is given by

$$Q^*(x) = T_k \left(\frac{b+a-2x}{b-a} \right) / T_k \left(\frac{b+a}{b-a} \right). \quad (12.32)$$

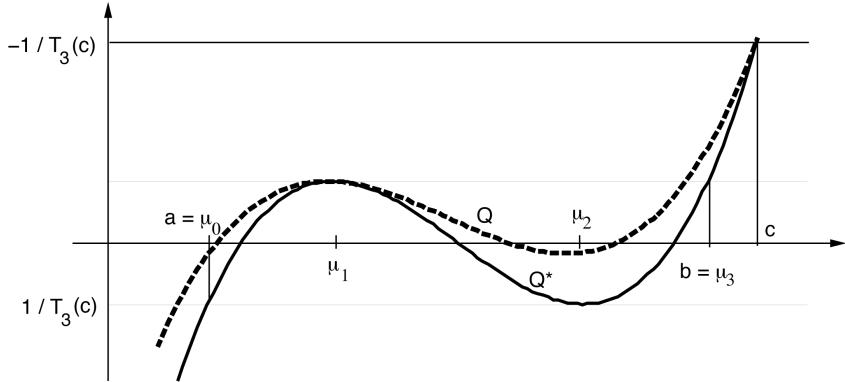


Figure 12.3: This is an illustration of the proof of Theorem 12.25 for $k = 3$. $f \equiv Q - Q^*$ has a double zero at μ_1 and one zero between μ_2 and μ_3 .

By Lemma 12.24

$$\max_{a \leq x \leq b} \left| T_k \left(\frac{b+a-2x}{b-a} \right) \right| = \max_{-1 \leq t \leq 1} |T_k(t)| = 1. \quad (12.33)$$

Moreover with $t = (b+a)/(b-a)$ we have

$$t + \sqrt{t^2 - 1} = \frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1}, \quad \kappa = b/a.$$

Thus again by Lemma 12.24 we find

$$T_k \left(\frac{b+a}{b-a} \right) = T_k \left(\frac{\kappa+1}{\kappa-1} \right) = \frac{1}{2} \left[\left(\frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1} \right)^k + \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k \right] \quad (12.34)$$

and (12.28) follows.

12.4.2 Convergence proof for steepest descent

For the proof of (12.18) the following inequality will be used.

Theorem 12.27 (Kantorovich inequality)

For any symmetric positive definite matrix $A \in \mathbb{R}^{n \times n}$

$$1 \leq \frac{(\mathbf{y}^T A \mathbf{y})(\mathbf{y}^T A^{-1} \mathbf{y})}{(\mathbf{y}^T \mathbf{y})^2} \leq \frac{(M+m)^2}{4Mm} \quad \mathbf{y} \neq \mathbf{0}, \quad \mathbf{y} \in \mathbb{R}^n, \quad (12.35)$$

where $M := \lambda_{\max}$ and $m := \lambda_{\min}$ are the largest and smallest eigenvalue of A , respectively.

Proof. If $(\lambda_j, \mathbf{u}_j)$ are orthonormal eigenpairs of \mathbf{A} then $(\lambda_j^{-1}, \mathbf{u}_j)$ are eigenpairs for \mathbf{A}^{-1} , $j = 1, \dots, n$. Let $\mathbf{y} = \sum_{j=1}^n c_j \mathbf{u}_j$ be the corresponding eigenvector expansion of a vector $\mathbf{y} \in \mathbb{R}^n$. By orthonormality, (cf. (5.15))

$$a := \frac{\mathbf{y}^T \mathbf{A} \mathbf{y}}{\mathbf{y}^T \mathbf{y}} = \sum_{i=1}^n t_i \lambda_i, \quad b := \frac{\mathbf{y}^T \mathbf{A}^{-1} \mathbf{y}}{\mathbf{y}^T \mathbf{y}} = \sum_{i=1}^n \frac{t_i}{\lambda_i}, \quad (12.36)$$

where

$$t_i = \frac{c_i^2}{\sum_{j=1}^n c_j^2} \geq 0, \quad i = 1, \dots, n \text{ and } \sum_{i=1}^n t_i = 1. \quad (12.37)$$

Thus a and b are **convex combinations** of the eigenvalues of \mathbf{A} and \mathbf{A}^{-1} , respectively. Let c be a positive constant to be chosen later. By the geometric/arithmetic mean inequality (7.34) and (12.36)

$$\sqrt{ab} = \sqrt{(ac)(b/c)} \leq (ac + b/c)/2 = \frac{1}{2} \sum_{i=1}^n t_i (\lambda_i c + 1/(\lambda_i c)) = \frac{1}{2} \sum_{i=1}^n t_i f(\lambda_i c),$$

where $f : [mc, Mc] \rightarrow \mathbb{R}$ is given by $f(x) := x + 1/x$. By (12.37)

$$\sqrt{ab} \leq \frac{1}{2} \max_{mc \leq x \leq Mc} f(x).$$

Since $f \in C^2$ and f'' is positive it follows from Lemma 7.39 that f is a convex function. But a convex function takes its maximum at one of the endpoints of the range (cf. Exercise 12.28) and we obtain

$$\sqrt{ab} \leq \frac{1}{2} \max\{f(mc), f(Mc)\}. \quad (12.38)$$

Choosing $c := 1/\sqrt{mM}$ we find $f(mc) = f(Mc) = \sqrt{\frac{M}{m}} + \sqrt{\frac{m}{M}} = \frac{M+m}{\sqrt{mM}}$. By (12.38) we obtain

$$\frac{(\mathbf{y}^T \mathbf{A} \mathbf{y})(\mathbf{y}^T \mathbf{A}^{-1} \mathbf{y})}{(\mathbf{y}^T \mathbf{y})^2} = ab \leq \frac{(M+m)^2}{4Mm},$$

the upper bound in (12.35). For the lower bound we use the Cauchy-Schwarz inequality as follows

$$1 = \left(\sum_{i=1}^n t_i \right)^2 = \left(\sum_{i=1}^n (t_i \lambda_i)^{1/2} (t_i / \lambda_i)^{1/2} \right)^2 \leq \left(\sum_{i=1}^n t_i \lambda_i \right) \left(\sum_{i=1}^n t_i / \lambda_i \right) = ab.$$

□

Exercise 12.28 (Maximum of a convex function)

Show that if $f : [a, b] \rightarrow \mathbb{R}$ is convex then $\max_{a \leq x \leq b} f(x) \leq \max\{f(a), f(b)\}$.

Proof of (12.18)

Let $\epsilon_j := \mathbf{x} - \mathbf{x}_j$, $j = 0, 1, \dots$, where $A\mathbf{x} = \mathbf{b}$. It is enough to show that

$$\frac{\|\epsilon_{k+1}\|_A^2}{\|\epsilon_k\|_A^2} \leq \left(\frac{\kappa-1}{\kappa+1}\right)^2, \quad k = 0, 1, 2, \dots, \quad (12.39)$$

for then $\|\epsilon_k\|_A \leq \left(\frac{\kappa-1}{\kappa+1}\right) \|\epsilon_{k-1}\| \leq \dots \leq \left(\frac{\kappa-1}{\kappa+1}\right)^k \|\epsilon_0\|$. It follows from (12.7) that

$$\epsilon_{k+1} = \epsilon_k - \alpha_k \mathbf{r}_k, \quad \alpha_k := \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{r}_k^T A \mathbf{r}_k}.$$

We find

$$\begin{aligned} \|\epsilon_k\|_A^2 &= \epsilon_k^T A \epsilon_k = \mathbf{r}_k^T A^{-1} \mathbf{r}_k, \\ \|\epsilon_{k+1}\|_A^2 &= (\epsilon_k - \alpha_k \mathbf{r}_k)^T A (\epsilon_k - \alpha_k \mathbf{r}_k) \\ &= \epsilon_k^T A \epsilon_k - 2\alpha_k \mathbf{r}_k^T A \epsilon_k + \alpha_k^2 \mathbf{r}_k^T A \mathbf{r}_k = \|\epsilon_k\|_A^2 - \frac{(\mathbf{r}_k^T \mathbf{r}_k)^2}{\mathbf{r}_k^T A \mathbf{r}_k}. \end{aligned}$$

Combining these and using Kantorovich inequality

$$\frac{\|\epsilon_{k+1}\|_A^2}{\|\epsilon_k\|_A^2} = 1 - \frac{(\mathbf{r}_k^T \mathbf{r}_k)^2}{(\mathbf{r}_k^T A \mathbf{r}_k)(\mathbf{r}_k^T A^{-1} \mathbf{r}_k)} \leq 1 - \frac{4\lambda_{min}\lambda_{max}}{(\lambda_{min} + \lambda_{max})^2} = \left(\frac{\kappa-1}{\kappa+1}\right)^2$$

and (12.39) is proved.

□

12.4.3 Monotonicity of the error

The error analysis for the conjugate gradient method is based on the A -norm. We end this chapter by considering the Euclidian norm of the error, and show that it is strictly decreasing.

Theorem 12.29 (The error in cg is strictly decreasing)

Let in the conjugate gradient method m be the smallest integer such that $\mathbf{r}_{m+1} = \mathbf{0}$. For $k \leq m$ we have $\|\epsilon_{k+1}\|_2 < \|\epsilon_k\|_2$. More precisely,

$$\|\epsilon_k\|_2^2 - \|\epsilon_{k+1}\|_2^2 = \frac{\|\mathbf{p}_k\|_2^2}{\|\mathbf{p}_k\|_A^2} (\|\epsilon_k\|_A^2 + \|\epsilon_{k+1}\|_A^2)$$

where $\epsilon_j = \mathbf{x} - \mathbf{x}_j$ and $A\mathbf{x} = b$.

Proof. For $j \leq m$

$$\epsilon_j = \mathbf{x}_{m+1} - \mathbf{x}_j = \mathbf{x}_m - \mathbf{x}_j + \alpha_m \mathbf{p}_m = \mathbf{x}_{m-1} - \mathbf{x}_j + \alpha_{m-1} \mathbf{p}_{m-1} + \alpha_m \mathbf{p}_m = \dots$$

so that

$$\epsilon_j = \sum_{i=j}^m \alpha_i \mathbf{p}_i, \quad \alpha_i = \frac{\mathbf{r}_i^T \mathbf{r}_i}{\mathbf{p}_i^T \mathbf{A} \mathbf{p}_i}. \quad (12.40)$$

By (12.40) and \mathbf{A} -orthogonality

$$\|\epsilon_j\|_{\mathbf{A}}^2 = \epsilon_j^T \mathbf{A} \epsilon_j = \sum_{i=j}^m \alpha_i^2 \mathbf{p}_i^T \mathbf{A} \mathbf{p}_i = \sum_{i=j}^m \frac{(\mathbf{r}_i^T \mathbf{r}_i)^2}{\mathbf{p}_i^T \mathbf{A} \mathbf{p}_i}. \quad (12.41)$$

By (12.16) and Lemma 12.19

$$\mathbf{p}_i^T \mathbf{p}_k = (\mathbf{r}_i + \beta_{i-1} \mathbf{p}_{i-1})^T \mathbf{p}_k = \beta_{i-1} \mathbf{p}_{i-1}^T \mathbf{p}_k = \dots = \beta_{i-1} \cdots \beta_k (\mathbf{p}_k^T \mathbf{p}_k),$$

and since $\beta_{i-1} \cdots \beta_k = (\mathbf{r}_i^T \mathbf{r}_i) / (\mathbf{r}_k^T \mathbf{r}_k)$ we obtain

$$\mathbf{p}_i^T \mathbf{p}_k = \frac{\mathbf{r}_i^T \mathbf{r}_i}{\mathbf{r}_k^T \mathbf{r}_k} \mathbf{p}_k^T \mathbf{p}_k, \quad i \geq k. \quad (12.42)$$

Since

$$\|\epsilon_k\|_2^2 = \|\epsilon_{k+1} + \mathbf{x}_{k+1} - \mathbf{x}_k\|_2^2 = \|\epsilon_{k+1} + \alpha_k \mathbf{p}_k\|_2^2,$$

we obtain

$$\begin{aligned} \|\epsilon_k\|_2^2 - \|\epsilon_{k+1}\|_2^2 &= \alpha_k (2 \mathbf{p}_k^T \epsilon_{k+1} + \alpha_k \mathbf{p}_k^T \mathbf{p}_k) \\ &\stackrel{(12.40)}{=} \alpha_k \left(2 \sum_{i=k+1}^m \alpha_i \mathbf{p}_i^T \mathbf{p}_k + \alpha_k \mathbf{p}_k^T \mathbf{p}_k \right) = \left(\sum_{i=k}^m + \sum_{i=k+1}^m \right) \alpha_k \alpha_i \mathbf{p}_i^T \mathbf{p}_k \\ &\stackrel{(12.42)}{=} \left(\sum_{i=k}^m + \sum_{i=k+1}^m \right) \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k} \frac{\mathbf{r}_i^T \mathbf{r}_i}{\mathbf{p}_i^T \mathbf{A} \mathbf{p}_i} \frac{\mathbf{r}_i^T \mathbf{r}_i}{\mathbf{r}_k^T \mathbf{r}_k} \mathbf{p}_k^T \mathbf{p}_k \\ &\stackrel{(12.41)}{=} \frac{\|\mathbf{p}_k\|_2^2}{\|\mathbf{p}_k\|_{\mathbf{A}}^2} (\|\epsilon_k\|_{\mathbf{A}}^2 + \|\epsilon_{k+1}\|_{\mathbf{A}}^2). \end{aligned}$$

and the Theorem is proved. \square

12.5 Preconditioning

For problems $\mathbf{A}\mathbf{x} = \mathbf{b}$ of size n , where both n and $\text{cond}_2(\mathbf{A})$ are large, it is often possible to improve the performance of the conjugate gradient method by using a

technique known as **preconditioning**. Instead of $\mathbf{Ax} = \mathbf{b}$ we consider an equivalent system $\mathbf{B}\mathbf{Ax} = \mathbf{B}\mathbf{b}$, where \mathbf{B} is nonsingular and $\text{cond}_2(\mathbf{BA})$ is smaller than $\text{cond}_2(\mathbf{A})$. The matrix \mathbf{B} will in many cases be the inverse of another matrix, $\mathbf{B} = \mathbf{M}^{-1}$. We cannot use CG on $\mathbf{B}\mathbf{Ax} = \mathbf{B}\mathbf{b}$ directly since \mathbf{BA} in general is not symmetric even if both \mathbf{A} and \mathbf{B} are. But if \mathbf{B} (and hence \mathbf{M}) is symmetric positive definite then we can apply CG to a symmetrized system and then transform the recurrence formulas to an iterative method for the original system $\mathbf{Ax} = \mathbf{b}$. This iterative method is known as the **preconditioned conjugate gradient method**. We shall see that the convergence properties of this method is determined by the eigenvalues of \mathbf{BA} .

Suppose \mathbf{B} is symmetric positive definite. By Theorem 3.18 there is a non-singular matrix \mathbf{C} such that $\mathbf{B} = \mathbf{C}^T\mathbf{C}$. (\mathbf{C} is only needed for the derivation and will not appear in the final formulas). Now

$$\mathbf{B}\mathbf{Ax} = \mathbf{B}\mathbf{b} \Leftrightarrow \mathbf{C}^T(\mathbf{CAC}^T)\mathbf{C}^{-T}\mathbf{x} = \mathbf{C}^T\mathbf{Cb} \Leftrightarrow (\mathbf{CAC}^T)\mathbf{y} = \mathbf{Cb}, \quad \& \quad \mathbf{x} = \mathbf{C}^T\mathbf{y}.$$

We have 3 linear systems

$$\mathbf{Ax} = \mathbf{b} \tag{12.43}$$

$$\mathbf{B}\mathbf{Ax} = \mathbf{B}\mathbf{b} \tag{12.44}$$

$$(\mathbf{CAC}^T)\mathbf{y} = \mathbf{Cb}, \quad \& \quad \mathbf{x} = \mathbf{C}^T\mathbf{y}. \tag{12.45}$$

Note that (12.43) and (12.45) are symmetric positive definite linear systems. In addition to being symmetric positive definite the matrix \mathbf{CAC}^T is similar to \mathbf{BA} . Indeed,

$$\mathbf{C}^T(\mathbf{CAC}^T)\mathbf{C}^{-T} = \mathbf{BA}.$$

Thus \mathbf{CAC}^T and \mathbf{BA} have the same eigenvalues. Therefore, if we apply the conjugate gradient method to (12.45) then the rate of convergence will be determined by the eigenvalues of \mathbf{BA} .

We apply the conjugate gradient method to $(\mathbf{CAC}^T)\mathbf{y} = \mathbf{Cb}$. Denoting the search direction by \mathbf{q}_k and the residual by $\mathbf{z}_k = \mathbf{Cb} - \mathbf{CAC}^T\mathbf{y}_k$ we obtain the following from (12.14), (12.15), and (12.16).

$$\begin{aligned} \mathbf{y}_{k+1} &= \mathbf{y}_k + \alpha_k \mathbf{q}_k, & \alpha_k &= \mathbf{z}_k^T \mathbf{z}_k / \mathbf{q}_k^T (\mathbf{CAC}^T) \mathbf{q}_k, \\ \mathbf{z}_{k+1} &= \mathbf{z}_k - \alpha_k (\mathbf{CAC}^T) \mathbf{q}_k, \\ \mathbf{q}_{k+1} &= \mathbf{z}_{k+1} + \beta_k \mathbf{q}_k, & \beta_k &= \mathbf{z}_{k+1}^T \mathbf{z}_{k+1} / \mathbf{z}_k^T \mathbf{z}_k. \end{aligned}$$

With

$$\mathbf{x}_k := \mathbf{C}^T \mathbf{y}_k, \quad \mathbf{p}_k := \mathbf{C}^T \mathbf{q}_k, \quad \mathbf{s}_k := \mathbf{C}^T \mathbf{z}_k, \quad \mathbf{r}_k := \mathbf{C}^{-1} \mathbf{z}_k \tag{12.46}$$

this can be transformed into

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k, \quad \alpha_k = \frac{\mathbf{s}_k^T \mathbf{r}_k}{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k}, \quad (12.47)$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A} \mathbf{p}_k, \quad (12.48)$$

$$\mathbf{s}_{k+1} = \mathbf{s}_k - \alpha_k \mathbf{B} \mathbf{A} \mathbf{p}_k, \quad (12.49)$$

$$\mathbf{p}_{k+1} = \mathbf{s}_{k+1} + \beta_k \mathbf{p}_k, \quad \beta_k = \frac{\mathbf{s}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{s}_k^T \mathbf{r}_k}. \quad (12.50)$$

Here \mathbf{x}_k will be an approximation to the solution \mathbf{x} of $\mathbf{A}\mathbf{x} = \mathbf{b}$, $\mathbf{r}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k$ is the residual in the original system, and $\mathbf{s}_k = \mathbf{B}\mathbf{b} - \mathbf{B}\mathbf{A}\mathbf{x}_k$ is the residual in the preconditioned system. This follows since by (12.46)

$$\mathbf{r}_k = \mathbf{C}^{-1} \mathbf{z}_k = \mathbf{b} - \mathbf{C}^{-1} \mathbf{C} \mathbf{A} \mathbf{C}^T \mathbf{y}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k$$

and $\mathbf{s}_k = \mathbf{C}^T \mathbf{z}_k = \mathbf{C}^T \mathbf{C} \mathbf{r}_k = \mathbf{B} \mathbf{r}_k$. We start with $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$, $\mathbf{p}_0 = \mathbf{s}_0 = \mathbf{B}\mathbf{r}_0$ and obtain the following preconditioned conjugate gradient algorithm for determining approximations \mathbf{x}_k to the solution of a symmetric positive definite system $\mathbf{A}\mathbf{x} = \mathbf{b}$.

Algorithm 12.30 (Preconditioned conjugate gradient)

The symmetric positive definite linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ is solved by the preconditioned conjugate gradient method on the system $\mathbf{B}\mathbf{A}\mathbf{x} = \mathbf{B}\mathbf{b}$, where \mathbf{B} is symmetric positive definite . \mathbf{x} is a starting vector for the iteration. The iteration is stopped when $\|\mathbf{r}_k\|_2/\|\mathbf{b}\|_2 \leq \text{tol}$ or $k > \text{itmax}$. K is the number of iterations used.

```

1 function [x,K]=pcg(A,B,b,x,tol,itmax)
2 r=b-A*x; p=B*r; s=p; rho=s'*r; rho0=b'*b;
3 for k=0:itmax
4   if sqrt(rho/rho0)<= tol
5     K=k; return
6   end
7   t=A*p; a=rho/(p'*t);
8   x=x+a*p; r=r-a*t;
9   w=B*t; s=s-a*w;
10  rhos=rho; rho=s'*r;
11  p=r+(rho/rhos)*p;
12 end
13 K=itmax+1;

```

Apart from the calculation of ρ this algorithm is quite similar to Algorithm 12.12. The main additional work is contained in $w = B * t$. We'll discuss this further in connection with an example. There the inverse of B is known and we have to solve a linear system to find w .

We have the following convergence result for this algorithm.

Theorem 12.31 (Error bound preconditioned cg)

Suppose we apply a symmetric positive definite preconditioner \mathbf{B} to the symmetric positive definite system $\mathbf{A}\mathbf{x} = \mathbf{b}$. Then the quantities \mathbf{x}_k computed in Algorithm 12.30 satisfy the following bound:

$$\frac{\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}}{\|\mathbf{x} - \mathbf{x}_0\|_{\mathbf{A}}} \leq 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k, \quad \text{for } k \geq 0,$$

where $\kappa = \lambda_{\max}/\lambda_{\min}$ is the ratio of the largest and smallest eigenvalue of \mathbf{BA} .

