MA4230: MATRIX COMPUTATION
(lecture notes for AY 2023/24)

Timo Sprekeler

last updated: November 5, 2023

Contents

Annotation: These lecture notes are based on the following book:
L. N. Trefethen and D. Bau III, Numerical Linear Algebra (SIAM, 1997).

1 Preliminaries

1.1 Matrices

Definition 1.1. For $m, n \in \mathbb{N} = \{1, 2, \dots\}$, we denote the class of real matrices of size $m \times n$ (m rows, n columns) by

$$\mathbb{R}^{m\times n} := \left\{ \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \middle| \, a_{ij} \in \mathbb{R} \; \forall i \in \{1, \dots, m\}, j \in \{1, \dots, n\} \right\}.$$

We set $\mathbb{R}^m := \mathbb{R}^{m\times 1}$ to denote the class of real column vectors of length m.

Notation: For a matrix $A \in \mathbb{R}^{m\times n}$, we often write $A = (a_{ij})$ with $a_{ij} \in \mathbb{R}$ denoting the (i, j)-th entry (row i, column j) of A, and $A = (a_1|a_2|\cdots|a_n)$ with $a_1, \dots, a_n \in \mathbb{R}^m$ denoting the column vectors of A.

1.1.1 Basic operations

For $A = (a_{ij}) \in \mathbb{R}^{m\times n}$, $B = (b_{ij}) \in \mathbb{R}^{m\times n}$, $C = (c_{ij}) \in \mathbb{R}^{n\times l}$ and $\alpha \in \mathbb{R}$, we define

- addition: $A + B \in \mathbb{R}^{m\times n}$, $(A + B)_{ij} := a_{ij} + b_{ij}$ $\forall i \in \{1, \dots, m\}, j \in \{1, \dots, n\}$,
- scalar multiplication: $\alpha A \in \mathbb{R}^{m\times n}$, $(\alpha A)_{ij} := \alpha a_{ij}$ $\forall i \in \{1, \dots, m\}, j \in \{1, \dots, n\}$,
- transposition: $A^{\mathrm{T}} \in \mathbb{R}^{n\times m}$, $(A^{\mathrm{T}})_{ij} := a_{ji}$ $\forall i \in \{1, \dots, n\}, j \in \{1, \dots, m\}$,
- matrix multiplication: $AC \in \mathbb{R}^{m\times l}$ given by

$$(AC)_{ij} := \sum_{k=1}^{n} a_{ik} c_{kj} \qquad \forall i \in \{1, \dots, m\}, j \in \{1, \dots, l\}.$$

Let us note that matrix multiplication allows us to form matrix-vector products, i.e., for a matrix $A = (a_{ij}) \in \mathbb{R}^{m\times n}$ and a vector $x = (x_1, x_2, \dots, x_n)^{\mathrm{T}} \in \mathbb{R}^n = \mathbb{R}^{n\times 1}$, we have $Ax \in \mathbb{R}^{m\times 1} = \mathbb{R}^m$ with entries $(Ax)_i = \sum_{k=1}^{n} a_{ik} x_k$ for $i \in \{1, \dots, m\}$.

Remark 1.1. (i) Matrix-vector product: For $A = (a_1|a_2|\cdots|a_n) \in \mathbb{R}^{m\times n}$ and $x = (x_1, x_2, \dots, x_n)^{\mathrm{T}} \in \mathbb{R}^n$, we have

$$Ax = \sum_{k=1}^{n} x_k a_k \in \operatorname{span}(a_1, \dots, a_n) \subseteq \mathbb{R}^m,$$

i.e., Ax is a linear combination of the columns a_k of A with coefficients x_k.

(ii) Matrix-matrix product: For matrices $A = (a_1|a_2|\cdots|a_n) \in \mathbb{R}^{m\times n}$ and $C = (c_1|c_2|\cdots|c_l) \in \mathbb{R}^{n\times l}$, let $B := AC = (b_1|b_2|\cdots|b_l) \in \mathbb{R}^{m\times l}$. We then have

$$b_i = Ac_i = \sum_{k=1}^{n} c_{ki} a_k \in \operatorname{span}(a_1, \dots, a_n) \subseteq \mathbb{R}^m \qquad \forall i \in \{1, \dots, l\},$$

i.e., b_i is a linear combination of the columns a_k of A with coefficients c_{ki}.

1.1.2 Connection to linear maps

Any matrix $A \in \mathbb{R}^{m\times n}$ induces a linear map via matrix-vector multiplication, that is, the map

$$L_A : \mathbb{R}^n \to \mathbb{R}^m, \qquad x \mapsto Ax \tag{1.1}$$

is linear. Recall that a map $L : \mathbb{R}^n \to \mathbb{R}^m$ is called linear, denoted $L \in \mathcal{L}(\mathbb{R}^n, \mathbb{R}^m)$, iff (short for "if, and only if,") it satisfies $L(\alpha x + y) = \alpha L(x) + L(y)$ for all $x, y \in \mathbb{R}^n$ and $\alpha \in \mathbb{R}$. Conversely, any linear map from $\mathbb{R}^n$ to $\mathbb{R}^m$ can be represented by a $m \times n$ matrix in the sense that for any $L \in \mathcal{L}(\mathbb{R}^n, \mathbb{R}^m)$ there exists $A \in \mathbb{R}^{m\times n}$ such that $L = L_A$.

Theorem 1.1. *There holds* $\mathcal{L}(\mathbb{R}^n, \mathbb{R}^m) = \{L_A : A \in \mathbb{R}^{m\times n}\}$ *with* L_A *as in* (1.1).

Proof. We have already observed that $\{L_A : A \in \mathbb{R}^{m\times n}\} \subseteq \mathcal{L}(\mathbb{R}^n, \mathbb{R}^m)$. For the converse inclusion, given $L \in \mathcal{L}(\mathbb{R}^n, \mathbb{R}^m)$, we have for any $x = (x_1, x_2, \ldots, x_n)^{\mathrm{T}} \in \mathbb{R}^n$ that

$$L(x) = L\left(\sum_{i=1}^n x_i e_i\right) = \sum_{i=1}^n x_i L(e_i) = Ax \quad \text{with} \quad A := (L(e_1)|L(e_2)|\cdots|L(e_n)) \in \mathbb{R}^{m\times n}.$$

Here, $e_1, \ldots, e_n$ denote the canonical basis vectors of $\mathbb{R}^n$. □

Let us point out the behavior of the linear map (1.1) under addition, scalar multiplication and matrix multiplication: For $A, B \in \mathbb{R}^{m\times n}$, $C \in \mathbb{R}^{n\times l}$ and $\alpha \in \mathbb{R}$, we have

$$L_{A+B} = L_A + L_B, \qquad L_{\alpha A} = \alpha L_A, \qquad L_{AC} = L_A \circ L_C.$$

1.1.3 Range and nullspace

Definition 1.2. Let $A \in \mathbb{R}^{m\times n}$. We then define

(i) the range of A to be $\mathcal{R}(A) := \{y \in \mathbb{R}^m | \exists\, x \in \mathbb{R}^n : y = Ax\}$,

(ii) the nullspace of A to be $\mathcal{N}(A) := \{x \in \mathbb{R}^n \mid Ax = 0\}$,

(iii) the rank of A to be $\mathrm{rk}(A) := \dim(\mathcal{R}(A))$,

(iv) the nullity of A to be $\mathrm{nullity}(A) := \dim(\mathcal{N}(A))$.

Remark 1.2. In view of Remark 1.1, we have for $A = (a_1|a_2|\cdots|a_n) \in \mathbb{R}^{m\times n}$ that $\mathcal{R}(A) = \mathrm{span}(a_1, \ldots, a_n)$. We also call $\mathcal{R}(A)$ the column space of A.

Theorem 1.2. *Let* $A, B \in \mathbb{R}^{m\times n}$ *and* $C \in \mathbb{R}^{n\times l}$. *Then the following assertions hold.*

(i) $0 \le \mathrm{rk}(A) = \mathrm{rk}(A^{\mathrm{T}}) \le \min\{m, n\}$ *("column rank equals row rank"),*

(ii) $\mathrm{rk}(A) + \mathrm{nullity}(A) = n$ *("rank-nullity theorem"),*

(iii) $\mathrm{rk}(A) + \mathrm{rk}(C) - n \le \mathrm{rk}(AC) \le \min\{\mathrm{rk}(A), \mathrm{rk}(C)\}$ *("Sylvester's inequalities"),*

(iv) $\mathrm{rk}(A + B) \le \mathrm{rk}(A) + \mathrm{rk}(B)$,

(v) $\mathrm{rk}(A^{\mathrm{T}}A) = \mathrm{rk}(A) = \mathrm{rk}(AA^{\mathrm{T}})$.

Proof. See undergraduate linear algebra. □

In view of (i), we say that a matrix $A \in \mathbb{R}^{m\times n}$ is of full rank iff it satisfies $\mathrm{rk}(A) = \min\{m,n\}$. Otherwise, when $\mathrm{rk}(A) < \min\{m,n\}$, we call A rank-deficient. For $m \geq n$, we can characterize matrices of full rank as follows.

Theorem 1.3. *Let $A = (a_1|a_2|\cdots|a_n) \in \mathbb{R}^{m\times n}$ with $m \geq n$. Then the following assertions are equivalent.*

(i) A is of full rank, i.e., $\mathrm{rk}(A) = n$.

(ii) The columns $a_1,\dots,a_n \in \mathbb{R}^m$ of A are linearly independent.

(iii) The associated linear map L_A given by (1.1) is injective.

Proof. (i)⇒(ii): If $\mathrm{rk}(A) = \dim(\mathrm{span}(a_1,\dots,a_n)) = n$, then clearly $a_1,\dots,a_n$ are linearly independent.
(ii)⇒(iii): Suppose that the columns $a_1,\dots,a_n \in \mathbb{R}^m$ of A are linearly independent, and let $x = (x_1,x_2,\dots,x_n)^{\mathrm{T}}, y = (y_1,y_2,\dots,y_n)^{\mathrm{T}} \in \mathbb{R}^n$ such that $L_A(x) = L_A(y)$, i.e., $Ax = Ay$. Then $A(x-y) = \sum_{i=1}^n (x_i - y_i)a_i = 0 \in \mathbb{R}^m$ and hence, $x_i - y_i = 0$ for all $i \in \{1,\dots,n\}$, i.e., $x = y$.
(iii)⇒(i): We show the contrapositive ¬(i)⇒ ¬(iii). To this end, suppose that A is not of full rank. Then, $\mathrm{rk}(A) = \dim(\mathrm{span}(a_1,\dots,a_n)) < n$ and hence, $a_1,\dots,a_n$ are linearly dependent. Then, there exists $c = (c_1,c_2,\dots,c_n)^{\mathrm{T}} \in \mathbb{R}^n\backslash\{0\}$ such that $\sum_{i=1}^n c_i a_i = 0$ and we conclude that L_A is not injective as $L_A(c) = Ac = 0 = L_A(0)$ and $c \neq 0$. □

1.1.4 Invertible matrices

We now turn our attention to square matrices $A \in \mathbb{R}^{n\times n}$. For $n \in \mathbb{N}$, we let

$$I_n := (e_1|e_2|\cdots|e_n) := \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \in \mathbb{R}^{n\times n}$$

denote the $n \times n$ identity matrix. Here, $e_1,\dots,e_n$ are the canonical basis vectors of $\mathbb{R}^n$.

Definition 1.3. A matrix $A \in \mathbb{R}^{n\times n}$ is said to be invertible (or non-singular) iff there exists a matrix $A^{-1} \in \mathbb{R}^{n\times n}$, called the inverse of A, such that $AA^{-1} = A^{-1}A = I_n$.

Remark 1.3. Let $A = (a_1|a_2|\cdots|a_n) \in \mathbb{R}^{n\times n}$ be invertible with inverse $A^{-1} \in \mathbb{R}^{n\times n}$, and let $b = \sum_{k=1}^n b_k e_k \in \mathbb{R}^n$. Further, let $x = A^{-1}b = \sum_{k=1}^n x_k e_k \in \mathbb{R}^n$. We regard x as the unique solution to $Ax = b$, i.e., $b = \sum_{k=1}^n x_k a_k$. Observe that $A^{-1}b$ is the vector containing the coefficients of the expansion of b in the basis $\{a_1,\dots,a_n\}$. Hence, multiplication by A^{-1} corresponds to a change of basis operation.

Observe that invertibility of a matrix $A \in \mathbb{R}^{n\times n}$ is equivalent to invertibility of the associated linear map L_A from (1.1). We state a few equivalent characterizations of invertibility.

Theorem 1.4. *For $A \in \mathbb{R}^{n\times n}$, we have the equivalences*

$$A \textit{ invertible} \Leftrightarrow \mathrm{rk}(A) = n \Leftrightarrow \mathcal{R}(A) = \mathbb{R}^n \Leftrightarrow \mathcal{N}(A) = \{0\} \Leftrightarrow \det(A) \neq 0 \Leftrightarrow 0 \notin \Lambda(A).$$

Proof. See undergraduate linear algebra. □

Here, $\det(A)$ denotes the determinant and $\Lambda(A) := \{\lambda \in \mathbb{C} : \det(A - \lambda I_n) = 0\}$ the spectrum (set of eigenvalues) of $A \in \mathbb{R}^{n\times n}$.

Theorem 1.5. *Let $A, C \in \mathbb{R}^{n\times n}$ be invertible matrices and let $\alpha \in \mathbb{R}\backslash\{0\}$. Then also $A^{-1}, AC, \alpha A, A^{\mathrm{T}} \in \mathbb{R}^{n\times n}$ are invertible and we have the following:*

(i) $(A^{-1})^{-1} = A, \quad (AC)^{-1} = C^{-1}A^{-1}, \quad (\alpha A)^{-1} = \frac{1}{\alpha}A^{-1}, \quad (A^{\mathrm{T}})^{-1} = (A^{-1})^{\mathrm{T}}.$

(ii) $\mathrm{rk}(A^{-1}) = \mathrm{rk}(A) = n, \quad \det(A^{-1}) = \frac{1}{\det(A)}.$

Proof. Assertion (i) is straightforward using Definition 1.3. Note $\mathrm{rk}(A^{-1}) = \mathrm{rk}(A) = n$ from Theorem 1.4 and invertibility of A and A^{-1}. Finally, note $\det(A) \neq 0$ from Theorem 1.4 and $\det(A^{-1})\det(A) = \det(A^{-1}A) = \det(I_n) = 1$ using the multiplicative property of the determinant. □

Remark 1.4. Let us provide the corresponding results for transposition. For matrices $A, B \in \mathbb{R}^{m\times n}$, $C \in \mathbb{R}^{n\times l}$, and a scalar $\alpha \in \mathbb{R}$, we have the following:

(i) $(A^{\mathrm{T}})^{\mathrm{T}} = A, \quad (AC)^{\mathrm{T}} = C^{\mathrm{T}}A^{\mathrm{T}}, \quad (\alpha A)^{\mathrm{T}} = \alpha A^{\mathrm{T}}, \quad (A + B)^{\mathrm{T}} = A^{\mathrm{T}} + B^{\mathrm{T}}.$

(ii) $\mathrm{rk}(A^{\mathrm{T}}) = \mathrm{rk}(A), \quad \det(A^{\mathrm{T}}) = \det(A).$

Definition 1.4. A matrix $A \in \mathbb{R}^{n\times n}$ is said to be symmetric iff $A^{\mathrm{T}} = A$. A matrix $Q \in \mathbb{R}^{n\times n}$ is said to be orthogonal iff Q is invertible and $Q^{-1} = Q^{\mathrm{T}}$.

1.1.5 Orthogonality

Definition 1.5. Let $x, y \in \mathbb{R}^n$. We define

(i) the Euclidean inner product $\langle x, y\rangle := x^{\mathrm{T}}y \in \mathbb{R}$, and

(ii) the Euclidean norm $\|x\|_2 := \sqrt{\langle x, x\rangle} \in \mathbb{R}$.

It can be shown that $\langle x, y\rangle = \|x\|_2\|y\|_2 \cos(\theta_{x,y})$ for any $x, y \in \mathbb{R}^n$, where $\theta_{x,y}$ denotes the angle between the vectors x and y. Further, it is straightforward to check that the inner product is bilinear, i.e., we have for any $x, x_1, x_2, y, y_1, y_2 \in \mathbb{R}^n$ and $\alpha \in \mathbb{R}$ that

$$\langle \alpha x_1 + x_2, y\rangle = \alpha\langle x_1, y\rangle + \langle x_2, y\rangle, \qquad \langle x, \alpha y_1 + y_2\rangle = \alpha\langle x, y_1\rangle + \langle x, y_2\rangle.$$

Further we have that $\langle\cdot,\cdot\rangle$ is symmetric, i.e., $\langle x, y\rangle = \langle y, x\rangle$ for any $x, y \in \mathbb{R}^n$.

Definition 1.6. We make the following definitions regarding orthogonality.

(i) Two vectors $x, y \in \mathbb{R}^n$ are called orthogonal, denoted $x \perp y$, iff $\langle x, y\rangle = 0$.

(ii) Two sets $X, Y \subseteq \mathbb{R}^n$ are called orthogonal, denoted $X \perp Y$, iff $x \perp y \ \forall x \in X, y \in Y$.

(iii) A set $S \subseteq \mathbb{R}^n\backslash\{0\}$ is called orthogonal iff $\forall x, y \in S : x \neq y \Longrightarrow x \perp y$.

(iv) A set $S \subseteq \mathbb{R}^n\backslash\{0\}$ is called orthonormal iff S is orthogonal and $\|x\|_2 = 1 \ \forall x \in S$.

Theorem 1.6. *The vectors in an orthogonal set $S \subseteq \mathbb{R}^n\backslash\{0\}$ are linearly independent. In particular, any orthogonal set $S \subseteq \mathbb{R}^n\backslash\{0\}$ containing n vectors is a basis for $\mathbb{R}^n$.*

Proof. Let $S = \{v_1, \dots, v_N\} \subseteq \mathbb{R}^n \backslash \{0\}$ be an orthogonal set and suppose that its elements were linearly dependent. Then, there exists a vector $v_k \in S$ which can be expressed as $v_k = \sum_{i \in \{1,\dots,N\} \backslash \{k\}} c_i v_i$ for some $c_i \in \mathbb{R}$, $i \in \{1, \dots, N\} \backslash \{k\}$, and we find that

$$\|v_k\|_2^2 = \langle v_k, v_k \rangle = \sum_{i \in \{1,\dots,N\} \backslash \{k\}} c_i \langle v_i, v_k \rangle = 0$$

as S is orthogonal. But this implies that $v_k = 0$, contradicting $v_k \in S \subseteq \mathbb{R}^n \backslash \{0\}$. □

With the concept of orthogonality at hand, we can decompose a given vector into orthogonal components. Indeed, given an arbitrary vector $x \in \mathbb{R}^n$ and an orthonormal set $\{q_1, q_2, \dots, q_N\} \subseteq \mathbb{R}^n \backslash \{0\}$, $1 \le N \le n$, we set $r := x - \sum_{k=1}^N \langle x, q_k \rangle q_k \in \mathbb{R}^n$ and write

$$x = \sum_{k=1}^N \langle x, q_k \rangle q_k + r = \sum_{k=1}^N (q_k q_k^{\mathrm{T}}) x + r.$$

Then $\{r\} \perp \{q_1, \dots, q_N\}$ as we have for any $i \in \{1, \dots, N\}$ that

$$\langle r, q_i \rangle = \langle x, q_i \rangle - \sum_{k=1}^N \langle x, q_k \rangle \langle q_k, q_i \rangle = \langle x, q_i \rangle - \langle x, q_i \rangle = 0, \tag{1.2}$$

and we deduce that r is the part of x orthogonal to the subspace $\operatorname{span}(q_1, \dots, q_N) \subseteq \mathbb{R}^n$, and $\langle x, q_k \rangle q_k = (q_k q_k^{\mathrm{T}}) x$ is the part of x in direction q_k for $k \in \{1, \dots, N\}$. We will see later that $P_q := qq^{\mathrm{T}}$ is an *orthogonal projector* isolating the component in direction $q \in \mathbb{R}^n$. Observe that if $N = n$, we have that $\{q_1, \dots, q_n\}$ is a basis of $\mathbb{R}^n$ and hence, $r = 0$.

Remark 1.5. Let $Q = (q_1 | q_2 | \cdots | q_n) \in \mathbb{R}^{n \times n}$ be orthogonal. Then $\{q_1, \dots, q_n\} \subset \mathbb{R}^n$ is an orthonormal basis of $\mathbb{R}^n$. Indeed, $Q^{\mathrm{T}} Q = I_n$ yields that $q_i^{\mathrm{T}} q_j = \delta_{ij}$ for all $i, j \in \{1, \dots, n\}$.

Here, δ_{ij} denotes the Kronecker delta, i.e., $\delta_{ij} = 1$ if $i = j$, and $\delta_{ij} = 0$ if $i \ne j$.

Remark 1.6. The Euclidean inner product is invariant under orthogonal transformations, i.e., for an orthogonal matrix $Q \in \mathbb{R}^{n \times n}$ there holds $\langle Qx, Qy \rangle = \langle x, y \rangle$ for any $x, y \in \mathbb{R}^n$. In particular, we have $\|Qx\|_2 = \|x\|_2$ for any $x \in \mathbb{R}^n$.