Proof. Since Algorithm 12.30 is equivalent to solving (12.45) by the conjugate gradient method Theorem 12.16 implies that

$$\frac{\|\mathbf{y} - \mathbf{y}_k\|_{\mathbf{CAC}^T}}{\|\mathbf{y} - \mathbf{y}_0\|_{\mathbf{CAC}^T}} \leq 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k, \quad \text{for } k \geq 0,$$

where \mathbf{y}_k is the conjugate gradient approximation to the solution \mathbf{y} of (12.45) and κ is the ratio of the largest and smallest eigenvalue of \mathbf{CAC}^T . Since \mathbf{BA} and \mathbf{CAC}^T are similar this is the same as the κ in the theorem. By (12.46) we have

$$\begin{aligned} \|\mathbf{y} - \mathbf{y}_k\|_{\mathbf{CAC}^T}^2 &= (\mathbf{y} - \mathbf{y}_k)^T (\mathbf{CAC}^T) (\mathbf{y} - \mathbf{y}_k) \\ &= (\mathbf{C}^T (\mathbf{y} - \mathbf{y}_k))^T \mathbf{A} (\mathbf{C}^T (\mathbf{y} - \mathbf{y}_k)) = \|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}^2 \end{aligned}$$

and the proof is complete. \square

We conclude that \mathbf{B} should satisfy the following requirements for a problem of size n :

1. The eigenvalues of \mathbf{BA} should be located in a narrow interval. Preferably one should be able to bound the length of the interval independently of n .
2. The evaluation of \mathbf{Bx} for a given vector \mathbf{x} should not be expensive in storage and arithmetic operations, ideally $O(n)$ for both.

In this book we only consider an example of a preconditioner. For a comprehensive treatment of preconditioners see [1].

12.6 Preconditioning Example

12.6.1 A variable coefficient problem

Consider the problem

$$\begin{aligned} -\frac{\partial}{\partial x} \left(c(x, y) \frac{\partial u}{\partial x} \right) - \frac{\partial}{\partial y} \left(c(x, y) \frac{\partial u}{\partial y} \right) &= f(x, y) & (x, y) \in \Omega = (0, 1)^2 \\ u(x, y) &= 0 & (x, y) \in \partial\Omega. \end{aligned} \tag{12.51}$$

Here Ω is the open unit square while $\partial\Omega$ is the boundary of Ω . The functions f and c are given and we seek a function $u = u(x, y)$ such that (12.51) holds. We assume that c and f are defined and continuous on Ω and that $c(x, y) > 0$ for all $(x, y) \in \Omega$. The problem (12.51) reduces to the Poisson problem in the special case where $c(x, y) = 1$ for $(x, y) \in \Omega$.

As for the Poisson problem we solve (12.51) numerically on a grid of points

$$\{(jh, kh) : j, k = 0, 1, \dots, m+1\}, \quad \text{where } h = 1/(m+1),$$

and where m is a positive integer. Let (x, y) be one of the interior grid points. For univariate functions f, g we use the central difference approximations

$$\begin{aligned} \frac{d}{dt} \left(f(t) \frac{d}{dt} g(t) \right) &\approx \left(f(t + \frac{h}{2}) \frac{d}{dt} g(t + h/2) - f(t - \frac{h}{2}) \frac{d}{dt} g(t - h/2) \right) / h \\ &\approx \left(f(t + \frac{h}{2})(g(t+h) - g(t)) - f(t - \frac{h}{2})(g(t) - g(t-h)) \right) / h^2 \end{aligned}$$

to obtain

$$\frac{\partial}{\partial x} \left(c \frac{\partial u}{\partial x} \right)_{j,k} \approx \frac{c_{j+\frac{1}{2},k}(v_{j+1,k} - v_{j,k}) - c_{j-\frac{1}{2},k}(v_{j,k} - v_{j-1,k})}{h^2}$$

and

$$\frac{\partial}{\partial y} \left(c \frac{\partial u}{\partial y} \right)_{j,k} \approx \frac{c_{j,k+\frac{1}{2}}(v_{j,k+1} - v_{j,k}) - c_{j,k-\frac{1}{2}}(v_{j,k} - v_{j,k-1})}{h^2},$$

where $c_{p,q} = c(ph, qh)$ and $v_{j,k} \approx u(jh, kh)$. With these approximations the discrete analog of (12.51) turns out to be

$$\begin{aligned} -(\mathbf{P}_h v)_{j,k} &= h^2 f_{j,k} \quad j, k = 1, \dots, m \\ v_{j,k} &= 0 \quad j = 0, m+1 \text{ all } k \text{ or } k = 0, m+1 \text{ all } j, \end{aligned} \tag{12.52}$$

where

$$\begin{aligned} -(\mathbf{P}_h v)_{j,k} &= (c_{j,k-\frac{1}{2}} + c_{j-\frac{1}{2},k} + c_{j+\frac{1}{2},k} + c_{j,k+\frac{1}{2}})v_{j,k} \\ &\quad - c_{j,k-\frac{1}{2}}v_{j,k-1} - c_{j-\frac{1}{2},k}v_{j-1,k} - c_{j+\frac{1}{2},k}v_{j+1,k} - c_{j,k+\frac{1}{2}}v_{j,k+1} \end{aligned} \tag{12.53}$$

and $f_{j,k} = f(jh, kh)$.

As before we let $\mathbf{V} = (v_{j,k}) \in \mathbb{R}^{m \times m}$ and $\mathbf{F} = (f_{j,k}) \in \mathbb{R}^{m \times m}$. The corresponding linear system can be written $\mathbf{A}\mathbf{x} = \mathbf{b}$ where $\mathbf{x} = \text{vec}(\mathbf{V})$, $\mathbf{b} = h^2 \text{vec}(\mathbf{F})$, and the n -by- n coefficient matrix \mathbf{A} is given by

$$\begin{aligned} a_{i,i} &= c_{j_i, k_i - \frac{1}{2}} + c_{j_i - \frac{1}{2}, k_i} + c_{j_i + \frac{1}{2}, k_i} + c_{j_i, k_i + \frac{1}{2}}, & i = 1, 2, \dots, n \\ a_{i+1,i} &= a_{i,i+1} = -c_{j_i + \frac{1}{2}, k_i}, & i \bmod m \neq 0 \\ a_{i+m,i} &= a_{i,i+m} = -c_{j_i, k_i + \frac{1}{2}}, & i = 1, 2, \dots, n-m \\ a_{i,j} &= 0 & \text{otherwise,} \end{aligned} \tag{12.54}$$

where (j_i, k_i) with $1 \leq j_i, k_i \leq m$ is determined uniquely from the equation $i = j_i + (k_i - 1)m$ for $i = 1, \dots, n$. If $c(x, y) = 1$ for all $(x, y) \in \Omega$ we recover the Poisson matrix.

In general we cannot write \mathbf{A} as a Kronecker sum. But we can show that \mathbf{A} is symmetric and it is positive definite as long as the function c is positive on Ω .

Theorem 12.32 (Positive definite matrix)

If $c(x, y) > 0$ for $(x, y) \in \Omega$ then the matrix \mathbf{A} given by (12.54) is symmetric positive definite.

Proof.

To each $x \in \mathbb{R}^n$ there corresponds a matrix $\mathbf{V} \in \mathbb{R}^{m \times m}$ such that $x = \text{vec}(\mathbf{V})$. We claim that

$$\mathbf{x}^T \mathbf{A} \mathbf{x} = \sum_{j=1}^m \sum_{k=0}^m c_{j,k+\frac{1}{2}} (v_{j,k+1} - v_{j,k})^2 + \sum_{k=1}^m \sum_{j=0}^m c_{j+\frac{1}{2},k} (v_{j+1,k} - v_{j,k})^2, \quad (12.55)$$

where $v_{0,k} = v_{m+1,k} = v_{j,0} = v_{j,m+1} = 0$ for $j, k = 0, 1, \dots, m+1$. Since $c_{j+\frac{1}{2},k}$ and $c_{j,k+\frac{1}{2}}$ correspond to values of c in Ω for the values of j, k in the sums it follows that they are positive and from (12.55) we see that $\mathbf{x}^T \mathbf{A} \mathbf{x} \geq 0$ for all $x \in \mathbb{R}^n$. Moreover if $\mathbf{x}^T \mathbf{A} \mathbf{x} = 0$ then all quadratic factors are zero and $v_{j,k+1} = v_{j,k}$ for $k = 0, 1, \dots, m$ and $j = 1, \dots, m$. Now $v_{j,0} = v_{j,m+1} = 0$ implies that $\mathbf{V} = \mathbf{0}$ and hence $x = 0$. Thus \mathbf{A} is symmetric positive definite.

It remains to prove (12.55). From the connection between (12.53) and (12.54) we have

$$\begin{aligned} \mathbf{x}^T \mathbf{A} \mathbf{x} &= \sum_{j=1}^m \sum_{k=1}^m -(P_h v)_{j,k} v_{j,k} \\ &= \sum_{j=1}^m \sum_{k=1}^m \left(c_{j,k-\frac{1}{2}} v_{j,k}^2 + c_{j-\frac{1}{2},k} v_{j,k}^2 + c_{j+\frac{1}{2},k} v_{j,k}^2 + c_{j,k+\frac{1}{2}} v_{j,k}^2 \right. \\ &\quad \left. - c_{j,k-\frac{1}{2}} v_{j,k-1} v_{j,k} - c_{j,k+\frac{1}{2}} v_{j,k} v_{j,k+1} \right. \\ &\quad \left. - c_{j-\frac{1}{2},k} v_{j-1,k} v_{j,k} - c_{j+\frac{1}{2},k} v_{j,k} v_{j+1,k} \right). \end{aligned}$$

Using the homogenous boundary conditions we obtain

$$\begin{aligned}
\sum_{j=1}^m \sum_{k=1}^m c_{j,k-\frac{1}{2}} v_{j,k}^2 &= \sum_{j=1}^m \sum_{k=0}^m c_{j,k+\frac{1}{2}} v_{j,k+1}^2, \\
\sum_{j=1}^m \sum_{k=1}^m c_{j,k-\frac{1}{2}} v_{j,k-1} v_{j,k} &= \sum_{j=1}^m \sum_{k=0}^m c_{j,k+\frac{1}{2}} v_{j,k+1} v_{j,k}, \\
\sum_{j=1}^m \sum_{k=1}^m c_{j-\frac{1}{2},k} v_{j,k}^2 &= \sum_{k=1}^m \sum_{j=0}^m c_{j+\frac{1}{2},k} v_{j+1,k}^2, \\
\sum_{j=1}^m \sum_{k=1}^m c_{j-\frac{1}{2},k} v_{j-1,k} v_{j,k} &= \sum_{k=1}^m \sum_{j=0}^m c_{j+\frac{1}{2},k} v_{j+1,k} v_{j,k}.
\end{aligned}$$

It follows that

$$\begin{aligned}
\mathbf{x}^T \mathbf{A} \mathbf{x} &= \sum_{j=1}^m \sum_{k=0}^m c_{j,k+\frac{1}{2}} (v_{j,k}^2 + v_{j,k+1}^2 - 2v_{j,k} v_{j,k+1}) \\
&\quad + \sum_{k=1}^m \sum_{j=0}^m c_{j+\frac{1}{2},k} (v_{j,k}^2 + v_{j+1,k}^2 - 2v_{j,k} v_{j+1,k})
\end{aligned}$$

and (12.55) follows. \square

12.6.2 Applying preconditioning

Consider solving $\mathbf{Ax} = \mathbf{b}$, where \mathbf{A} is given by (12.54) and $\mathbf{b} \in \mathbb{R}^n$. Since \mathbf{A} is positive definite it is nonsingular and the system has a unique solution $\mathbf{x} \in \mathbb{R}^n$. Moreover we can use either Cholesky factorization or the block tridiagonal solver to find \mathbf{x} . Since the bandwidth of \mathbf{A} is $m = \sqrt{n}$ both of these methods require $O(n^2)$ arithmetic operations for large n .

If we choose $c(x, y) \equiv 1$ in (12.51), we get the Poisson problem. With this in mind, we may think of the coefficient matrix \mathbf{A}_p arising from the discretization of the Poisson problem as an approximation to the matrix (12.54). This suggests using $\mathbf{B} = \mathbf{A}_p^{-1}$, the inverse of the discrete Poisson matrix as a preconditioner for the system (12.52).

Consider Algorithm 12.30. With this preconditioner the calculation $\mathbf{w} = \mathbf{B}\mathbf{t}$ takes the form $\mathbf{A}_p \mathbf{w}_k = \mathbf{t}_k$.

In Section 10.2 we developed a Simple fast Poisson Solver, Cf. Algorithm 10.1. This method can be utilized to solve $\mathbf{A}_p \mathbf{w} = \mathbf{t}$.

Consider the specific problem where

$$c(x, y) = e^{-x+y} \text{ and } f(x, y) = 1.$$

n	2500	10000	22500	40000	62500
K	222	472	728	986	1246
K/\sqrt{n}	4.44	4.72	4.85	4.93	4.98
K_{pre}	22	23	23	23	23

Table 12.33: The number of iterations K (no preconditioning) and K_{pre} (with preconditioning) for the problem (12.51) using the discrete Poisson problem as a preconditioner.

We have used Algorithm 12.12 (conjugate gradient without preconditioning), and Algorithm 12.30 (conjugate gradient with preconditioning) to solve the problem (12.51). We used $\mathbf{x}_0 = 0$ and $\epsilon = 10^{-8}$. The results are shown in Table 12.33.

Without preconditioning the number of iterations still seems to be more or less proportional to \sqrt{n} although the convergence is slower than for the constant coefficient problem. Using preconditioning speeds up the convergence considerably. The number of iterations appears to be bounded independently of n .

Using a preconditioner increases the work in each iteration. For the present example the number of arithmetic operations in each iteration changes from $O(n)$ without preconditioning to $O(n^{3/2})$ or $O(n \log_2 n)$ with preconditioning. This is not a large increase and both the number of iterations and the computing time is reduced significantly.

Let us finally show that the number $\kappa = \lambda_{max}/\lambda_{min}$ which determines the rate of convergence for the preconditioned conjugate gradient method applied to (12.51) can be bounded independently of n .

Theorem 12.34 (Eigenvalues of preconditioned matrix)

Suppose $0 < c_0 \leq c(x, y) \leq c_1$ for all $(x, y) \in [0, 1]^2$. For the eigenvalues of the matrix $\mathbf{B}\mathbf{A} = \mathbf{A}_p^{-1}\mathbf{A}$ just described we have

$$\kappa = \frac{\lambda_{max}}{\lambda_{min}} \leq \frac{c_1}{c_0}.$$

Proof.

Suppose $\mathbf{A}_p^{-1}\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ for some $\mathbf{x} \in \mathbb{R}^n \setminus \{0\}$. Then $\mathbf{A}\mathbf{x} = \lambda\mathbf{A}_p\mathbf{x}$. Multiplying this by \mathbf{x}^T and solving for λ we find

$$\lambda = \frac{\mathbf{x}^T \mathbf{A} \mathbf{x}}{\mathbf{x}^T \mathbf{A}_p \mathbf{x}}.$$

We computed $\mathbf{x}^T \mathbf{A} \mathbf{x}$ in (12.55) and we obtain $\mathbf{x}^T \mathbf{A}_p \mathbf{x}$ by setting all the c 's there

equal to one

$$\mathbf{x}^T \mathbf{A}_p \mathbf{x} = \sum_{i=1}^m \sum_{j=0}^m (v_{i,j+1} - v_{i,j})^2 + \sum_{j=1}^m \sum_{i=0}^m (v_{i+1,j} - v_{i,j})^2.$$

Thus $\mathbf{x}^T \mathbf{A}_p \mathbf{x} > 0$ and bounding all the c 's in (12.55) from below by c_0 and above by c_1 we find

$$c_0(\mathbf{x}^T \mathbf{A}_p \mathbf{x}) \leq \mathbf{x}^T \mathbf{A} \mathbf{x} \leq c_1(\mathbf{x}^T \mathbf{A}_p \mathbf{x})$$

which implies that $c_0 \leq \lambda \leq c_1$ for all eigenvalues λ of $\mathbf{B}\mathbf{A} = \mathbf{A}_p^{-1}\mathbf{A}$. \square

Using $c(x, y) = e^{-x+y}$ as above, we find $c_0 = e^{-2}$ and $c_1 = 1$. Thus $\kappa \leq e^2 \approx 7.4$, a quite acceptable matrix condition number which explains the convergence results from our numerical experiment.

12.7 Review Questions

12.7.1 Does the steepest descent and conjugate gradient method always converge?

12.7.2 What kind of orthogonalities occur in the conjugate gradient method?

12.7.3 What is a Krylov space?

12.7.4 What is a convex function?

12.7.5 How do SOR and conjugate gradient compare?

Part V

Eigenvalues and Eigenvectors

Chapter 13

Numerical Eigenvalue Problems

13.1 Eigenpairs

Eigenpairs have applications in quantum mechanics, differential equations, elasticity in mechanics,

Consider the eigenpair problem for some classes of matrices $\mathbf{A} \in \mathbb{C}^{n \times n}$.

Diagonal Matrices. The eigenpairs are easily determined. Since $\mathbf{A}\mathbf{e}_i = a_{ii}\mathbf{e}_i$ the eigenpairs are $(\lambda_i, \mathbf{e}_i)$, where $\lambda_i = a_{ii}$ for $i = 1, \dots, n$. Moreover, the eigenvectors of \mathbf{A} are linearly independent.

Triangular Matrices Suppose \mathbf{A} is upper or lower triangular. Consider finding the eigenvalues Since $\det(\mathbf{A} - \lambda \mathbf{I}) = \prod_{i=1}^n (a_{ii} - \lambda)$ the eigenvalues are $\lambda_i = a_{ii}$ for $i = 1, \dots, n$, the diagonal elements of \mathbf{A} . To determine the eigenvectors can be challenging.

Block Diagonal Matrices Suppose

$$\mathbf{A} = \text{diag}(\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_r), \quad \mathbf{A}_i \in \mathbb{C}^{m_i \times m_i}.$$

Here the eigenpair problem reduces to r smaller problems. Let $\mathbf{A}_i \mathbf{X}_i = \mathbf{X}_i \mathbf{D}_i$ define the eigenpairs of \mathbf{A}_i for $i = 1, \dots, r$ and let $\mathbf{X} := \text{diag}(\mathbf{X}_1, \dots, \mathbf{X}_r)$, $\mathbf{D} := \text{diag}(\mathbf{D}_1, \dots, \mathbf{D}_r)$. Then the eigenpairs for \mathbf{A} are given by

$$\begin{aligned} \mathbf{AD} &= \text{diag}(\mathbf{A}_1, \dots, \mathbf{A}_r) \text{diag}(\mathbf{X}_1, \dots, \mathbf{X}_r) = \text{diag}(\mathbf{A}_1 \mathbf{X}_1, \dots, \mathbf{A}_r \mathbf{X}_r) \\ &= \text{diag}(\mathbf{X}_1 \mathbf{D}_1, \dots, \mathbf{X}_r \mathbf{D}_r) = \mathbf{XD}. \end{aligned}$$

Block Triangular matrices Let $\mathbf{A}_{11}, \mathbf{A}_{22}, \dots, \mathbf{A}_{rr}$ be the diagonal

blocks of \mathbf{A} . By Property 8. of determinants

$$\det(\mathbf{A} - \lambda \mathbf{I}) = \prod_{i=1}^r \det(\mathbf{A}_{ii} - \lambda \mathbf{I})$$

and the eigenvalues are found from the eigenvalues of the diagonal blocks.

In this and the next chapter we consider some numerical methods for finding one or more of the eigenvalues and eigenvectors of a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$. Maybe the first method which comes to mind is to form the characteristic polynomial $\pi_{\mathbf{A}}$ of \mathbf{A} , and then use a polynomial root finder, like Newton's method to determine one or several of the eigenvalues.

It turns out that this is not suitable as an all purpose method. One reason is that a small change in one of the coefficients of $\pi_{\mathbf{A}}(\lambda)$ can lead to a large change in the roots of the polynomial. For example, if $\pi_{\mathbf{A}}(\lambda) = \lambda^{16}$ and $q(\lambda) = \lambda^{16} - 10^{-16}$ then the roots of $\pi_{\mathbf{A}}$ are all equal to zero, while the roots of q are $\lambda_j = 10^{-1} e^{2\pi i j / 16}$, $j = 1, \dots, 16$. The roots of q have absolute value 0.1 and a perturbation in one of the polynomial coefficients of magnitude 10^{-16} has led to an error in the roots of approximately 0.1. The situation can be somewhat remedied by representing the polynomials using a different basis.

In this text we will only consider methods which work directly with the matrix. But before that, in Section 13.3 we consider how much the eigenvalues change when the elements in the matrix are perturbed. We start with a simple but useful result for locating the eigenvalues.

13.2 Gerschgorin's Theorem

The following theorem is useful for locating eigenvalues of an arbitrary square matrix.

Theorem 13.1 (Gerschgorin's circle theorem)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$. Define for $i = 1, 2, \dots, n$

$$R_i = \{z \in \mathbb{C} : |z - a_{ii}| \leq r_i\}, \quad r_i := \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|,$$

$$C_j = \{z \in \mathbb{C} : |z - a_{jj}| \leq c_j\}, \quad c_j := \sum_{\substack{i=1 \\ i \neq j}}^n |a_{ij}|.$$

Then any eigenvalue of \mathbf{A} lies in $R \cap C$ where $R = R_1 \cup R_2 \cup \dots \cup R_n$ and $C = C_1 \cup C_2 \cup \dots \cup C_n$.

Proof. Suppose (λ, \mathbf{x}) is an eigenpair for \mathbf{A} . We claim that $\lambda \in R_i$, where i is such that $|x_i| = \|\mathbf{x}\|_\infty$. Indeed, $\mathbf{Ax} = \lambda\mathbf{x}$ implies that $\sum_j a_{ij}x_j = \lambda x_i$ or $(\lambda - a_{ii})x_i = \sum_{j \neq i} a_{ij}x_j$. Dividing by x_i and taking absolute values we find

$$|\lambda - a_{ii}| = \left| \sum_{j \neq i} a_{ij}x_j/x_i \right| \leq \sum_{j \neq i} |a_{ij}| |x_j/x_i| \leq r_i$$

since $|x_j/x_i| \leq 1$ for all j . Thus $\lambda \in R_i$.

Since λ is also an eigenvalue of \mathbf{A}^T , it must be in one of the row disks of \mathbf{A}^T . But these are the column disks C_j of \mathbf{A} . Hence $\lambda \in C_j$ for some j . \square

The set R_i is a subset of the complex plane consisting of all points inside a circle with center at a_{ii} and radius r_i , c.f. Figure 13.1. R_i is called a (Gershgorin) row disk.