Let us also note that $|\det(Q)| = 1$ for an orthognal matrix $Q \in \mathbb{R}^{n \times n}$. The associated linear map L_Q is an orthogonal transformation preserving the inner product on $\mathbb{R}^n$, and corresponds to a rigid rotation (when $\det(Q) = 1$) or a reflection (when $\det(Q) = -1$) of the space. In dimension $n = 2$, we can characterize orthogonal matrices as follows.

Remark 1.7. Any orthogonal 2×2 matrix $Q \in \mathbb{R}^{2 \times 2}$ with $\det(Q) = 1$ can be written as

$$Q = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}, \quad \theta \in [0, 2\pi),$$

with L_Q rotating the plane anticlockwise by the angle θ, and any orthogonal 2×2 matrix $Q \in \mathbb{R}^{2 \times 2}$ with $\det(Q) = -1$ can be written as

$$Q = \begin{pmatrix} \cos(\beta) & \sin(\beta) \\ \sin(\beta) & -\cos(\beta) \end{pmatrix}, \quad \beta \in [0, 2\pi),$$

with L_Q reflecting the plane across $y = \tan(\beta/2)x$ if $\beta \ne \pi$, and across $x = 0$ if $\beta = \pi$.

1.2 Norms

Definition 1.7. A map $\|\cdot\|: V \to [0,\infty)$ from a vector space V over $\mathbb{R}$ (or $\mathbb{C}$) into the set of non-negative real numbers is called a norm iff there holds

(i) definiteness: $\forall v \in V : \|v\| = 0 \Longrightarrow v = 0$,

(ii) absolute homogeneity: $\|\alpha v\| = |\alpha|\|v\| \quad \forall v \in V, \alpha \in \mathbb{R}$ (or $\mathbb{C}$),

(iii) triangle inequality: $\|v_1 + v_2\| \leq \|v_1\| + \|v_2\| \quad \forall v_1, v_2 \in V$.

If $V = \mathbb{R}^n$, we say $\|\cdot\|$ is a vector norm, and if $V = \mathbb{R}^{m\times n}$, we say $\|\cdot\|$ is a matrix norm.

1.2.1 Vector norms

The most important vector norms are the p-norms $\|\cdot\|_p$, including the Euclidean norm for $p = 2$.

Definition 1.8. For $p \in [1,\infty)$, we define the p-norm $\|\cdot\|_p : \mathbb{R}^n \to [0,\infty)$ given by

$$\|x\|_p := \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}, \qquad x = (x_1, x_2, \ldots, x_n)^{\mathrm{T}} \in \mathbb{R}^n.$$

Further, we define the ∞-norm (or maximum norm) $\|\cdot\|_\infty : \mathbb{R}^n \to [0,\infty)$ given by

$$\|x\|_\infty := \max_{i\in\{1,\ldots,n\}} |x_i|, \qquad x = (x_1, x_2, \ldots, x_n)^{\mathrm{T}} \in \mathbb{R}^n.$$

Remark 1.8. In dimension $n = 1$, we have $\|x\|_p = |x| \; \forall x \in \mathbb{R}$ for any $p \in [1,\infty) \cup \{\infty\}$.

Lemma 1.1. *Let $p, q \in (1,\infty)$ be such that there holds $\frac{1}{p} + \frac{1}{q} = 1$. Then, there holds $ab \leq \frac{1}{p}a^p + \frac{1}{q}b^q$ for any $a, b \in [0,\infty)$. This inequality is called Young's inequality.*

Proof. Let us assume that $a, b \in (0,\infty)$ as the claim is trivial if $a = 0$ or $b = 0$. Let us note that the exponential function $\exp : \mathbb{R} \to \mathbb{R}$ is convex, i.e., for any $\alpha \in [0,1]$ and $x, y \in \mathbb{R}$ we have $\exp(\alpha x + (1-\alpha)y) \leq \alpha \exp(x) + (1-\alpha)\exp(y)$. We find that

$$\begin{aligned} ab = \exp(\log(ab)) &= \exp(p^{-1}(p\log(a)) + (1-p^{-1})(q\log(b))) \\ &\leq p^{-1}\exp(p\log(a)) + (1-p^{-1})\exp(q\log(b)) = \frac{a^p}{p} + \frac{b^q}{q}, \end{aligned}$$

where log denotes the natural logarithm. □

Theorem 1.7. *Let $p, q \in (1,\infty)$ be such that there holds $\frac{1}{p} + \frac{1}{q} = 1$. Then, for any two vectors $x = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}, y = (y_1, y_2, \ldots, y_n)^{\mathrm{T}} \in \mathbb{R}^n$ there holds*

$$|\langle x, y\rangle| = \left|\sum_{i=1}^{n} x_i y_i\right| \leq \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}} \left(\sum_{i=1}^{n} |y_i|^q\right)^{\frac{1}{q}} = \|x\|_p \|y\|_q.$$

This inequality is called Hölder's inequality. The special case $p = q = 2$ is also known as the Cauchy–Schwarz inequality.

Proof. Let us assume that $x, y \in \mathbb{R}^n \backslash \{0\}$ as the claim is trivial if $x = 0$ or $y = 0$. Then we have

$$\frac{|\langle x, y\rangle|}{\|x\|_p\|y\|_q} \leq \sum_{i=1}^{n} \frac{|x_i|}{\|x\|_p} \frac{|y_i|}{\|y\|_q} \leq \frac{1}{p} \frac{\sum_{i=1}^{n}|x_i|^p}{\|x\|_p^p} + \frac{1}{q} \frac{\sum_{i=1}^{n}|y_i|^q}{\|y\|_q^q} = \frac{1}{p} + \frac{1}{q} = 1,$$

where we have used Young's inequality from Lemma 1.1. □

Remark 1.9. We also have that $|\langle x, y\rangle| \leq \|x\|_1\|y\|_\infty$ for any $x, y \in \mathbb{R}^n$.

Theorem 1.8. *The map $\|\cdot\|_p : \mathbb{R}^n \to [0, \infty)$ is indeed a norm for any $p \in [1, \infty) \cup \{\infty\}$.*

Proof. The tricky part of the proof is the triangle inequality for $p \in (1, \infty)$, and we leave the remaining parts as an exercise. Let $p \in (1, \infty)$ and set $q := \frac{p}{p-1}$. Then $\frac{1}{p} + \frac{1}{q} = 1$ and, using Hölder's inequality from Theorem 1.7, we find for any $x = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}, y = (y_1, y_2, \ldots, y_n)^{\mathrm{T}} \in \mathbb{R}^n$ that there holds

$$\begin{aligned}
\|x + y\|_p^p = \sum_{i=1}^{n}|x_i + y_i|^p &\leq \sum_{i=1}^{n}|x_i||x_i + y_i|^{p-1} + \sum_{i=1}^{n}|y_i||x_i + y_i|^{p-1} \\
&\leq (\|x\|_p + \|y\|_p) \left(\sum_{i=1}^{n}|x_i + y_i|^{(p-1)q}\right)^{\frac{1}{q}} \\
&= (\|x\|_p + \|y\|_p)\|x + y\|_{(p-1)q}^{p-1} = (\|x\|_p + \|y\|_p)\|x + y\|_p^{p-1},
\end{aligned}$$

and hence, $\|x + y\|_p \leq \|x\|_p + \|y\|_p$. □

All vector norms are equivalent in the sense of the following result.

Theorem 1.9. *Let $\|\cdot\|, |\!|\!|\cdot|\!|\!| : \mathbb{R}^n \to [0, \infty)$ be norms on $\mathbb{R}^n$. Then, $\|\cdot\|$ and $|\!|\!|\cdot|\!|\!|$ are equivalent, that is, there exist constants $C_1, C_2 > 0$ such that*

$$C_1\|x\| \leq |\!|\!|x|\!|\!| \leq C_2\|x\| \qquad \forall x \in \mathbb{R}^n.$$

Proof. See undergraduate linear algebra. Actually, any two norms on a finite dimensional space are equivalent. □

1.2.2 Induced matrix norms

As a first observation, note that for $A = (a_1|a_2|\cdots|a_n) \in \mathbb{R}^{m\times n}$, we have that

$$\operatorname{vec}(A) := \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} \in \mathbb{R}^{mn}, \qquad (\text{note } a_i \in \mathbb{R}^m \ \forall i \in \{1, \ldots, n\})$$

and we can use the aforementioned vector norms to measure its size. However, it is more useful to view $A \in \mathbb{R}^{m\times n}$ in terms of the associated linear operator $L_A \in \mathcal{L}(\mathbb{R}^n, \mathbb{R}^m)$ from (1.1) and use the operator norm induced by given vector norms on $\mathbb{R}^n$ and $\mathbb{R}^m$.

Definition 1.9. Consider the normed vector spaces $(\mathbb{R}^n, \|\cdot\|_{(n)})$ and $(\mathbb{R}^m, \|\cdot\|_{(m)})$, i.e., $\|\cdot\|_{(n)}$ is a vector norm on $\mathbb{R}^n$ and $\|\cdot\|_{(m)}$ is a vector norm on $\mathbb{R}^m$. Then we define the induced matrix norm $\|\cdot\|_{(m,n)} : \mathbb{R}^{m\times n} \to [0,\infty)$ by

$$\|A\|_{(m,n)} := \sup_{x\in\mathbb{R}^n\setminus\{0\}} \frac{\|Ax\|_{(m)}}{\|x\|_{(n)}} = \sup_{\substack{x\in\mathbb{R}^n \\ \|x\|_{(n)}=1}} \|Ax\|_{(m)}, \qquad A \in \mathbb{R}^{m\times n}.$$

In the case that $\|\cdot\|_{(n)} = \|\cdot\|_{(m)} = \|\cdot\|_p$ for $p \in [1,\infty) \cup \{\infty\}$, we call

$$\|A\|_p := \sup_{x\in\mathbb{R}^n\setminus\{0\}} \frac{\|Ax\|_p}{\|x\|_p} = \sup_{\substack{x\in\mathbb{R}^n \\ \|x\|_p=1}} \|Ax\|_p, \qquad A \in \mathbb{R}^{m\times n}$$

the p-norm of a matrix.

Theorem 1.10. *The map $\|\cdot\|_{(m,n)} : \mathbb{R}^{m\times n} \to [0,\infty)$ is a norm on $\mathbb{R}^{m\times n}$ for any choice of vector norms $\|\cdot\|_{(n)}$ on $\mathbb{R}^n$ and $\|\cdot\|_{(m)}$ on $\mathbb{R}^m$.*

Proof. Exercise. □

Remark 1.10. For $A \in \mathbb{R}^{m\times n}$, the number $\|A\|_{(m,n)}$ is the smallest constant $C \geq 0$ such that $\|L_A(x)\|_{(m)} = \|Ax\|_{(m)} \leq C\|x\|_{(n)}$ $\forall x \in \mathbb{R}^n$, i.e., it is the greatest factor by which L_A can stretch a vector in $\mathbb{R}^n$.

Remark 1.11. For $n_1, n_2, n_3 \in \mathbb{N}$ let $\|\cdot\|_{(n_k)}$ be a norm on $\mathbb{R}^{n_k}$, and let $A \in \mathbb{R}^{n_1\times n_2}$ and $C \in \mathbb{R}^{n_2\times n_3}$. Then we have that

$$\|AC\|_{(n_1,n_3)} \leq \|A\|_{(n_1,n_2)}\|C\|_{(n_2,n_3)}.$$

Indeed, this follows from

$$\|ACx\|_{(n_1)} \leq \|A\|_{(n_1,n_2)}\|Cx\|_{(n_2)} \leq \|A\|_{(n_1,n_2)}\|C\|_{(n_2,n_3)}\|x\|_{(n_3)} \qquad \forall x \in \mathbb{R}^{n_3}.$$

Therefore, induced matrix norms are said to be submultiplicative. Note that general matrix norms do not need to be submultiplicative (exercise).

Example 1.1. For a diagonal matrix

$$A := \operatorname{diag}(\alpha_1, \alpha_2, \ldots, \alpha_n) := \begin{pmatrix} \alpha_1 & 0 & \cdots & 0 \\ 0 & \alpha_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha_n \end{pmatrix} \in \mathbb{R}^{n\times n},$$

we have that $\|A\|_p = \max_{i\in\{1,\ldots,n\}}|\alpha_i|$ for all $p \in [1,\infty) \cup \{\infty\}$.

Proof. First, consider $p \in [1,\infty)$. Then, for any $x = (x_1, x_2, \ldots, x_n)^{\mathrm{T}} \in \mathbb{R}^n$, we have that

$$\|Ax\|_p^p = \sum_{i=1}^n |\alpha_i x_i|^p \leq \left(\max_{i\in\{1,\ldots,n\}}|\alpha_i|^p\right) \sum_{i=1}^n |x_i|^p = \left(\max_{i\in\{1,\ldots,n\}}|\alpha_i|\right)^p \|x\|_p^p,$$

and hence, $\|A\|_p \leq \max_{i\in\{1,\dots,n\}}|\alpha_i|$. For the converse inequality, we use the canonical basis vectors to find

$$\|A\|_p \geq \frac{\|Ae_i\|_p}{\|e_i\|_p} = \frac{\|\alpha_i e_i\|_p}{\|e_i\|_p} = |\alpha_i| \qquad \forall i \in \{1,\dots,n\},$$

and hence, $\|A\|_p \geq \max_{i\in\{1,\dots,n\}}|\alpha_i|$ and we can conclude that $\|A\|_p = \max_{i\in\{1,\dots,n\}}|\alpha_i|$. Now consider $p=\infty$. Then, for any $x=(x_1,x_2,\dots,x_n)^{\mathrm{T}} \in \mathbb{R}^n$, we have that

$$\|Ax\|_\infty = \max_{i\in\{1,\dots,n\}}|\alpha_i x_i| \leq \left(\max_{i\in\{1,\dots,n\}}|\alpha_i|\right)\left(\max_{i\in\{1,\dots,n\}}|x_i|\right) = \left(\max_{i\in\{1,\dots,n\}}|\alpha_i|\right)\|x\|_\infty,$$

and hence, $\|A\|_\infty \leq \max_{i\in\{1,\dots,n\}}|\alpha_i|$. The converse is shown as before. □

Example 1.2. For a matrix $A=(a_1|a_2|\cdots|a_n)=(b_1|b_2|\cdots|b_m)^{\mathrm{T}} \in \mathbb{R}^{m\times n}$ there holds

$$\|A\|_\infty = \max_{i\in\{1,\dots,m\}}\|b_i\|_1, \qquad \|A\|_1 = \max_{j\in\{1,\dots,n\}}\|a_j\|_1,$$

i.e., $\|A\|_\infty$ is the "maximum row sum" and $\|A\|_1$ the "maximum column sum" of A.

Proof. We leave the claimed result for the ∞-norm as an exercise and only prove that $\|A\|_1 = \max_{i\in\{1,\dots,n\}}\|a_i\|_1$ for $A=(a_1|a_2|\cdots|a_n) \in \mathbb{R}^{m\times n}$. For any vector $x=(x_1,x_2,\dots,x_n)^{\mathrm{T}} \in \mathbb{R}^n$, we have that

$$\|Ax\|_1 = \left\|\sum_{i=1}^n x_i a_i\right\|_1 \leq \sum_{i=1}^n |x_i|\|a_i\|_1 \leq \left(\max_{i\in\{1,\dots,n\}}\|a_i\|_1\right)\sum_{i=1}^n|x_i| = \left(\max_{i\in\{1,\dots,n\}}\|a_i\|_1\right)\|x\|_1$$

and hence, $\|A\|_1 \leq \max_{i\in\{1,\dots,n\}}\|a_i\|_1$. For the converse inequality, note that

$$\|A\|_1 \geq \frac{\|Ae_i\|_1}{\|e_i\|_1} = \frac{\|a_i\|_1}{\|e_i\|_1} = \|a_i\|_1 \qquad \forall i \in \{1,\dots,n\},$$

and hence, $\|A\|_1 \geq \max_{i\in\{1,\dots,n\}}\|a_i\|_1$. We conclude that $\|A\|_1 = \max_{i\in\{1,\dots,n\}}\|a_i\|_1$. □

Example 1.3. For a row vector $A=a^{\mathrm{T}} \in \mathbb{R}^{1\times n}$ there holds $\|A\|_2 = \|a\|_2$.

Proof. For $A=a^{\mathrm{T}} \in \mathbb{R}^{1\times n}$, we have $\|Ax\|_2 = |\langle a,x\rangle| \leq \|a\|_2\|x\|_2$ for any $x\in\mathbb{R}^n$ and hence, $\|A\|_2 \leq \|a\|_2$. If $a=0\in\mathbb{R}^n$, we have $\|A\|_2 \leq 0$ which yields $\|A\|_2 = 0 = \|a\|_2$. If $a\in\mathbb{R}^n\backslash\{0\}$, there holds $\|A\|_2 \geq \frac{\|Aa\|_2}{\|a\|_2} = \frac{|\langle a,a\rangle|}{\|a\|_2} = \|a\|_2$ and we conclude $\|A\|_2 = \|a\|_2$. □

Example 1.4. Let $u\in\mathbb{R}^m$ and $v\in\mathbb{R}^n$. Then, for the *outer product* $A=uv^{\mathrm{T}} \in \mathbb{R}^{m\times n}$ we have that $\|A\|_2 = \|u\|_2\|v\|_2$.

Proof. We have $\|Ax\|_2 = \|uv^{\mathrm{T}}x\|_2 = \|u\|_2|\langle v,x\rangle| \leq \|u\|_2\|v\|_2\|x\|_2$ for any $x\in\mathbb{R}^n$ and hence, $\|A\|_2 \leq \|u\|_2\|v\|_2$. If $v=0$, we have $\|A\|_2 \leq 0$ which yields $\|A\|_2 = 0 = \|u\|_2\|v\|_2$. If $v\in\mathbb{R}^n\backslash\{0\}$, there holds $\|A\|_2 \geq \frac{\|Av\|_2}{\|v\|_2} = \frac{\|uv^{\mathrm{T}}v\|_2}{\|v\|_2} = \frac{\|u\|_2|\langle v,v\rangle|}{\|v\|_2} = \|u\|_2\|v\|_2$ and we conclude that $\|A\|_2 = \|u\|_2\|v\|_2$. □

The matrix 2-norm is also known as the spectral norm. We will see later that there holds $\|A\|_2 = \sqrt{\lambda_{\max}(A^{\mathrm{T}}A)}$ for $A\in\mathbb{R}^{m\times n}$, where $\lambda_{\max}(A^{\mathrm{T}}A)$ denotes the largest eigenvalue of $A^{\mathrm{T}}A$.

1.2.3 Frobenius norm

Let us note that not all matrix norms are induced by vector norms. The most important example of such a norm is the Frobenius norm.

Definition 1.10. The map $\|\cdot\|_F : \mathbb{R}^{m\times n} \to [0,\infty)$ given by

$$\|A\|_F := \sqrt{\operatorname{tr}(A^{\mathrm{T}}A)} = \sqrt{\operatorname{tr}(AA^{\mathrm{T}})} = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}, \qquad A = (a_{ij}) \in \mathbb{R}^{m\times n},$$

is called the Frobenius norm.

Here, $\operatorname{tr}(B)$ denotes the trace of a square matrix B, that is, the sum of its diagonal entries.

Theorem 1.11. *The map $\|\cdot\|_F$ is indeed a norm on $\mathbb{R}^{m\times n}$. Further, the Frobenius norm is submultiplicative, that is,*

$$\|AC\|_F \le \|A\|_F\|C\|_F \qquad \forall A \in \mathbb{R}^{m\times n}, C \in \mathbb{R}^{n\times l}.$$

Proof. Exercise. □

Remark 1.12. The Frobenius norm $\|\cdot\|_F$ is induced by the *Frobenius inner product* $\langle\cdot,\cdot\rangle_F : \mathbb{R}^{m\times n} \times \mathbb{R}^{m\times n} \to \mathbb{R}$ given by

$$\langle A, B\rangle_F := \operatorname{tr}(A^{\mathrm{T}}B) = \operatorname{tr}(BA^{\mathrm{T}}) = \sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}b_{ij}, \qquad A = (a_{ij}), B = (b_{ij}) \in \mathbb{R}^{m\times n},$$

i.e., $\|A\|_F = \sqrt{\langle A, A\rangle_F}$ $\forall A \in \mathbb{R}^{m\times n}$. Further, we have the Cauchy–Schwarz inequality

$$|\langle A, B\rangle_F| \le \|A\|_F\|B\|_F \qquad \forall A, B \in \mathbb{R}^{m\times n}.$$

We point out that, as for vector norms, also any two matrix norms are equivalent in the sense of the following result.