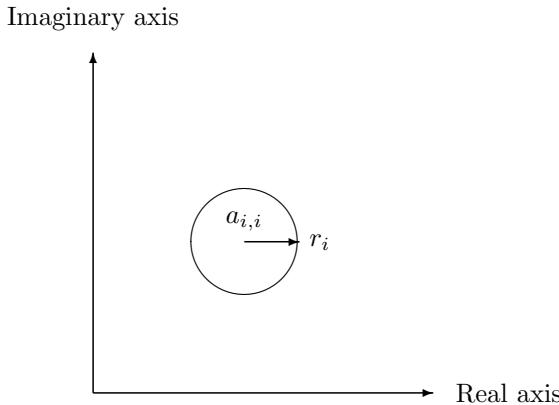


Figure 13.1: The Gershgorin disk R_i .

An eigenvalue λ lies in the union of the row disks R_1, \dots, R_n and also in the union of the column disks C_1, \dots, C_n . If \mathbf{A} is Hermitian then $R_i = C_i$ for $i = 1, 2, \dots, n$. Moreover, in this case the eigenvalues of \mathbf{A} are real, and the Gershgorin disks can be taken to be intervals on the real line.

Example 13.2 (Gershgorin)

Let $\mathbf{T} = \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{m \times m}$ be the second derivative matrix. Since \mathbf{A} is Hermitian we have $R_i = C_i$ for all i and the eigenvalues are real. We find

$$R_1 = R_m = \{z \in \mathbb{R} : |z-2| \leq 1\}, \text{ and } R_i = \{z \in \mathbb{R} : |z-2| \leq 2\}, \quad i = 2, 3, \dots, m-1.$$

We conclude that $\lambda \in [0, 4]$ for any eigenvalue λ of \mathbf{T} . To check this, we recall that by Lemma 1.31 the eigenvalues of \mathbf{T} are given by

$$\lambda_j = 4 \left[\sin \frac{j\pi}{2(m+1)} \right]^2, \quad j = 1, 2, \dots, m.$$

When m is large the smallest eigenvalue $4 \left[\sin \frac{\pi}{2(m+1)} \right]^2$ is very close to zero and the largest eigenvalue $4 \left[\sin \frac{m\pi}{2(m+1)} \right]^2$ is very close to 4. Thus Gershgorin's theorem gives a remarkably good estimate for large m .

Sometimes some of the Gershgorin disks are distinct and we have

Corollary 13.3 (Disjoint Gershgorin disks)

If p of the Gershgorin row disks are disjoint from the others, the union of these disks contains precisely p eigenvalues. The same result holds for the column disks.

Proof. Consider a family of matrices

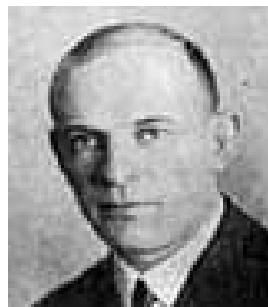
$$\mathbf{A}(t) := \mathbf{D} + t(\mathbf{A} - \mathbf{D}), \quad \mathbf{D} := \text{diag}(a_{11}, \dots, a_{nn}), \quad t \in [0, 1].$$

We have $\mathbf{A}(0) = \mathbf{D}$ and $\mathbf{A}(1) = \mathbf{A}$. As a function of t , every eigenvalue of $\mathbf{A}(t)$ is a continuous function of t . This follows from Theorem 13.7, see Exercise 13.8. The row disks $R_i(t)$ of $\mathbf{A}(t)$ have radius proportional to t , indeed

$$R_i(t) = \{z \in \mathbb{C} : |z - a_{ii}| \leq tr_i\}, \quad r_i := \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|.$$

Clearly $0 \leq t_1 < t_2 \leq 1$ implies $R_i(t_1) \subset R_i(t_2)$ and $R_i(1)$ is a row disk of \mathbf{A} for all i . Suppose $\bigcup_{k=1}^p R_{i_k}(1)$ are disjoint from the other disks of \mathbf{A} and set $R^p(t) := \bigcup_{k=1}^p R_{i_k}(t)$ for $t \in [0, 1]$. Now $R^p(0)$ contains only the p eigenvalues $a_{i_1, i_1}, \dots, a_{i_p, i_p}$ of $\mathbf{A}(0) = \mathbf{D}$. As t increases from zero to one the set $R^p(t)$ is disjoint from the other row disks of \mathbf{A} and by the continuity of the eigenvalues cannot loose or gain eigenvalues. It follows that $R^p(1)$ must contain p eigenvalues of \mathbf{A} . \square

Example 13.4 Consider the matrix $\mathbf{A} = \begin{bmatrix} 1 & \epsilon_1 & \epsilon_2 \\ \epsilon_3 & 2 & \epsilon_4 \\ \epsilon_5 & \epsilon_6 & 3 \end{bmatrix}$, where $|\epsilon_i| \leq 10^{-15}$ all i . By Corollary 13.3 the eigenvalues $\lambda_1, \lambda_2, \lambda_3$ of \mathbf{A} are distinct and satisfy $|\lambda_j - j| \leq 2 \times 10^{-15}$ for $j = 1, 2, 3$.



Semyon Aranovich Gershgorin, 1901-1933 (left), Jacques Salomon Hadamard, 1865-1963 (right).

Exercise 13.5 (Nonsingularity using Gerschgorin)

Consider the matrix

$$\mathbf{A} = \begin{pmatrix} 4 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 \\ 0 & 1 & 4 & 1 \\ 0 & 0 & 1 & 4 \end{pmatrix}.$$

Show using Gerschgorin's theorem that \mathbf{A} is nonsingular.

Exercise 13.6 (Gerschgorin, strictly diagonally dominant matrix)

Show using Gerschgorin's theorem that a strictly diagonally dominant matrix \mathbf{A} ($|a_{i,i}| > \sum_{j \neq i} |a_{i,j}|$ for all i) is nonsingular.

13.3 Perturbation of Eigenvalues

In this section we study the following problem. Given matrices $\mathbf{A}, \mathbf{E} \in \mathbb{C}^{n \times n}$, where we think of \mathbf{E} as a perturbation of \mathbf{A} . By how much do the eigenvalues of \mathbf{A} and $\mathbf{A} + \mathbf{E}$ differ? Not surprisingly this problem is more complicated than the corresponding problem for linear systems.

We illustrate this by considering two examples. Suppose $\mathbf{A}_0 := \mathbf{0}$ is the zero matrix. If $\lambda \in \sigma(\mathbf{A}_0 + \mathbf{E}) = \sigma(\mathbf{E})$, then $|\lambda| \leq \|\mathbf{E}\|_\infty$ by Theorem 11.28, and any zero eigenvalue of \mathbf{A}_0 is perturbed by at most $\|\mathbf{E}\|_\infty$. On the other hand consider for $\epsilon > 0$ the matrices

$$\mathbf{A}_1 := \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}, \quad \mathbf{E} := \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \epsilon & 0 & 0 & \cdots & 0 & 0 \end{bmatrix} = \epsilon \mathbf{e}_n \mathbf{e}_1^T.$$

The characteristic polynomial of $\mathbf{A}_1 + \mathbf{E}$ is $\pi(\lambda) := (-1)^n(\lambda^n - \epsilon)$, and the zero eigenvalues of \mathbf{A}_1 are perturbed by the amount $|\lambda| = \|\mathbf{E}\|_\infty^{1/n}$. Thus, for $n = 16$, a perturbation of say $\epsilon = 10^{-16}$ gives a change in eigenvalue of 0.1.

The following theorem shows that a dependence $\|\mathbf{E}\|_\infty^{1/n}$ is the worst that can happen.

Theorem 13.7 (Elsner's theorem(1985))

Suppose $\mathbf{A}, \mathbf{E} \in \mathbb{C}^{n \times n}$. To every $\mu \in \sigma(\mathbf{A} + \mathbf{E})$ there is a $\lambda \in \sigma(\mathbf{A})$ such that

$$|\mu - \lambda| \leq K\|\mathbf{E}\|_2^{1/n}, \quad K = (\|\mathbf{A}\|_2 + \|\mathbf{A} + \mathbf{E}\|_2)^{1-1/n}. \quad (13.1)$$

Proof. Suppose \mathbf{A} has eigenvalues $\lambda_1, \dots, \lambda_n$ and let λ_1 be one which is closest to μ . Let \mathbf{u}_1 with $\|\mathbf{u}_1\|_2 = 1$ be an eigenvector corresponding to μ , and extend \mathbf{u}_1 to an orthonormal basis $\{\mathbf{u}_1, \dots, \mathbf{u}_n\}$ of \mathbb{C}^n . Note that

$$\begin{aligned} \|(\mu\mathbf{I} - \mathbf{A})\mathbf{u}_1\|_2 &= \|(\mathbf{A} + \mathbf{E})\mathbf{u}_1 - \mathbf{A}\mathbf{u}_1\|_2 = \|\mathbf{E}\mathbf{u}_1\|_2 \leq \|\mathbf{E}\|_2, \\ \prod_{j=2}^n \|(\mu\mathbf{I} - \mathbf{A})\mathbf{u}_j\|_2 &\leq \prod_{j=2}^n (|\mu| + \|\mathbf{A}\mathbf{u}_j\|_2) \leq (\|(\mathbf{A} + \mathbf{E})\|_2 + \|\mathbf{A}\|_2)^{n-1}. \end{aligned}$$

Using this and Hadamard's inequality (4.15) we find

$$\begin{aligned} |\mu - \lambda_1|^n &\leq \prod_{j=1}^n |\mu - \lambda_j| = |\det(\mu\mathbf{I} - \mathbf{A})| = |\det((\mu\mathbf{I} - \mathbf{A})[\mathbf{u}_1, \dots, \mathbf{u}_n])| \\ &\leq \|(\mu\mathbf{I} - \mathbf{A})\mathbf{u}_1\|_2 \prod_{j=2}^n \|(\mu\mathbf{I} - \mathbf{A})\mathbf{u}_j\|_2 \leq \|\mathbf{E}\|_2 (\|(\mathbf{A} + \mathbf{E})\|_2 + \|\mathbf{A}\|_2)^{n-1}. \end{aligned}$$

The result follows by taking n th roots in this inequality. \square

It follows from this theorem that the eigenvalues depend continuously on the elements of the matrix. The factor $\|\mathbf{E}\|_2^{1/n}$ shows that this dependence is almost, but not quite, differentiable. As an example, the eigenvalues of the matrix $\begin{bmatrix} 1 & 1 \\ \epsilon & 1 \end{bmatrix}$ are $1 \pm \sqrt{\epsilon}$ and this expression is not differentiable at $\epsilon = 0$.

Exercise 13.8 (Continuity of eigenvalues)

Suppose

$$\mathbf{A}(t) := \mathbf{D} + t(\mathbf{A} - \mathbf{D}), \quad \mathbf{D} := \text{diag}(a_{11}, \dots, a_{nn}), \quad t \in \mathbb{R}.$$

$0 \leq t_1 < t_2 \leq 1$ and that μ is an eigenvalue of $\mathbf{A}(t_2)$. Show, using Theorem 13.7 with $\mathbf{A} = \mathbf{A}(t_1)$ and $\mathbf{E} = \mathbf{A}(t_2) - \mathbf{A}(t_1)$, that $\mathbf{A}(t_1)$ has an eigenvalue λ such that

$$|\lambda - \mu| \leq C(t_2 - t_1)^{1/n}, \quad \text{where } C \leq 2(\|\mathbf{D}\|_2 + \|\mathbf{A} - \mathbf{D}\|_2).$$

Thus, as a function of t , every eigenvalue of $\mathbf{A}(t)$ is a continuous function of t .

13.3.1 Nondefective matrices

Recall that a matrix is nondefective if the eigenvectors form a basis for \mathbb{C}^n . For nondefective matrices we can get rid of the annoying exponent $1/n$ in $\|\mathbf{E}\|_2$. For a more general discussion than the one in the following theorem see [30].

Theorem 13.9 (Absolute errors)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ has linearly independent eigenvectors $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ and let $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]$ be the eigenvector matrix. To any $\mu \in \mathbb{C}$ and $\mathbf{x} \in \mathbb{C}^n$ with $\|\mathbf{x}\|_p = 1$ we can find an eigenvalue λ of \mathbf{A} such that

$$|\lambda - \mu| \leq K_p(\mathbf{X})\|\mathbf{r}\|_p, \quad 1 \leq p \leq \infty, \quad (13.2)$$

where $\mathbf{r} := \mathbf{Ax} - \mu\mathbf{x}$ and $K_p(\mathbf{X}) := \|\mathbf{X}\|_p \|\mathbf{X}^{-1}\|_p$. If for some $\mathbf{E} \in \mathbb{C}^{n \times n}$ it holds that (μ, \mathbf{x}) is an eigenpair for $\mathbf{A} + \mathbf{E}$, then we can find an eigenvalue λ of \mathbf{A} such that

$$|\lambda - \mu| \leq K_p(\mathbf{X})\|\mathbf{E}\|_p, \quad 1 \leq p \leq \infty, \quad (13.3)$$

Proof. If $\mu \in \sigma(\mathbf{A})$ then we can take $\lambda = \mu$ and (13.2), (13.3) hold trivially. So assume $\mu \notin \sigma(\mathbf{A})$. Since \mathbf{A} is nondefective it can be diagonalized, we have $\mathbf{A} = \mathbf{XD}\mathbf{X}^{-1}$, where $\mathbf{D} = \text{diag}(\lambda_1, \dots, \lambda_n)$ and $(\lambda_j, \mathbf{x}_j)$ are the eigenpairs of \mathbf{A} for $j = 1, \dots, n$. Define $\mathbf{D}_1 := \mathbf{D} - \mu\mathbf{I}$. Then $\mathbf{D}_1^{-1} = \text{diag}((\lambda_1 - \mu)^{-1}, \dots, (\lambda_n - \mu)^{-1})$ exists and

$$\mathbf{XD}_1^{-1}\mathbf{X}^{-1}\mathbf{r} = (\mathbf{X}(\mathbf{D} - \mu\mathbf{I})\mathbf{X}^{-1})^{-1}\mathbf{r} = (\mathbf{A} - \mu\mathbf{I})^{-1}(\mathbf{A} - \mu\mathbf{I})\mathbf{x} = \mathbf{x}.$$

Using this and Lemma 13.11 below we obtain

$$1 = \|\mathbf{x}\|_p = \|\mathbf{XD}_1^{-1}\mathbf{X}^{-1}\mathbf{r}\|_p \leq \|\mathbf{D}_1^{-1}\|_p K_p(\mathbf{X})\|\mathbf{r}\|_p = \frac{K_p(\mathbf{X})\|\mathbf{r}\|_p}{\min_j |\lambda_j - \mu|}.$$

But then (13.2) follows. If $(\mathbf{A} + \mathbf{E})\mathbf{x} = \mu\mathbf{x}$ then $\mathbf{0} = \mathbf{Ax} - \mu\mathbf{x} + \mathbf{Ex} = \mathbf{r} + \mathbf{Ex}$. But then $\|\mathbf{r}\|_p = \|-\mathbf{Ex}\|_p \leq \|\mathbf{E}\|_p$. Inserting this in (13.2) proves (13.3). \square

The equation (13.3) shows that for a nondefective matrix the absolute error can be magnified by at most $K_p(\mathbf{X})$, the condition number of the eigenvector matrix with respect to inversion. If $K_p(\mathbf{X})$ is small then a small perturbation changes the eigenvalues by small amounts.

Even if we get rid of the exponent $1/n$, the equation (13.3) illustrates that it can be difficult or sometimes impossible to compute accurate eigenvalues and eigenvectors of matrices with almost linearly dependent eigenvectors. On the other hand the eigenvalue problem for normal matrices is better conditioned. Indeed, if \mathbf{A} is normal then it has a set of orthonormal eigenvectors and the eigenvector matrix is unitary. If we restrict attention to the 2-norm then $K_2(\mathbf{X}) = 1$ and (13.3) implies the following result.

Theorem 13.10 (Perturbations, normal matrix)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is normal and let μ be an eigenvalue of $\mathbf{A} + \mathbf{E}$ for some $\mathbf{E} \in \mathbb{C}^{n \times n}$. Then we can find an eigenvalue λ of \mathbf{A} such that $|\lambda - \mu| \leq \|\mathbf{E}\|_2$.

For an even stronger result for Hermitian matrices see Corollary 5.45. We conclude that the situation for the absolute error in an eigenvalue of a Hermitian matrix is quite satisfactory. Small perturbations in the elements are not magnified in the eigenvalues.

In the proof of Theorem 13.9 we used that the p -norm of a diagonal matrix is equal to its spectral radius.

Lemma 13.11 (p -norm of a diagonal matrix)

If $\mathbf{A} = \text{diag}(\lambda_1, \dots, \lambda_n)$ is a diagonal matrix then $\|\mathbf{A}\|_p = \rho(\mathbf{A})$ for $1 \leq p \leq \infty$.

Proof. For $p = \infty$ the proof is left as an exercise. For any $\mathbf{x} \in \mathbb{C}^n$ and $p < \infty$ we have

$$\|\mathbf{Ax}\|_p = \|[\lambda_1 x_1, \dots, \lambda_n x_n]^T\|_p = \left(\sum_{j=1}^n |\lambda_j|^p |x_j|^p \right)^{1/p} \leq \rho(\mathbf{A}) \|\mathbf{x}\|_p.$$

Thus $\|\mathbf{A}\|_p = \max_{\mathbf{x} \neq \mathbf{0}} \frac{\|\mathbf{Ax}\|_p}{\|\mathbf{x}\|_p} \leq \rho(\mathbf{A})$. But from Theorem 11.28 we have $\rho(\mathbf{A}) \leq \|\mathbf{A}\|_p$ and the proof is complete. \square

Exercise 13.12 (∞ -norm of a diagonal matrix)

Give a direct proof that $\|\mathbf{A}\|_\infty = \rho(\mathbf{A})$ if \mathbf{A} is diagonal.

For the accuracy of an eigenvalue of small magnitude we are interested in the size of the relative error.

Theorem 13.13 (Relative errors)

Suppose in Theorem 13.9 that $\mathbf{A} \in \mathbb{C}^{n \times n}$ is nonsingular. To any $\mu \in \mathbb{C}$ and $\mathbf{x} \in \mathbb{C}^n$ with $\|\mathbf{x}\|_p = 1$, we can find an eigenvalue λ of \mathbf{A} such that

$$\frac{|\lambda - \mu|}{|\lambda|} \leq K_p(\mathbf{X}) K_p(\mathbf{A}) \frac{\|\mathbf{r}\|_p}{\|\mathbf{A}\|_p}, \quad 1 \leq p \leq \infty, \quad (13.4)$$

where $\mathbf{r} := \mathbf{Ax} - \mu\mathbf{x}$. If for some $\mathbf{E} \in \mathbb{C}^{n \times n}$ it holds that (μ, \mathbf{x}) is an eigenpair for $\mathbf{A} + \mathbf{E}$, then we can find an eigenvalue λ of \mathbf{A} such that

$$\frac{|\lambda - \mu|}{|\lambda|} \leq K_p(\mathbf{X}) \|\mathbf{A}^{-1} \mathbf{E}\|_p \leq K_p(\mathbf{X}) K_p(\mathbf{A}) \frac{\|\mathbf{E}\|_p}{\|\mathbf{A}\|_p}, \quad 1 \leq p \leq \infty, \quad (13.5)$$

Proof. Applying Theorem 11.28 to \mathbf{A}^{-1} we have for any $\lambda \in \sigma(\mathbf{A})$

$$\frac{1}{\lambda} \leq \|\mathbf{A}^{-1}\|_p = \frac{K_p(\mathbf{A})}{\|\mathbf{A}\|_p}$$

and (13.4) follows from (13.2). To prove (13.5) we define the matrices $\mathbf{B} := \mu\mathbf{A}^{-1}$ and $\mathbf{F} := -\mathbf{A}^{-1}\mathbf{E}$. If (λ_j, \mathbf{x}) are the eigenpairs for \mathbf{A} then $(\frac{\mu}{\lambda_j}, \mathbf{x})$ are the eigenpairs for \mathbf{B} for $j = 1, \dots, n$. Since (μ, \mathbf{x}) is an eigenpair for $\mathbf{A} + \mathbf{E}$ we find

$$(\mathbf{B} + \mathbf{F} - \mathbf{I})\mathbf{x} = (\mu\mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{E} - \mathbf{I})\mathbf{x} = \mathbf{A}^{-1}(\mu\mathbf{I} - (\mathbf{E} + \mathbf{A}))\mathbf{x} = \mathbf{0}.$$

Thus $(1, \mathbf{x})$ is an eigenpair for $\mathbf{B} + \mathbf{F}$. Applying Theorem 13.9 to this eigenvalue we can find $\lambda \in \sigma(\mathbf{A})$ such that $|\frac{\mu}{\lambda} - 1| \leq K_p(\mathbf{X})\|\mathbf{F}\|_p = K_p(\mathbf{X})\|\mathbf{A}^{-1}\mathbf{E}\|_p$ which proves the first estimate in (13.5). The second inequality in (13.5) follows from the submultiplicativity of the p -norm. \square

13.4 Unitary Similarity Transformation of a Matrix into Upper Hessenberg Form

Before attempting to find eigenvalues and eigenvectors of a matrix (exceptions are made for certain sparse matrices), it is often advantageous to reduce it by similarity transformations to a simpler form. Orthogonal or unitary similarity transformations are particularly important since they are insensitive to round-off errors in the elements of the matrix. In this section we show how this reduction can be carried out.

Recall that a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ is upper Hessenberg if $a_{i,j} = 0$ for $j = 1, 2, \dots, i-2$, $i = 3, 4, \dots, n$. We will reduce $\mathbf{A} \in \mathbb{C}^{n \times n}$ to upper Hessenberg form by unitary similarity transformations. Let $\mathbf{A}_1 = \mathbf{A}$ and define $\mathbf{A}_{k+1} = \mathbf{H}_k \mathbf{A}_k \mathbf{H}_k$ for $k = 1, 2, \dots, n-2$. Here \mathbf{H}_k is a Householder transformation chosen to introduce zeros in the elements of column k of \mathbf{A}_k under the subdiagonal. The final matrix \mathbf{A}_{n-1} will be upper Hessenberg. Householder transformations were used in Chapter 4 to reduce a matrix to upper triangular form. To preserve eigenvalues similarity transformations are needed and then the final matrix in the reduction cannot in general be upper triangular.

If $\mathbf{A}_1 = \mathbf{A}$ is Hermitian, the matrix \mathbf{A}_{n-1} will be Hermitian and tridiagonal. For if $\mathbf{A}_k^* = \mathbf{A}_k$ then

$$\mathbf{A}_{k+1}^* = (\mathbf{H}_k \mathbf{A}_k \mathbf{H}_k)^* = \mathbf{H}_k \mathbf{A}_k^* \mathbf{H}_k = \mathbf{A}_{k+1}.$$

Since \mathbf{A}_{n-1} is upper Hessenberg and Hermitian, it must be tridiagonal.

To describe the reduction to upper Hessenberg or tridiagonal form in more detail we partition \mathbf{A}_k as follows

$$\mathbf{A}_k = \begin{bmatrix} \mathbf{B}_k & \mathbf{C}_k \\ \mathbf{D}_k & \mathbf{E}_k \end{bmatrix}.$$

Suppose $\mathbf{B}_k \in \mathbb{C}^{k,k}$ is upper Hessenberg, and the first $k - 1$ columns of $\mathbf{D}_k \in \mathbb{C}^{n-k,k}$ are zero, i.e. $\mathbf{D}_k = [\mathbf{0}, \mathbf{0}, \dots, \mathbf{0}, \mathbf{d}_k]$. Let $\mathbf{V}_k = \mathbf{I} - \mathbf{v}_k \mathbf{v}_k^* \in \mathbb{C}^{n-k,n-k}$ be a Householder transformation such that $\mathbf{V}_k \mathbf{d}_k = \alpha_k \mathbf{e}_1$. Define

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_k \end{bmatrix} \in \mathbb{C}^{n \times n}.$$

The matrix \mathbf{H}_k is a Householder transformation, and we find

$$\begin{aligned} \mathbf{A}_{k+1} &= \mathbf{H}_k \mathbf{A}_k \mathbf{H}_k = \begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_k \end{bmatrix} \begin{bmatrix} \mathbf{B}_k & \mathbf{C}_k \\ \mathbf{D}_k & \mathbf{E}_k \end{bmatrix} \begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_k \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{B}_k & \mathbf{C}_k \mathbf{V}_k \\ \mathbf{V}_k \mathbf{D}_k & \mathbf{V}_k \mathbf{E}_k \mathbf{V}_k \end{bmatrix}. \end{aligned}$$

Now $\mathbf{V}_k \mathbf{D}_k = [\mathbf{V}_k \mathbf{0}, \dots, \mathbf{V}_k \mathbf{0}, \mathbf{V}_k \mathbf{d}_k] = (\mathbf{0}, \dots, \mathbf{0}, \alpha_k \mathbf{e}_1)$. Moreover, the matrix \mathbf{B}_k is not affected by the \mathbf{H}_k transformation. Therefore the upper left $(k+1) \times (k+1)$ corner of \mathbf{A}_{k+1} is upper Hessenberg and the reduction is carried one step further. The reduction stops with \mathbf{A}_{n-1} which is upper Hessenberg.

To find \mathbf{A}_{k+1} we use Algorithm 4.17 to find \mathbf{v}_k and α_k . We store \mathbf{v}_k in the k th column of a matrix \mathbf{L} as $\mathbf{L}(k+1 : n, k) = \mathbf{v}_k$. This leads to the following algorithm.

Algorithm 13.14 (Householder reduction to Hessenberg form) This algorithm uses Householder similarity transformations to reduce a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ to upper Hessenberg form. The reduced matrix \mathbf{B} is tridiagonal if \mathbf{A} is symmetric. Details of the transformations are stored in a lower triangular matrix \mathbf{L} . The elements of \mathbf{L} can be used to assemble a unitary matrix \mathbf{Q} such that $\mathbf{B} = \mathbf{Q}^* \mathbf{A} \mathbf{Q}$. Algorithm 4.17 is used in each step of the reduction.

```

1 function [L,B] = hesshousegen(A)
2 n=length(A); L=zeros(n,n); B=A;
3 for k=1:n-2
4     [v,B(k+1,k)]=housegen(B(k+1:n,k));
5     L((k+1):n,k)=v; B((k+2):n,k)=zeros(n-k-1,1);
6     C=B((k+1):n,(k+1):n); B((k+1):n,(k+1):n)=C-v*(v'*C);
7     C=B(1:n,(k+1):n); B(1:n,(k+1):n)=C-(C*v)*v';
8 end

```

Exercise 13.15 (Number of arithmetic operations, Hessenberg reduction)

Show that the number of arithmetic operations for Algorithm 13.14 is $\frac{10}{3}n^3 = 5G_n$.

13.4.1 Assembling Householder transformations

We can use the output of Algorithm 13.14 to assemble the matrix $\mathbf{Q} \in \mathbb{R}^{n \times n}$ such that \mathbf{Q} is orthonormal and $\mathbf{Q}^* \mathbf{A} \mathbf{Q}$ is upper Hessenberg. We need to compute the

product $\mathbf{Q} = \mathbf{H}_1 \mathbf{H}_2 \cdots \mathbf{H}_{n-2}$, where $\mathbf{H}_k = \begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{I} - \mathbf{v}_k \mathbf{v}_k^T \end{bmatrix}$ and $\mathbf{v}_k \in \mathbb{R}^{n-k}$. Since $\mathbf{v}_1 \in \mathbb{R}^{n-1}$ and $\mathbf{v}_{n-2} \in \mathbb{R}^2$ it is most economical to assemble the product from right to left. We compute

$$\mathbf{Q}_{n-1} = \mathbf{I} \text{ and } \mathbf{Q}_k = \mathbf{H}_k \mathbf{Q}_{k+1} \text{ for } k = n-2, n-3, \dots, 1.$$

Suppose \mathbf{Q}_{k+1} has the form $\begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{U}_k \end{bmatrix}$, where $\mathbf{U}_k \in \mathbb{R}^{n-k, n-k}$. Then

$$\mathbf{Q}_k = \begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{I} - \mathbf{v}_k \mathbf{v}_k^T \end{bmatrix} * \begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{U}_k \end{bmatrix} = \begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{U}_k - \mathbf{v}_k (\mathbf{v}_k^T \mathbf{U}_k) \end{bmatrix}.$$

This leads to the following algorithm.

Algorithm 13.16 (Assemble Householder transformations)

Suppose $[\mathbf{L}, \mathbf{B}] = \text{hesshousegen}(\mathbf{A})$ is the output of Algorithm 13.14. This algorithm assembles an orthonormal matrix \mathbf{Q} from the columns of \mathbf{L} such that $\mathbf{B} = \mathbf{Q}^* \mathbf{A} \mathbf{Q}$ is upper Hessenberg.

```

1 function Q = accumulateQ(L)
2 n=length(L); Q=eye(n);
3 for k=n-2:-1:1
4     v=L((k+1):n,k); C=Q((k+1):n,(k+1):n);
5     Q((k+1):n,(k+1):n)=C-v*(v'*C);
6 end

```

Exercise 13.17 (Assemble Householder transformations)

Show that the number of arithmetic operations required by Algorithm 13.16 is $\frac{4}{3}n^3 = 2G_n$.

Exercise 13.18 (Tridiagonalize a symmetric matrix)

If \mathbf{A} is real and symmetric we can modify Algorithm 13.14 as follows. To find \mathbf{A}_{k+1} from \mathbf{A}_k we have to compute $\mathbf{V}_k \mathbf{E}_k \mathbf{V}_k$ where \mathbf{E}_k is symmetric. Dropping subscripts we have to compute a product of the form $\mathbf{G} = (\mathbf{I} - \mathbf{v}\mathbf{v}^T)\mathbf{E}(\mathbf{I} - \mathbf{v}\mathbf{v}^T)$. Let $\mathbf{w} := \mathbf{E}\mathbf{v}$, $\beta := \frac{1}{2}\mathbf{v}^T\mathbf{w}$ and $\mathbf{z} := \mathbf{w} - \beta\mathbf{v}$. Show that $\mathbf{G} = \mathbf{E} - \mathbf{v}\mathbf{z}^T - \mathbf{z}\mathbf{v}^T$. Since \mathbf{G} is symmetric, only the sub- or superdiagonal elements of \mathbf{G} need to be computed. Computing \mathbf{G} in this way, it can be shown that we need $O(4n^3/3)$ operations to tridiagonalize a symmetric matrix by orthonormal similarity transformations. This is less than half the work to reduce a nonsymmetric matrix to upper Hessenberg form. We refer to [29] for a detailed algorithm.

13.5 Computing a Selected Eigenvalue of a Symmetric Matrix

Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be symmetric with eigenvalues $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$. In this section we consider a method to compute an approximation to the m th eigenvalue λ_m for

some $1 \leq m \leq n$. Using Householder similarity transformations as outlined in the previous section we can assume that \mathbf{A} is symmetric and tridiagonal.

$$\mathbf{A} = \begin{bmatrix} d_1 & c_1 & & & \\ c_1 & d_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & c_{n-2} & d_{n-1} & c_{n-1} \\ & & & c_{n-1} & d_n \end{bmatrix}. \quad (13.6)$$

Suppose one of the off-diagonal elements is equal to zero, say $c_i = 0$. We then have $\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 \end{bmatrix}$, where

$$\mathbf{A}_1 = \begin{bmatrix} d_1 & c_1 & & & \\ c_1 & d_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & c_{i-2} & d_{i-1} & c_{i-1} \\ & & & c_{i-1} & d_i \end{bmatrix} \text{ and } \mathbf{A}_2 = \begin{bmatrix} d_{i+1} & c_{i+1} & & & \\ c_{i+1} & d_{i+2} & c_{i+2} & & \\ & \ddots & \ddots & \ddots & \\ & & c_{n-2} & d_{n-1} & c_{n-1} \\ & & & c_{n-1} & d_n \end{bmatrix}.$$

Thus \mathbf{A} is block diagonal and we can split the eigenvalue problem into two smaller problems involving \mathbf{A}_1 and \mathbf{A}_2 . We assume that this reduction has been carried out so that \mathbf{A} is irreducible, i.e., $c_i \neq 0$ for $i = 1, \dots, n-1$.

We first show that irreducibility implies that the eigenvalues are distinct.

Lemma 13.19 (Distinct eigenvalues of a tridiagonal matrix)

An irreducible, tridiagonal and symmetric matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ has n real and distinct eigenvalues.

Proof. Let \mathbf{A} be given by (13.6). By Theorem 5.39 the eigenvalues are real. Define for $x \in \mathbb{R}$ the polynomial $p_k(x) := \det(x\mathbf{I}_k - \mathbf{A}_k)$ for $k = 1, \dots, n$, where \mathbf{A}_k is the upper left $k \times k$ corner of \mathbf{A} (the leading principal submatrix of order k). The eigenvalues of \mathbf{A} are the roots of the polynomial p_n . Using the last column to expand for $k \geq 2$ the determinant $p_{k+1}(x)$ we find

$$p_{k+1}(x) = (x - d_{k+1})p_k(x) - c_k^2 p_{k-1}(x). \quad (13.7)$$

Since $p_1(x) = x - d_1$ and $p_2(x) = (x - d_2)(x - d_1) - c_1^2$ this also holds for $k = 0, 1$ if we define $p_{-1}(x) = 0$ and $p_0(x) = 1$. For M sufficiently large we have

$$p_2(-M) > 0, \quad p_2(d_1) < 0, \quad p_2(+M) > 0.$$

Since p_2 is continuous there are $y_1 \in (-M, d_1)$ and $y_2 \in (d_1, M)$ such that $p_2(y_1) = p_2(y_2) = 0$. It follows that the root d_1 of p_1 separates the roots of p_2 , so y_1 and y_2 must be distinct. Consider next

$$p_3(x) = (x - d_3)p_2(x) - c_2^2 p_1(x) = (x - d_3)(x - y_1)(x - y_2) - c_2^2(x - d_1).$$

Since $y_1 < d_1 < y_2$ we have for M sufficiently large

$$p_3(-M) < 0, \quad p_3(y_1) > 0, \quad p_3(y_2) < 0, \quad p_3(+M) > 0.$$

Thus the roots x_1, x_2, x_3 of p_3 are separated by the roots y_1, y_2 of p_2 . In the general case suppose for $k \geq 2$ that the roots z_1, \dots, z_{k-1} of p_{k-1} separate the roots y_1, \dots, y_k of p_k . Choose M so that $y_0 := -M < y_1, y_{k+1} := M > y_k$. Then

$$y_0 < y_1 < z_1 < y_2 < z_2 \cdots < z_{k-1} < y_k < y_{k+1}.$$

We claim that for M sufficiently large

$$p_{k+1}(y_j) = (-1)^{k+1-j} |p_{k+1}(y_j)| \neq 0, \text{ for } j = 0, 1, \dots, k+1.$$

This holds for $j = 0, k+1$, and for $j = 1, \dots, k$ since

$$p_{k+1}(y_j) = -c_k^2 p_{k-1}(y_j) = -c_k^2 (y_j - z_1) \cdots (y_j - z_{k-1}).$$

It follows that the roots x_1, \dots, x_{k+1} are separated by the roots y_1, \dots, y_k of p_k and by induction the roots of p_n (the eigenvalues of \mathbf{A}) are distinct. \square

13.5.1 The inertia theorem

We say that two matrices $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n \times n}$ are **congruent** if $\mathbf{A} = \mathbf{E}^* \mathbf{B} \mathbf{E}$ for some nonsingular matrix $\mathbf{E} \in \mathbb{C}^{n \times n}$. By Theorem 5.32 a Hermitian matrix \mathbf{A} is both congruent and similar to a diagonal matrix \mathbf{D} , $\mathbf{U}^* \mathbf{A} \mathbf{U} = \mathbf{D}$ where \mathbf{U} is unitary. The eigenvalues of \mathbf{A} are the diagonal elements of \mathbf{D} . Let $\pi(\mathbf{A})$, $\zeta(\mathbf{A})$ and $v(\mathbf{A})$ denote the number of positive, zero and negative eigenvalues of \mathbf{A} . If \mathbf{A} is Hermitian then all eigenvalues are real and $\pi(\mathbf{A}) + \zeta(\mathbf{A}) + v(\mathbf{A}) = n$.

Theorem 13.20 (Sylvester's inertia theorem)

If $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n \times n}$ are Hermitian and congruent then $\pi(\mathbf{A}) = \pi(\mathbf{B})$, $\zeta(\mathbf{A}) = \zeta(\mathbf{B})$ and $v(\mathbf{A}) = v(\mathbf{B})$.

Proof. Suppose $\mathbf{A} = \mathbf{E}^* \mathbf{B} \mathbf{E}$, where \mathbf{E} is nonsingular. Assume first that \mathbf{A} and \mathbf{B} are diagonal matrices. Suppose $\pi(\mathbf{A}) = k$ and $\pi(\mathbf{B}) = m < k$. We shall show that this leads to a contradiction. Let \mathbf{E}_1 be the upper left $m \times k$ corner of \mathbf{E} . Since $m < k$, we can find a nonzero \mathbf{x} such that $\mathbf{E}_1 \mathbf{x} = \mathbf{0}$ (cf. Lemma 0.16). Let $\mathbf{y}^T = [\mathbf{x}^T, \mathbf{0}^T] \in \mathbb{C}^n$, and $\mathbf{z} = [z_1, \dots, z_n]^T = \mathbf{E} \mathbf{y}$. Then $z_i = 0$ for $i = 1, 2, \dots, m$. If \mathbf{A} has positive eigenvalues $\lambda_1, \dots, \lambda_k$ and \mathbf{B} has eigenvalues μ_1, \dots, μ_n , where $\mu_i \leq 0$ for $i \geq m+1$ then

$$\mathbf{y}^* \mathbf{A} \mathbf{y} = \sum_{i=1}^n \lambda_i |y_i|^2 = \sum_{i=1}^k \lambda_i |x_i|^2 > 0.$$

But

$$\mathbf{y}^* \mathbf{A} \mathbf{y} = \mathbf{y}^* \mathbf{E}^* \mathbf{B} \mathbf{E} \mathbf{y} = \mathbf{z}^* \mathbf{B} \mathbf{z} = \sum_{i=m+1}^n \mu_i |z_i|^2 \leq 0,$$

a contradiction.

We conclude that $\pi(\mathbf{A}) = \pi(\mathbf{B})$ if \mathbf{A} and \mathbf{B} are diagonal. Moreover, $v(\mathbf{A}) = \pi(-\mathbf{A}) = \pi(-\mathbf{B}) = v(\mathbf{B})$ and $\zeta(\mathbf{A}) = n - \pi(\mathbf{A}) - v(\mathbf{A}) = n - \pi(\mathbf{B}) - v(\mathbf{B}) = \zeta(\mathbf{B})$. This completes the proof for diagonal matrices.

Let in the general case \mathbf{U}_1 and \mathbf{U}_2 be unitary matrices such that $\mathbf{U}_1^* \mathbf{A} \mathbf{U}_1 = \mathbf{D}_1$ and $\mathbf{U}_2^* \mathbf{B} \mathbf{U}_2 = \mathbf{D}_2$ where \mathbf{D}_1 and \mathbf{D}_2 are diagonal matrices. Since $\mathbf{A} = \mathbf{E}^* \mathbf{B} \mathbf{E}$, we find $\mathbf{D}_1 = \mathbf{F}^* \mathbf{D}_2 \mathbf{F}$ where $\mathbf{F} = \mathbf{U}_2^* \mathbf{E} \mathbf{U}_1$ is nonsingular. Thus \mathbf{D}_1 and \mathbf{D}_2 are congruent diagonal matrices. But since \mathbf{A} and \mathbf{D}_1 , \mathbf{B} and \mathbf{D}_2 have the same eigenvalues, we find $\pi(\mathbf{A}) = \pi(\mathbf{D}_1) = \pi(\mathbf{D}_2) = \pi(\mathbf{B})$. Similar results hold for ζ and v . \square

Corollary 13.21 (Counting eigenvalues using the LDL* factorization)

Suppose $\mathbf{A} = \text{tridiag}(c_i, d_i, c_i) \in \mathbb{R}^{n \times n}$ is symmetric and that $\alpha \in \mathbb{R}$ is such that $\mathbf{A} - \alpha \mathbf{I}$ has a symmetric LU factorization, i.e. $\mathbf{A} - \alpha \mathbf{I} = \mathbf{L} \mathbf{D} \mathbf{L}^T$ where \mathbf{L} is unit lower triangular and \mathbf{D} is diagonal. Then the number of eigenvalues of \mathbf{A} strictly less than α equals the number of negative diagonal elements in \mathbf{D} . The diagonal elements $d_1(\alpha), \dots, d_n(\alpha)$ in \mathbf{D} can be computed recursively as follows

$$d_1(\alpha) = d_1 - \alpha, \quad d_k(\alpha) = d_k - \alpha - c_{k-1}^2 / d_{k-1}(\alpha), \quad k = 2, 3, \dots, n. \quad (13.8)$$

Proof. Since the diagonal elements in \mathbf{L} in an LU factorization equal the diagonal elements in \mathbf{D} in an \mathbf{LDL}^T factorization we see that the formulas in (13.8) follows immediately from (1.16). Since \mathbf{L} is nonsingular, $\mathbf{A} - \alpha \mathbf{I}$ and \mathbf{D} are congruent. By the previous theorem $v(\mathbf{A} - \alpha \mathbf{I}) = v(\mathbf{D})$, the number of negative diagonal elements in \mathbf{D} . If $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ then $(\mathbf{A} - \alpha \mathbf{I})\mathbf{x} = (\lambda - \alpha)\mathbf{x}$, and $\lambda - \alpha$ is an eigenvalue of $\mathbf{A} - \alpha \mathbf{I}$. But then $v(\mathbf{A} - \alpha \mathbf{I})$ equals the number of eigenvalues of \mathbf{A} which are less than α . \square

Exercise 13.22 (Counting eigenvalues)

Consider the matrix in Exercise 13.5. Determine the number of eigenvalues greater than 4.5.

Exercise 13.23 (Overflow in LDL* factorization)

Let for $n \in \mathbb{N}$

$$\mathbf{A}_n = \begin{bmatrix} 10 & 1 & 0 & \cdots & 0 \\ 1 & 10 & 1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & 1 & 10 & 1 \\ 0 & \cdots & 0 & 1 & 10 \end{bmatrix} \in \mathbb{R}^{n \times n}.$$

- a) Let d_k be the diagonal elements of \mathbf{D} in an LDL^* factorization of \mathbf{A}_n . Show that $5 + \sqrt{24} < d_k \leq 10$, $k = 1, 2, \dots, n$.
- b) Show that $D_n := \det(\mathbf{A}_n) > (5 + \sqrt{24})^n$. Give $n_0 \in \mathbb{N}$ such that your computer gives an overflow when D_{n_0} is computed in floating point arithmetic.

Exercise 13.24 (Simultaneous diagonalization)

(Simultaneous diagonalization of two symmetric matrices by a congruence transformation). Let $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}$ where $\mathbf{A}^T = \mathbf{A}$ and \mathbf{B} is symmetric positive definite. Let $\mathbf{B} = \mathbf{U}^T \mathbf{D} \mathbf{U}$ where \mathbf{U} is orthonormal and $\mathbf{D} = \text{diag}(d_1, \dots, d_n)$. Let $\hat{\mathbf{A}} = \mathbf{D}^{-1/2} \mathbf{U} \mathbf{A} \mathbf{U}^T \mathbf{D}^{-1/2}$ where

$$\mathbf{D}^{-1/2} := \text{diag}(d_1^{-1/2}, \dots, d_n^{-1/2}).$$

- a) Show that $\hat{\mathbf{A}}$ is symmetric.

Let $\hat{\mathbf{A}} = \hat{\mathbf{U}}^T \hat{\mathbf{D}} \hat{\mathbf{U}}$ where $\hat{\mathbf{U}}$ is orthonormal and $\hat{\mathbf{D}}$ is diagonal. Set $\mathbf{E} = \mathbf{U}^T \mathbf{D}^{-1/2} \hat{\mathbf{U}}^T$.

- b) Show that \mathbf{E} is nonsingular and that $\mathbf{E}^T \mathbf{A} \mathbf{E} = \hat{\mathbf{D}}$, $\mathbf{E}^T \mathbf{B} \mathbf{E} = \mathbf{I}$.

For a more general result see Theorem 10.1 in [19].