Theorem 1.12. *Let $\|\cdot\|, |||\cdot||| : \mathbb{R}^{m\times n} \to [0,\infty)$ be norms on $\mathbb{R}^{m\times n}$. Then, $\|\cdot\|$ and $|||\cdot|||$ are equivalent, that is, there exist constants $C_1, C_2 > 0$ such that*

$$C_1\|A\| \le |||A||| \le C_2\|A\| \qquad \forall A \in \mathbb{R}^{m\times n}.$$

Proof. See undergraduate linear algebra (any two norms on a finite dimensional space are equivalent). □

1.2.4 Orthogonal invariance

The spectral norm and the Frobenius norm are invariant under multiplication by orthogonal matrices:

Theorem 1.13. *Let $A \in \mathbb{R}^{m\times n}$. Further, let $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ be two orthogonal matrices. Then, we have that*

(i) $\|UA\|_2 = \|A\|_2$ *and* $\|AV\|_2 = \|A\|_2$,

(ii) $\|UA\|_F = \|A\|_F$ *and* $\|AV\|_F = \|A\|_F$.

Proof. (i) In view of Remark 1.6 there holds $\|Vx\|_2 = \|x\|_2$ for any $x \in \mathbb{R}^n$, and $\|Uy\|_2 = \|y\|_2$ for any $y \in \mathbb{R}^m$. We have that

$$\|UA\|_2 = \sup_{x \in \mathbb{R}^n \setminus \{0\}} \frac{\|UAx\|_2}{\|x\|_2} = \sup_{x \in \mathbb{R}^n \setminus \{0\}} \frac{\|Ax\|_2}{\|x\|_2} = \|A\|_2$$

and, using that $L_V : \mathbb{R}^n \to \mathbb{R}^n$ is a bijection (as V is invertible),

$$\|AV\|_2 = \sup_{x \in \mathbb{R}^n \setminus \{0\}} \frac{\|AVx\|_2}{\|x\|_2} = \sup_{x \in \mathbb{R}^n \setminus \{0\}} \frac{\|AVx\|_2}{\|Vx\|_2} = \sup_{\tilde{x} \in \mathbb{R}^n \setminus \{0\}} \frac{\|A\tilde{x}\|_2}{\|\tilde{x}\|_2} = \|A\|_2.$$

(ii) Note that $\|B\|_F^2 = \mathrm{tr}(B^{\mathrm{T}}B) = \mathrm{tr}(BB^{\mathrm{T}})$ for any $B \in \mathbb{R}^{m \times n}$. Hence, we have that

$$\begin{aligned}
\|UA\|_F^2 &= \mathrm{tr}((UA)^{\mathrm{T}}(UA)) = \mathrm{tr}(A^{\mathrm{T}}U^{\mathrm{T}}UA) = \mathrm{tr}(A^{\mathrm{T}}A) = \|A\|_F^2, \\
\|AV\|_F^2 &= \mathrm{tr}((AV)(AV)^{\mathrm{T}}) = \mathrm{tr}(AVV^{\mathrm{T}}A^{\mathrm{T}}) = \mathrm{tr}(AA^{\mathrm{T}}) = \|A\|_F^2.
\end{aligned}$$

□

2 Singular Value Decomposition

2.1 Definition and geometric interpretation

In this section, we introduce the singular value decomposition and provide a geometric interpretation.

Definition of full and reduced SVD

We start by introducing some notation.

Definition 2.1. A matrix $A \in \mathbb{R}^{m\times n}$ is called diagonal iff there exist $p := \min(m, n)$ real numbers $\alpha_1, \ldots, \alpha_p \in \mathbb{R}$ such that $A = \operatorname{diag}_{m\times n}(\alpha_1, \alpha_2, \ldots, \alpha_p)$, where $\operatorname{diag}_{m\times n}$ is defined as follows.

(i) For $m, n \in \mathbb{N}$ with $m \geq n$ and $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$, we define

$$\operatorname{diag}_{m\times n}(\alpha_1, \alpha_2, \ldots, \alpha_n) := \begin{pmatrix} \alpha_1 & 0 & \cdots & 0 \\ 0 & \alpha_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha_n \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} \in \mathbb{R}^{m\times n}.$$

(ii) For $m, n \in \mathbb{N}$ with $m < n$ and $\alpha_1, \ldots, \alpha_m \in \mathbb{R}$, we define

$$\operatorname{diag}_{m\times n}(\alpha_1, \alpha_2, \ldots, \alpha_m) := \begin{pmatrix} \alpha_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \alpha_2 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha_m & 0 & \cdots & 0 \end{pmatrix} \in \mathbb{R}^{m\times n}.$$

A singular value decomposition, abbreviated SVD, is defined as follows.

Definition 2.2. Let $A \in \mathbb{R}^{m\times n}$ for some $m, n \in \mathbb{N}$ and set $p := \min(m, n)$. If there exist

$$\begin{aligned} U &= (u_1|u_2|\cdots|u_m) && \in \mathbb{R}^{m\times m} && \text{orthogonal}, \\ V &= (v_1|v_2|\cdots|v_n) && \in \mathbb{R}^{n\times n} && \text{orthogonal}, \\ \Sigma &= \operatorname{diag}_{m\times n}(\sigma_1, \sigma_2, \ldots, \sigma_p) && \in \mathbb{R}^{m\times n} && \text{with } \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_p \geq 0, \end{aligned}$$

such that there holds $U^{\mathrm{T}} A V = \Sigma$, or equivalently,

$$A = U\Sigma V^{\mathrm{T}}, \tag{2.1}$$

then we call (2.1) a singular value decomposition (SVD) of A with singular values $\sigma_1, \ldots, \sigma_p \geq 0$, left singular vectors $u_1, \ldots, u_m \in \mathbb{R}^m$, and right singular vectors $v_1, \ldots, v_n \in \mathbb{R}^n$.

Remark 2.1. The SVD (2.1) can be simplified to

$$A = \hat{U}\hat{\Sigma}\hat{V}^{\mathrm{T}}$$

with $\hat{U} := (u_1|\cdots|u_p) \in \mathbb{R}^{m\times p}$, $\hat{\Sigma} := \mathrm{diag}_{p\times p}(\sigma_1,\ldots,\sigma_p) \in \mathbb{R}^{p\times p}$, $\hat{V} = (v_1|\cdots|v_p) \in \mathbb{R}^{n\times p}$. We call such a decomposition $A = \hat{U}\hat{\Sigma}\hat{V}^{\mathrm{T}}$ with $\hat{U} \in \mathbb{R}^{m\times p}$ and $\hat{V} \in \mathbb{R}^{n\times p}$ having orthonormal columns, and $\hat{\Sigma} \in \mathbb{R}^{p\times p}$ being diagonal with non-negative and non-increasing diagonal entries, a *reduced SVD* of A.

Proof. If $m \geq n$ (i.e., $p = n$), we have with $\hat{U} = (u_1|\cdots|u_n)$ and $\hat{\Sigma} := \mathrm{diag}_{n\times n}(\sigma_1,\ldots,\sigma_n)$ that (note $\hat{V} = V$)

$$U\Sigma V^{\mathrm{T}} = (u_1|\cdots|u_m)\begin{pmatrix}\hat{\Sigma} \\ \hline 0_{(m-n)\times n}\end{pmatrix}(v_1|\cdots|v_n)^{\mathrm{T}} = \hat{U}\hat{\Sigma}V^{\mathrm{T}}.$$

If $m < n$ (i.e., $p = m$), we have with $\hat{\Sigma} := \mathrm{diag}_{m\times m}(\sigma_1,\ldots,\sigma_m)$ and $\hat{V} = (v_1|\cdots|v_m)$ that (note $\hat{U} = U$)

$$U\Sigma V^{\mathrm{T}} = (u_1|\cdots|u_m)\left(\hat{\Sigma}\,\middle|\,0_{m\times(n-m)}\right)(v_1|\cdots|v_n)^{\mathrm{T}} = U\hat{\Sigma}\hat{V}^{\mathrm{T}}.$$

□

Note that in the proof of Remark 2.1, we have used the notation $0_{r\times s}$ to denote the $r \times s$ zero-matrix $0 \in \mathbb{R}^{r\times s}$.

Example 2.1. Two examples of SVDs for rectangular matrices are

$$\begin{pmatrix}1 & 0 & -1 & 1\\ -1 & 1 & 0 & 1\end{pmatrix} = \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}\begin{pmatrix}\sqrt{3} & 0 & 0 & 0\\ 0 & \sqrt{3} & 0 & 0\end{pmatrix}\begin{pmatrix}\frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}}\\ 0 & \frac{1}{\sqrt{3}} & -\frac{2}{\sqrt{6}} & 0\\ -\frac{1}{\sqrt{3}} & 0 & 0 & \frac{2}{\sqrt{6}}\\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}}\end{pmatrix}^{\mathrm{T}},$$

$$\begin{pmatrix}1 & 1\\ -1 & 1\\ 1 & 1\\ 0 & 0\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}} & 0 & 0 & -\frac{1}{\sqrt{2}}\\ 0 & 1 & 0 & 0\\ \frac{1}{\sqrt{2}} & 0 & 0 & \frac{1}{\sqrt{2}}\\ 0 & 0 & 1 & 0\end{pmatrix}\begin{pmatrix}2 & 0\\ 0 & \sqrt{2}\\ 0 & 0\\ 0 & 0\end{pmatrix}\begin{pmatrix}\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\end{pmatrix}^{\mathrm{T}},$$

with corresponding reduced SVDs

$$\begin{pmatrix}1 & 0 & -1 & 1\\ -1 & 1 & 0 & 1\end{pmatrix} = \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}\begin{pmatrix}\sqrt{3} & 0\\ 0 & \sqrt{3}\end{pmatrix}\begin{pmatrix}\frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}}\\ 0 & \frac{1}{\sqrt{3}}\\ -\frac{1}{\sqrt{3}} & 0\\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}}\end{pmatrix}^{\mathrm{T}},$$

$$\begin{pmatrix}1 & 1\\ -1 & 1\\ 1 & 1\\ 0 & 0\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}} & 0\\ 0 & 1\\ \frac{1}{\sqrt{2}} & 0\\ 0 & 0\end{pmatrix}\begin{pmatrix}2 & 0\\ 0 & \sqrt{2}\end{pmatrix}\begin{pmatrix}\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\end{pmatrix}^{\mathrm{T}}.$$

Geometric interpretation

The geometric interpretation of the SVD is that the image of the 2-norm unit sphere $\{x \in \mathbb{R}^n | \, \|x\|_2 = 1\}$ under any $m \times n$ matrix is a hyperellipse. A hyperellipse in $\mathbb{R}^m$ is the m-dimensional generalization of en ellipse: it is the surface obtained by stretching the 2-norm unit sphere in $\mathbb{R}^m$ by some factors $\sigma_1, \ldots, \sigma_m \geq 0$ in the directions of orthonormal vectors $u_1, \ldots, u_m \in \mathbb{R}^m$.

Indeed, observe that a SVD $A = U\Sigma V^{\mathrm{T}}$ of a matrix $A \in \mathbb{R}^{m\times n}$ can be rewritten as

$$AV = (a_1|\cdots|a_n)(v_1|\cdots|v_n) = U\Sigma = (u_1|\cdots|u_m)\mathrm{diag}_{m\times n}(\sigma_1, \sigma_2, \ldots, \sigma_p)$$

due to orthogonality of V (recall that $p := \min(m, n)$). Therefore,

$$\begin{aligned} &\text{if } m \geq n: && Av_i = \sigma_i u_i \quad \forall i \in \{1, \ldots, n\}, \\ &\text{if } m < n: && Av_i = \sigma_i u_i \quad \forall i \in \{1, \ldots, m\}, \quad Av_j = 0 \quad \forall j \in \{m+1, \ldots, n\}. \end{aligned} \tag{2.2}$$

Let us consider the following explicit example of a SVD for a 2×2 square matrix:

$$A := \begin{pmatrix} 2 & 11 \\ 10 & -5 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 4\sqrt{10} & 0 \\ 0 & 3\sqrt{10} \end{pmatrix} \begin{pmatrix} -\frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \end{pmatrix}^{\mathrm{T}} =: U\Sigma V^{\mathrm{T}}, \tag{2.3}$$

with singular values $\sigma_1 = 4\sqrt{10}$, $\sigma_2 = 3\sqrt{10}$, left singular vectors $u_1 = (\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}})^{\mathrm{T}}$, $u_2 = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^{\mathrm{T}}$, and right singular vectors $v_1 = (-\frac{1}{\sqrt{5}}, \frac{2}{\sqrt{5}})^{\mathrm{T}}$, $v_2 = (\frac{2}{\sqrt{5}}, \frac{1}{\sqrt{5}})^{\mathrm{T}}$.

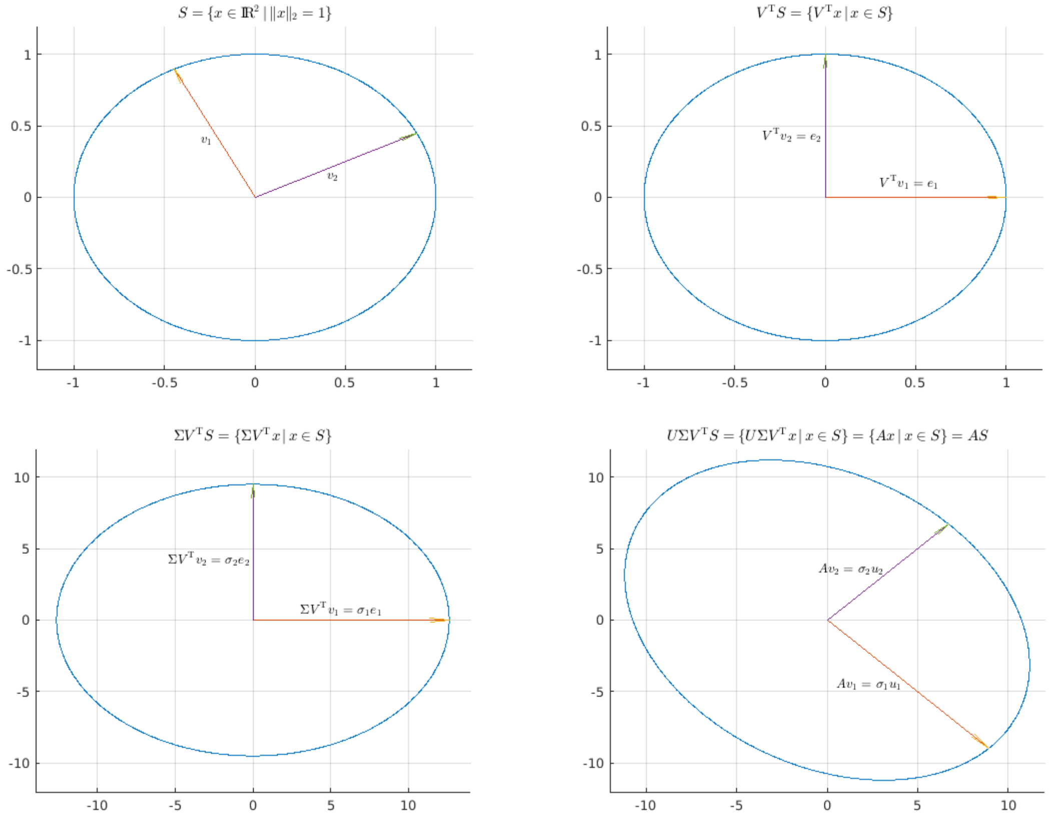

Figure 1: Illustration of the SVD $A = U\Sigma V^{\mathrm{T}}$ from (2.3).

In this example, first V^{T} reflects the unit sphere across the line $y = \frac{1+\sqrt{5}}{2}x$, preserving its shape, then Σ stretches the sphere into an ellipse aligned with e_1, e_2 by scaling the x-coordinate by σ_1 and the y-coordinate by σ_2, and finally U rotates the ellipse clockwise by the angle $\frac{\pi}{4}$ (recall Remark 1.7). Thus, we see that the image of $A = U\Sigma V^{\mathrm{T}}$, or more precisely the image of the associated linear map $L_A = L_U \circ L_\Sigma \circ L_{V^{\mathrm{T}}}$ is indeed an ellipse with principal semiaxes $\sigma_1 u_1$ and $\sigma_2 u_2$.

2.2 Existence and uniqueness

Theorem 2.1 (Existence result for SVD)**.** *Every matrix $A \in \mathbb{R}^{m\times n}$ has a SVD* (2.1)*, as defined in Definition 2.2.*

Proof. Step 1: We begin by setting

$$\sigma_1 := \|A\|_2 = \sup_{\substack{x\in\mathbb{R}^n \\ \|x\|_2=1}} \|Ax\|_2.$$

Observe that the 2-norm unit sphere $S := \{x \in \mathbb{R}^n : \|x\|_2 = 1\}$ is a compact subset of $\mathbb{R}^n$ and that the map $S \ni x \mapsto \|Ax\|_2 \in \mathbb{R}$ is continuous (exercise). Therefore, there exists a unit vector $v_1 \in \mathbb{R}^n$ with $\|v_1\|_2 = 1$ such that $\|Av_1\|_2 = \sigma_1$, and hence, there holds

$$Av_1 = \sigma_1 u_1 \tag{2.4}$$

for some unit vector $u_1 \in \mathbb{R}^m$ with $\|u_1\|_2 = 1$.

Step 2: Next, we construct orthogonal matrices $V_1 \in \mathbb{R}^{n\times n}$ and $U_1 \in \mathbb{R}^{m\times m}$ such that

$$U_1^{\mathrm{T}} A V_1 = \left(\begin{array}{c|c} \sigma_1 & 0_{1\times(n-1)} \\ \hline 0_{(m-1)\times 1} & B \end{array}\right) \in \mathbb{R}^{m\times n} \tag{2.5}$$

for some $B \in \mathbb{R}^{(m-1)\times(n-1)}$.

To this end, we extend v_1 to an orthonormal basis $\{v_1, \dots, v_n\} \subseteq \mathbb{R}^n$ of the space $\mathbb{R}^n$ and u_1 to an orthonormal basis $\{u_1, \dots, u_m\} \subseteq \mathbb{R}^m$ of the space $\mathbb{R}^m$, and set

$$V_1 := (v_1|\cdots|v_n) \in \mathbb{R}^{n\times n}, \qquad U_1 := (u_1|\cdots|u_m) \in \mathbb{R}^{m\times m}.$$

Then, the matrices V_1 and U_1 are orthogonal by construction, and we have from (2.4) that

$$U_1^{\mathrm{T}} A V_1 = \left(\begin{array}{c|c} \sigma_1 & w^{\mathrm{T}} \\ \hline 0_{(m-1)\times 1} & B \end{array}\right) =: A_1 \in \mathbb{R}^{m\times n}$$

for some $w \in \mathbb{R}^{n-1}$ and $B \in \mathbb{R}^{(m-1)\times(n-1)}$. We are done with Step 2 if we can show that $w = 0 \in \mathbb{R}^{n-1}$. In order to do so, we set $\tilde{w} := \begin{pmatrix} \sigma_1 \\ w \end{pmatrix} \in \mathbb{R}^n$ and note that

$$\begin{aligned} &\|A_1\tilde{w}\|_2^2 \\ &= \left\| \left(\begin{array}{c|c} \sigma_1 & w^{\mathrm{T}} \\ \hline 0_{(m-1)\times 1} & B \end{array}\right) \begin{pmatrix} \sigma_1 \\ w \end{pmatrix} \right\|_2^2 = \left\| \begin{pmatrix} \sigma_1^2 + w^{\mathrm{T}} w \\ Bw \end{pmatrix} \right\|_2^2 \geq (\sigma_1^2 + w^{\mathrm{T}}w)^2 = (\sigma_1^2 + w^{\mathrm{T}}w)\|\tilde{w}\|_2^2, \end{aligned}$$

which yields $\|A_1\|_2 \geq \sqrt{\sigma_1^2 + w^{\mathrm{T}}w}$. Since V_1 and U_1 are orthogonal (thus also U_1^{T} is orthogonal), we have from Theorem 1.13 that $\|A_1\|_2 = \|U_1^{\mathrm{T}} A V_1\|_2 = \|A\|_2 = \sigma_1$. Hence, $\sigma_1 \geq \sqrt{\sigma_1^2 + w^{\mathrm{T}}w} \geq \sigma_1$ and consequently, $w = 0 \in \mathbb{R}^{n-1}$ and we have (2.5).