13.5.2 Approximating λ_m

Corollary 13.21 can be used to determine the m th eigenvalue of \mathbf{A} , where $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$. Using Gershgorin's theorem we first find an interval $[a, b]$, such that (a, b) contains the eigenvalues of \mathbf{A} . Let for $x \in [a, b]$

$$\rho(x) := \#\{k : d_k(x) > 0 \text{ for } k = 1, \dots, n\}$$

be the number of eigenvalues of \mathbf{A} which are strictly greater than x . Clearly $\rho(a) = n$, $\rho(b) = 0$. Choosing a tolerance ϵ and using bisection we proceed as

follows:

```

 $h = b - a;$ 
 $for j = 1 : itmax$ 
 $c = (a + b)/2;$ 
 $if b - a < eps * h$ 
 $\lambda = (a + b)/2; return$ 
 $end$ 
 $k = \rho(c);$ 
 $if k \geq m a = c else b = c;$ 
 $end$ 
```

(13.9)

We generate a sequence $\{[a_j, b_j]\}$ of intervals, each containing λ_m and $b_j - a_j = 2^{-j}(b - a)$.

As it stands this method will fail if in (13.8) one of the $d_k(\alpha)$ is zero. One possibility is to replace such a $d_k(\alpha)$ by a suitable small number, say $\delta_k = c_k \epsilon_M$, where ϵ_M is the Machine epsilon, typically 2×10^{-16} for Matlab. This replacement is done if $|d_k(\alpha)| < |\delta_k|$.

Exercise 13.25 (Program code for one eigenvalue)

Suppose $\mathbf{A} = \text{tridiag}(\mathbf{c}, \mathbf{d}, \mathbf{c})$ is symmetric and tridiagonal with elements d_1, \dots, d_n on the diagonal and c_1, \dots, c_{n-1} on the neighboring subdiagonals. Let $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ be the eigenvalues of \mathbf{A} . We shall write a program to compute one eigenvalue λ_m for a given m using bisection and the method outlined in (13.9).

- a) Write a function $k=\text{count}(\mathbf{c}, \mathbf{d}, x)$ which for given x counts the number of eigenvalues of \mathbf{A} strictly greater than x . Use the replacement described above if one of the $d_j(x)$ is close to zero.
- b) Write a function $\lambda = \text{findeigv}(\mathbf{c}, \mathbf{d}, m)$ which first estimates an interval $(a, b]$ containing all eigenvalues of \mathbf{A} and then generates a sequence $\{[a_j, b_j]\}$ of intervals each containing λ_m . Iterate until $b_j - a_j \leq (b - a)\epsilon_M$, where ϵ_M is Matlab's machine epsilon eps . Typically $\epsilon_M \approx 2.22 \times 10^{-16}$.
- c) Test the program on $\mathbf{T} := \text{tridiag}(-1, 2, -1)$ of size 100. Compare the exact value of λ_5 with your result and the result obtained by using Matlab's built-in function eig .

Exercise 13.26 (Determinant of upper Hessenberg matrix)

Suppose $\mathbf{A} \in \mathbb{C}^{n \times n}$ is upper Hessenberg and $x \in \mathbb{C}$. We will study two algorithms to compute $f(x) = \det(\mathbf{A} - x\mathbf{I})$.

- a) Show that Gaussian elimination without pivoting requires $O(n^2)$ arithmetic operations.
- b) Show that the number of arithmetic operations is the same if partial pivoting is used.
- c) Estimate the number of arithmetic operations if Given's rotations are used.
- d) Compare the two methods discussing advantages and disadvantages.

13.6 Review Questions

13.6.1 Suppose $\mathbf{A}, \mathbf{E} \in \mathbb{C}^{n \times n}$. To every $\mu \in \sigma(\mathbf{A} + \mathbf{E})$ there is a $\lambda \in \sigma(\mathbf{A})$ which is in some sense close to μ .

- What is the general result (Elsner's theorem)?
- what if \mathbf{A} is non defective?
- what if \mathbf{A} is normal?
- what if \mathbf{A} is Hermitian?

13.6.2 Can Gershgorin's theorem be used to check if a matrix is nonsingular?

13.6.3 How many arithmetic operation does it take to reduce a matrix by similarity transformations to upper Hessenberg form by Householder transformations?

13.6.4 Give a condition ensuring that a tridiagonal symmetric matrix has real and distinct eigenvalues:

13.6.5 What is the content of Sylvester's inertia theorem?

13.6.6 Give an application of this theorem.

Chapter 14

The QR Algorithm

The QR algorithm is a method to find all eigenvalues and eigenvectors of a matrix. It is related to a simpler method called the power method and we start studying this method and its variants.



Vera Nikolaevna Kublanovskaya, 1920-2012.

14.1 The Power Method and its variants

These methods can be used to compute a single eigenpair of a matrix. They also play a role when studying properties of the QR algorithm.

14.1.1 The power method

The **power method** in its basic form is a technique to compute the eigenvector corresponding to the largest (in absolute value) eigenvalue of a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$. As a by product we can also find the corresponding eigenvalue. We define a sequence $\{\mathbf{z}_k\}$ of vectors in \mathbb{C}^n by

$$\mathbf{z}_k := \mathbf{A}^k \mathbf{z}_0 = \mathbf{A} \mathbf{z}_{k-1}, \quad k = 1, 2, \dots \quad (14.1)$$

Example 14.1 (Power method)

Let

$$\mathbf{A} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}, \quad \mathbf{z}_0 := \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$

We find

$$\mathbf{z}_1 = \mathbf{A} \mathbf{z}_0 = \begin{bmatrix} 2 \\ -1 \end{bmatrix}, \quad \mathbf{z}_2 = \mathbf{A} \mathbf{z}_1 = \begin{bmatrix} 5 \\ -4 \end{bmatrix}, \quad \dots, \quad \mathbf{z}_k = \frac{1}{2} \begin{bmatrix} 1 + 3^k \\ 1 - 3^k \end{bmatrix}, \quad \dots$$

It follows that $2\mathbf{z}_k/3^k$ converges to the eigenvector $[1, -1]$ corresponding to the dominant eigenvalue $\lambda = 3$. The sequence of Rayleigh quotients $\{\mathbf{z}_k^T \mathbf{A} \mathbf{z}_k / \mathbf{z}_k^T \mathbf{z}_k\}$ will converge to the dominant eigenvalue $\lambda = 3$.

To understand better what happens we expand \mathbf{z}_0 in terms of the eigenvectors

$$\mathbf{z}_0 = \frac{1}{2} \begin{bmatrix} 1 \\ -1 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2.$$

Since \mathbf{A}^k has eigenpairs $(\lambda_j^k, \mathbf{v}_j)$, $j = 1, 2$ we find

$$\mathbf{z}_k = c_1 \lambda_1^k \mathbf{v}_1 + c_2 \lambda_2^k \mathbf{v}_2 = c_1 3^k \mathbf{v}_1 + c_2 1^k \mathbf{v}_2.$$

Thus $3^{-k} \mathbf{z}_k = c_1 \mathbf{v}_1 + 3^{-k} c_2 \mathbf{v}_2 \rightarrow c_1 \mathbf{v}_1$. Since $c_1 \neq 0$ we obtain convergence to the dominant eigenvector.

Let $\mathbf{A} \in \mathbb{C}^{n \times n}$ have eigenpairs $(\lambda_j, \mathbf{v}_j)$, $j = 1, \dots, n$ with $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$.

Given $\mathbf{z}_0 \in \mathbb{C}^n$ we assume that

- (i) $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n|$,
 - (ii) $\mathbf{z}_0^T \mathbf{v}_1 \neq 0$
 - (iii) \mathbf{A} has linearly independent eigenvectors.
- (14.2)

The first assumption means that \mathbf{A} has a dominant eigenvalue λ_1 of algebraic multiplicity one. The second assumption says that \mathbf{z}_0 has a component in the

direction \mathbf{v}_1 . The third assumption is not necessary, but is included in order to simplify the analysis.

To see what happens let $\mathbf{z}_0 = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \cdots + c_n \mathbf{v}_n$, where by assumption (ii) of (14.2) we have $c_1 \neq 0$. Since $\mathbf{A}^k \mathbf{v}_j = \lambda_j^k \mathbf{v}_j$ for all j we see that

$$\mathbf{z}_k = c_1 \lambda_1^k \mathbf{v}_1 + c_2 \lambda_2^k \mathbf{v}_2 + \cdots + c_n \lambda_n^k \mathbf{v}_n, \quad k = 0, 1, 2, \dots \quad (14.3)$$

Dividing by λ_1^k we find

$$\frac{\mathbf{z}_k}{\lambda_1^k} = c_1 \mathbf{v}_1 + c_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k \mathbf{v}_2 + \cdots + c_n \left(\frac{\lambda_n}{\lambda_1} \right)^k \mathbf{v}_n, \quad k = 0, 1, 2, \dots \quad (14.4)$$

Assumption (i) of (14.2) implies that $(\lambda_j / \lambda_1)^k \rightarrow 0$ as $k \rightarrow \infty$ for all $j \geq 2$ and we obtain

$$\lim_{k \rightarrow \infty} \frac{\mathbf{z}_k}{\lambda_1^k} = c_1 \mathbf{v}_1, \quad (14.5)$$

the dominant eigenvector of \mathbf{A} . It can be shown that this also holds for defective matrices as long as (i) and (ii) of (14.2) hold, see for example page 58 of [29].

In practice we need to scale the iterates \mathbf{z}_k somehow and we normally do not know λ_1 . Instead we choose a norm on \mathbb{C}^n , set $\mathbf{x}_0 = \mathbf{z}_0 / \|\mathbf{z}_0\|$ and generate for $k = 1, 2, \dots$ unit vectors as follows:

$$\begin{aligned} (i) \quad & \mathbf{y}_k = \mathbf{A} \mathbf{x}_{k-1} \\ (ii) \quad & \mathbf{x}_k = \mathbf{y}_k / \|\mathbf{y}_k\|. \end{aligned} \quad (14.6)$$

Lemma 14.2 (Convergence of the power method)

Suppose (14.2) holds. Then

$$\lim_{k \rightarrow \infty} \left(\frac{|\lambda_1|}{\lambda_1} \right)^k \mathbf{x}_k = \frac{c_1}{|c_1|} \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|}.$$

In particular, if $\lambda_1 > 0$ and $c_1 > 0$ then the sequence $\{\mathbf{x}_k\}$ will converge to the eigenvector $\mathbf{u}_1 := \mathbf{v}_1 / \|\mathbf{v}_1\|$ of unit length.

Proof. By induction on k it follows that $\mathbf{x}_k = \mathbf{z}_k / \|\mathbf{z}_k\|$ for all $k \geq 0$, where $\mathbf{z}_k = \mathbf{A}^k \mathbf{z}_0$. Indeed, this holds for $k = 1$, and if it holds for $k - 1$ then $\mathbf{y}_k = \mathbf{A} \mathbf{x}_{k-1} = \mathbf{A} \mathbf{z}_{k-1} / \|\mathbf{z}_{k-1}\| = \mathbf{z}_k / \|\mathbf{z}_{k-1}\|$ and $\mathbf{x}_k = (\mathbf{z}_k / \|\mathbf{z}_{k-1}\|)(\|\mathbf{z}_{k-1}\| / \|\mathbf{z}_k\|) = \mathbf{z}_k / \|\mathbf{z}_k\|$. But then

$$\mathbf{x}_k = \frac{\mathbf{z}_k}{\|\mathbf{z}_k\|} = \frac{c_1 \lambda_1^k}{|c_1 \lambda_1^k|} \frac{\mathbf{v}_1 + \frac{c_2}{c_1} \left(\frac{\lambda_2}{\lambda_1} \right)^k \mathbf{v}_2 + \cdots + \frac{c_n}{c_1} \left(\frac{\lambda_n}{\lambda_1} \right)^k \mathbf{v}_n}{\left\| \mathbf{v}_1 + \frac{c_2}{c_1} \left(\frac{\lambda_2}{\lambda_1} \right)^k \mathbf{v}_2 + \cdots + \frac{c_n}{c_1} \left(\frac{\lambda_n}{\lambda_1} \right)^k \mathbf{v}_n \right\|}, \quad k = 0, 1, 2, \dots,$$

and this implies the lemma. \square

Suppose we know an approximate eigenvector \mathbf{u} of \mathbf{A} , but not the corresponding eigenvalue μ . One way of estimating μ is to minimize the Euclidian norm of the residual $r(\lambda) := \mathbf{A}\mathbf{u} - \lambda\mathbf{u}$.

Theorem 14.3 (The Rayleigh quotient minimizes the residual)

Let $\mathbf{A} \in \mathbb{C}^{n \times n}$, $\mathbf{u} \in \mathbb{C}^n \setminus \{\mathbf{0}\}$, and let $\rho : \mathbb{C} \rightarrow \mathbb{R}$ be given by $\rho(\lambda) = \|\mathbf{A}\mathbf{u} - \lambda\mathbf{u}\|_2$. Then ρ is minimized when $\lambda := \frac{\mathbf{u}^* \mathbf{A} \mathbf{u}}{\mathbf{u}^* \mathbf{u}}$, the Rayleigh quotient for \mathbf{A} .

Proof. Assume $\mathbf{u}^* \mathbf{u} = 1$ and extend \mathbf{u} to an orthonormal basis $\{\mathbf{u}, \mathbf{U}\}$ for \mathbb{C}^n . Then $\mathbf{U}^* \mathbf{u} = \mathbf{0}$ and

$$\begin{bmatrix} \mathbf{u}^* \\ \mathbf{U}^* \end{bmatrix} (\mathbf{A}\mathbf{u} - \lambda\mathbf{u}) = \begin{bmatrix} \mathbf{u}^* \mathbf{A} \mathbf{u} - \lambda \mathbf{u}^* \mathbf{u} \\ \mathbf{U}^* \mathbf{A} \mathbf{u} - \lambda \mathbf{U}^* \mathbf{u} \end{bmatrix} = \begin{bmatrix} \mathbf{u}^* \mathbf{A} \mathbf{u} - \lambda \\ \mathbf{U}^* \mathbf{A} \mathbf{u} \end{bmatrix}.$$

By unitary invariance of the Euclidian norm

$$\rho(\lambda)^2 = |\mathbf{u}^* \mathbf{A} \mathbf{u} - \lambda|^2 + \|\mathbf{U}^* \mathbf{A} \mathbf{u}\|_2^2,$$

and ρ has a global minimum at $\lambda = \mathbf{u}^* \mathbf{A} \mathbf{u}$. \square

Exercise 14.4 (Orthogonal vectors)

Show that \mathbf{u} and $\mathbf{A}\mathbf{u} - \lambda\mathbf{u}$ are orthogonal when $\lambda = \frac{\mathbf{u}^* \mathbf{A} \mathbf{u}}{\mathbf{u}^* \mathbf{u}}$.

Using Rayleigh quotients we can incorporate the calculation of the eigenvalue into the power iteration. We can then compute the residual and stop the iteration when the residual is sufficiently small. But what does it mean to be sufficiently small? Recall that if \mathbf{A} is nonsingular with a nonsingular eigenvector matrix \mathbf{X} and (μ, \mathbf{u}) is an approximate eigenpair with $\|\mathbf{u}\|_2 = 1$, then by (13.4) we can find an eigenvalue λ of \mathbf{A} such that

$$\frac{|\lambda - \mu|}{|\lambda|} \leq K_2(\mathbf{X})K_2(\mathbf{A}) \frac{\|\mathbf{A}\mathbf{u} - \mu\mathbf{u}\|_2}{\|\mathbf{A}\|_2}.$$

Thus if the relative residual is small and both \mathbf{A} and \mathbf{X} are well conditioned then the relative error in the eigenvalue will be small.

This discussion leads to the power method with Rayleigh quotient computation. Given $\mathbf{A} \in \mathbb{C}^{n \times n}$, a starting vector $\mathbf{z} \in \mathbb{C}^n$, a maximum number K of iterations, and a convergence tolerance tol . The power method combined with a Rayleigh quotient estimate for the eigenvalue is used to compute a dominant eigenpair (l, \mathbf{x}) of \mathbf{A} with $\|\mathbf{x}\|_2 = 1$. The integer it returns the number of iterations needed in order for $\|\mathbf{Ax} - l\mathbf{x}\|_2 / \|\mathbf{A}\|_F < tol$. If no such eigenpair is found in K iterations the value $it = K + 1$ is returned.

Algorithm 14.5 (The power method)

```

1 function [l,x,it]=powerit(A,z,K,tol)
2 af=norm(A,'fro'); x=z/norm(z);
3 for k=1:K
4     y=A*x; l=x'*y;
5     if norm(y-l*x)/af<tol
6         it=k; x=y/norm(y); return
7     end
8     x=y/norm(y);
9 end
10 it=K+1;

```

Example 14.6 (Power method)

We try powerit on the three matrices

$$\mathbf{A}_1 := \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad \mathbf{A}_2 := \begin{bmatrix} 1.7 & -0.4 \\ 0.15 & 2.2 \end{bmatrix}, \text{ and } \mathbf{A}_3 = \begin{bmatrix} 1 & 2 \\ -3 & 4 \end{bmatrix}.$$

In each case we start with the random vector $\mathbf{z} = [0.6602, 0.3420]$ and $\text{tol} = 10^{-6}$. For \mathbf{A}_1 we get convergence in 7 iterations, for \mathbf{A}_2 it takes 174 iterations, and for \mathbf{A}_3 we do not get convergence.

The matrix \mathbf{A}_3 does not have a dominant eigenvalue since the two eigenvalues are complex conjugate of each other. Thus the basic condition (i) of (14.2) is not satisfied and the power method diverges. The enormous difference in the rate of convergence for \mathbf{A}_1 and \mathbf{A}_2 can be explained by looking at (14.4). The rate of convergence depends on the ratio $\frac{|\lambda_2|}{|\lambda_1|}$. If this ratio is small then the convergence is fast, while it can be quite slow if the ratio is close to one. The eigenvalues of \mathbf{A}_1 are $\lambda_1 = 5.3723$ and $\lambda_2 = -0.3723$ giving a quite small ratio of 0.07 and the convergence is fast. On the other hand the eigenvalues of \mathbf{A}_2 are $\lambda_1 = 2$ and $\lambda_2 = 1.9$ and the corresponding ratio is 0.95 resulting in slow convergence.

A variant of the power method is the **shifted power method**. In this method we choose a number s and apply the power method to the matrix $\mathbf{A} - s\mathbf{I}$. The number s is called a shift since it shifts an eigenvalue λ of \mathbf{A} to $\lambda - s$ of $\mathbf{A} - s\mathbf{I}$. Sometimes the convergence can be faster if the shift is chosen intelligently. For example, if we apply the shifted power method to \mathbf{A}_2 in Example 14.6 with shift 1.8, then with the same starting vector and tol as above, we get convergence in 17 iterations instead of 174 for the unshifted algorithm.

14.1.2 The inverse power method

Another variant of the power method with Rayleigh quotient is the **inverse power method**. This method can be used to determine any eigenpair (λ, \mathbf{x}) of \mathbf{A} as long

as λ has algebraic multiplicity one. In the inverse power method we apply the power method to the inverse matrix $(\mathbf{A} - s\mathbf{I})^{-1}$, where s is a shift. If \mathbf{A} has eigenvalues $\lambda_1, \dots, \lambda_n$ in no particular order then $(\mathbf{A} - s\mathbf{I})^{-1}$ has eigenvalues

$$\mu_1(s) = (\lambda_1 - s)^{-1}, \mu_2(s) = (\lambda_2 - s)^{-1}, \dots, \mu_n(s) = (\lambda_n - s)^{-1}.$$

Suppose λ_1 is a simple eigenvalue of \mathbf{A} . Then $\lim_{s \rightarrow \lambda_1} |\mu_1(s)| = \infty$, while $\lim_{s \rightarrow \lambda_1} \mu_j(s) = (\lambda_j - \lambda_1)^{-1} < \infty$ for $j = 2, \dots, n$. Hence, by choosing s sufficiently close to λ_1 the inverse power method will converge to that eigenvalue.

For the inverse power method (14.6) is replaced by

$$\begin{aligned} (i) \quad & (\mathbf{A} - s\mathbf{I})\mathbf{y}_k = \mathbf{x}_{k-1} \\ (ii) \quad & \mathbf{x}_k = \mathbf{y}_k / \|\mathbf{y}_k\|. \end{aligned} \tag{14.7}$$

Note that we solve the linear system rather than computing the inverse matrix. Normally the PLU factorization of $\mathbf{A} - s\mathbf{I}$ is precomputed in order to speed up the computation.

14.1.3 Rayleigh quotient iteration

A variant of the inverse power method is known simply as **Rayleigh quotient iteration**. In this method we change the shift from iteration to iteration, using the previous Rayleigh quotient s_{k-1} as the current shift. In each iteration we need to compute the following quantities

$$\begin{aligned} (i) \quad & (\mathbf{A} - s_{k-1}\mathbf{I})\mathbf{y}_k = \mathbf{x}_{k-1}, \\ (ii) \quad & \mathbf{x}_k = \mathbf{y}_k / \|\mathbf{y}_k\|, \\ (iii) \quad & s_k = \mathbf{x}_k^* \mathbf{A} \mathbf{x}_k, \\ (iv) \quad & \mathbf{r}_k = \mathbf{A} \mathbf{x}_k - s_k \mathbf{x}_k. \end{aligned}$$

We can avoid the calculation of $\mathbf{A} \mathbf{x}_k$ in (iii) and (iv). Let

$$\rho_k := \frac{\mathbf{y}_k^* \mathbf{x}_{k-1}}{\mathbf{y}_k^* \mathbf{y}_k}, \quad \mathbf{w}_k := \frac{\mathbf{x}_{k-1}}{\|\mathbf{y}_k\|_2}.$$

Then

$$\begin{aligned} s_k &= \frac{\mathbf{y}_k^* \mathbf{A} \mathbf{y}_k}{\mathbf{y}_k^* \mathbf{y}_k} = s_{k-1} + \frac{\mathbf{y}_k^* (\mathbf{A} - s_{k-1}\mathbf{I})\mathbf{y}_k}{\mathbf{y}_k^* \mathbf{y}_k} = s_{k-1} + \frac{\mathbf{y}_k^* \mathbf{x}_{k-1}}{\mathbf{y}_k^* \mathbf{y}_k} = s_{k-1} + \rho_k, \\ \mathbf{r}_k &= \mathbf{A} \mathbf{x}_k - s_k \mathbf{x}_k = \frac{\mathbf{A} \mathbf{y}_k - (s_{k-1} + \rho_k) \mathbf{y}_k}{\|\mathbf{y}_k\|_2} = \frac{\mathbf{x}_{k-1} - \rho_k \mathbf{y}_k}{\|\mathbf{y}_k\|_2} = \mathbf{w}_k - \rho_k \mathbf{x}_k. \end{aligned}$$

Another problem is that the linear system in i) becomes closer and closer to singular as s_k converges to the eigenvalue. Thus the system becomes more and

more ill-conditioned and we can expect large errors in the computed \mathbf{y}_k . This is indeed true, but we are lucky. Most of the error occurs in the direction of the eigenvector and this error disappears when we normalize \mathbf{y}_k in *ii*). Miraculously, the normalized eigenvector will be quite accurate.

Given an approximation (s, \mathbf{x}) to an eigenpair (λ, \mathbf{v}) of a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$. The following algorithm computes a hopefully better approximation to (λ, \mathbf{v}) by doing one Rayleigh quotient iteration. The length nr of the new residual is also returned.

Algorithm 14.7 (Rayleigh quotient iteration)

```

1 function [x,s,nr]=rayleight(A,x,s)
2 n=length(x);
3 y=(A-s*eye(n,n))\x;
4 yn=norm(y);
5 w=x/yn;
6 x=y/yn;
7 rho=x'*w;
8 s=s+rho;
9 nr=norm(w-rho*x);

```

Since the shift changes from iteration to iteration the computation of \mathbf{y} in `rayleight` will require $O(n^3)$ arithmetic operations for a full matrix. For such a matrix it might be useful to reduce it to an upper Hessenberg form, or tridiagonal form, before starting the iteration. However, if we have a good approximation to an eigenpair then only a few iterations are necessary to obtain close to machine accuracy.

If Rayleigh quotient iteration converges the convergence will be quadratic and sometimes even cubic. We illustrate this with an example.

Example 14.8 (Rayleigh quotient iteration)

The smallest eigenvalue of the matrix $\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ is $\lambda_1 = (5 - \sqrt{33})/2 \approx -0.37$. Starting with $\mathbf{x} = [1, 1]^T$ and $s = 0$ `rayleight` converges to this eigenvalue and corresponding eigenvector. In Table 14.9 we show the rate of convergence by iterating `rayleight` 5 times. The errors are approximately squared in each iteration indicating quadratic convergence.

14.2 The basic QR Algorithm

The QR algorithm is an iterative method to compute all eigenvalues and eigenvectors of a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$. The matrix is reduced to triangular form by a sequence of unitary similarity transformations computed from the QR factorization of \mathbf{A} . Recall that for a square matrix the QR factorization and the QR

k	1	2	3	4	5
$\ \mathbf{r}\ _2$	1.0e+000	7.7e-002	1.6e-004	8.2e-010	2.0e-020
$ s - \lambda_1 $	3.7e-001	-1.2e-002	-2.9e-005	-1.4e-010	-2.2e-016

Table 14.9: Quadratic convergence of Rayleigh quotient iteration.

decomposition are the same. If $\mathbf{A} = \mathbf{Q}\mathbf{R}$ is a QR factorization then $\mathbf{Q} \in \mathbb{C}^{n \times n}$ is unitary, $\mathbf{Q}^*\mathbf{Q} = \mathbf{I}$ and $\mathbf{R} \in \mathbb{C}^{n \times n}$ is upper triangular.

The basic QR algorithm takes the following form:

$$\begin{aligned} \mathbf{A}_1 &= \mathbf{A} \\ \text{for } k &= 1, 2, \dots \\ \mathbf{Q}_k \mathbf{R}_k &= \mathbf{A}_k \quad (\text{QR factorization of } \mathbf{A}_k) \\ \mathbf{A}_{k+1} &= \mathbf{R}_k \mathbf{Q}_k. \\ \text{end} \end{aligned} \tag{14.8}$$

The determination of the QR factorization of \mathbf{A}_k and the computation of $\mathbf{R}_k \mathbf{Q}_k$ is called a QR step. It is not at all clear that a QR step does anything useful. At this point, since $\mathbf{R}_k = \mathbf{Q}_k^* \mathbf{A}_k$ we find

$$\mathbf{A}_{k+1} = \mathbf{R}_k \mathbf{Q}_k = \mathbf{Q}_k^* \mathbf{A}_k \mathbf{Q}_k, \tag{14.9}$$

so \mathbf{A}_{k+1} is unitary similar to \mathbf{A}_k . By induction \mathbf{A}_{k+1} is unitary similar to \mathbf{A} . Thus, each \mathbf{A}_k has the same eigenvalues as \mathbf{A} . We shall see that the basic QR algorithm is related to the power method.

Here are two examples to illustrate what happens.

Example 14.10 (QR iteration; real eigenvalues)

We start with

$$\mathbf{A}_1 = \mathbf{A} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} = \left(\frac{1}{\sqrt{5}} \begin{bmatrix} 2 & -1 \\ 1 & 2 \end{bmatrix} \right) * \left(\frac{1}{\sqrt{5}} \begin{bmatrix} 5 & 4 \\ 0 & 3 \end{bmatrix} \right) = \mathbf{Q}_1 \mathbf{R}_1$$

and obtain

$$\mathbf{A}_2 = \mathbf{R}_1 \mathbf{Q}_1 = \frac{1}{5} \begin{bmatrix} 5 & 4 \\ 0 & 3 \end{bmatrix} * \begin{bmatrix} 2 & -1 \\ 1 & 2 \end{bmatrix} = \frac{1}{5} \begin{bmatrix} 14 & 3 \\ 3 & 6 \end{bmatrix} = \begin{bmatrix} 2.8 & 0.6 \\ 0.6 & 1.2 \end{bmatrix}.$$

Continuing we find

$$\mathbf{A}_4 \approx \begin{bmatrix} 2.997 & -0.074 \\ -0.074 & 1.0027 \end{bmatrix}, \quad \mathbf{A}_{10} \approx \begin{bmatrix} 3.0000 & -0.0001 \\ -0.0001 & 1.0000 \end{bmatrix}$$

\mathbf{A}_{10} is almost diagonal and contains approximations to the eigenvalues $\lambda_1 = 3$ and $\lambda_2 = 1$ on the diagonal.

Example 14.11 (QR iteration; complex eigenvalues)

Applying the QR iteration (14.8) to the matrix

$$A_1 = A = \begin{bmatrix} 0.9501 & 0.8913 & 0.8214 & 0.9218 \\ 0.2311 & 0.7621 & 0.4447 & 0.7382 \\ 0.6068 & 0.4565 & 0.6154 & 0.1763 \\ 0.4860 & 0.0185 & 0.7919 & 0.4057 \end{bmatrix}$$

we obtain

$$A_{14} = \left[\begin{array}{c|cc|c} 2.323 & 0.047223 & -0.39232 & -0.65056 \\ \hline -2.1e-10 & 0.13029 & 0.36125 & 0.15946 \\ -4.1e-10 & -0.58622 & 0.052576 & -0.25774 \\ \hline 1.2e-14 & 3.3e-05 & -1.1e-05 & 0.22746 \end{array} \right].$$

This matrix is almost quasi-triangular and estimates for the eigenvalues $\lambda_1, \dots, \lambda_4$ of A can now easily be determined from the diagonal blocks of A_{14} . The 1×1 blocks give us two real eigenvalues $\lambda_1 \approx 2.323$ and $\lambda_4 \approx 0.2275$. The middle 2×2 block has complex eigenvalues resulting in $\lambda_2 \approx 0.0914 + 0.4586i$ and $\lambda_3 \approx 0.0914 - 0.4586i$. From Gershgorin's circle theorem 13.1 and Corollary 13.3 it follows that the approximations to the real eigenvalues are quite accurate. We would also expect the complex eigenvalues to have small absolute errors.

These two examples illustrate what happens in general. The sequence $(A_k)_k$ converges to the triangular Schur form (Cf. Theorem 5.28) if all the eigenvalues are real or the quasi-triangular Schur form (Cf. Definition 5.37) if some of the eigenvalues are complex.

14.2.1 Relation to the power method

Let us show that the basic QR algorithm is related to the power method. We obtain the QR factorization of the powers A^k as follows:

Theorem 14.12 (QR and power)

For $k = 1, 2, 3, \dots$, the QR factorization of A^k is $A^k = \tilde{Q}_k \tilde{R}_k$, where

$$\tilde{Q}_k := Q_1 \cdots Q_k \text{ and } \tilde{R}_k := R_k \cdots R_1, \quad (14.10)$$

and $Q_1, \dots, Q_k, R_1, \dots, R_k$ are the matrices generated by the basic QR algorithm (14.8).

Proof. By (14.9)

$$A_k = Q_{k-1}^* A_{k-1} Q_{k-1} = Q_{k-1}^* Q_{k-2}^* A_{k-2} Q_{k-2} Q_{k-1} = \cdots = \tilde{Q}_{k-1}^* A \tilde{Q}_{k-1}. \quad (14.11)$$

$$\mathbf{A} = \begin{bmatrix} x & x & x & x \\ 0 & x & x & x \\ 0 & 0 & x & x \\ 0 & 0 & 0 & x \end{bmatrix} \xrightarrow{\mathbf{P}_{12}^*} \begin{bmatrix} x & x & x & x \\ \mathbf{x} & x & x & x \\ 0 & 0 & x & x \\ 0 & 0 & 0 & x \end{bmatrix} \xrightarrow{\mathbf{P}_{23}^*} \begin{bmatrix} x & x & x & x \\ x & x & x & x \\ 0 & \mathbf{x} & x & x \\ 0 & 0 & 0 & x \end{bmatrix} \xrightarrow{\mathbf{P}_{34}^*} \begin{bmatrix} x & x & x & x \\ x & x & x & x \\ 0 & x & x & x \\ 0 & 0 & \mathbf{x} & x \end{bmatrix}.$$

Figure 14.1: Post multiplication in a QR step.

The proof is by induction on k . Clearly $\tilde{\mathbf{Q}}_1 \tilde{\mathbf{R}}_1 = \mathbf{Q}_1 \mathbf{R}_1 = \mathbf{A}_1$. Suppose $\tilde{\mathbf{Q}}_{k-1} \tilde{\mathbf{R}}_{k-1} = \mathbf{A}^{k-1}$ for some $k \geq 2$. Since $\mathbf{Q}_k \mathbf{R}_k = \mathbf{A}_k$ and using (14.11)

$$\tilde{\mathbf{Q}}_k \tilde{\mathbf{R}}_k = \tilde{\mathbf{Q}}_{k-1} (\mathbf{Q}_k \mathbf{R}_k) \tilde{\mathbf{R}}_{k-1} = \tilde{\mathbf{Q}}_{k-1} \mathbf{A}_k \tilde{\mathbf{R}}_{k-1} = (\tilde{\mathbf{Q}}_{k-1} \tilde{\mathbf{Q}}_{k-1}^*) \mathbf{A} \tilde{\mathbf{Q}}_{k-1} \tilde{\mathbf{R}}_{k-1} = \mathbf{A}^k.$$

□

Since $\tilde{\mathbf{R}}_k$ is upper triangular, its first column is a multiple of \mathbf{e}_1 so that

$$\mathbf{A}^k \mathbf{e}_1 = \tilde{\mathbf{Q}}_k \tilde{\mathbf{R}}_k \mathbf{e}_1 = \tilde{r}_{11}^{(k)} \tilde{\mathbf{Q}}_k \mathbf{e}_1 \text{ or } \tilde{\mathbf{q}}_1^{(k)} := \tilde{\mathbf{Q}}_k \mathbf{e}_1 = \frac{1}{\tilde{r}_{11}^{(k)}} \mathbf{A}^k \mathbf{e}_1.$$

Since $\|\tilde{\mathbf{q}}_1^{(k)}\|_2 = 1$ the first column of $\tilde{\mathbf{Q}}_k$ is the result of applying the normalized power iteration (14.6) to the starting vector $\mathbf{x}_0 = \mathbf{e}_1$. If this iteration converges we conclude that the first column of $\tilde{\mathbf{Q}}_k$ must converge to a dominant eigenvector of \mathbf{A} . It can be shown that the first column of \mathbf{A}_k must then converge to $\lambda_1 \mathbf{e}_1$, where λ_1 is a dominant eigenvalue of \mathbf{A} . This is clearly what happens in Examples 14.10 and 14.11. Indeed, what is observed in practice is that the sequence $(\tilde{\mathbf{Q}}_k^* \mathbf{A} \tilde{\mathbf{Q}}_k)_k$ converges to a (quasi-triangular) Schur form of \mathbf{A} .

14.2.2 Invariance of the Hessenberg form

One QR step requires $O(n^3)$ arithmetic operations for a matrix \mathbf{A} of order n . By an initial reduction of \mathbf{A} to upper Hessenberg form \mathbf{H}_1 using Algorithm 13.14, the cost of a QR step can be reduced to $O(n^2)$. Consider a QR step on \mathbf{H}_1 . We first determine plane rotations $\mathbf{P}_{i,i+1}$, $i = 1, \dots, n-1$ so that $\mathbf{P}_{n-1,n} \cdots \mathbf{P}_{1,2} \mathbf{H}_1 = \mathbf{R}_1$ is upper triangular. The details were described in Section 4.5. Thus $\mathbf{H}_1 = \mathbf{Q}_1 \mathbf{R}_1$, where $\mathbf{Q}_1 = \mathbf{P}_{1,2}^* \cdots \mathbf{P}_{n-1,n}^*$ is a QR factorization of \mathbf{H}_1 . To finish the QR step we compute $\mathbf{R}_1 \mathbf{Q}_1 = \mathbf{R}_1 \mathbf{P}_{1,2}^* \cdots \mathbf{P}_{n-1,n}^*$. This postmultiplication step is illustrated by the Wilkinson diagram in Figure 14.1.

The postmultiplication by $\mathbf{P}_{i,i+1}$ introduces a nonzero in position $(i+1, i)$ leaving the other elements marked by a zero in Figure 14.1 unchanged. Thus the final matrix $\mathbf{R}_1 \mathbf{P}_{1,2}^* \cdots \mathbf{P}_{n-1,n}^*$ is upper Hessenberg and a QR step leaves the Hessenberg form invariant.

In conclusion, to compute \mathbf{A}_{k+1} from \mathbf{A}_k requires $O(n^2)$ arithmetic operations if \mathbf{A}_k is upper Hessenberg and $O(n)$ arithmetic operations if \mathbf{A}_k is tridiagonal.

14.2.3 Deflation

If a subdiagonal element $a_{i+1,i}$ of an upper Hessenberg matrix \mathbf{A} is equal to zero, then the eigenvalues of \mathbf{A} are the union of the eigenvalues of the two smaller matrices $A(1:i, 1:i)$ and $A(i+1:n, i+1:n)$. Thus if during the iteration the $(i+1,i)$ element of \mathbf{A}_k is sufficiently small then we can continue the iteration on the two smaller submatrices separately.

To see what effect this can have on the eigenvalues of \mathbf{A} suppose $|a_{i+1,i}^{(k)}| \leq \epsilon$. Let $\hat{\mathbf{A}}_k := \mathbf{A}_k - a_{i+1,i}^{(k)} \mathbf{e}_{i+1} \mathbf{e}_i^T$ be the matrix obtained from \mathbf{A}_k by setting the $(i+1,i)$ element equal to zero. Since $\mathbf{A}_k = \tilde{\mathbf{Q}}_{k-1}^* \mathbf{A} \tilde{\mathbf{Q}}_{k-1}$ we have

$$\hat{\mathbf{A}}_k = \tilde{\mathbf{Q}}_{k-1}^* (\mathbf{A} + \mathbf{E}) \tilde{\mathbf{Q}}_{k-1}, \quad \mathbf{E} = \tilde{\mathbf{Q}}_{k-1} (a_{i+1,i}^{(k)} \mathbf{e}_{i+1} \mathbf{e}_i^T) \tilde{\mathbf{Q}}_{k-1}^*.$$

Since $\tilde{\mathbf{Q}}_{k-1}$ is unitary, $\|\mathbf{E}\|_F = \|a_{i+1,i}^{(k)} \mathbf{e}_{i+1} \mathbf{e}_i^T\|_F = |a_{i+1,i}^{(k)}| \leq \epsilon$ and setting $a_{i+1,i}^{(k)} = 0$ amounts to a perturbation in the original \mathbf{A} of at most ϵ . For how to chose ϵ see the discussion on page 94-95 in [29].

This deflation occurs often in practice and can with a proper implementation reduce the computation time considerably. It should be noted that to find the eigenvectors of the original matrix one has to continue with some care, see [29].

14.3 The Shifted QR Algorithms

Like in the inverse power method it is possible to speed up the convergence by introducing shifts. The **explicitly shifted QR algorithm** works as follows:

$$\mathbf{A}_1 = \mathbf{A}$$

for $k = 1, 2, \dots$

Choose a shift s_k

$$\mathbf{Q}_k \mathbf{R}_k = \mathbf{A}_k - s_k \mathbf{I} \quad (\text{QR factorization of } \mathbf{A}_k - s_k \mathbf{I})$$

$$\mathbf{A}_{k+1} = \mathbf{R}_k \mathbf{Q}_k + s_k \mathbf{I}.$$

end

Since $\mathbf{R}_k = \mathbf{Q}_k^* (\mathbf{A}_k - s_k \mathbf{I})$ we find

$$\mathbf{A}_{k+1} = \mathbf{Q}_k^* (\mathbf{A}_k - s_k \mathbf{I}) \mathbf{Q}_k + s_k \mathbf{I} = \mathbf{Q}_k^* \mathbf{A}_k \mathbf{Q}_k$$

and \mathbf{A}_{k+1} and \mathbf{A}_k are unitary similar.

The shifted QR algorithm is related to the power method with shift, cf. Theorem 14.12 and also the inverse power method. In fact the last column of \mathbf{Q}_k is the result of one iteration of the inverse power method to \mathbf{A}^* with shift s_k . Indeed, since $\mathbf{A} - s_k \mathbf{I} = \mathbf{Q}_k \mathbf{R}_k$ we have $(\mathbf{A} - s_k \mathbf{I})^* = \mathbf{R}_k^* \mathbf{Q}_k^*$ and $(\mathbf{A} - s_k \mathbf{I})^* \mathbf{Q}_k = \mathbf{R}_k^*$.

Thus, since \mathbf{R}_k^* is lower triangular with n, n element $\bar{r}_{nn}^{(k)}$ we find $(\mathbf{A} - s_k \mathbf{I})^* \mathbf{Q}_k \mathbf{e}_n = \mathbf{R}_k^* \mathbf{e}_n = \bar{r}_{nn}^{(k)} \mathbf{e}_n$ from which the conclusion follows.

The shift $s_k := \mathbf{e}_n^T \mathbf{A}_k \mathbf{e}_n$ is called the **Rayleigh quotient shift**, while the eigenvalue of the lower right 2×2 corner of \mathbf{A}_k closest to the n, n element of \mathbf{A}_k is called the **Wilkinson shift**. This shift can be used to find complex eigenvalues of a real matrix. The convergence is very fast and at least quadratic both for the Rayleigh quotient shift and the Wilkinson shift.

By doing two QR iterations at a time it is possible to find both real and complex eigenvalues without using complex arithmetic. The corresponding algorithm is called the **implicitly shifted QR algorithm**

After having computed the eigenvalues we can compute the eigenvectors in steps. First we find the eigenvectors of the triangular or quasi-triangular matrix. We then compute the eigenvectors of the upper Hessenberg matrix and finally we get the eigenvectors of \mathbf{A} .

Practical experience indicates that only $O(n)$ iterations are needed to find all eigenvalues of \mathbf{A} . Thus both the explicit- and implicit shift QR algorithms are normally $O(n^3)$ algorithms.

For further remarks and detailed algorithms see [29].

14.4 Review Questions

14.4.1 What is the main use of the power method?

14.4.2 Can the QR method be used to find all eigenvectors of a matrix?

14.4.3 Can the power method be used to find an eigenvalue?

14.4.4 Do the power method converge to an eigenvector corresponding to a complex eigenvalue?

14.4.5 What is the inverse power method?

14.4.6 Give a relation between the QR algorithm and the power method.

14.4.7 How can we make the basic QR algorithm converge faster?

Part VI

Appendix

Appendix A

Computer Arithmetic

A.1 Absolute and Relative Errors

Suppose a and b are real or complex scalars. If b is an approximation to a then there are different ways of measuring the error in b .

Definition A.1 (Absolute error)

The absolute error in b as an approximation to a is the number $\epsilon := |a - b|$. The number $e := b - a$ is called the error in b as an approximation to a . This is what we have to add to a to get b .

Note that the absolute error is symmetric in a and b , so that ϵ is also the absolute error in a as an approximation to b

Definition A.2 (Relative error) If $a \neq 0$ then the relative error in b as an approximation to a is defined by

$$\rho = \rho_b := \frac{|b - a|}{|a|}.$$

We say that a and b agree to approximately $-\log_{10} \rho$ digits.

As an example, if $a := 31415.9265$ and $b := 31415.8951$, then $\rho = 0.999493 * 10^{-6}$ and a and b agree to approximately 6 digits.

We have $b = a(1 + r)$ for some r if and only if $\rho = |r|$.

We can also consider the relative error $\rho_a := |a - b|/|b|$ in a as an approximation to b .

Lemma A.3 (Relative errors)

If $a, b \neq 0$ and $\rho_b < 1$ then $\rho_a \leq \rho_b/(1 - \rho_b)$.

Proof. Since $|a|\rho_b = |b - a| \geq |a| - |b|$ we obtain $|b| \geq |a| - |a - b| = (1 - \rho_b)|a|$. Then

$$\rho_a = \frac{|b - a|}{|b|} \leq \frac{|b - a|}{(1 - \rho_b)|a|} = \frac{\rho_b}{1 - \rho_b}.$$

□

If ρ_b is small then ρ_a is small and it does not matter whether we choose ρ_a or ρ_b to discuss relative error.