Step 3: We conclude the proof using induction on the dimension of A. Note that, from (2.5) we have

$$\begin{aligned} U_1^{\mathrm{T}} A V_1 &= \left(\frac{\sigma_1}{O_{(m-1)\times 1}}\right) = \operatorname{diag}_{m\times 1}(\sigma_1) \quad \text{if } n = 1, \\ U_1^{\mathrm{T}} A V_1 &= (\sigma_1 |\, 0_{1\times(n-1)}) = \operatorname{diag}_{1\times n}(\sigma_1) \quad \text{if } m = 1, \end{aligned}$$

i.e., every $A \in \mathbb{R}^{m\times n}$ with $m = 1$ or $n = 1$ has a SVD. Now, let $A \in \mathbb{R}^{m\times n}$ with $m, n \in \mathbb{N}$ and $m, n \geq 2$. Then, in view of (2.5), and assuming the matrix $B \in \mathbb{R}^{(m-1)\times(n-1)}$ has a SVD $B = U_2\Sigma_2 V_2^{\mathrm{T}}$ with orthogonal $U_2 \in \mathbb{R}^{(m-1)\times(m-1)}$, $V_2 \in \mathbb{R}^{(n-1)\times(n-1)}$ and diagonal $\Sigma_2 \in \mathbb{R}^{(m-1)\times(n-1)}$ with non-increasing non-negative diagonal entries, we find that

$$\begin{aligned} U_1^{\mathrm{T}} A V_1 &= \left(\begin{array}{c|c} \sigma_1 & 0_{1\times(n-1)} \\ \hline 0_{(m-1)\times 1} & U_2\Sigma_2 V_2^{\mathrm{T}} \end{array}\right) \\ &= \left(\begin{array}{c|c} 1 & 0_{1\times(m-1)} \\ \hline 0_{(m-1)\times 1} & U_2 \end{array}\right) \left(\begin{array}{c|c} \sigma_1 & 0_{1\times(n-1)} \\ \hline 0_{(m-1)\times 1} & \Sigma_2 \end{array}\right) \left(\begin{array}{c|c} 1 & 0_{1\times(n-1)} \\ \hline 0_{(n-1)\times 1} & V_2 \end{array}\right)^{\mathrm{T}}, \end{aligned}$$

i.e., we conclude that

$$U^{\mathrm{T}} A V = \left(\begin{array}{c|c} \sigma_1 & 0_{1\times(n-1)} \\ \hline 0_{(m-1)\times 1} & \Sigma_2 \end{array}\right) =: \Sigma \in \mathbb{R}^{m\times n}$$

is diagonal with non-increasing non-negative entries, where the orthogonal matrices $U \in \mathbb{R}^{m\times m}$, $V \in \mathbb{R}^{n\times n}$ are given by

$$U := U_1 \left(\begin{array}{c|c} 1 & 0_{1\times(m-1)} \\ \hline 0_{(m-1)\times 1} & U_2 \end{array}\right) \in \mathbb{R}^{m\times m}, \qquad V := V_1 \left(\begin{array}{c|c} 1 & 0_{1\times(n-1)} \\ \hline 0_{(n-1)\times 1} & V_2 \end{array}\right) \in \mathbb{R}^{n\times n}.$$

It is quickly seen that U and V are indeed orthogonal, using that the product of two orthogonal matrices is orthogonal. (Note σ_1 is greater or equal than the diagonal entries of Σ_2 as, using orthogonal invariance of the spectral norm, $\sigma_1 = \|A\|_2 = \|U^{\mathrm{T}}AV\|_2 = \|\Sigma\|_2$, which equals to the maximum of the absolute values of the diagonal entries of Σ; see also Remark 2.3.) □

The natural question to ask is if the SVD to a given matrix is unique. This is not the case as can be seen from, e.g., the one-dimensional case.

Remark 2.2. Note that a matrix $Q = (q) \in \mathbb{R}^{1\times 1}$ is orthogonal iff $q \in \{-1, 1\}$. Therefore, a matrix $A = (a) \in \mathbb{R}^{1\times 1}$ must have the unique singular value $\sigma_1 = |a|$. The SVD is not unique as

$$\begin{aligned} (a) &= (1)(a)(1)^{\mathrm{T}} &&= (-1)(a)(-1)^{\mathrm{T}} && \text{if } a > 0 \\ (a) &= (1)(0)(1)^{\mathrm{T}} &&= (-1)(0)(-1)^{\mathrm{T}} = (-1)(0)(1)^{\mathrm{T}} = (1)(0)(-1)^{\mathrm{T}} && \text{if } a = 0 \\ (a) &= (-1)(-a)(1)^{\mathrm{T}} &&= (1)(-a)(-1)^{\mathrm{T}} && \text{if } a < 0 \end{aligned}$$

are SVDs for $A = (a)$. However, the left and right singular vectors are unique up to signs.

We can show that the singular values are uniquely determined for any given matrix $A \in \mathbb{R}^{m\times n}$. For square matrices $A \in \mathbb{R}^{n\times n}$ with distinct singular values, we can prove uniqueness for the left and right singular vectors up to signs.

Remark 2.3. Let us note that the largest singular value σ_1 is uniquely determined as for any $A \in \mathbb{R}^{m\times n}$ with SVD $A = U\Sigma V^{\mathrm{T}}$ we have that (recall $p := \min(m,n)$)

$$\|A\|_2 = \|U\Sigma V^{\mathrm{T}}\|_2 = \|\Sigma\|_2 = \|\mathrm{diag}_{m\times n}(\sigma_1, \sigma_2, \ldots, \sigma_p)\|_2 = \sigma_1,$$

where we have used orthogonal invariance of the spectral norm (Theorem 1.13) and the fact that $\|\mathrm{diag}_{m\times n}(\sigma_1, \sigma_2, \ldots, \sigma_p)\|_2 = \max_{i\in\{1,\ldots,p\}}|\sigma_i|$ (exercise, see Example 1.1 for square diagonal matrices).

Theorem 2.2 (Uniqueness result for SVD)**.** *The singular values $\{\sigma_i\}$ of any given matrix $A \in \mathbb{R}^{m\times n}$ are unique and we have*

$$\{\sigma_1^2, \ldots, \sigma_p^2\} = \begin{cases} \Lambda(A^{\mathrm{T}}A) & , \text{ if } m \geq n, \\ \Lambda(AA^{\mathrm{T}}) & , \text{ if } m < n. \end{cases} \tag{2.6}$$

Further, if $A \in \mathbb{R}^{n\times n}$ is square and the singular values are positive and distinct, then the left singular vectors $\{u_i\}$ and right singular vectors $\{v_i\}$ are unique up to signs.

Remark 2.4. What do we mean by "unique up to signs"? Recall that a SVD $A = U\Sigma V^{\mathrm{T}} = (u_1|\cdots|u_m)[\mathrm{diag}_{m\times n}(\sigma_1, \ldots, \sigma_p)](v_1|\cdots|v_n)^{\mathrm{T}}$ is equivalent to (2.2). So, one can always find another SVD by replacing a chosen v_i by $-v_i$ when also replacing u_i by $-u_i$. We claim that, if $A \in \mathbb{R}^{n\times n}$ is square and the singular values are positive and distinct, this is the only way of obtaining other SVDs.

Proof of Theorem 2.2. Uniqueness of the singular values: For $A \in \mathbb{R}^{m\times n}$ with SVD $A = U\Sigma V^{\mathrm{T}}$, we have

$$\begin{aligned} A^{\mathrm{T}}A &= V\Sigma^{\mathrm{T}}U^{\mathrm{T}}U\Sigma V^{\mathrm{T}} = V\Sigma^{\mathrm{T}}\Sigma V^{\mathrm{T}} \in \mathbb{R}^{n\times n}, \\ AA^{\mathrm{T}} &= U\Sigma V^{\mathrm{T}}V\Sigma^{\mathrm{T}}U^{\mathrm{T}} = U\Sigma\Sigma^{\mathrm{T}}U^{\mathrm{T}} \in \mathbb{R}^{m\times m}. \end{aligned}$$

Thus, $A^{\mathrm{T}}A$ is similar to $\Sigma^{\mathrm{T}}\Sigma$, and AA^{T} is similar to $\Sigma\Sigma^{\mathrm{T}}$. Note that

$$\begin{aligned} \Sigma^{\mathrm{T}}\Sigma &= \mathrm{diag}_{n\times n}(\sigma_1^2, \ldots, \sigma_n^2), \quad \Sigma\Sigma^{\mathrm{T}} = \mathrm{diag}_{m\times m}(\sigma_1^2, \ldots, \sigma_n^2, 0, \ldots, 0) \quad \text{if } m \geq n, \\ \Sigma^{\mathrm{T}}\Sigma &= \mathrm{diag}_{n\times n}(\sigma_1^2, \ldots, \sigma_m^2, 0, \ldots, 0), \quad \Sigma\Sigma^{\mathrm{T}} = \mathrm{diag}_{m\times m}(\sigma_1^2, \ldots, \sigma_m^2) \quad \text{if } m < n. \end{aligned} \tag{2.7}$$

As similar matrices have the same set of eigenvalues, we obtain (2.6). In particular, the squares of the singular values $\sigma_1^2, \ldots, \sigma_p^2$ are uniquely determined. As singular values are non-negative and non-increasing, this yields that $\sigma_1, \ldots, \sigma_p$ are uniquely determined.

Uniqueness of $\{u_i\}, \{v_i\}$ up to signs: We do not give a rigorous proof, but note that – geometrically – if the lengths of the semiaxes of a hyperellipse (i.e., the singular values $\{\sigma_i\}$) are distinct, then the semiaxes (i.e., the vectors $\{\sigma_i u_i\}$) are determined uniquely up to signs from the geometry of the hyperellipse. Note that if Σ and U is uniquely determined, then also V must be uniquely determined from $A = U\Sigma V^{\mathrm{T}}$ as U and Σ are invertible (singular values were assumed to be positive).

□

Now that we know that there exists a SVD with uniquely determined Σ to any arbitrary matrix, we can transform any given matrix into a diagonal matrix via a change of bases.

Remark 2.5. Any matrix $A \in \mathbb{R}^{m\times n}$ with a SVD $A = U\Sigma V^{\mathrm{T}}$ reduces to the diagonal matrix Σ when the range is expressed in the basis of left singular vectors (columns of U) and the domain in the basis of right singular vectors (columns of V). More precisely, for any $x \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$, there holds

$$Ax = b \quad\Longleftrightarrow\quad U\Sigma V^{\mathrm{T}}x = b \quad\Longleftrightarrow\quad \Sigma V^{\mathrm{T}}x = U^{\mathrm{T}}b \quad\Longleftrightarrow\quad \Sigma x' = b',$$

where $x' = V^{\mathrm{T}}x$ is the coordinate vector for the expansion of x in the basis of right singular vectors and $b' = U^{\mathrm{T}}b$ the coordinate vector for the expansion of b in the basis of left singular vectors.

Let us recall that for diagonalizable (also called non-defective) square matrices, we can also use its eigenvalue decomposition to transform into a diagonal matrix.

Remark 2.6. If $A \in \mathbb{R}^{n\times n}$ is diagonalizable with eigenvalue decomposition $A = XDX^{-1}$ for some invertible $X \in \mathbb{R}^{n\times n}$ and a diagonal matrix $D \in \mathbb{C}^{n\times n}$ containing the eigenvalues of A on its diagonal, then for any $x, b \in \mathbb{C}^n$ we have

$$Ax = b \quad\Longleftrightarrow\quad XDX^{-1}x = b \quad\Longleftrightarrow\quad DX^{-1}x = X^{-1}b \quad\Longleftrightarrow\quad Dx' = b',$$

where $x' = X^{-1}x, b' = X^{-1}b$ are the coordinate vectors for the expansions of x, b in the basis of columns of X (eigenvectors).

Note that the SVD uses two orthonormal bases (left and right singular vectors), whereas the eigenvalue decomposition uses only one – not necessarily orthogonal – basis (eigenvectors). The huge advantage of the SVD is that any matrix has a SVD. In contrast, an eigenvalue decomposition only exists for certain square matrices, i.e., for diagonalizable matrices (geometric multiplicity equals algebraic multiplicity for all eigenvalues).

2.3 Computation

We have seen that any matrix $A \in \mathbb{R}^{m\times n}$ has a SVD, and that the singular values are uniquely determined from (2.6).

Remark 2.7. Observe that we have

$$\begin{aligned} & A = U[\operatorname{diag}_{m\times n}(\sigma_1, \dots, \sigma_p)]V^{\mathrm{T}} \text{ is a SVD for } A \\ \Longleftrightarrow\quad & A^{\mathrm{T}} = V[\operatorname{diag}_{n\times m}(\sigma_1, \dots, \sigma_p)]U^{\mathrm{T}} \text{ is a SVD for } A^{\mathrm{T}} \end{aligned}$$

for any $A \in \mathbb{R}^{m\times n}$.

In view of this remark, we can restrict our attention to matrices $A \in \mathbb{R}^{m\times n}$ with $m \geq n$.

Algorithm 2.1 (Computation of SVD)**.** Let $A = (a_1|\cdots|a_n) \in \mathbb{R}^{m\times n}$ with $m \geq n$. Then, a SVD for A can be computed as follows.

1) Compute the so-called Gram matrix of $a_1, \dots, a_n \in \mathbb{R}^m$ for the Euclidean inner product on $\mathbb{R}^m$, that is,

$$A^{\mathrm{T}}A = (a_1|\cdots|a_n)^{\mathrm{T}}(a_1|\cdots|a_n) = \begin{pmatrix} \langle a_1, a_1\rangle & \langle a_1, a_2\rangle & \cdots & \langle a_1, a_n\rangle \\ \langle a_2, a_1\rangle & \langle a_2, a_2\rangle & \cdots & \langle a_2, a_n\rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle a_n, a_1\rangle & \langle a_n, a_2\rangle & \cdots & \langle a_n, a_n\rangle \end{pmatrix} \in \mathbb{R}^{n\times n}.$$

This matrix is symmetric (note $\langle x, y\rangle = \langle y, x\rangle \ \forall x, y \in \mathbb{R}^m$), thus orthogonally diagonalizable, and its eigenvalues are non-negative numbers (see (2.6)).

2) Compute an eigenvalue decomposition

$$A^{\mathrm{T}}A = VDV^{\mathrm{T}},$$

where $V = (v_1|\cdots|v_n) \in \mathbb{R}^{n\times n}$ is orthogonal, and $D = \operatorname{diag}_{n\times n}(\lambda_1, \ldots, \lambda_n) \in \mathbb{R}^{n\times n}$ is diagonal with $\lambda_1, \ldots, \lambda_n \in \Lambda(A^{\mathrm{T}}A)$ satisfying $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$.

3) Set $\sigma_i := \sqrt{\lambda_i}$ for $i \in \{1, \ldots, n\}$, and set

$$\Sigma := \operatorname{diag}_{m\times n}(\sigma_1, \ldots, \sigma_n) \in \mathbb{R}^{m\times n}$$

Note that $\Sigma^{\mathrm{T}}\Sigma = D$.

4) Find an orthogonal matrix $U = (u_1|\cdots|u_m) \in \mathbb{R}^{m\times m}$ such that

$$U\Sigma = AV, \quad \text{i.e.,} \quad \sigma_i u_i = Av_i \quad \forall i \in \{1, \ldots, n\}.$$

Then, we have that $A = U\Sigma V^{\mathrm{T}}$ is a SVD for A.

Example 2.2. We compute a SVD of

$$M := \begin{pmatrix} 1 & 0 & -1 & 1 \\ -1 & 1 & 0 & 1 \end{pmatrix} \in \mathbb{R}^{2\times 4}.$$

To this end, we set $A := M^{\mathrm{T}} \in \mathbb{R}^{4\times 2}$ and apply Algorithm 2.1 to A.

1) We compute $A^{\mathrm{T}}A = \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix} \in \mathbb{R}^{2\times 2}$.

2) We set $V := (v_1|v_2) := \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \in \mathbb{R}^{2\times 2}$ and $D := \operatorname{diag}_{2\times 2}(\lambda_1, \lambda_2) := \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix} \in \mathbb{R}^{2\times 2}$. Then, V is orthogonal, D is diagonal with $\lambda_1 \geq \lambda_2$, and $A^{\mathrm{T}}A = VDV^{\mathrm{T}}$.

3) Set $\Sigma := \operatorname{diag}_{4\times 2}(\sigma_1, \sigma_2) := \operatorname{diag}_{4\times 2}(\sqrt{\lambda_1}, \sqrt{\lambda_2}) = \begin{pmatrix} \sqrt{3} & 0 \\ 0 & \sqrt{3} \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \in \mathbb{R}^{4\times 2}$.

4) Find $U = (u_1|u_2|u_3|u_4) \in \mathbb{R}^{4\times 4}$ orthogonal with $\sigma_i u_i = Av_i$ for $i \in \{1, 2\}$. Then, $u_1 = \frac{1}{\sqrt{3}}(1, 0, -1, 1)^{\mathrm{T}}$ and $u_2 = \frac{1}{\sqrt{3}}(-1, 1, 0, 1)^{\mathrm{T}}$, which we can extend to an orthonormal basis u_1, u_2, u_3, u_4 of $\mathbb{R}^4$. We can take

$$U := \begin{pmatrix} \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} \\ 0 & \frac{1}{\sqrt{3}} & -\frac{2}{\sqrt{6}} & 0 \\ -\frac{1}{\sqrt{3}} & 0 & 0 & \frac{2}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} \end{pmatrix} \in \mathbb{R}^{4\times 4}.$$

(The vectors u_3, u_4 can be found using the fact that $\{u_3, u_4\}$ needs to be an orthonormal basis of $\mathcal{N}((u_1|u_2)^{\mathrm{T}})$.)

We obtain that $A = U\Sigma V^{\mathrm{T}}$ is a SVD of A, and hence, transposing this equation,

$$\begin{pmatrix} 1 & 0 & -1 & 1 \\ -1 & 1 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \sqrt{3} & 0 & 0 & 0 \\ 0 & \sqrt{3} & 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} \\ 0 & \frac{1}{\sqrt{3}} & -\frac{2}{\sqrt{6}} & 0 \\ -\frac{1}{\sqrt{3}} & 0 & 0 & \frac{2}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} \end{pmatrix}^{\mathrm{T}}$$

is a SVD for $M = A^{\mathrm{T}}$. Note that this is the SVD given in Example 2.1.

Further examples/exercises can be found on the problem sheets, where we will also discuss a useful alternative to compute SVDs for square matrices $A \in \mathbb{R}^{n\times n}$ which is based on an eigenvalue decomposition of the symmetric matrix $\left(\begin{array}{c|c} 0_{n\times n} & A^{\mathrm{T}} \\ \hline A & 0_{n\times n} \end{array}\right) \in \mathbb{R}^{2n\times 2n}$.

2.4 Matrix properties

We now state and prove some crucial results on the connection of the SVD to matrix properties.

Theorem 2.3. *Let $A \in \mathbb{R}^{m\times n}$, set $p := \min(m,n)$, and let*

$$A = U\Sigma V^{\mathrm{T}} = (u_1|\cdots|u_m)[\mathrm{diag}_{m\times n}(\sigma_1,\ldots,\sigma_p)](v_1|\cdots|v_n)^{\mathrm{T}}$$

be a SVD for A. Further, let $0 \le r \le p$ denote the number of non-zero singular values of A, so that $\sigma_1,\ldots,\sigma_r > 0$ and $\sigma_{r+1},\ldots,\sigma_p = 0$. Then, we have the following assertions.

(i) $\mathrm{rk}(A) = r$.

(ii) $\mathcal{R}(A) = \mathrm{span}(u_1,\ldots,u_r)$ *and* $\mathcal{N}(A) = \mathrm{span}(v_{r+1},\ldots,v_n)$.

(iii) $\|A\|_2 = \sigma_1$ *and* $\|A\|_F = \sqrt{\sum_{i=1}^r \sigma_i^2}$.

(iv) $\{\sigma_1,\ldots,\sigma_r\} = \{\sqrt{\lambda}\,|\,\lambda \in \Lambda(A^{\mathrm{T}}A)\}\backslash\{0\} = \{\sqrt{\lambda}\,|\,\lambda \in \Lambda(AA^{\mathrm{T}})\}\backslash\{0\}$.

(v) If $m = n$, then $|\det(A)| = \prod_{i=1}^n \sigma_i$.

(vi) If $m = n$ and $A = A^{\mathrm{T}}$, then $\{\sigma_1,\ldots,\sigma_n\} = \{|\lambda|\,|\,\lambda \in \Lambda(A)\}$.

Notation: We define $\{x_i,\ldots,x_j\} := \emptyset$ for $i,j \in \mathbb{N}_0$ with $i > j$. We define $\mathrm{span}(\emptyset) := \{0\}$.