A.2 Floating Point Numbers

We shall assume that the reader is familiar with different number systems (binary, octal, decimal, hexadecimal) and how to convert from one number system to another. We use $(x)_\beta$ to indicate a number written to the base β . If no parenthesis and subscript are used, the base 10 is understood. For instance,

$$\begin{aligned}(100)_2 &= 4, \\ (0.1)_2 &= 0.5, \\ 0.1 &= (0.1)_{10} = (0.0001100110011001\dots)_2.\end{aligned}$$

In general,

$$x = (c_m c_{m-1} \dots c_0.d_1 d_2 \dots d_n)_\beta$$

means

$$x = \sum_{i=0}^m c_i \beta^i + \sum_{i=1}^n d_i \beta^{-i}, \quad 0 \leq c_i, d_i \leq \beta - 1.$$

We can move the decimal point by adding an exponent:

$$y = x \cdot \beta^e,$$

for example

$$(0.1)_{10} = (1.100110011001\dots)_2 \cdot 2^{-4}.$$

We turn now to a description of the floating-point numbers. We will only describe a **standard system**, namely the binary IEEE floating-point standard. Although it is not used by all systems, it has been widely adopted and is used in MATLAB. For a more complete introduction to the subject see [13],[28].

We denote the real numbers which are represented in our computer by \mathcal{F} . The set \mathcal{F} are characterized by three integers t , and \underline{e}, \bar{e} . We define

$$\epsilon_M := 2^{-t}, \quad \text{machine epsilon,} \tag{A.1}$$

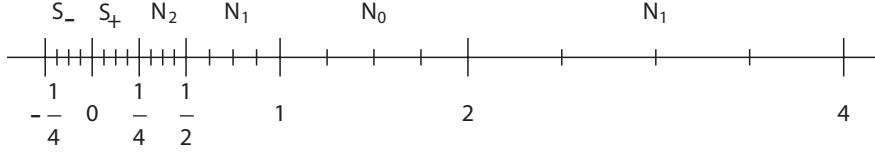


Figure A.1: Distribution of some positive floating-point numbers

and

$$\begin{aligned} \mathcal{F} &:= \{0\} \cup \mathcal{S} \cup \mathcal{N}, \text{ where} \\ \mathcal{N} &:= \mathcal{N}_+ \cup \mathcal{N}_-, \quad \mathcal{N}_+ := \cup_{e=\underline{e}}^{\bar{e}} \mathcal{N}_e, \quad \mathcal{N}_- := -\mathcal{N}_+, \\ \mathcal{N}_e &:= \{(1.d_1d_2 \cdots d_t)_2 * 2^e = \{1, 1 + \epsilon_M, 1 + 2\epsilon_M, \dots, 2 - \epsilon_M\} * 2^e, \\ \mathcal{S} &:= \mathcal{S}_+ \cup \mathcal{S}_-, \quad \mathcal{S}_+ := \{\epsilon_M, 2\epsilon_M, 3\epsilon_M, \dots, 1 - \epsilon_M\} * 2^{\underline{e}}, \quad \mathcal{S}_- := -\mathcal{S}_+. \end{aligned} \quad (\text{A.2})$$

Example A.4 (Floating numbers)

Suppose $t := 2$, $\bar{e} = 3$ and $\underline{e} := -2$. Then $\epsilon_M = 1/4$ and we find

$$\begin{aligned} \mathcal{N}_{-2} &= \left\{ \frac{1}{4}, \frac{5}{16}, \frac{3}{8}, \frac{7}{16} \right\}, \quad \mathcal{N}_{-1} = \left\{ \frac{1}{2}, \frac{5}{8}, \frac{3}{4}, \frac{7}{8} \right\}, \quad \mathcal{N}_0 = \left\{ 1, \frac{5}{4}, \frac{3}{2}, \frac{3}{4}, \frac{7}{4} \right\}, \\ \mathcal{N}_1 &= \left\{ 2, \frac{5}{2}, 3, \frac{7}{2} \right\}, \quad \mathcal{N}_2 = \{4, 5, 6, 7\}, \quad \mathcal{N}_3 = \{8, 10, 12, 14\}, \\ \mathcal{S}_+ &= \left\{ \frac{1}{16}, \frac{1}{8}, \frac{3}{16} \right\}, \quad \mathcal{S}_- = \left\{ -\frac{3}{16}, -\frac{1}{8}, -\frac{1}{16} \right\}. \end{aligned}$$

The position of some of these sets on the real line is shown in Figure A.1

1. The elements of \mathcal{N} are called **normalized (floating-point) numbers**. They consists of three parts, the sign +1 or -1, the **mantissa** $(1.d_1d_2 \cdots d_t)_2$, and the **exponent part** 2^e .
2. the elements in \mathcal{N}_+ has the sign +1 indicated by the bit $\sigma = 0$ and the elements in \mathcal{N}_- has the sign bit $\sigma = 1$. Thus the sign of a number is $(-1)^\sigma$. The standard system has two zeros +0 and -0.
3. The mantissa is a number between 1 and 2. It consists of $t + 1$ binary digits.
4. The number e in the exponent part is restricted to the range $\underline{e} \leq e \leq \bar{e}$.
5. The positive normalized numbers are located in the interval $[r_m, r_M]$, where

$$r_m := 2^{\underline{e}}, \quad r_M := (2 - \epsilon_M) * 2^{\bar{e}}. \quad (\text{A.3})$$

6. The elements in \mathcal{S} are called **subnormal** or **denormalized**. As for normalized numbers they consists of three parts, but the mantissa is less than one in size. The main use of subnormal numbers is to soften the effect of underflow. If a number is in the range $(0, (1 - \epsilon_M/2) * 2^{\underline{e}})$, then it is rounded to the nearest subnormal number or to zero.
7. Two additional symbols "Inf" and "NaN" are used for special purposes.
8. The symbol **Inf** is used to represent numbers outside the interval $[-r_M, r_M]$ (**overflow**), and results of arithmetic operations of the form $x/0$, where $x \in \mathcal{N}$. Inf has a sign, +Inf and -Inf.
9. The symbol **NaN** stands for "not a number". a NaN results from illegal operations of the form $0/0, 0 * \text{Inf}, \text{Inf}/\text{Inf}, \text{Inf} - \text{Inf}$ and so on.
10. The choices of t , \bar{e} , and \underline{e} are to some extent determined by the architecture of the computer. A floating-point number, say x , occupies $n := 1 + \tau + t$ bits, where 1 bit is used for the sign, τ bits for the exponent, and t bits for the fractional part of the mantissa.

τ	t	
σ	exp	frac

Here $\sigma = 0$ if $x > 0$ and $\sigma = 1$ if $x < 0$, and $\text{exp} \in \{0, 1, 2, 3, \dots, 2^\tau - 1\}$ is an integer. The integer frac is the fractional part $d_1 d_2 \dots d_t$ of the mantissa. The value of a normalized number in the standard system is

$$x = (-1)^\sigma * (1.\text{frac})_2 * 2^{\text{exp}-b}, \text{ where } b := 2^{\tau-1} - 1. \quad (\text{A.4})$$

The integer b is called the **bias**.

11. To explain the choice of b we note that the extreme values $\text{exp} = 0$ and $\text{exp} = 2^\tau - 1$ are used for special purposes. The value $\text{exp} = 0$ is used for the number zero and the subnormal numbers, while $\text{exp} = 2^\tau - 1$ is used for Inf and NaN. Since $2b = 2^\tau - 2$, the remaining numbers of exp , i.e., $\text{exp} \in \{1, 2, \dots, 2^\tau - 2\}$ correspond to e in the set $\{1 - b, 2 - b, \dots, b\}$. Thus in a standard system we have

$$\underline{e} = 1 - b, \quad \bar{e} = b := 2^{\tau-1} - 1. \quad (\text{A.5})$$

12. The most common choices of τ and t are shown in the following table

precision	τ	t	b	$\epsilon_M = 2^{-t}$	$r_m = 2^{1-b}$	r_M
half	5	10	15	9.8×10^{-4}	6.1×10^{-5}	6.6×10^4
single	8	23	127	1.2×10^{-7}	1.2×10^{-38}	3.4×10^{38}
double	11	52	1023	2.2×10^{-16}	2.2×10^{-308}	1.8×10^{308}
quad	15	112	16383	1.9×10^{-34}	3.4×10^{-4932}	1.2×10^{4932}

Here b is given by (A.5) and r_M by (A.3). The various lines correspond to a normalized number occupying **half** a word of 32 bits, one word (**single precision**), two words (**double precision**), and 4 words (**quad precision**).

A.3 Rounding and Arithmetic Operations

The standard system is a closed system. Every $x \in \mathbb{R}$ has a representation as either a floating-point number, or Inf or NaN, and every arithmetic operation produces a result. We denote the computer representation of a real number x by $\text{fl}(x)$.

A.3.1 Rounding

To represent a real number x there are three cases.

$$\text{fl}(x) = \begin{cases} \text{Inf}, & \text{if } x > r_M, \\ -\text{Inf}, & \text{if } x < -r_M, \\ \text{round to zero}, & \text{otherwise.} \end{cases}$$

To represent a real number with $|x| \leq r_M$ the system chooses a machine number $\text{fl}(x)$ closest to x . This is known as **rounding**. When x is midway between two numbers in \mathcal{F} we can either choose the one of larger magnitude (**round away from zero**), or pick the one with a zero last bit (**round to zero**). The standard system uses round to zero. As an example, if $x = 1 + \epsilon_M/2$, then x is midway between 1 and $1 + \epsilon_M$. Therefore $\text{fl}(x) = 1 + \epsilon_M$ if round away from zero is used, while $\text{fl}(x) = 1$ if x is rounded to zero. This is because the machine representation of 1 has `frac` = 0.

The following lemma gives a bound for the relative error in rounding.

Theorem A.5 (Relative error in rounding)

If $r_m \leq |x| \leq r_M$ then

$$\text{fl}(x) = x(1 + \delta), \quad |\delta| \leq u_M := \frac{1}{2}\epsilon_M = 2^{-t-1}.$$

Proof. Suppose $2^e < x < 2^{e+1}$. Then $\text{fl}(x) \in \{1, 1 + \epsilon_M, 1 + 2\epsilon_M, \dots, 2 - \epsilon_M\} * 2^e$. These numbers are uniformly spaced with spacing $\epsilon_M * 2^e$ and therefore $|\text{fl}(x) - x| \leq \frac{1}{2}\epsilon_M 2^e \leq \frac{1}{2}\epsilon_M * |x|$. The proof for a negative x is similar. \square

The number u_M is called the **rounding unit**.

A.3.2 Arithmetic operations

Suppose $x, y \in \mathcal{N}$. In a standard system we have

$$\text{fl}(x \circ y) = (x \circ y)(1 + \delta), \quad |\delta| \leq u_M, \quad \circ \in \{+, -, *, /, \sqrt{\}\}, \quad (\text{A.6})$$

where u_M is the rounding unit of the system. This means that the computed value is as good as the rounded exact answer. This is usually achieved by using one or several extra digits known as **guard digits** in the calculation.

Appendix B

Differentiation of Vector Functions

For any sufficiently differentiable $f : \mathbb{R}^n \rightarrow \mathbb{R}$ we recall that the partial derivative with respect to the i th variable of f is defined by

$$D_i f(\mathbf{x}) := \frac{\partial f(\mathbf{x})}{\partial x_i} := \lim_{h \rightarrow 0} \frac{f(\mathbf{x} + h\mathbf{e}_i) - f(\mathbf{x})}{h}, \quad \mathbf{x} \in \mathbb{R}^n,$$

where \mathbf{e}_i is the i th unit vector in \mathbb{R}^n . For each $\mathbf{x} \in \mathbb{R}^n$ we define the **gradient** $\nabla f(\mathbf{x}) \in \mathbb{R}^n$, and the **hessian** $\mathbf{H}f = \nabla \nabla^T f(\mathbf{x}) \in \mathbb{R}^{n,n}$ of f by

$$\nabla f := \begin{bmatrix} D_1 f \\ \vdots \\ D_n f \end{bmatrix}, \quad \mathbf{H}f := \nabla \nabla^T f := \begin{bmatrix} D_1 D_1 f & \cdots & D_1 D_n f \\ \vdots & & \vdots \\ D_n D_1 & \cdots & D_n D_n f \end{bmatrix}, \quad (\text{B.1})$$

where $\nabla^T f := (\nabla f)^T$ is the row vector gradient. The operators $\nabla \nabla^T$ and $\nabla^T \nabla$ are quite different. Indeed, $\nabla^T \nabla f = D_1^2 f + \cdots + D_n^2 f =: \nabla^2 f$ the **Laplacian** of f , while $\nabla \nabla^T$ can be thought of as an outer product resulting in a matrix.

Lemma B.1 (Product rules)

For $f, g : \mathbb{R}^n \rightarrow \mathbb{R}$ we have the product rules

1. $\nabla(fg) = f\nabla g + g\nabla f, \quad \nabla^T(fg) = f\nabla^T g + g\nabla^T f,$
2. $\nabla \nabla^T(fg) = \nabla f \nabla^T g + \nabla g \nabla^T f + f \nabla \nabla^T g + g \nabla \nabla^T f.$
3. $\nabla^2(fg) = 2\nabla^T f \nabla g + f \nabla^2 g + g \nabla^2 f.$

We define the **Jacobian** of a vector function $\mathbf{f} = [f_1, \dots, f_m]^T : \mathbb{R}^n \rightarrow \mathbb{R}^m$ as the m, n matrix

$$\nabla^T \mathbf{f} := \begin{bmatrix} D_1 f_1 & \cdots & D_n f_1 \\ \vdots & & \vdots \\ D_1 f_m & \cdots & D_n f_m \end{bmatrix}.$$

As an example, if $f(\mathbf{x}) = f(x, y) = x^2 - xy + y^2$ and $\mathbf{g}(x, y) := [f(x, y), x - y]^T$ then

$$\begin{aligned} \nabla f(x, y) &= \begin{bmatrix} 2x - y \\ -x + 2y \end{bmatrix}, & \nabla^T \mathbf{g}(x, y) &= \begin{bmatrix} 2x - y & -x + 2y \\ 1 & -1 \end{bmatrix}, \\ \mathbf{H}f(x, y) &= \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}. \end{aligned}$$

The second order Taylor expansion in n variables can be expressed in terms of the gradient and the hessian.

Lemma B.2 (Second order Taylor expansion)

Suppose $f \in C^2(\Omega)$, where $\Omega \subset \mathbb{R}^n$ contains two points $\mathbf{x}, \mathbf{x} + \mathbf{h} \in \Omega$, such that the line segment $L := \{\mathbf{x} + t\mathbf{h} : t \in (0, 1)\} \subset \Omega$. Then

$$f(\mathbf{x} + \mathbf{h}) = f(\mathbf{x}) + \mathbf{h}^T \nabla f(\mathbf{x}) + \frac{1}{2} \mathbf{h}^T \nabla \nabla^T f(\mathbf{c}) \mathbf{h}, \text{ for some } \mathbf{c} \in L. \quad (\text{B.2})$$

Proof. Let $g : [0, 1] \rightarrow \mathbb{R}$ be defined by $g(t) := f(\mathbf{x} + t\mathbf{h})$. Then $g \in C^2[0, 1]$ and by the chain rule

$$\begin{aligned} g(0) &= f(\mathbf{x}), & g(1) &= f(\mathbf{x} + \mathbf{h}), \\ g'(t) &= \sum_{i=1}^n h_i \frac{\partial f(\mathbf{x} + t\mathbf{h})}{\partial x_i} = \mathbf{h}^T \nabla f(\mathbf{x} + t\mathbf{h}), \\ g''(t) &= \sum_{i=1}^n \sum_{j=1}^n h_i h_j \frac{\partial^2 f(\mathbf{x} + t\mathbf{h})}{\partial x_i \partial x_j} = \mathbf{h}^T \nabla \nabla^T f(\mathbf{x} + t\mathbf{h}) \mathbf{h}. \end{aligned}$$

Inserting these expressions in the second order Taylor expansion

$$g(1) = g(0) + g'(0) + \frac{1}{2} g''(u), \text{ for some } u \in (0, 1),$$

we obtain (B.2) with $\mathbf{c} = \mathbf{x} + u\mathbf{h}$. \square

The gradient and hessian of some functions involving matrices can be found from the following lemma.

Lemma B.3 (Functions involving matrices)

For any $m, n \in \mathbb{N}$, $\mathbf{B} \in \mathbb{R}^{n,n}$, $\mathbf{C} \in \mathbb{R}^{m,n}$, and $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^m$ we have

1. $\nabla(\mathbf{y}^T \mathbf{C}) = \nabla^T(\mathbf{C}\mathbf{x}) = \mathbf{C}$,
2. $\nabla(\mathbf{x}^T \mathbf{B}\mathbf{x}) = (\mathbf{B} + \mathbf{B}^T)\mathbf{x}$, $\nabla^T(\mathbf{x}^T \mathbf{B}\mathbf{x}) = \mathbf{x}^T(\mathbf{B} + \mathbf{B}^T)$,
3. $\nabla\nabla^T(\mathbf{x}^T \mathbf{B}\mathbf{x}) = \mathbf{B} + \mathbf{B}^T$.

Proof.

1. We find $D_i(\mathbf{y}^T \mathbf{C}) = \lim_{h \rightarrow 0} \frac{1}{h} ((\mathbf{y} + h\mathbf{e}_i)^T \mathbf{C} - \mathbf{y}^T \mathbf{C}) = \mathbf{e}_i^T \mathbf{C}$ and $D_i(\mathbf{C}\mathbf{x}) = \lim_{h \rightarrow 0} \frac{1}{h} (\mathbf{C}(\mathbf{x} + h\mathbf{e}_i) - \mathbf{C}\mathbf{x}) = \mathbf{C}\mathbf{e}_i$ and 1. follows.

2. Here we find

$$\begin{aligned} D_i(\mathbf{x}^T \mathbf{B}\mathbf{x}) &= \lim_{h \rightarrow 0} \frac{1}{h} ((\mathbf{x} + h\mathbf{e}_i)^T \mathbf{B}(\mathbf{x} + h\mathbf{e}_i) - \mathbf{x}^T \mathbf{B}\mathbf{x}) \\ &= \lim_{h \rightarrow 0} (\mathbf{e}_i^T \mathbf{B}\mathbf{x} + \mathbf{x}^T \mathbf{B}\mathbf{e}_i + h\mathbf{e}_i^T \mathbf{e}_i) = \mathbf{e}_i^T (\mathbf{B} + \mathbf{B}^T) \mathbf{x}, \end{aligned}$$

and the first part of 2. follows. Taking transpose we obtain the second part.

3. Combining 1. and 2. we obtain 3.

□

Bibliography

- [1] Axelsson, Owe, *Iterative Solution Methods*, Cambridge University Press, Cambridge, 1994.
- [2] Beckenbach, E. F. and R. Bellman, *Inequalities*, Springer Verlag, Berlin, Fourth Printing, 1983.
- [3] Björck, Åke, *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia, 1995.
- [4] E. Cohen, R. F. Riesenfeld, G. Elber, *Geometric Modeling with Splines: An Introduction*, A.K. Peters, Ltd., 2001,
- [5] Demmel, J. W., *Applied Numerical Linear Algebra*, SIAM, Philadelphia, 1997.
- [6] Golub, G. H., and C. F. Van Loan, *Matrix Computations*, John Hopkins University Press, Baltimore, MD, third edition, 1996.
- [7] Grcar, Joseph F., Mathematicians of Gaussian elimination, Notices of the AMS, **58** (2011), 782–792.
- [8] Greenbaum, Anne, *Iterative Methods for Solving Linear Systems*, SIAM, Philadelphia, 1997.
- [9] Hackbusch, Wolfgang, *Iterative Solution of Large Sparse Systems of Equations*, Springer-Verlag, Berlin, 1994.
- [10] Hall, C. A. and W. W. Meyer, Optimal error bounds for cubic spline interpolation. J. Approx. Theory, **16** (1976), 105122.
- [11] Hestenes, Magnus, *Conjugate Direction Methods in Optimization*, Springer-Verlag, Berlin, 1980.
- [12] Hestenes, M. and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, Journal of Research of the National Bureau of Standards **29**(1952), 409–439.

- [13] Higham, Nicloas J., *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia, 1996.
- [14] Hirsch, Morris W. and Stephen Smale, *Differential Equations, Dynamical Systems, and Linear Algebra*, Academic Press, INC., San Diego, 1974.
- [15] Horn, Roger A. and Charles R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1985.
- [16] Horn, Roger A. and Charles R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1991.
- [17] Ipsen, I.C.F, Numerical Matrix Analysis: Linear Systems and Least Squares, SIAM, Philadelphia, 2009.
- [18] Kato, *Perturbation Theory for Linear Operators*, Pringer.
- [19] Lancaster, P., and Rodman, L., Canonical forms for hermitian matrix pairs under strict equivalence and congruence”, SIAM Review, vol. 47, 2005, 407-443.
- [20] Laub, A. J., Matrix Analysis for Scientists and Engineers, SIAM Philadelphia, 2005.
- [21] Laub, A. J., Computational Matrix Analysis, SIAM Philadelphia, 2012.
- [22] Lawson, C.L. and R. J. Hanson, *Solving Least Squares Problems*, Prentice-Hall, Englewood Cliffs, N.J, 1974.
- [23] Lax, Peter D., *Linear Algebra*, John Wiley & Sons, New York, 1997.
- [24] Lay, D.C: Linear algebra and its applications, 2012. Addison Wesley / Pearson. Fourth edition.
- [25] Leon, Steven J., *Linear Algebra with Applications*, Prentice Hall, NJ, Seventh Edition, 2006.
- [26] Meyer, Carl D., *Matrix Analysis and Applied Linear Algebra* , Siam Philadelphia, 2000.
- [27] Steel, J. Michael, *The Cauchy-Schwarz Master Class*, Cambridge University Press, Cambridge, UK, 2004.
- [28] Stewart, G. W., *Matrix Algorithms Volume I: Basic Decompositions*, Siam Philadelphia, 1998.
- [29] Stewart, G. W., *Matrix Algorithms Volume II: Eigensystems*, Siam Philadelphia, 2001.