Proof. (i) Observe that for any invertible matrices $M_m \in \mathbb{R}^{m\times m}$ and $M_n \in \mathbb{R}^{n\times n}$ there holds $\mathrm{rk}(M_m A) = \mathrm{rk}(A) = \mathrm{rk}(AM_n)$ (exercise). Further, observe that $\mathrm{rk}(\Sigma) = r$. Hence,

$$\mathrm{rk}(A) = \mathrm{rk}(U\Sigma V^{\mathrm{T}}) = \mathrm{rk}(\Sigma) = r.$$

(ii) Note that there holds

$$\mathcal{R}(U\Sigma V^{\mathrm{T}}) = \{U\Sigma V^{\mathrm{T}}x\,|\,x \in \mathbb{R}^n\} = \{U\Sigma y\,|\,y \in \mathbb{R}^n\} = \{Uz\,|\,z \in \mathcal{R}(\Sigma)\},$$

where we have used in the second equality that $V^{\mathrm{T}} \in \mathbb{R}^{n\times n}$ is invertible, and

$$\mathcal{N}(U\Sigma V^{\mathrm{T}}) = \{x \in \mathbb{R}^n\,|\,U\Sigma V^{\mathrm{T}}x = 0\} = \{x \in \mathbb{R}^n\,|\,\Sigma V^{\mathrm{T}}x = 0\} = \{x \in \mathbb{R}^n\,|\,V^{\mathrm{T}}x \in \mathcal{N}(\Sigma)\},$$

where we have used in the second equality that $U \in \mathbb{R}^{m\times m}$ is invertible. Observing that

$$\mathcal{R}(\Sigma) = \operatorname{span}(e_1, \dots, e_r) \subseteq \mathbb{R}^m, \qquad \mathcal{N}(\Sigma) = \operatorname{span}(e_{r+1}, \dots, e_n) \subseteq \mathbb{R}^n,$$

it follows that $\mathcal{R}(A) = \operatorname{span}(u_1, \dots, u_r)$ and $\mathcal{N}(A) = \operatorname{span}(v_{r+1}, \dots, v_n)$.
(iii) We have already shown that $\|A\|_2 = \sigma_1$ in Remark 2.3. It remains to prove the claim that $\|A\|_F^2 = \sum_{i=1}^r \sigma_i^2$. To this end, recall from Theorem 1.13 that the Frobenius norm is invariant under multiplication by orthogonal matrices. Therefore, we have

$$\|A\|_F^2 = \|U\Sigma V^{\mathrm{T}}\|_F^2 = \|\Sigma\|_F^2 = \sum_{i=1}^r \sigma_i^2.$$

(iv) Recall from the proof of Theorem 2.2 that $A^{\mathrm{T}}A$ is similar to $\Sigma^{\mathrm{T}}\Sigma$, and that AA^{T} is similar to $\Sigma\Sigma^{\mathrm{T}}$. The claim now follows in view of (2.7).
(v) Assume $m = n$. Then, by the multiplicative property of the determinant we have

$$\det(A) = \det(U\Sigma V^{\mathrm{T}}) = \det(U)\det(\Sigma)\det(V^{\mathrm{T}}) = \det(U)\det(\Sigma)\det(V),$$

where we have used that $\det(M^{\mathrm{T}}) = \det(M)$ for any $M \in \mathbb{R}^{n\times n}$ in the last step. Recalling that $|\det(Q)| = 1$ for any orthogonal matrix $Q \in \mathbb{R}^{n\times n}$, we deduce that

$$|\det(A)| = |\det(\Sigma)| = \prod_{i=1}^n \sigma_i.$$

(vi) Assume $m = n$ and that A is symmetric. Then, all of its eigenvalues are real, and A is orthogonally diagonalizable, i.e., there exists an orthogonal matrix $Q \in \mathbb{R}^{n\times n}$ and a diagonal matrix $D = \operatorname{diag}_{n\times n}(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^{n\times n}$ with $\{\lambda_1, \dots, \lambda_n\} = \Lambda(A)$ such that $A = QDQ^{\mathrm{T}}$. We assume that the diagonal entries of D are ordered such that $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$. We then define the matrices

$$\tilde{\Sigma} := \operatorname{diag}_{n\times n}(|\lambda_1|, \dots, |\lambda_n|) \in \mathbb{R}^{n\times n}, \qquad S := \operatorname{diag}_{n\times n}(\operatorname{sign}(\lambda_1), \dots, \operatorname{sign}(\lambda_n)) \in \mathbb{R}^{n\times n},$$

and note that $D = \tilde{\Sigma}S = \tilde{\Sigma}S^{\mathrm{T}}$. Setting $\tilde{U} := Q$ and $\tilde{V} := QS$, this yields

$$A = Q\tilde{\Sigma}S^{\mathrm{T}}Q^{\mathrm{T}} = \tilde{U}\tilde{\Sigma}\tilde{V}^{\mathrm{T}}.$$

Observe that the matrices $\tilde{U}, \tilde{V}$ are orthogonal (note $SS^{\mathrm{T}} = S^{\mathrm{T}}S = I_n$) and thus, this is a SVD of A. Recalling that singular values of a matrix are uniquely determined, we conclude that $|\lambda_1|, \dots, |\lambda_n|$ are the singular values of A. □

Remark 2.8. The proof of Theorem 2.3(vi) yields a method to obtain a SVD of a symmetric matrix from its eigenvalue decomposition. A short alternative proof of (vi) goes as follows: If $A \in \mathbb{R}^{n\times n}$ is a symmetric matrix with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$, we have by Theorem 2.2 that $\{\sigma_1^2, \dots, \sigma_n^2\} = \Lambda(A^{\mathrm{T}}A) = \Lambda(A^2) = \{\lambda^2 \,|\, \lambda \in \Lambda(A)\}$, proving (vi).

Note the last equality can be shown as follows for A symmetric: A is orthogonally diagonalizable, i.e., $A = QDQ^{\mathrm{T}}$ for some $Q \in \mathbb{R}^{n\times n}$ orthogonal and $D = \operatorname{diag}_{n\times n}(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^{n\times n}$ with $\{\lambda_1, \dots, \lambda_n\} = \Lambda(A)$. Then, $A^2 = QD^2Q^{\mathrm{T}}$ and thus, $\Lambda(A^2) = \Lambda(D^2) = \{\lambda_1^2, \dots, \lambda_n^2\}$.

Theorem 2.3 lays the foundation for many practical algorithms. In particular, from a computational point of view, the standard way to compute the rank of a matrix is to count the number of singular values greater than some very small tolerance, the most accurate method for computing orthonormal bases of the range and the nullspace of a matrix is via (ii), and the standard way to compute the spectral norm of a matrix A is via $\|A\|_2 = \sigma_1$.

2.5 Low-rank approximation

The problem of low-rank approximation deals with the following optimization problem: Given some non-zero matrix $A \in \mathbb{R}^{m\times n}\backslash\{0\}$ and some $\nu \in \mathbb{N}_0$ with $0 \leq \nu < \mathrm{rk}(A)$, we want to find the best approximation to A in the class $\{B \in \mathbb{R}^{m\times n} \,|\, \mathrm{rk}(B) \leq \nu\}$, that is,

$$\begin{cases} \text{minimize} & \|A - B\|, \\ \text{subject to} & B \in \mathbb{R}^{m\times n},\ \mathrm{rk}(B) \leq \nu, \end{cases} \tag{2.8}$$

for some given matrix norm $\|\cdot\| : \mathbb{R}^{m\times n} \to [0, \infty)$. We are going to solve this optimization problem for the spectral and Frobenius norms by using the SVD.

As a first step, let us observe that any matrix A can be written as the sum of r rank-one matrices, where $r := \mathrm{rk}(A)$, using the SVD.

Remark 2.9. Let $A \in \mathbb{R}^{m\times n}$, set $p := \min(m, n)$, and let

$$A = U\Sigma V^{\mathrm{T}} = (u_1|\cdots|u_m)[\mathrm{diag}_{m\times n}(\sigma_1, \ldots, \sigma_p)](v_1|\cdots|v_n)^{\mathrm{T}}$$

be a SVD for A. Setting $r := \mathrm{rk}(A)$, we have that A can be written as

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^{\mathrm{T}}. \tag{2.9}$$

Indeed, this follows from the fact that we can write Σ as the sum of the r matrices $\mathrm{diag}_{m\times n}(\sigma_1, 0, \ldots, 0)$, $\mathrm{diag}_{m\times n}(0, \sigma_2, 0, \ldots, 0)$, $\ldots$, $\mathrm{diag}_{m\times n}(0, \ldots, 0, \sigma_r)$.

You can find other, more simple, ways to express $A \in \mathbb{R}^{m\times n}$ as a sum of rank-one matrices (exercise). However, the decomposition (2.9) has the property that its ν-th partial sum captures the largest possible amount of "energy" of A, that is, it is a minimizer of the optimization problem (2.8) for the spectral and Frobenius norms.

Theorem 2.4 (Eckart–Young–Mirsky theorem). *Let $A \in \mathbb{R}^{m\times n}\backslash\{0\}$, set $p := \min(m, n)$, and let*

$$A = U\Sigma V^{\mathrm{T}} = (u_1|\cdots|u_m)[\mathrm{diag}_{m\times n}(\sigma_1, \ldots, \sigma_p)](v_1|\cdots|v_n)^{\mathrm{T}}$$

be a SVD for A. Further, let $\nu \in \mathbb{N}_0$ with $0 \leq \nu < \mathrm{rk}(A)$, and set

$$A_\nu = \sum_{i=1}^{\nu} \sigma_i u_i v_i^{\mathrm{T}}.$$

Then, A_ν is the best approximation to A in the class $\{B \in \mathbb{R}^{m\times n} \,|\, \mathrm{rk}(B) \leq \nu\}$ with respect to the spectral norm, i.e.,

$$\inf_{\substack{B\in\mathbb{R}^{m\times n} \\ \mathrm{rk}(B)\leq\nu}} \|A - B\|_2 = \|A - A_\nu\|_2 = \sigma_{\nu+1}, \tag{2.10}$$

and with respect to the Frobenius norm, i.e.,

$$\inf_{\substack{B\in\mathbb{R}^{m\times n} \\ \mathrm{rk}(B)\leq\nu}} \|A - B\|_F = \|A - A_\nu\|_F = \sqrt{\sum_{i=\nu+1}^{r} \sigma_i^2}.$$

We only prove the result for the spectral norm, that is, (2.10), and omit the proof of the result for the Frobenius norm.

Proof of (2.10). Let us write $r := \mathrm{rk}(A)$ and note that $1 \leq r \leq p$ as $A \in \mathbb{R}^{m\times n}\backslash\{0\}$. Further, note $0 \leq \nu \leq r-1$.

Step 1: We start by showing that $\|A - A_\nu\|_2 = \sigma_{\nu+1}$. To this end, we use Remark 2.9 to obtain

$$\|A - A_\nu\|_2 = \left\|\sum_{i=1}^{r} \sigma_i u_i v_i^{\mathrm{T}} - \sum_{i=1}^{\nu} \sigma_i u_i v_i^{\mathrm{T}}\right\|_2 = \left\|\sum_{i=\nu+1}^{r} \sigma_i u_i v_i^{\mathrm{T}}\right\|_2 = \sigma_{\nu+1},$$

where we have used in the last step that the the largest singular value of the matrix $\sum_{i=\nu+1}^{r} \sigma_i u_i v_i^{\mathrm{T}}$ is given by $\sigma_{\nu+1}$. In particular, as $\mathrm{rk}(A_\nu) \leq \nu$, we find that

$$\inf_{\substack{B\in\mathbb{R}^{m\times n}\\ \mathrm{rk}(B)\leq\nu}} \|A - B\|_2 \leq \|A - A_\nu\|_2 = \sigma_{\nu+1}.$$

Step 2: It remains to prove that

$$\inf_{\substack{B\in\mathbb{R}^{m\times n}\\ \mathrm{rk}(B)\leq\nu}} \|A - B\|_2 \geq \sigma_{\nu+1}.$$

Suppose that there exists a matrix $B \in \mathbb{R}^{m\times n}$ with $\mathrm{rk}(B) \leq \nu$ and $\|A - B\|_2 < \sigma_{\nu+1}$. Then, by the rank-nullity theorem (see Theorem 1.2), there holds

$$\dim(\mathcal{N}(B)) = \mathrm{nullity}(B) = n - \mathrm{rk}(B) \geq n - \nu$$

and we have that

$$\|Ax\|_2 = \|(A - B)x\|_2 \leq \|A - B\|_2\|x\|_2 < \sigma_{\nu+1}\|x\|_2 \qquad \forall x \in \mathcal{N}(B)\backslash\{0\}. \tag{2.11}$$

We also have for any $v = \sum_{i=1}^{\nu+1} \alpha_i v_i \in \mathrm{span}(v_1, \ldots, v_{\nu+1})$ that

$$\|Av\|_2^2 = \left\|\sum_{i=1}^{\nu+1} \alpha_i\sigma_i u_i\right\|_2^2 = \sum_{i=1}^{\nu+1} \alpha_i^2\sigma_i^2 \geq \sigma_{\nu+1}^2 \sum_{i=1}^{\nu+1} \alpha_i^2 = \sigma_{\nu+1}^2 \left\|\sum_{i=1}^{\nu+1} \alpha_i v_i\right\|_2^2 = \sigma_{\nu+1}^2\|v\|_2^2, \tag{2.12}$$

where we have used that $Av_i = \sigma_i u_i$ for all $i \in \{1, \ldots, \nu+1\}$ (note $\nu + 1 \leq r \leq p$), and the Pythagorean theorem for orthogonal vectors, that is, for any two orthogonal vectors $a, b \in \mathbb{R}^n$ there holds $\|a + b\|_2^2 = \|a\|_2^2 + \|b\|_2^2$. Note that for the subspaces $N := \mathcal{N}(B)$ and $S := \mathrm{span}(v_1, \ldots, v_{\nu+1})$ of the vector space $\mathbb{R}^n$, we have

$$\dim(N \cap S) \geq \dim(N) + \dim(S) - \dim(N + S) \geq (n - \nu) + (\nu + 1) - n = 1,$$

and hence, there exists a nonzero vector which is contained in both N and S. In view of (2.11) and (2.12), we obtain a contradiction and the result is proved. □

A practical application of low-rank approximation is image compression.

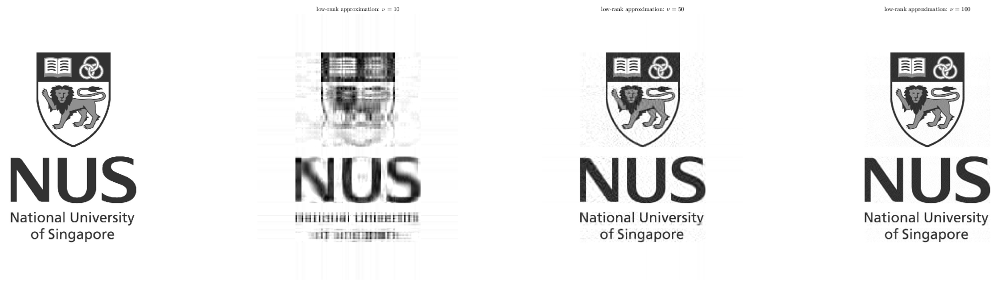
low-rank approximation: $\nu = 10$
low-rank approximation: $\nu = 50$
low-rank approximation: $\nu = 100$
NUS
National University
of Singapore

Figure 2: Low-rank approximation applied to an image. From left to right: original image (991×751 matrix of rank 481), low-rank approximation with $\nu = 10, 50, 100$.

Remark 2.10. Let $A \in \mathbb{R}^{m\times n}\backslash\{0\}$ and write $r := \mathrm{rk}(A) \in \{1, \dots, p\}$, where $p := \min(m, n)$. Then, there exist matrices $M \in \mathbb{R}^{m\times r}$ and $N \in \mathbb{R}^{n\times r}$ such that

$$A = MN^{\mathrm{T}}.$$

Proof. Let $A = U\Sigma V^{\mathrm{T}} = (u_1|\cdots|u_m)[\mathrm{diag}_{m\times n}(\sigma_1, \dots, \sigma_p)](v_1|\cdots|v_n)^{\mathrm{T}}$ be a SVD of A. Noting that $\sigma_1, \dots, \sigma_r > 0$ and $\sigma_j = 0$ $\forall r < j \leq p$, we see that we can write $A = MN^{\mathrm{T}}$ with $M := (\sigma_1 u_1|\cdots|\sigma_r u_r) \in \mathbb{R}^{m\times r}$ and $N := (v_1|\cdots|v_r) \in \mathbb{R}^{n\times r}$. □

In view of Remark 2.10, if the rank r of a matrix $A \in \mathbb{R}^{m\times n}$ is small compared to m and n, we only need $r(m + n)$ numbers to describe A instead of mn numbers (e.g., when $m = n$, storing $r \cdot 2n$ numbers is saving storage compared to storing n^2 numbers if $r < \frac{n}{2}$). In the particular example of Figure 2, we have the following:

- We need $991 \cdot 751 = 744241$ numbers to describe the original matrix/image. (Rk: better to store mn numbers instead of $r(m+n)$ numbers as $481(991+751) > 991{\cdot}751$.)
- For the rank-100 approximation, we only need $100(991 + 751) = 174200$ numbers to describe the matrix/image.
- For the rank-50 approximation, we only need $50(991 + 751) = 87100$ numbers to describe the matrix/image.
- For the rank-10 approximation, we only need $10(991 + 751) = 17420$ numbers to describe the matrix/image.

We see that it is much cheaper to store these three low-rank approximations compared to the original image (which is great as it is quite hard to notice a difference between the rank-100 approximation and the original image by the human eye).

3 QR Factorization

3.1 Definition of full and reduced QR factorization

For simplicity, we restrict ourselves to “tall” matrices $A \in \mathbb{R}^{m\times n}$ with $m \geq n$.

Definition 3.1. Let $m, n \in \mathbb{N}$ with $m \geq n$. A matrix $R = (r_{ij}) \in \mathbb{R}^{m\times n}$ is called upper-triangular iff $r_{ij} = 0$ whenever $i > j$, i.e., iff it is of the form

$$R = \begin{pmatrix} \hat{R} \\ \hline 0_{(m-n)\times n} \end{pmatrix} \in \mathbb{R}^{m\times n}, \quad \text{where} \quad \hat{R} = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ 0 & r_{22} & \cdots & r_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & r_{nn} \end{pmatrix} \in \mathbb{R}^{n\times n}.$$

Definition 3.2. Let $m, n \in \mathbb{N}$ with $m \geq n$, and let $A \in \mathbb{R}^{m\times n}$. If there exist

$$\begin{aligned} Q &= (q_1|\cdots|q_m) \quad \in \mathbb{R}^{m\times m} \text{ orthogonal}, \\ R &= \begin{pmatrix} \hat{R} \\ \hline 0_{(m-n)\times n} \end{pmatrix} \in \mathbb{R}^{m\times n} \ \text{ upper-triangular}, \end{aligned}$$

such that there holds

$$A = QR, \tag{3.1}$$

then we call (3.1) a QR factorization of A.

Remark 3.1. The QR factorization (3.1) can be simplified to

$$A = QR = (q_1|\cdots|q_m) \begin{pmatrix} \hat{R} \\ \hline 0_{(m-n)\times n} \end{pmatrix} = (q_1|\cdots|q_n)\hat{R}.$$

We call such a factorization $A = \hat{Q}\hat{R}$ with $\hat{Q} \in \mathbb{R}^{m\times n}$ having orthonormal columns and $\hat{R} \in \mathbb{R}^{n\times n}$ being upper-triangular a *reduced QR factorization* of A.

Example 3.1. An example of a QR factorization is

$$\begin{pmatrix} 1 & 1 \\ -1 & 1 \\ 1 & 1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{2}} & 0 \\ -\frac{1}{\sqrt{3}} & \frac{2}{\sqrt{6}} & 0 & 0 \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \sqrt{3} & \frac{1}{\sqrt{3}} \\ 0 & \frac{4}{\sqrt{6}} \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

with corresponding reduced QR factorization

$$\begin{pmatrix} 1 & 1 \\ -1 & 1 \\ 1 & 1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{6}} \\ -\frac{1}{\sqrt{3}} & \frac{2}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{6}} \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \sqrt{3} & \frac{1}{\sqrt{3}} \\ 0 & \frac{4}{\sqrt{6}} \end{pmatrix}.$$

3.2 Existence and uniqueness

Remark 3.2. Let $A \in \mathbb{R}^{m\times n}$ with $m \geq n$. Then, finding a reduced QR factorization $A = \hat{Q}\hat{R}$ with $\hat{Q} \in \mathbb{R}^{m\times n}$ having orthonormal columns and $\hat{R} \in \mathbb{R}^{n\times n}$ upper-triangular, i.e.,

$$A = (a_1|\cdots|a_n) = (q_1|\cdots|q_n)\begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ 0 & r_{22} & \cdots & r_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & r_{nn} \end{pmatrix} = \hat{Q}\hat{R},$$

is equivalent to finding n orthonormal vectors $q_1, \ldots, q_n \in \mathbb{R}^m$ and $\frac{n(n+1)}{2}$ real numbers $\{r_{ij}\}_{1\leq i\leq j\leq n} \subseteq \mathbb{R}$ such that

$$\begin{cases} a_1 &= r_{11}q_1, \\ a_2 &= r_{12}q_1 + r_{22}q_2, \\ \vdots & \\ a_n &= r_{1n}q_1 + r_{2n}q_2 + \cdots + r_{nn}q_n. \end{cases} \tag{3.2}$$

We now describe a procedure, called *Gram–Schmidt orthogonalization*, for obtaining a reduced QR factorization to a matrix $A = (a_1|\cdots|a_n) \in \mathbb{R}^{m\times n}$, $m \geq n$, of full rank $\mathrm{rk}(A) = n$. We want to find orthonormal vectors $q_1, \ldots, q_n \in \mathbb{R}^m$ such that

$$\mathrm{span}(q_1, \ldots, q_i) = \mathrm{span}(a_1, \ldots, a_i) \qquad \forall i \in \{1, \ldots, n\}.$$

As a first step, let us set $q_1 := \frac{a_1}{\|a_1\|_2}$ and $r_{11} := \|a_1\|_2$ so that q_1 is a unit vector and we have $a_1 = r_{11}q_1$ (thus also $q_1 = r_{11}^{-1}a_1$ as $r_{11} > 0$). We now make the following observation.

Remark 3.3. Let $A = (a_1|\cdots|a_n) \in \mathbb{R}^{m\times n}$, $m \geq n \geq 2$, and assume $\mathrm{rk}(A) = n$. Suppose, for some $k \in \{2, \ldots, n\}$, we are given orthonormal vectors $q_1, \ldots, q_{k-1} \in \mathbb{R}^m$ such that $q_i \in \mathrm{span}(a_1, \ldots, a_i)$ for all $i \in \{1, \ldots, k-1\}$. Then, the vector

$$q_k := \pm\frac{\tilde{q}_k}{\|\tilde{q}_k\|_2}, \quad \text{where} \quad \tilde{q}_k := a_k - \sum_{l=1}^{k-1}\langle q_l, a_k\rangle q_l$$

(note $\tilde{q}_k \neq 0$ as A has full rank) is a unit vector satisfying $q_k \in \mathrm{span}(a_1, \ldots, a_k)$ and $\{q_k\} \perp \{q_1, \ldots, q_{k-1}\}$ (see (1.2)). Observe that this allows us to write

$$a_k = \sum_{l=1}^{k} r_{lk}q_l, \qquad r_{lk} := \begin{cases} \langle q_l, a_k\rangle & , \text{ if } 1 \leq l \leq k-1, \\ \pm\|\tilde{q}_k\|_2 & , \text{ if } l = k. \end{cases}$$

As desired, this procedure yields orthonormal vectors $q_1, \ldots, q_n \in \mathbb{R}^m$ and real numbers $\{r_{ij}\}_{1\leq i\leq j\leq n} \subseteq \mathbb{R}$ satisfying (3.2). These are given by

$$\begin{aligned} \forall\, 1 \leq j \leq n: \quad & q_j = \frac{1}{r_{jj}}\left(a_j - \sum_{l=1}^{j-1} r_{lj}q_l\right), \\ \forall\, 1 \leq i \leq j \leq n: \quad & r_{ij} = \begin{cases} \langle q_i, a_j\rangle & , \text{ if } i \leq j-1, \\ \pm\|a_j - \sum_{l=1}^{j-1} r_{lj}q_l\|_2 & , \text{ if } i = j. \end{cases} \end{aligned} \tag{3.3}$$

The sign of the values r_{jj}, $1 \leq j \leq n$, is not determined and we use the convention to choose $r_{jj} > 0$ so that the upper-triangular matrix $\hat{R}$ in the resulting reduced QR factorization $A = \hat{Q}\hat{R}$ has positive diagonal entries.

Algorithm 3.1 (Gram–Schmidt)**.** Let $m, n \in \mathbb{N}$, $m \geq n$, and $A = (a_1|\cdots|a_n) \in \mathbb{R}^{m\times n}$ with $\mathrm{rk}(A) = n$. Then, A has the reduced QR factorization $A = \hat{Q}\hat{R}$ with

$$\hat{Q} := (q_1|\cdots|q_n) \in \mathbb{R}^{m\times n}, \qquad \hat{R} := \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ 0 & r_{22} & \cdots & r_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & r_{nn} \end{pmatrix} \in \mathbb{R}^{n\times n}$$

determined as follows.

Step 1) Compute

$$\tilde{q}_1 := a_1 \in \mathbb{R}^m, \qquad r_{11} := \|\tilde{q}_1\|_2 > 0, \qquad q_1 := \frac{1}{r_{11}}\tilde{q}_1 \in \mathbb{R}^m.$$

If $n = 1$, we stop. If $n \geq 2$, we continue as follows.

Step 2) Compute $r_{12} := \langle q_1, a_2 \rangle \in \mathbb{R}$. Then, compute

$$\tilde{q}_2 := a_2 - r_{12}q_1 \in \mathbb{R}^m, \qquad r_{22} := \|\tilde{q}_2\|_2 > 0, \qquad q_2 := \frac{1}{r_{22}}\tilde{q}_2 \in \mathbb{R}^m.$$

$\vdots$

Step j) Compute $r_{ij} := \langle q_i, a_j \rangle \in \mathbb{R}$ for $i \in \{1, \dots, j-1\}$. Then, compute

$$\tilde{q}_j := a_j - \sum_{l=1}^{j-1} r_{lj}q_l \in \mathbb{R}^m, \qquad r_{jj} := \|\tilde{q}_j\|_2 > 0, \qquad q_j := \frac{1}{r_{jj}}\tilde{q}_j \in \mathbb{R}^m.$$

$\vdots$

Step n) Compute $r_{in} := \langle q_i, a_n \rangle \in \mathbb{R}$ for $i \in \{1, \dots, n-1\}$. Then, compute

$$\tilde{q}_n := a_n - \sum_{l=1}^{n-1} r_{ln}q_l \in \mathbb{R}^m, \qquad r_{nn} := \|\tilde{q}_n\|_2 > 0, \qquad q_n := \frac{1}{r_{nn}}\tilde{q}_n \in \mathbb{R}^m.$$

Remark 3.4. Observe that this is well-defined for full-rank matrices, i.e., we have that $r_{ii} \neq 0$ for all $i \in \{1, \dots, n\}$. Indeed, if we would have $r_{jj} = 0$ for some $j \in \{1, \dots, n\}$, then $\tilde{q}_j = a_j - \sum_{l=1}^{j-1} r_{lj}q_l = 0$ and thus, $a_j \in \mathrm{span}(q_1, \dots, q_{j-1}) = \mathrm{span}(a_1, \dots, a_{j-1})$, a contradiction to the assumption that A is of full rank.

Example 3.2. Consider the matrix

$$A := (a_1|a_2|a_3) := \begin{pmatrix} 1 & 0 & 1 \\ -1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & 2 & 1 \end{pmatrix} \in \mathbb{R}^{4\times 3}. \tag{3.4}$$

Note that A is of full rank, i.e., $\mathrm{rk}(A) = 3$, and we can apply Algorithm 3.1 to A to obtain a reduced QR factorization.

1) Set $\tilde{q}_1 := a_1 = (1,-1,1,1)^{\mathrm{T}}$. Then, we have that $r_{11} := \|\tilde{q}_1\|_2 = 2$ and we set $q_1 := r_{11}^{-1}\tilde{q}_1 = (\frac{1}{2}, -\frac{1}{2}, \frac{1}{2}, \frac{1}{2})^{\mathrm{T}}$.

2) Compute $r_{12} := \langle q_1, a_2\rangle = 1$ and set $\tilde{q}_2 := a_2 - r_{12}q_1 = (-\frac{1}{2}, \frac{3}{2}, \frac{1}{2}, \frac{3}{2})^{\mathrm{T}}$. Then, $r_{22} := \|\tilde{q}_2\|_2 = \sqrt{5}$ and we set $q_2 := r_{22}^{-1}\tilde{q}_2 = (-\frac{1}{2\sqrt{5}}, \frac{3}{2\sqrt{5}}, \frac{1}{2\sqrt{5}}, \frac{3}{2\sqrt{5}})^{\mathrm{T}}$.

3) Compute $r_{13} := \langle q_1, a_3\rangle = 0$ and $r_{23} := \langle q_2, a_3\rangle = \frac{2}{\sqrt{5}}$. Then, we have that $\tilde{q}_3 := a_3 - r_{13}q_1 - r_{23}q_2 = (\frac{6}{5}, \frac{2}{5}, -\frac{6}{5}, \frac{2}{5})^{\mathrm{T}}$ with $r_{33} := \|\tilde{q}_3\|_2 = \frac{4}{\sqrt{5}}$, and we set $q_3 := r_{33}^{-1}\tilde{q}_3 = (\frac{3}{2\sqrt{5}}, \frac{1}{2\sqrt{5}}, -\frac{3}{2\sqrt{5}}, \frac{1}{2\sqrt{5}})^{\mathrm{T}}$.

We deduce that $A = \hat{Q}\hat{R}$ with

$$\hat{Q} := \begin{pmatrix} \frac{1}{2} & -\frac{1}{2\sqrt{5}} & \frac{3}{2\sqrt{5}} \\ -\frac{1}{2} & \frac{3}{2\sqrt{5}} & \frac{1}{2\sqrt{5}} \\ \frac{1}{2} & \frac{1}{2\sqrt{5}} & -\frac{3}{2\sqrt{5}} \\ \frac{1}{2} & \frac{3}{2\sqrt{5}} & \frac{1}{2\sqrt{5}} \end{pmatrix}, \qquad \hat{R} := \begin{pmatrix} 2 & 1 & 0 \\ 0 & \sqrt{5} & \frac{2}{\sqrt{5}} \\ 0 & 0 & \frac{4}{\sqrt{5}} \end{pmatrix}$$

is a reduced QR factorization of A. Note that a full QR factorization can be obtained by "filling up" $\hat{Q}$ with an additional orthonormal column and $\hat{R}$ with an additional row of zeros. We can take, e.g.,

$$Q := \begin{pmatrix} \frac{1}{2} & -\frac{1}{2\sqrt{5}} & \frac{3}{2\sqrt{5}} & \frac{1}{2} \\ -\frac{1}{2} & \frac{3}{2\sqrt{5}} & \frac{1}{2\sqrt{5}} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2\sqrt{5}} & -\frac{3}{2\sqrt{5}} & \frac{1}{2} \\ \frac{1}{2} & \frac{3}{2\sqrt{5}} & \frac{1}{2\sqrt{5}} & -\frac{1}{2} \end{pmatrix}, \qquad R := \begin{pmatrix} 2 & 1 & 0 \\ 0 & \sqrt{5} & \frac{2}{\sqrt{5}} \\ 0 & 0 & \frac{4}{\sqrt{5}} \\ 0 & 0 & 0 \end{pmatrix}$$

to find that $A = QR$ is a (full) QR factorization of A.

Remark 3.5. From a reduced QR factorization, we can always obtain a full QR factorization. More precisely, let $A \in \mathbb{R}^{m\times n}$, $m \geq n$, and suppose we have a reduced QR factorization $A = \hat{Q}\hat{R}$ with $\hat{Q} = (q_1|\cdots|q_n) \in \mathbb{R}^{m\times n}$ having orthonormal columns and $\hat{R} \in \mathbb{R}^{n\times n}$ upper-triangular. Note that if $m = n$, this is already a full QR factorization. If $m > n$, we can choose arbitrary orthonormal vectors $q_{n+1}, \ldots, q_m \in \mathbb{R}^m$ satisfying $\{q_{n+1}, \ldots, q_m\} \perp \{q_1, \ldots, q_n\}$, and obtain with $Q = (\hat{Q}|q_{n+1}|\cdots|q_m) \in \mathbb{R}^{m\times m}$ and $R = \begin{pmatrix} \hat{R} \\ \hline 0_{(m-n)\times n} \end{pmatrix} \in \mathbb{R}^{m\times n}$ that $A = QR$ is a (full) QR factorization of A.

We can now prove that any arbitrary matrix $A \in \mathbb{R}^{m\times n}$, $m \geq n$, has a full QR factorization.

Theorem 3.1 (Existence result for QR)**.** *Let $m, n \in \mathbb{N}$ with $m \geq n$. Then, every matrix $A \in \mathbb{R}^{m\times n}$ has a (full) QR factorization.*

Proof. We know that every full-rank matrix $A \in \mathbb{R}^{m\times n}$, $m \geq n$, has a reduced QR factorization (by the Gram–Schmidt Algorithm 3.1 and Remark 3.4) and hence, by Remark 3.5, also a full QR factorization. It remains to consider the case of rank-deficient matrices. To this end, let $A \in \mathbb{R}^{m\times n}$, $m \geq n$, with $0 \leq \mathrm{rk}(A) < n$. Then, running Algorithm 3.1, there will be at least one step j, where $\tilde{q}_j = 0$. Whenever this happens, we set $r_{jj} = 0$ and take $q_j \in \mathbb{R}^m$, $\|q_j\|_2 = 1$, to be any arbitrary unit vector satisfying $\{q_j\} \perp \{q_1, \ldots, q_{j-1}\}$, and continue Algorithm 3.1. This yields a reduced QR factorization for A, from which we can then obtain a full QR factorization from Remark 3.5. □

Remark 3.6. In particular, every matrix $A \in \mathbb{R}^{m\times n}$, $m \geq n$, has a reduced QR factorization.

Note that we now have a way to compute reduced and full QR factorizations to arbitrary real $m \times n$ matrices with $m \geq n$. Exercises can be found on the problem sheets.

Let us observe that the QR factorization is not unique. Indeed, suppose we are given a QR factorization $A = QR$ and choose $s_1, \dots, s_m \in \{-1, 1\}$. Then, for $i = 1, \dots, m$ we can multiply the i-th column of Q and the i-th row of R by s_i without changing the product QR, thus yielding a new QR factorization. However, we can prove that the reduced QR factorization $A = \hat{Q}\hat{R}$ of full-rank matrices $A \in \mathbb{R}^{m\times n}$, $m \geq n$, is unique upon fixing the sign of the diagonal entries of $\hat{R}$.

Theorem 3.2 (Uniqueness result for QR). *Let $m, n \in \mathbb{N}$ with $m \geq n$. Then, every matrix $A \in \mathbb{R}^{m\times n}$ with* $\operatorname{rk}(A) = n$ *has a unique reduced QR factorization $A = \hat{Q}\hat{R}$ with $\hat{R}$ having positive diagonal entries.*

Proof. Let $A \in \mathbb{R}^{m\times n}$, $m \geq n$, be a matrix of full rank, i.e., $\operatorname{rk}(A) = n$. Then, in view of Remarks 3.2 and 3.3, any reduced QR factorization $A = \hat{Q}\hat{R}$ with $\hat{Q} = (q_1|\cdots|q_n) \in \mathbb{R}^{m\times n}$ having orthonormal columns and $\hat{R} = (r_{ij}) \in \mathbb{R}^{n\times n}$ being upper-triangular must satisfy (3.3). We have already observed in Remark 3.4 that the values r_{ii}, $1 \leq i \leq n$, given by (3.3) are non-zero since A is of full rank, and hence, the vectors $q_1, \dots, q_n \in \mathbb{R}^m$ and numbers $\{r_{ij}\}_{1\leq i\leq j\leq n} \subseteq \mathbb{R}$ are uniquely determined except for the sign of the values r_{ii}, $1 \leq i \leq n$. Once we fix those signs to $r_{ii} > 0$, $1 \leq i \leq n$, by imposing that $\hat{R}$ should have positive diagonal entries, we have shown the claim. □

Remark 3.7 (Application of QR factorization: solution of linear systems). The QR factorization provides a method to solve linear systems. For given $A \in \mathbb{R}^{m\times n}$, $m \geq n$, and $b \in \mathbb{R}^m$, consider the problem of finding $x \in \mathbb{R}^n$ such that $Ax = b$. Observe that, if we have a QR factorization $A = QR$ with $Q \in \mathbb{R}^{m\times m}$ orthogonal and $R \in \mathbb{R}^{m\times n}$ upper triangular, we have

$$Ax = b \quad\iff\quad QRx = b \quad\iff\quad Rx = Q^{\mathrm{T}}b.$$

Therefore, once we have computed a QR factorization $A = QR$, we can compute $\tilde{b} := Q^{\mathrm{T}}b \in \mathbb{R}^m$ and solve the upper-triangular system $Rx = \tilde{b}$ by backward substitution.

3.3 Projectors

We now introduce the concept of projectors, which is crucial to many algorithms in numerical linear algebra.

Definition 3.3. A square matrix $P \in \mathbb{R}^{n\times n}$ is called a projector, or a projection matrix, iff it is idempotent, that is, $P^2 = P$.

Note that, in terms of the associated linear map $L_P : \mathbb{R}^n \to \mathbb{R}^n$, $x \mapsto Px$, to a matrix $P \in \mathbb{R}^{n\times n}$, the condition $P^2 = P$ means that $L_P \circ L_P = L_P$.

Remark 3.8. Let $P \in \mathbb{R}^{n\times n}$ be a projector. We make the following two observations.

(i) We have $Py = y$ for any $y \in \mathcal{R}(P)$.
Indeed, if $y \in \mathcal{R}(P)$ then $y = Px$ for some $x \in \mathbb{R}^n$, and hence, $Py = P^2x = Px = y$.

(ii) We have $Px - x \in \mathcal{N}(P)$ for any $x \in \mathbb{R}^n$.
Indeed, there holds $P(Px - x) = P^2x - Px = Px - Px = 0$ for any $x \in \mathbb{R}^n$.

The projector P projects onto $\mathcal{R}(P)$ along $\mathcal{N}(P)$.

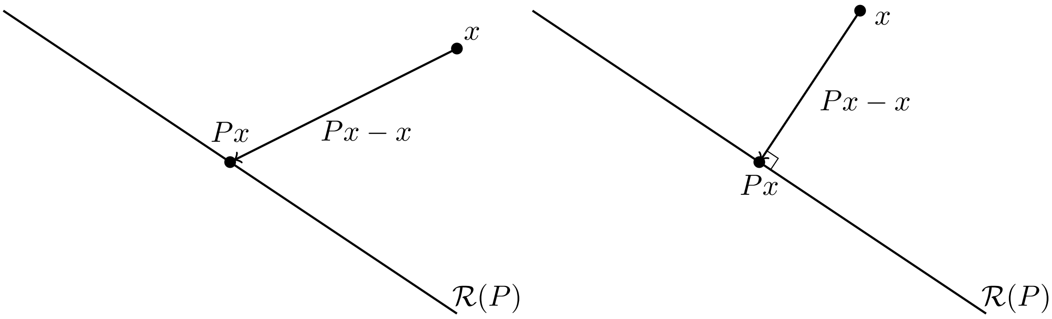

Figure 3: Left: Action of an oblique (i.e., a non-orthogonal) projector $P \in \mathbb{R}^{n\times n}$ on a vector $x \in \mathbb{R}^n$. Right: Action of an orthogonal projector $P \in \mathbb{R}^{n\times n}$ on a vector $x \in \mathbb{R}^n$.

Remark 3.9. Let $P \in \mathbb{R}^{n\times n}$ be a projector. Then,

$$(I_n - P)^2 = I_n^2 - 2P + P^2 = I_n - 2P + P = I_n - P,$$

i.e., $I_n - P \in \mathbb{R}^{n\times n}$ is a projector.

Definition 3.4. Let $P \in \mathbb{R}^{n\times n}$ be a projector. Then, $I_n - P \in \mathbb{R}^{n\times n}$ is called the complementary projector to P.

We are going to see that the complementary projector to P is the projector onto $\mathcal{N}(P)$ along $\mathcal{R}(P)$. Let us introduce the notion of complementary subspaces.

Definition 3.5. Let $S_1, S_2 \subseteq \mathbb{R}^n$ be subspaces of $\mathbb{R}^n$. Then, S_1 and S_2 are called complementary subspaces of $\mathbb{R}^n$ iff there holds $S_1 + S_2 = \mathbb{R}^n$ and $S_1 \cap S_2 = \{0\}$.

We can now show the following results.

Theorem 3.3. *Let $P \in \mathbb{R}^{n\times n}$ be a projector. Then, we have the following assertions.*

(i) $\mathcal{R}(I_n - P) = \mathcal{N}(P)$ and $\mathcal{N}(I_n - P) = \mathcal{R}(P)$.

(ii) $\mathcal{R}(P)$ and $\mathcal{N}(P)$ are complementary subspaces of $\mathbb{R}^n$. Further, for any $x \in \mathbb{R}^n$,

$$x = Px + (I_n - P)x \in \mathcal{R}(P) + \mathcal{N}(P)$$

is the unique way of writing $x = x_1 + x_2$ with $x_1 \in \mathcal{R}(P)$ and $x_2 \in \mathcal{N}(P)$.

Proof. (i) Let us start by proving that $\mathcal{R}(I_n - P) = \mathcal{N}(P)$. We have already observed in Remark 3.8(ii) that for any $x \in \mathbb{R}^n$ there holds $x - Px \in \mathcal{N}(P)$, and thus, $\mathcal{R}(I_n - P) \subseteq \mathcal{N}(P)$. To see that also $\mathcal{N}(P) \subseteq \mathcal{R}(I_n - P)$, note that if $x \in \mathcal{N}(P)$ we have $x = x - Px \in \mathcal{R}(I_n - P)$. It remains to prove that $\mathcal{N}(I_n - P) = \mathcal{R}(P)$. Since $I_n - P \in \mathbb{R}^{n\times n}$ is a projector, we have by the first part that $\mathcal{N}(I_n - P) = \mathcal{R}(I_n - (I_n - P)) = \mathcal{R}(P)$.

(ii) First, note $\mathcal{R}(P)$ and $\mathcal{N}(P)$ are subspaces of $\mathbb{R}^n$. Let us show that $\mathcal{R}(P) + \mathcal{N}(P) = \mathbb{R}^n$. Clearly, $\mathcal{R}(P) + \mathcal{N}(P) \subseteq \mathbb{R}^n$ since $\mathcal{R}(P) \subseteq \mathbb{R}^n$ and $\mathcal{N}(P) \subseteq \mathbb{R}^n$. For the converse, let $x \in \mathbb{R}^n$. Then, $x = Px + (I_n - P)x \in \mathcal{R}(P) + \mathcal{R}(I_n - P)$, i.e., $x \in \mathcal{R}(P) + \mathcal{N}(P)$ (note $\mathcal{R}(I_n - P) = \mathcal{N}(P)$ by (i)). We conclude that $\mathcal{R}(P) + \mathcal{N}(P) = \mathbb{R}^n$. Next, let us show that $\mathcal{R}(P) \cap \mathcal{N}(P) = \{0\}$. Clearly $0 \in \mathcal{R}(P) \cap \mathcal{N}(P)$ and it remains to show $\mathcal{R}(P) \cap \mathcal{N}(P) \subseteq \{0\}$. To this end, let $x \in \mathcal{R}(P) \cap \mathcal{N}(P)$. Then, $x = P\tilde{x}$ for some $\tilde{x} \in \mathbb{R}^n$, and there holds $Px = 0$. Hence, $P^2\tilde{x} = 0$ and since $P^2 = P$, we have $x = P\tilde{x} = 0$. We conclude that $\mathcal{R}(P) \cap \mathcal{N}(P) = \{0\}$.