- [30] Stewart, G. W. and Ji-guang Sun, *Matrix Perturbation Theory*, Academic Press, San Diego, 1990.
- [31] Stewart, G. W., *Introduction to Matrix Computations*, Academic press, New York, 1973.
- [32] Trefethen, Lloyd N., and David Bau III, *Numerical Linear Algebra*, Siam Philadelphia, 1997.
- [33] Tveito, A., and R. Winther, *Partial Differential Equations*, Springer, Berlin.
- [34] Van Loan, Charles, *Computational Frameworks for the Fast Fourier Transform*, Siam Philadelphia, 1992.
- [35] Varga, R. S., *Matrix Iterative Analysis/ 2nd Edn.*, Springer Verlag, New York, 2000.
- [36] Wilkinson, J. H., *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965.
- [37] Young, D. M., *Iterative Solution of Large Linear Systems*, Academic Press, New York, 1971.
- [38] Zhang, F., *Matrix Theory*, Springer, New York, 1999.

List of Exercises

0.25	The inverse of a general 2×2 matrix	15
0.26	The inverse of a special 2×2 matrix	16
0.27	Sherman-Morrison formula	16
0.29	Cramer's rule; special case	19
0.30	Adjoint matrix; special case	19
0.31	Determinant equation for a plane	20
0.32	Signed area of a triangle	20
0.33	Vandermonde matrix	20
0.34	Cauchy determinant (1842)	20
0.35	Inverse of the Hilbert matrix	22
1.12	The shifted power basis is a basis	39
1.13	The natural spline, $n = 1$	39
1.14	Bounding the moments	39
1.15	Moment equations for 1. derivative boundary conditions	39
1.16	Minimal norm property of the natural spline	39
1.18	Computing the D_2 -spline	41
1.20	Spline evaluation	41
1.25	LU factorization of 2. derivative matrix	45
1.26	Inverse of the 2. derivative matrix	46
1.27	Central difference approximation of 2. derivative	46
1.28	Two point boundary value problem	46
1.29	Two point boundary value problem; computation	47
1.30	Approximate force	48
1.33	Eigenpairs \mathbf{T} of order 2	50
1.38	Matrix element as a quadratic form	55
1.39	Outer product expansion of a matrix	55
1.40	The product $\mathbf{A}^T \mathbf{A}$	55
1.41	Outer product expansion	55
1.42	System with many right hand sides; compact form	55
1.43	Block multiplication example	55
1.44	Another block multiplication example	55

2.8	Column oriented backsolve	64
2.11	Computing the inverse of a triangular matrix	65
2.13	Finite sums of integers	67
2.14	Multiplying triangular matrices	67
2.19	# operations for banded triangular systems	71
2.21	L1U and LU1	72
2.22	LU of nonsingular matrix	72
2.23	Row interchange	72
2.24	LU and determinant	72
2.25	Diagonal elements in U	72
2.26	Proof of LDU theorem	72
2.27	Proof of LU1 theorem	72
2.31	Making block LU into LU	74
2.36	Using PLU for \mathbf{A}^T	78
2.37	Using PLU for determinant	78
2.38	Using PLU for \mathbf{A}^{-1}	79
3.20	Positive definite characterizations	90
3.29	A counterexample	95
4.4	The $\mathbf{A}^* \mathbf{A}$ inner product	100
4.5	Angle between vectors in complex case	100
4.18	What does algorithm housegen do when $\mathbf{x} = \mathbf{e}_1$?	107
4.19	Examples of Householder transformations	107
4.20	2×2 Householder transformation	107
4.28	QR decomposition	113
4.29	Householder triangulation	113
4.32	QR using Gram-Schmidt, II	114
4.34	Plane rotation	115
4.35	Solving upper Hessenberg system using rotations	117
5.5	Eigenvalues of a block triangular matrix	124
5.6	Characteristic polynomial of transpose	124
5.7	Characteristic polynomial of inverse	124
5.8	The power of the eigenvector expansion	124
5.9	Eigenvalues of an idempotent matrix	124
5.10	Eigenvalues of a nilpotent matrix	124
5.11	Eigenvalues of a unitary matrix	124
5.12	Nonsingular approximation of a singular matrix	125
5.13	Companion matrix	125
5.17	Find eigenpair example	127
5.22	Jordan example	130
5.23	A nilpotent matrix	130
5.24	Properties of the Jordan form	130
5.25	Powers of a Jordan block	130

5.26	The minimal polynomial	130
5.27	Big Jordan example	131
5.30	Schur decomposition example	133
5.31	Deflation example	133
5.34	Skew-Hermitian matrix	135
5.35	Eigenvalues of a skew-Hermitian matrix	135
5.36	Eigenvector expansion using orthogonal eigenvectors	135
5.46	Eigenvalue perturbation for Hermitian matrices	139
5.48	Hoffman-Wielandt	139
5.51	Biorthogonal expansion	141
5.54	Generalized Rayleigh quotient	142
6.2	SVD1	144
6.3	SVD2	144
6.7	SVD examples	146
6.8	More SVD examples	147
6.9	Singular values of a normal matrix	147
6.10	The matrices $\mathbf{A}^* \mathbf{A}$, $\mathbf{A} \mathbf{A}^*$ and SVD	147
6.13	Nonsingular matrix	150
6.14	Full row rank	150
6.16	Counting dimensions of fundamental subspaces	151
6.17	Rank and nullity relations	151
6.18	Orthonormal bases example	151
6.19	Some spanning sets	151
6.20	Singular values and eigenpair of composite matrix	151
6.26	Rank example	155
6.27	Another rank example	156
7.7	Consistency of sum norm?	161
7.8	Consistency of max norm?	161
7.9	Consistency of modified max norm	161
7.11	What is the sum norm subordinate to?	162
7.12	What is the max norm subordinate to?	162
7.19	Spectral norm	166
7.20	Spectral norm of the inverse	166
7.21	p -norm example	167
7.24	Unitary invariance of the spectral norm	167
7.25	$\ AU\ _2$ rectangular \mathbf{A}	167
7.26	p -norm of diagonal matrix	168
7.27	spectral norm of a column vector	168
7.28	Norm of absolute value matrix	168
7.35	Sharpness of perturbation bounds	173
7.36	Condition number of 2. derivative matrix	173
7.47	When is a complex norm an inner product norm?	179

7.48	p norm for $p = 1$ and $p = \infty$	179
7.49	The p -norm unit sphere	179
7.50	Sharpness of p -norm inequality	179
7.51	p -norm inequalities for arbitrary p	179
8.10	Fitting a circle to points	186
8.18	The generalized inverse	194
8.19	Uniqueness of generalized inverse	194
8.20	Verify that a matrix is a generalized inverse	195
8.21	Linearly independent columns and generalized inverse	195
8.22	The generalized inverse of a vector	195
8.23	The generalized inverse of an outer product	195
8.24	The generalized inverse of a diagonal matrix	195
8.25	Properties of the generalized inverse	195
8.26	The generalized inverse of a product	196
8.27	The generalized inverse of the conjugate transpose	196
8.28	Linearly independent columns	196
8.29	Analysis of the general linear system	196
8.30	Fredholm's alternative	196
8.33	Condition number	199
8.34	Equality in perturbation bound	199
8.36	Problem using normal equations	200
9.2	4×4 Poisson matrix	210
9.5	Properties of Kronecker products	212
9.9	2. derivative matrix is positive definite	216
9.10	1D test matrix is positive definite?	216
9.11	Eigenvalues for 2D test matrix of order 4	216
9.12	Nine point scheme for Poisson problem	216
9.13	Matrix equation for nine point scheme	217
9.14	Biharmonic equation	217
10.5	Fourier matrix	229
10.6	Sine transform as Fourier transform	229
10.7	Explicit solution of the discrete Poisson equation	229
10.8	Improved version of Algorithm 10.1	230
10.9	Fast solution of 9 point scheme	230
10.10	Algorithm for fast solution of 9 point scheme	230
10.11	Fast solution of biharmonic equation	231
10.12	Algorithm for fast solution of biharmonic equation	231
10.13	Check algorithm for fast solution of biharmonic equation	231
10.14	Fast solution of biharmonic equation using 9 point rule	231
11.12	Richardson and Jacobi	245
11.13	R-method when eigenvalues have positive real part	246
11.16	Example: GS converges, J diverges	247

11.17	Divergence example for J and GS	247
11.18	Strictly diagonally dominance; The J method	247
11.19	Strictly diagonally dominance; The GS method	247
11.23	Convergence example for fix point iteration	251
11.24	Estimate in Lemma 11.22 can be exact	251
11.25	Slow spectral radius convergence	251
11.31	A special norm	255
11.33	When is $\mathbf{A} + \mathbf{E}$ nonsingular?	256
12.1	\mathbf{A} -norm	262
12.2	Paraboloid	262
12.5	Steepest descent iteration	265
12.8	Conjugate gradient iteration, II	268
12.9	Conjugate gradient iteration, III	268
12.10	The cg step length is optimal	268
12.11	Starting value in cg	268
12.17	Program code for testing steepest descent	273
12.18	Using cg to solve normal equations	273
12.23	Krylov space and cg iterations	276
12.26	Another explicit formula for the Chebyshev polynomial	279
12.28	Maximum of a convex function	281
13.5	Nonsingularity using Gerschgorin	299
13.6	Gerschgorin, strictly diagonally dominant matrix	299
13.8	Continuity of eigenvalues	300
13.12	∞ -norm of a diagonal matrix	302
13.15	Number of arithmetic operations, Hessenberg reduction	304
13.17	Assemble Householder transformations	305
13.18	Tridiagonalize a symmetric matrix	305
13.22	Counting eigenvalues	308
13.23	Overflow in LDL* factorization	308
13.24	Simultaneous diagonalization	309
13.25	Program code for one eigenvalue	310
13.26	Determinant of upper Hessenberg matrix	310
14.4	Orthogonal vectors	316

Index

- 1D test matrix, 47
- 2D test matrix, 209
- convex combinations, 279
- eigenvector expansion, 120
- positive definite, 91
- positive semidefinite, 91
- A-inner product, 260
- A-norm, 260
- absolute error, 169, 325
- algebraic multiplicity, 124
- algorithms
 - assemble Householder transformations, 303
 - backsolve, 62
 - backsolve column oriented, 63
 - bandcholesky, 85
 - cg, 267
 - fastpoisson, 221
 - findsubintervals, 40
 - forwardsolve, 61
 - forwardsolve column oriented, 63
 - housegen, 104
 - Householder reduction to Hessenberg form, 302
 - Householder triangulation, 107
 - Jacobi, 237
 - L1U factorization, 69
 - LDL* factorization, 81
 - preconditioned cg, 283
- Rayleigh quotient iteration, 317
- SOR, 238
- spline evaluation, 39
- splineint, 39
- testing conjugate gradient, 268
- the power method, 315
- trifactor, 35
- trisolve, 35
- upper Hessenberg linear system, 115
- Anorm, 260
- back substituion, 57
- banded matrix, 4
- bandsemi-cholesky, 90
- biharmonic equation, 215
 - fast solution method, 229
 - nine point rule, 229
- block LU theorem, 71
- Cauchy determinant, 21
- Cauchy-Binet formula, 19
- Cauchy-Schwarz inequality, 97
- Cayley Hamilton Theorem, 129
- central difference, 44
- central difference approximation
 - second derivative, 44
- change of basis matrix, 10
- characteristic equation, 23, 119
- characteristic polynomial, 23, 119
- Chebyshev polynomial, 276
- column operations, 17

- column space (span), 11
companion matrix, 123
complete pivoting, 76
complexity of an algorithm, 64
computer arithmetic, 325
condition number
 ill-conditioned, 169
congruent matrices, 305
conjugate gradient method, 259
 convergence, 269
 derivation, 263
 Krylov spaces, 271
 least squares problem, 271
 preconditioning, 281
 preconditioning algorithm, 283
 preconditioning convergence,
 284
convex combination, 135, 174
convex function, 174
Courant-Fischer theorem, 137
Cramer's rule, 18
cubic spline
 minimal norm, 38
- defective matrix, 119
deflation, 131
determinant, 16
 area of a triangle, 20
 Cauchy, 21
 Cauchy-Binet, 19
 cofactor expansion, 17
 plane equation, 20
 straight line equation, 17
 Vandermonde, 20
- dirac delta, 3
direct sum, 187
discrete cosine transform, 222
discrete Fourier transform, 222,
 223
 Fourier matrix, 223
discrete sine transform, 222
double precision, 329
- eigenpair, 22, 45, 119
 left eigenpair, 133, 138
 right eigenpair, 138
eigenvalue, 22, 45, 119
 algebraic multiplicity, 124
 characteristic equation, 23,
 119
 characteristic polynomial, 23,
 119
Courant-Fischer theorem, 137
geometric multiplicity, 124
Hoffman-Wielandt theorem,
 138
left eigenvalue, 138
location, 294
Rayleigh quotient, 135
right eigenvalue, 138
spectral theorem, 134
spectrum, 22, 119
triangular matrix, 24
- eigenvector, 22, 45, 119
 left eigenvector, 138
 right eigenvector, 138
- elementary reflector, 102
Elsner's theorem, 298
equivalent norms, 159
extension of basis, 100
- fast Fourier transform, 222, 224
 recursive FFT, 226
- fill-inn, 218
- finite difference method, 41
- finite dimensional vector space,
 6
- fixed point form of discrete Pois-
 son equation, 236
- fixed-point, 239
- fixed-point iteration, 239
- floating-point number
 bias, 328
 denormalized, 328
 double precision, 329

- exponent part, 327
- guard digits, 330
- half precision, 329
- Inf, 328
- mantissa, 327
- NaN, 328
- normalized, 327
- overflow, 328
- quadruple precision, 329
- round away from zero, 329
- round to zero, 329
- rounding, 329
- rounding unit, 329
- single precision, 329
- subnormal, 328
- Fourier matrix, 223
- Fredholm's alternative, 196
- Frobenius norm, 154
- Gaussian elimination, 57
 - complete pivoting, 76
 - interchange matrix, 73
 - pivot, 72
 - pivot vector, 73
 - pivoting, 72
- generalized inverse, 194
- geometric multiplicity, 124
- Gershgorin's theorem, 294
- Given's rotation, 113
- gradient, 82, 331
- gradient method, 262
- Gram-Schmidt, 99
- guard digits, 330
- Hölder's inequality, 158, 176
- Hadamard's inequality, 110
- half precision, 329
- Hermitian matrix, 48
- hessian, 82, 331
- Hilbert matrix, 22, 185
- Hoffman-Wielandt theorem, 138
- Householder transformation, 102
- identity matrix, 3
- ill-conditioned problem, 169
- inequality
 - geometric/arithmetic mean, 176
 - Hölder, 176
 - Kantorovich, 278
 - Minkowski, 177
- Inf, 328
- inner product, 96
 - inner product norm, 96
 - Pythagoras' theorem, 99
 - standard inner product in \mathbb{C}^n , 96
- inner product space
 - orthogonal basis, 99
 - orthonormal basis, 99
- interchange matrix, 73
- inverse power method, 315
- inverse triangle inequality, 159
- iterative method
 - convergence, 241
 - Gauss-Seidel, 235
 - Jacobi, 235
 - SOR, 235
 - SOR, convergence, 244
 - SSOR, 236
- iterative methods, 233
- Jacobian, 332
- Jordan factors, 126
- Jordan form, 126
 - Jordan block, 125
 - Jordan canonical form, 125
 - principal vectors, 126
- Kronecker product, 209
 - eigenvectors, 211
 - inverse, 211
 - left product, 209
 - mixed product rule, 210
 - nonsingular, 211

- positive definite, 211
properties, 211
right product, 209
symmetry, 211
transpose, 211
- Kronecker sum, 210
nonsingular, 211
positive definite, 211
symmetry, 211
- Krylov space, 271
- Laplacian, 331
- LDL theorem , 80
- leading principal block submatrices, 71
- leading principal minor, 58
- leading principal submatrices, 58
- least squares
error analysis, 196
- least squares problem, 181
- least squares solution, 181
- left eigenpair, 133, 138
- left eigenvalue, 138
- left eigenvector, 138
- left triangular, 65
- linear combination, 6
- linear interpolation polynomial, 28
- linear system
existence and uniqueness, 12, 13
homogeneous, 12
overdetermined, 12
residual vector, 170
square, 12
underdetermined, 12
- linearly dependent, 7
- linearly independent, 7
- LSQ, 181
- LU factorization, 65
- LU theorem, 67
- mantissa, 327
- matrix
addition, 3
adjoint, 18
adjoint formula for the inverse, 18
block matrix, 49
block triangular, 51
blocks, 49
cofactor, 18
column space (span), 11
companion matrix, 123
computing inverse, 63
deflation, 131
diagonal, 3
element-by-element operations, 3
Hadamard product, 3
Hilbert, 22
idempotent, 122
ill-conditioned, 170
inverse, 14
inverse Hilbert matrix, 22
invertible, 14
Kronecker product, 209
leading principal minor, 58
leading principal submatrices, 58
- left inverse, 14
- left triangular, 4
- lower Hessenberg, 4
- lower triangular, 4
- LU theorem, 67
- multiplication, 3
- negative (semi)definite, 82
- Neumann series, 253
- nilpotent, 122
- nonsingular, 12, 13
- null space (ker), 11
- operator norm, 163
- outer product, 53
- outer product expansion, 53
- permutation, 73

-
- positive definite, 82
 positive semidefinite, 82
 principal minor, 58
 principal submatrix, 58
 product of triangular matrices, 52
 quasi-triangular, 134
 right inverse, 14
 right triangular, 4
 row space, 11
 scalar multiplication, 3
 Schur product, 3
 second derivative, 42
 similar matrices, 121
 similarity transformation, 121
 singular, 12, 13
 spectral radius, 242, 251
 strictly diagonally dominant, 32
 test matrix, 2D, 209
 test matrix, 1D, 47
 trace, 23
 triangular, 52
 tridiagonal, 4
 unit triangular, 52
 upper Hessenberg, 4
 upper trapezoidal, 105
 upper triangular, 4
 vec operation, 207
 weakly diagonally dominant, 42
 well-conditioned, 170
matrix norm
 consistent norm, 161
 Frobenius norm, 154, 160
 max norm, 160
 operator norm, 162
 spectral norm, 164
 subordinate norm, 162
 sum norm, 160
 two-norm, 164
 minimal polynomial, 128
 Minkowski's inequality, 158, 177
 mixed product rule, 210
 NaN, 328
 natural ordering, 207
 negative (semi)definite, 82
 Neumann series, 253
 nilpotent matrix, 122
 nondefective matrix, 119
 nonsingular matrix, 12
 nontrivial subspaces, 9
norm, 157
 l_1 -norm, 158
 l_2 -norm, 158
 l_∞ -norm, 158
 absolute norm, 168
 continuity, 159
 Euclidian norm, 158
 infinity-norm, 158
 max norm, 158
 monotone norm, 168
 one-norm, 158
 triangle inequality, 157
 two-norm, 158
 normal matrix, 131
 normal equations, 182
 null space (ker), 11
 operation count, 63
 operator norm, 163
 optimal relaxation parameter, 247
 optimal step length, 262
 ordered SVD, 146
 orthogonal decomposition, 192
 orthogonal matrix, see orthonormal matrix, 101
 orthogonal projections, 187
 orthogonal sum, 187
 orthonormal matrix, 101
 outer product, 53
 overflow, 328
p-norms, 158

- paraboloid, 260
- parallelogram identity, 177
- partial pivoting, 75
- permutation matrix, 73
- perpendicular vectors, 98
- pivot row, 72
- pivot vector, 73
- pivots, 59
- plane rotation, 113
- Poisson matrix, 208
- Poisson problem, 205
 - five point stencil, 207
 - nine point scheme, 214
 - Poisson matrix, 208
 - variable coefficients, 284
- Poisson problem (1D), 41
- polarization identity, 179
- positive definite, 82
- positive semidefinite, 82
- power method, 312
 - inverse, 315
 - Rayleigh quotient iteration, 316
 - shifted, 315
- preconditioned conjugate gradient method, 259
- preconditioning, 281
- preconditioning matrix, 239
- principal minor, 58
- principal submatrix, 58
- principal vectors, 126
- pseudo inverse, 194
- QR algorithm
 - implicit shift, 322
 - Rayleigh quotient shift, 322
 - shifted, 321
 - Wilkinson shift, 322
- QR decomposition, 109
- QR factorization, 109
- quadratic form, 82
- quadruple precision, 329
- rate of convergence, 247
- Rayleigh quotient, 135
 - generalized, 140
- Rayleigh quotient iteration, 316
- relative error, 169, 325
- residual vector, 170
- Richardson's method, 242
- right eigenpair, 138
- right eigenvalue, 138
- right eigenvector, 138
- rotation in the i,j -plane, 114
- rounding unit, 329
- row operations, 17
- row space, 11
- Runge phenomenon, 28
- scalar product, 96
- scaled partial pivoting, 76
- Schur factorization, 130
- Schur factors, 130
- second derivative matrix, 42
- semi-Cholesky factorization, 88
- Sherman-Morrison formula, 16
- shifted power method, 315
- similar matrices, 121
- similarity transformation, 121
- single precision, 329
- singular value factorization, 148
- singular values, 143
 - error analysis, 199
- singular vectors, 143
- span, 6
- spectral radius, 242, 251
- spectral theorem, 134
- spectrum, 22, 119
- splitting matrices for J, GS and SOR, 240
- splitting matrix, 239
- standard inner product in \mathbb{C}^n , 101
- steepest descent, 262
- stencil, 207

- sum of subspaces, 187
- sums of integers, 65
- SVF, 148
- Sylvester's inertia theorem, 305
- trace, 23
- triangle inequality, 157
- triangular matrix
 - lower triangular, 65
 - upper triangular, 65
- trivial subspace, 9
- two point boundary value problem, 41
- unit vectors, 3
- unitary matrix, 101
- upper trapezoidal matrix, 105
- upper triangular, 65
- vector
 - angle, 98
 - linearly dependent, 7
 - linearly independent, 7
 - nontrivial subspaces, 9
 - orthogonal, 98
 - orthonormal, 98
- vector norm, 98, 157
- vector space
 - basis, 7
 - change of basis matrix, 10
 - complementary, 9
 - complex inner product space, 96
 - dimension, 8
 - dimension formula for sums of subspaces, 9
 - direct sum, 9
 - enlarging vectors to a basis, 8
 - examples of subspaces, 9
 - existence of basis, 8
 - intersection, 9
 - normed, 157
- orthogonal vectors, 98
- real, 5
- span, 7
- subspace, 8
- sum, 9
- union, 9
- vectorization, 206
- weights, 185