Since $\mathcal{R}(P)$ and $\mathcal{N}(P)$ are complementary subspaces of $\mathbb{R}^n$, we deduce that any $x \in \mathbb{R}^n$ can be uniquely written as $x = x_1 + x_2$ with $x_1 \in \mathcal{R}(P)$ and $x_2 \in \mathcal{N}(P)$ (will follow from Step 1 in the proof of Theorem 3.4). □

We observe that a projector separates $\mathbb{R}^n$ into two complementary subspaces, namely $\mathcal{R}(P)$ and $\mathcal{N}(P)$. Conversely, for any two arbitrary complementary subspaces, we can find a suitable projector in the following sense.

Theorem 3.4. *Let $S_1, S_2 \subseteq \mathbb{R}^n$ be two complementary subspaces of $\mathbb{R}^n$. Then, there exists a unique projector $P \in \mathbb{R}^{n\times n}$ such that $\mathcal{R}(P) = S_1$ and $\mathcal{N}(P) = S_2$. We call this projector the projector onto S_1 along S_2.*

Proof. Let $S_1, S_2 \subseteq \mathbb{R}^n$ be two complementary subspaces of $\mathbb{R}^n$, i.e., $S_1 + S_2 = \mathbb{R}^n$ and $S_1 \cap S_2 = \{0\}$.

Step 1: We start by showing that any $x \in \mathbb{R}^n$ has a unique decomposition

$$x = x_1 + x_2 \quad \text{with} \quad x_1 \in S_1,\ x_2 \in S_2.$$

The existence of such a decomposition is guaranteed since $S_1 + S_2 = \mathbb{R}^n$, and it only remains to show uniqueness. To this end, suppose $x = x_1 + x_2 = \tilde{x}_1 + \tilde{x}_2$ for some $x_1, \tilde{x}_1 \in S_1$ and $x_2, \tilde{x}_2 \in S_2$. Then, we have $x_1 - \tilde{x}_1 \in S_1$, $\tilde{x}_2 - x_2 \in S_2$, and hence,

$$x_1 - \tilde{x}_1 = \tilde{x}_2 - x_2 \in S_1 \cap S_2 = \{0\},$$

i.e., $x_1 = \tilde{x}_1$ and $x_2 = \tilde{x}_2$.

Step 2: We construct a projector $P \in \mathbb{R}^{n\times n}$ such that $\mathcal{R}(P) = S_1$ and $\mathcal{N}(P) = S_2$. To this end, we define a linear map $L : \mathbb{R}^n \to \mathbb{R}^n$, $x \mapsto L(x)$, as follows: For a vector $x \in \mathbb{R}^n$ with $x = x_1 + x_2$ where $x_1 \in S_1$ and $x_2 \in S_2$, we define $L(x) := x_1$. Note that by Step 1 (existence and uniqueness of such a decomposition), this yields a well-defined map. We claim that L is linear, i.e., $L \in \mathcal{L}(\mathbb{R}^n, \mathbb{R}^n)$. Indeed, given $x = x_1 + x_2 \in \mathbb{R}^n$, $y = y_1 + y_2 \in \mathbb{R}^n$ with $x_1, y_1 \in S_1$, $x_2, y_2 \in S_2$, and $\alpha \in \mathbb{R}$, we have that

$$L(\alpha x + y) = L((\alpha x_1 + y_1) + (\alpha x_2 + y_2)) = \alpha x_1 + y_1 = \alpha L(x) + L(y),$$

since $\alpha x + y = (\alpha x_1 + y_1) + (\alpha x_2 + y_2)$ with $(\alpha x_i + y_i) \in S_i$ for $i \in \{1, 2\}$. As L is linear, we deduce that there exists a matrix $P \in \mathbb{R}^{n\times n}$ such that $L(x) = Px$ for any $x \in \mathbb{R}^n$ (see Theorem 1.1).

We check that P is a projector: For any $x = x_1 + x_2 \in \mathbb{R}^n$ with $x_1 \in S_1$, $x_2 \in S_2$, we have

$$P^2 x = L(L(x)) = L(x_1) = L(x_1 + 0) = x_1 = L(x) = Px.$$

It follows that $P^2 = P$ and hence, P is a projector.

We check that $\mathcal{R}(P) = S_1$: We have that $\mathcal{R}(P) = \{L(x) \,|\, x \in \mathbb{R}^n\} \subseteq S_1$ by definition of the map L. Conversely, for $y \in S_1$ note $y = y + 0$ with $y \in S_1$, $0 \in S_2$, so that $y = L(y) = Py \in \mathcal{R}(P)$.

We check that $\mathcal{N}(P) = S_2$: This holds as for any $x = x_1 + x_2 \in \mathbb{R}^n$ with $x_1 \in S_1$ and $x_2 \in S_2$ there holds $Px = 0$ iff $L(x) = 0$ iff $x_1 = 0$.

Step 3: We show that P is unique. To this end, suppose there exists another projector $\tilde{P} \in \mathbb{R}^{n\times n}$ such that $\mathcal{R}(\tilde{P}) = S_1$ and $\mathcal{N}(\tilde{P}) = S_2$. Then, in view of Remark 3.8, we must have $\tilde{P}y = y$ for any $y \in \mathcal{R}(\tilde{P}) = S_1$. Hence, for any $x \in \mathbb{R}^n$ with decomposition $x = x_1 + x_2$ where $x_1 \in S_1$, $x_2 \in S_2$, we must have that

$$\tilde{P}x = \tilde{P}x_1 + \tilde{P}x_2 = x_1 + 0 = x_1 = L(x) = Px.$$

We deduce that $\tilde{P} = P$ and the claim is proved. □

Let us now turn our attention to the important class of orthogonal projectors.

Definition 3.6. A projector $P \in \mathbb{R}^{n\times n}$ is called an orthogonal projector iff it projects onto S_1 along S_2 for some subspaces S_1, S_2 of $\mathbb{R}^n$ with $S_1 \perp S_2$. A projector which is not an orthogonal projector is called an oblique projector.

Note that an orthogonal projector does not need to be an orthogonal matrix. Actually, if $P \in \mathbb{R}^{n\times n}$ is an orthogonal projector that is also an orthogonal matrix ($PP^{\mathrm{T}} = P^{\mathrm{T}}P = I_n$), it follows from the next result that then, P must be the identity matrix $P = I_n$.

We have the following characterization of orthogonal projectors:

Theorem 3.5. *A square matrix $P \in \mathbb{R}^{n\times n}$ is an orthogonal projector iff there holds*

$$P^2 = P = P^{\mathrm{T}}.$$

Proof. "$\Longleftarrow$": Let $P \in \mathbb{R}^{n\times n}$ with $P^2 = P = P^{\mathrm{T}}$. Then, as $P^2 = P$, we have that P is a projector (onto $\mathcal{R}(P)$ along $\mathcal{N}(P)$). We need to show that $\mathcal{R}(P) \perp \mathcal{N}(P)$. To this end, let $x \in \mathcal{N}(P)$ and $y \in \mathcal{R}(P)$, and recall from Theorem 3.3 that $\mathcal{N}(P) = \mathcal{R}(I_n - P)$. Hence, we have $x = (I_n - P)u = u - Pu$ and $y = Pv$ for some $u, v \in \mathbb{R}^n$, and we find

$$\langle x, y\rangle = \langle u - Pu, Pv\rangle = \langle u, Pv\rangle - \langle Pu, Pv\rangle = \langle u, Pv\rangle - \langle u, P^{\mathrm{T}}Pv\rangle = 0,$$

where we have used in the last equality that $P^{\mathrm{T}}P = P^2 = P$. Therefore, P is an orthogonal projector.

"$\Longrightarrow$": Let $P \in \mathbb{R}^{n\times n}$ be an orthogonal projector, i.e., $P^2 = P$ and P projects onto S_1 along S_2 for some subspaces S_1, S_2 of $\mathbb{R}^n$ with $S_1 \perp S_2$. Then, writing $r := \dim(S_1)$, there exists an orthonormal basis $\{q_1, \ldots, q_n\}$ of $\mathbb{R}^n$ such that $\{q_1, \ldots, q_r\}$ is a basis of S_1 and $\{q_{r+1}, \ldots, q_n\}$ is a basis of S_2. Let us set $Q := (q_1|\cdots|q_n) \in \mathbb{R}^{n\times n}$ and note that Q is orthogonal. Noting that

$$Pq_i = \begin{cases} q_i & \text{, if } 1 \le i \le r, \\ 0 & \text{, if } r < i \le n, \end{cases}$$

we have that

$$Q^{\mathrm{T}}PQ = \mathrm{diag}_{n\times n}(\underbrace{1, \ldots, 1}_{\text{r times}}, \underbrace{0, \ldots, 0}_{\text{(n-r) times}}) =: \Sigma \in \mathbb{R}^{n\times n}.$$

This yields that $P = Q\Sigma Q^{\mathrm{T}}$ is a SVD (and an eigenvalue decomposition) of P. We find that

$$P^{\mathrm{T}} = Q\Sigma^{\mathrm{T}}Q^{\mathrm{T}} = Q\Sigma Q^{\mathrm{T}} = P$$

since Σ is symmetric. □

Theorem 3.6. *Let $P \in \mathbb{R}^{n\times n}\backslash\{0\}$ be a projector. Then, we have the following:*

(i) all non-zero singular values of P are greater than or equal to 1.

(ii) P is an orthogonal projector iff $\|P\|_2 = 1$.

Proof. The proof of (i) is an exercise. Assuming that (i) holds, let us prove (ii).

"$\Longrightarrow$": If P is an orthogonal projector, we have seen in the proof of Theorem 3.5 that all non-zero singular values of P are equal to 1. We find that $\|P\|_2 = 1$ (note since $P \neq 0_{n\times n}$, there is at least one non-zero singular value).

"$\Longleftarrow$": Suppose $P \in \mathbb{R}^{n\times n}\backslash\{0\}$ is a projector with $\|P\|_2 = 1$, and write $r := \mathrm{rk}(P)$. Let $P = U\Sigma V^{\mathrm{T}} = (u_1|\cdots|u_n)\mathrm{diag}_{n\times n}(\sigma_1,\ldots,\sigma_n)(v_1|\cdots|v_n)^{\mathrm{T}}$ be a SVD of P. Since $1 = \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$, it follows from (i) that $\sigma_i = 1$ for all $i \in \{1,\ldots,r\}$. We deduce that $P = \sum_{i=1}^{r}\sigma_i u_i v_i^{\mathrm{T}} = \sum_{i=1}^{r} u_i v_i^{\mathrm{T}}$ and $P^{\mathrm{T}} = \sum_{i=1}^{r} v_i u_i^{\mathrm{T}}$. Note that as $Pu_j = u_j$ for all $j \in \{1,\ldots,r\}$ (since $u_1,\ldots,u_r \in \mathcal{R}(P)$), we have $\sum_{i=1}^{r} u_i\langle v_i, u_j\rangle = u_j$ and hence (left-multiply by u_j^{T}) $\langle v_j, u_j\rangle = 1$ for all $j \in \{1,\ldots,r\}$. This gives $v_j = u_j$ for all $j \in \{1,\ldots,r\}$ (note $\|v_j - u_j\|_2^2 = \|v_j\|_2^2 + \|u_j\|_2^2 - 2\langle v_j, u_j\rangle = 0$) and we conclude that $P = P^{\mathrm{T}}$. In view of Theorem 3.5, this implies that P is an orthogonal projector. □

Remark 3.10 (Projection with orthonormal basis). Let $\{q_1,\ldots,q_n\}$ be an orthonormal basis of $\mathbb{R}^n$, and consider the complementary subspaces $S_1 := \mathrm{span}(q_1,\ldots,q_r)$ and $S_2 := \mathrm{span}(q_{r+1},\ldots,q_n)$ of $\mathbb{R}^n$, where $1 \leq r \leq n-1$. Then, the projector onto S_1 along S_2 is given by

$$P = \hat{Q}\hat{Q}^{\mathrm{T}} \in \mathbb{R}^{n\times n},$$

where $\hat{Q} := (q_1|\cdots|q_r) \in \mathbb{R}^{n\times r}$ (note $\mathcal{R}(P) = \mathcal{R}(\hat{Q}) = S_1$ and $\mathcal{N}(P) = \mathcal{N}(\hat{Q}^{\mathrm{T}}) = S_2$). Note that P is an orthogonal projector as $S_1 \perp S_2$ (or note $P^2 = P = P^{\mathrm{T}}$). The corresponding linear map

$$L_P : \mathbb{R}^n \to \mathbb{R}^n, \quad x \mapsto \hat{Q}\hat{Q}^{\mathrm{T}}x = \sum_{i=1}^{r}(q_i q_i^{\mathrm{T}})x = \sum_{i=1}^{r}\langle x, q_i\rangle q_i$$

projects the vector space $\mathbb{R}^n$ orthogonally onto S_1 along S_2, i.e., it isolates the components of a vector in directions $q_1,\ldots,q_r$. Note that the complementary projector $I_n - P$ is also an orthogonal projector: it is the projector onto S_2 along S_1, i.e., it isolates the components of a vector in directions $q_{r+1},\ldots,q_n$. The corresponding linear map is

$$L_{I_n-P} : \mathbb{R}^n \to \mathbb{R}^n, \quad x \mapsto (I_n - \hat{Q}\hat{Q}^{\mathrm{T}})x = \sum_{i=r+1}^{n}(q_i q_i^{\mathrm{T}})x = \sum_{i=r+1}^{n}\langle x, q_i\rangle q_i.$$

Observe that we can decompose any $x \in \mathbb{R}^n$ uniquely into $x = x_1 + x_2$ with $x_1 \in S_1$, $x_2 \in S_2$, where $x_1 = \hat{Q}\hat{Q}^{\mathrm{T}}x$ and $x_2 = (I_n - \hat{Q}\hat{Q}^{\mathrm{T}})x$.

Remark 3.11 (Projection with arbitrary basis). Let S_1 be a subspace of $\mathbb{R}^m$ spanned by $n \leq m$ linearly independent vectors $a_1, \dots, a_n \in \mathbb{R}^m$. We set $A := (a_1|\cdots|a_n) \in \mathbb{R}^{m\times n}$ so that $S_1 = \mathcal{R}(A)$, and we want to construct an orthogonal projector $P \in \mathbb{R}^{m\times m}$ onto S_1. For $x \in \mathbb{R}^m$ we must have $Px \in S_1$, i.e., $Px = Ay$ for some $y \in \mathbb{R}^n$, and $\{Px - x\} \perp S_1$, i.e.,

$$0_{n\times 1} = \begin{pmatrix} \langle a_1, Px - x\rangle \\ \vdots \\ \langle a_n, Px - x\rangle \end{pmatrix} = A^{\mathrm{T}}(Px - x) = A^{\mathrm{T}}Ay - A^{\mathrm{T}}x.$$

Note that, in view of Theorem 1.2, we have $\operatorname{rk}(A^{\mathrm{T}}A) = \operatorname{rk}(A) = n$ and hence, $A^{\mathrm{T}}A \in \mathbb{R}^{n\times n}$ is invertible. We find that $y = (A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}}x$ and thus $Px = Ay = A(A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}}x$. We conclude that the orthogonal projector onto $S_1 = \mathcal{R}(A)$ is given by

$$P = A(A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}} \in \mathbb{R}^{m\times m}.$$

Observe that if $A = \hat{Q}$ has orthonormal columns, this reduces to $P = \hat{Q}\hat{Q}^{\mathrm{T}}$.

Remark 3.12 (Uniqueness of the orthogonal projector onto a given subspace). Let $S \subseteq \mathbb{R}^n$ be a subspace of $\mathbb{R}^n$. Then, there exists a unique orthogonal projector $P \in \mathbb{R}^{n\times n}$ onto S. In order to show uniqueness, suppose that $P_1, P_2 \in \mathbb{R}^{n\times n}$ are orthogonal projectors with $\mathcal{R}(P_1) = \mathcal{R}(P_2) = S$. Then, we have that $(P_1 - P_2)x = P_1x - P_2x \in S$ for all $x \in \mathbb{R}^n$, and also $(P_1 - P_2)x = (x - P_2x) - (x - P_1x) \in S^\perp$ (note that for $i \in \{1, 2\}$ there holds $\langle x - P_ix, y\rangle = \langle x, y - P_i^{\mathrm{T}}y\rangle = \langle x, y - P_iy\rangle = 0$ for all $y \in \mathcal{R}(P_i) = S$). It follows that $(P_1 - P_2)x = 0 \in \mathbb{R}^n$ for any $x \in \mathbb{R}^n$ and thus, $P_1 = P_2$.

Observe that for the unique orthogonal projector $P \in \mathbb{R}^{n\times n}$ onto S we have that $\mathcal{N}(P) = S^\perp$ (exercise), i.e., P is the projector onto S along $S^\perp$.

3.4 QR via Gram–Schmidt orthogonalization

Recall the Gram–Schmidt algorithm 3.1 for full-rank matrices, as well as the adjustments for rank-deficient matrices discussed in Section 3.2. Let us now provide a re-interpretation of Algorithm 3.1 in terms of orthogonal projections.

Remark 3.13 (Gram–Schmidt via projectors). Let $A = (a_1|\cdots|a_n) \in \mathbb{R}^{m\times n}$, $m \geq n$, and assume $\operatorname{rk}(A) = n$. Let $q_1, \dots, q_n \in \mathbb{R}^m$ be the orthonormal vectors obtained through Algorithm 3.1 and define

$$\begin{aligned} &P_1 := I_m \\ &P_i := I_m - \hat{Q}_{i-1}\hat{Q}_{i-1}^{\mathrm{T}}, \quad \text{where} \quad \hat{Q}_{i-1} := (q_1|\cdots|q_{i-1}) \in \mathbb{R}^{m\times i}, \qquad 2 \leq i \leq n. \end{aligned}$$

Note that $P_i \in \mathbb{R}^{m\times m}$ projects the vector space $\mathbb{R}^m$ onto the space orthogonal to $\operatorname{span}(q_1, \dots, q_{i-1})$. Then, we observe that $q_1, \dots, q_n$ are given by

$$q_1 = \frac{P_1a_1}{\|P_1a_1\|_2}, \quad q_2 = \frac{P_2a_2}{\|P_2a_2\|_2}, \quad \cdots \quad , \quad q_n = \frac{P_na_n}{\|P_na_n\|_2},$$

i.e., q_i is precisely the normalized orthogonal projection of a_i onto the space orthogonal to $\operatorname{span}(q_1, \dots, q_{i-1})$.

Written down in an algorithmic way, we have the following algorithm.

Algorithm 3.2 (Classical Gram–Schmidt iteration)**.** Let $A = (a_1|\cdots|a_n) \in \mathbb{R}^{m\times n}$, $m \geq n$, and $\mathrm{rk}(A) = n$. The classical Gram–Schmidt iteration does the following:

```
for $j = 1, \dots, n$ do
  $\tilde{q}_j = a_j$
  for $i = 1, \dots, j-1$ do
    $r_{ij} = \langle q_i, a_j \rangle$
    $\tilde{q}_j = \tilde{q}_j - r_{ij} q_i$
  end for
  $r_{jj} = \|\tilde{q}_j\|_2$
  $q_j = \frac{1}{r_{jj}} \tilde{q}_j$
end for
```

We call this the classical Gram–Schmidt iteration as, unfortunately, it is numerically unstable (sensitive to rounding errors, we will discuss numerical stability later). However, a simple modification leads to improved stability. The key observation is that the projector $P_i = I_m - \hat{Q}_{i-1}\hat{Q}_{i-1}^{\mathrm{T}} \in \mathbb{R}^{m\times m}$ of rank $m-(i-1)$ from Remark 3.13 can be decomposed as the product of $i-1$ projectors of rank $m-1$:

$$P_i = (I_m - q_{i-1}q_{i-1}^{\mathrm{T}})(I_m - q_{i-2}q_{i-2}^{\mathrm{T}}) \cdots (I_m - q_1 q_1^{\mathrm{T}}), \qquad 2 \leq i \leq n.$$

The modified Gram–Schmidt iteration is given below.

Algorithm 3.3 (Modified Gram–Schmidt iteration)**.** Let $A = (a_1|\cdots|a_n) \in \mathbb{R}^{m\times n}$, $m \geq n$, and $\mathrm{rk}(A) = n$. The modified Gram–Schmidt iteration does the following:

```
for $i = 1, \dots, n$ do
  $\tilde{q}_i = a_i$
end for
for $i = 1, \dots, n$ do
  $r_{ii} = \|\tilde{q}_i\|_2$
  $q_i = \frac{1}{r_{ii}} \tilde{q}_i$
  for $j = i+1, \dots, n$ do
    $r_{ij} = \langle q_i, \tilde{q}_j \rangle$
    $\tilde{q}_j = \tilde{q}_j - r_{ij} q_i$
  end for
end for
```

We can assess the work of algorithms 3.2 and 3.3 by counting the number of floating point operations, abbreviated flops. Every addition, subtraction, multiplication, division and square root is counted as one flop.

Theorem 3.7. *Algorithms 3.2 and 3.3 require* $\sim 2mn^2$ *flops.*

Proof. Let us only look at Algorithm 3.3. We have the following number of additions:

$$\begin{aligned}\sum_{i=1}^{n}\left((m-1) + \sum_{j=i+1}^{n}(m-1)\right) &= (m-1)n + \sum_{i=1}^{n}(m-1)(n-i) \\ &= (m-1)\left(n + n^2 - \frac{n(n+1)}{2}\right) = \frac{1}{2}(m-1)n(n+1).\end{aligned}$$

We have the following number of subtractions:

$$\sum_{i=1}^{n}\sum_{j=i+1}^{n} m = m\sum_{i=1}^{n}(n-i) = m\left(n^2 - \frac{n(n+1)}{2}\right) = \frac{1}{2}mn(n-1).$$

We have the following number of multiplications:

$$\sum_{i=1}^{n}\left(m + \sum_{j=i+1}^{n}(m+m)\right) = m\left(n + 2\sum_{i=1}^{n}(n-i)\right) = m\left(n + 2n^2 - n(n+1)\right) = mn^2.$$

Further, we have $\sum_{i=1}^{n} m = mn$ divisions, and $\sum_{i=1}^{n} 1 = n$ square roots. In total, we have

$$\#\text{flops} = \frac{1}{2}(m-1)n(n+1) + \frac{1}{2}mn(n-1) + mn^2 + mn + n = 2mn^2 + (m - \tfrac{n-1}{2})n,$$

from which we see that $\#\text{flops} \sim 2mn^2$. □

Here, $\#\text{flops} \sim 2mn^2$ means that $\lim_{m,n\to\infty} \frac{\#\text{flops}}{2mn^2} = 1$.

Remark 3.14 (Gram–Schmidt = triangular orthogonalization). The outer steps of Algorithm 3.3 can be regarded as right-multiplication by an upper-triangular square matrix: Schematically, the method does the following:

1. $AR_1 = (a_1|\cdots|a_n)\begin{pmatrix} \frac{1}{r_{11}} & -\frac{r_{12}}{r_{11}} & -\frac{r_{13}}{r_{11}} & \cdots & -\frac{r_{1n}}{r_{11}} \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & 0 \\ 0 & \cdots & \cdots & 0 & 1 \end{pmatrix} = (q_1|*|\cdots|*)$,

2. $AR_1R_2 = (q_1|*|\cdots|*)\begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & \frac{1}{r_{22}} & -\frac{r_{23}}{r_{22}} & \cdots & -\frac{r_{2n}}{r_{22}} \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & 0 \\ 0 & \cdots & \cdots & 0 & 1 \end{pmatrix} = (q_1|q_2|*|\cdots|*)$

$\vdots$

n. $AR_1R_2\cdots R_n = (q_1|q_2|\cdots|q_n) = \hat{Q}$, i.e., we have $A = \hat{Q}\hat{R}$ with $\hat{R} = (R_1\cdots R_n)^{-1}$.

Gram–Schmidt is a *triangular orthogonalization* method.

Next, we will discuss an *orthogonal triangulation* method for obtaining QR factorizations, the so-called Householder triangularization.

3.5 QR via Householder triangularization

An alternative method for computing QR factorizations is the so-called Householder triangularization. Recall from Remark 3.14 that Gram–Schmidt is a method of triangular orthogonalization, i.e., A is transformed into a matrix with orthonormal columns via right-multiplication by upper-triangular matrices ($AR_1R_2\cdots R_n = \hat{Q}$, giving a reduced QR factorization $A = \hat{Q}\hat{R}$ with $\hat{R} = (R_1\cdots R_n)^{-1}$). On the contrary, Householder triangularization is a method of orthogonal triangulation, i.e., A is transformed into an upper-triangular matrix via left-multiplication by orthogonal matrices:

$$Q_n\cdots Q_2Q_1A = R. \tag{3.5}$$

Remark 3.15. Let $A \in \mathbb{R}^{m\times n}$, $m \geq n$, and suppose we have found orthogonal matrices $Q_1,\ldots,Q_n \in \mathbb{R}^{m\times m}$ and an upper-triangular matrix $R \in \mathbb{R}^{m\times n}$ such that (3.5) holds. Then, $A = QR$ with $Q := Q_1^{\mathrm{T}}Q_2^{\mathrm{T}}\cdots Q_n^{\mathrm{T}} \in \mathbb{R}^{m\times m}$ is a (full) QR factorization of A (note that the product of orthogonal matrices is orthogonal).

We are going to construct the matrices $Q_1,\ldots,Q_n \in \mathbb{R}^{m\times m}$ in a way so that $A \in \mathbb{R}^{m\times n}$, $m \geq n$ is transformed as follows: (illustration for $m = 4$, $n = 3$; l.m. short for left-multiply)

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \\ a_{41} & a_{42} & a_{43} \end{pmatrix} \underset{\text{l.m.}Q_1}{\Longrightarrow} \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ 0 & * & * \\ 0 & * & * \\ 0 & * & * \end{pmatrix} \underset{\text{l.m.}Q_2}{\Longrightarrow} \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ 0 & r_{22} & r_{23} \\ 0 & 0 & * \\ 0 & 0 & * \end{pmatrix} \underset{\text{l.m.}Q_3}{\Longrightarrow} \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ 0 & r_{22} & r_{23} \\ 0 & 0 & r_{33} \\ 0 & 0 & 0 \end{pmatrix}.$$

So, left-multiplication by Q_k should leave the first $(k-1)$ rows unchanged and introduce zeros below the k-th main diagonal entry, thus leading to an upper-triangular matrix $R = Q_n\cdots Q_2Q_1A$ after n such steps. We choose Q_i, $i \in \{1,\ldots,n\}$, to be an orthogonal matrix of the form

$$Q_i = \left(\begin{array}{c|c} I_{i-1} & 0_{(i-1)\times(m-i+1)} \\ \hline 0_{(m-i+1)\times(i-1)} & F \end{array}\right) \in \mathbb{R}^{m\times m},$$

where $F \in \{F_-, F_+\} \in \mathbb{R}^{(m-i+1)\times(m-i+1)}$ should act on vectors in $\mathbb{R}^{m-i+1}$ as follows:

$$x = \begin{pmatrix} \langle x, e_1\rangle \\ \langle x, e_2\rangle \\ \vdots \\ \langle x, e_{m-i+1}\rangle \end{pmatrix} \qquad \underset{\text{l.m.}F_\pm}{\Longrightarrow} \qquad F_\pm x = \begin{pmatrix} \pm\|x\|_2 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \pm\|x\|_2 e_1.$$

The idea is illustrated below: $F_\pm$ reflects the space $\mathbb{R}^{m-i+1}$ along the hyperplane $H_\pm$ orthogonal to the vector $v_\pm = \pm\|x\|_2e_1 - x$.

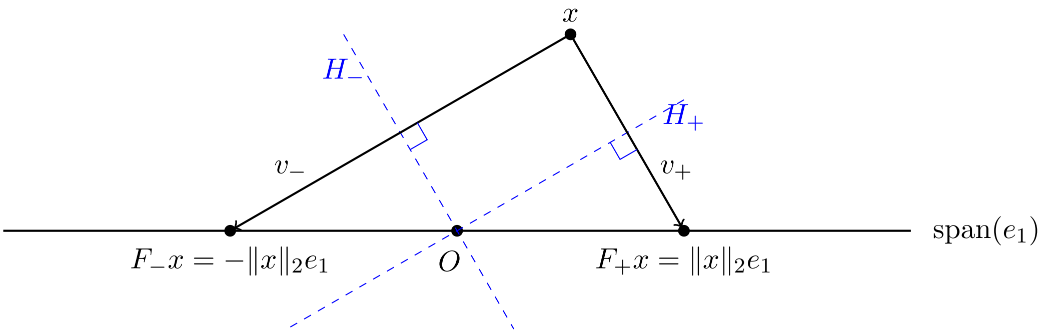

Figure 4: Illustration of Householder reflectors $F_\pm$.

Noting that $I_{m-i+1} - \frac{vv^{\mathrm{T}}}{\|v\|_2^2} \in \mathbb{R}^{(m-i+1)\times(m-i+1)}$ is the orthogonal projector onto the hyperplane orthogonal to $v \in \mathbb{R}^{m-i+1}$, we find that (need to go twice as far)

$$F = I_{m-i+1} - 2\frac{vv^{\mathrm{T}}}{\|v\|_2^2}$$

is as required. We call F a Householder reflector.

In view of numerical stability, one should choose the reflector which moves x the larger distance, i.e., we choose

$$v = \operatorname{sign}(\langle x, e_1\rangle)\|x\|_2 e_1 + x,$$

where we define $\operatorname{sign}(\alpha) = 1$ for $\alpha \geq 0$ and $\operatorname{sign}(\alpha) = -1$ otherwise.

Example 3.3. We compute a QR factorization of the full-rank matrix $A \in \mathbb{R}^{4\times 3}$ defined in (3.4) via Householder triangularization.

Q_1: Set $x_1 := a_1 = (1, -1, 1, 1)^{\mathrm{T}}$ and $v_1 := \operatorname{sign}(\langle x_1, e_1\rangle)\|x_1\|_2 e_1 + x_1 = (3, -1, 1, 1)^{\mathrm{T}}$. Then, compute

$$Q_1 := I_4 - 2\frac{v_1 v_1^{\mathrm{T}}}{\|v_1\|_2^2} = \frac{1}{6}\begin{pmatrix} -3 & 3 & -3 & -3 \\ 3 & 5 & 1 & 1 \\ -3 & 1 & 5 & -1 \\ -3 & 1 & -1 & 5 \end{pmatrix}, \qquad Q_1 A = \begin{pmatrix} -2 & -1 & 0 \\ 0 & \frac{4}{3} & \frac{4}{3} \\ 0 & \frac{2}{3} & -\frac{4}{3} \\ 0 & \frac{5}{3} & \frac{2}{3} \end{pmatrix}.$$

Q_2: Set $x_2 := (\frac{4}{3}, \frac{2}{3}, \frac{5}{3})^{\mathrm{T}}$ and $v_2 := \operatorname{sign}(\langle x_2, e_1\rangle)\|x_2\|_2 e_1 + x_2 = (\sqrt{5} + \frac{4}{3}, \frac{2}{3}, \frac{5}{3})^{\mathrm{T}}$. Then,

$$Q_2 := \left(\begin{array}{c|c} 1 & 0_{1\times 3} \\ \hline 0_{3\times 1} & I_3 - 2\frac{v_2 v_2^{\mathrm{T}}}{\|v_2\|_2^2} \end{array}\right) = \frac{\sqrt{5}}{435}\begin{pmatrix} \frac{435}{\sqrt{5}} & 0 & 0 & 0 \\ 0 & -116 & -58 & -145 \\ 0 & -58 & 75\sqrt{5} + 16 & -(30\sqrt{5} - 40) \\ 0 & -145 & -(30\sqrt{5} - 40) & 12\sqrt{5} + 100 \end{pmatrix}$$

and we have $Q_2 Q_1 A = \begin{pmatrix} -2 & -1 & 0 \\ 0 & -\sqrt{5} & -\frac{2}{\sqrt{5}} \\ 0 & 0 & -\frac{24\sqrt{5}+200}{145} \\ 0 & 0 & -\frac{12\sqrt{5}-16}{29} \end{pmatrix}$.

Q_3: Set $x_3 := (-\frac{24\sqrt{5}+200}{145}, -\frac{12\sqrt{5}-16}{29})^{\mathrm{T}}$ and compute $v_3 := \operatorname{sign}(\langle x_3, e_1\rangle)\|x_3\|_2 e_1 + x_3 = -\frac{4}{29}(7\sqrt{5} + 10, 3\sqrt{5} - 4)^{\mathrm{T}}$. Then,

$$Q_3 := \left(\begin{array}{c|c} I_2 & 0_{2\times 2} \\ \hline 0_{2\times 2} & I_2 - 2\frac{v_3 v_3^{\mathrm{T}}}{\|v_3\|_2^2} \end{array}\right) = \frac{1}{29}\begin{pmatrix} 29 & 0 & 0 & 0 \\ 0 & 29 & 0 & 0 \\ 0 & 0 & -10\sqrt{5} - 6 & 4\sqrt{5} - 15 \\ 0 & 0 & 4\sqrt{5} - 15 & 10\sqrt{5} + 6 \end{pmatrix}.$$

and we have $Q_3 Q_2 Q_1 A = \begin{pmatrix} -2 & -1 & 0 \\ 0 & -\sqrt{5} & -\frac{2}{\sqrt{5}} \\ 0 & 0 & \frac{4}{\sqrt{5}} \\ 0 & 0 & 0 \end{pmatrix}$.

Noting that Q_1, Q_2, Q_3 are symmetric orthogonal matrices, we find that $A = QR$ with

$$Q := Q_1Q_2Q_3 = \begin{pmatrix} -\frac{1}{2} & \frac{1}{2\sqrt{5}} & \frac{3}{2\sqrt{5}} & -\frac{1}{2} \\ \frac{1}{2} & -\frac{3}{2\sqrt{5}} & \frac{1}{2\sqrt{5}} & -\frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2\sqrt{5}} & -\frac{3}{2\sqrt{5}} & -\frac{1}{2} \\ -\frac{1}{2} & -\frac{3}{2\sqrt{5}} & \frac{1}{2\sqrt{5}} & \frac{1}{2} \end{pmatrix}, \qquad R := \begin{pmatrix} -2 & -1 & 0 \\ 0 & -\sqrt{5} & -\frac{2}{\sqrt{5}} \\ 0 & 0 & \frac{4}{\sqrt{5}} \\ 0 & 0 & 0 \end{pmatrix}$$

is a QR factorization of A.

Let us note that in practice, one would not form all of the above matrices explicitly. To compute the factor R of a QR factorization of A, we can do the following:

Algorithm 3.4 (Householder triangularization)**.** Let $m, n \in \mathbb{N}$ with $m \geq n$. For a matrix $A \in \mathbb{R}^{m\times n}$, the Householder triangularization produces the factor R of a QR factorization $A = QR$ and goes as follows:

```
for $i = 1, \dots, n$ do
    $x = A_{i:m,i}$
    $v_i = \operatorname{sign}(x_1)\|x\|_2 e_1 + x$      ($x_1$ denotes the first entry of $x$)
    $v_i = \frac{1}{\|v_i\|_2} v_i$
    $A_{i:m,i:n} = A_{i:m,i:n} - 2v_i(v_i^{\mathrm{T}} A_{i:m,i:n})$
end for
```

This algorithm stores the result R in place of A. The reflection vectors $v_1, \dots, v_n$ are stored for applying and forming Q (see Algorithms 3.5 and 3.6).

Theorem 3.8. *Algorithm 3.4 requires* $\sim 2mn^2 - \frac{2}{3}n^3$ *flops.*

Proof. Omitted. □

Notation: Here, we have written $A_{i_1:i_2,j_1:j_2}$ to denote the $(i_2 - i_1 + 1) \times (j_2 - j_1 + 1)$ sub-matrix of A with top-left corner $a_{i_1j_1}$ and bottom-right corner $a_{i_2j_2}$.

For practical applications, there is often no need to construct Q explicitly. However, in view of Remark 3.7, we need to be able to compute matrix-vector products $Q^{\mathrm{T}}b$. This can be achieved with the following algorithm: (note $Q = Q_1Q_2 \cdots Q_n$, $Q^{\mathrm{T}} = Q_n \cdots Q_2Q_1$ since the matrices Q_i are symmetric orthogonal matrices)

Algorithm 3.5 (Computing $Q^{\mathrm{T}}b$ implicitly)**.** After running Algorithm 3.4, a product $Q^{\mathrm{T}}b$ with a given $b \in \mathbb{R}^m$ can be calculated via:

```
for $i = 1, \dots, n$ do
    $b_{i:m} = b_{i:m} - 2v_i(v_i^{\mathrm{T}} b_{i:m})$
end for,
```

leaving the result $Q^{\mathrm{T}}b$ in place of b.

If it is required to explicitly form Q, this can be done by computing the columns $Qe_1, \dots Qe_m$ via the following algorithm:

Algorithm 3.6 (Computing Qx implicitly)**.** After running Algorithm 3.4, a product Qx with a given $x \in \mathbb{R}^m$ can be calculated via:

```
for $i = n, n-1, \dots, 1$ do
    $x_{i:m} = x_{i:m} - 2v_i(v_i^{\mathrm{T}} x_{i:m})$
end for,
```

leaving the result Qx in place of x.

3.6 QR via Givens rotations

Finally, we give a brief overview of a third method for computing QR factorizations: Givens rotations. This method is particularly useful for sparse matrices (i.e., if there are only few entries below the diagonal which need to be eliminated to reach upper-triangular form). The key observation is the following:

Recall from Remark 1.7 that any orthogonal matrix $Q \in \mathbb{R}^{2\times 2}$ with $\det(Q) = 1$ is of the form

$$Q(\theta) = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}, \quad \theta \in [0, 2\pi),$$

and that $L_{Q(\theta)}$ rotates the plane $\mathbb{R}^2$ anticlockwise by the angle θ. Now, given some vector $x = (x_1, x_2)^{\mathrm{T}} \in \mathbb{R}^2$ with $x_2 \neq 0$, we can eliminate its second component by rotating x onto the vector $Q(\theta)x = (\|x\|_2, 0)^{\mathrm{T}}$ using a suitable angle θ. Indeed, using the matrix $Q(\theta)$ with $\theta \in [0, 2\pi)$ satisfying

$$\cos(\theta) = \frac{x_1}{\|x\|_2}, \qquad \sin(\theta) = -\frac{x_2}{\|x\|_2} \tag{3.6}$$

(note such a θ exists as $\|(\frac{x_1}{\|x\|_2}, -\frac{x_2}{\|x\|_2})^{\mathrm{T}}\|_2 = 1$), we have that

$$Q(\theta)x = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} \sqrt{x_1^2 + x_2^2} \\ 0 \end{pmatrix}.$$

For simplicity, we will illustrate how to transform a matrix into upper-triangular form using Givens rotations at the following explicit example:

$$A = \begin{pmatrix} -2 & -1 & 1 \\ 3 & 2 & -1 \\ 4 & 1 & 4 \end{pmatrix}.$$

Givens rotations in 3D are the following matrices (or rather their associated linear maps):

$$G_1(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{pmatrix}, \quad L_{G_1(\theta)} : \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \mapsto \begin{pmatrix} x_1 \\ \tilde{x}_2 \\ \tilde{x}_3 \end{pmatrix} \text{ where } \begin{pmatrix} \tilde{x}_2 \\ \tilde{x}_3 \end{pmatrix} = Q(\theta) \begin{pmatrix} x_2 \\ x_3 \end{pmatrix},$$

$$G_2(\theta) = \begin{pmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{pmatrix}, \quad L_{G_2(\theta)} : \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \mapsto \begin{pmatrix} \tilde{x}_1 \\ x_2 \\ \tilde{x}_3 \end{pmatrix} \text{ where } \begin{pmatrix} \tilde{x}_1 \\ \tilde{x}_3 \end{pmatrix} = Q(\theta) \begin{pmatrix} x_1 \\ x_3 \end{pmatrix},$$

$$G_3(\theta) = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad L_{G_3(\theta)} : \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \mapsto \begin{pmatrix} \tilde{x}_1 \\ \tilde{x}_2 \\ x_3 \end{pmatrix} \text{ where } \begin{pmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{pmatrix} = Q(\theta) \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}.$$

Note that the matrices $G_i(\theta)$, $i \in \{1, 2, 3\}$, are orthogonal matrices.

Step 1: Look at the first column of A and choose an entry below the diagonal which we would like to eliminate, and choose an entry you would like to use for this elimination. Say, we would like to eliminate the entry $a_{31} = 4$ by using the entry $a_{21} = 3$, thus leaving the first row of A unchanged. To this end, we will use the Givens rotation $G_1(\theta)$ with θ such that $Q(\theta)$ rotates $(3, 4)^{\mathrm{T}}$ onto the vector $(\sqrt{3^2 + 4^2}, 0)^{\mathrm{T}} = (5, 0)^{\mathrm{T}}$. We know from

(3.6) what to do: we take $\theta \in [0, 2\pi)$ such that $\cos(\theta) = \frac{3}{5}$ and $\sin(\theta) = -\frac{4}{5}$ (we are not interested in the precise value of θ). Then,

$$G_1 := \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{3}{5} & \frac{4}{5} \\ 0 & -\frac{4}{5} & \frac{3}{5} \end{pmatrix}, \qquad G_1 A = \begin{pmatrix} -2 & -1 & 1 \\ 5 & 2 & \frac{13}{5} \\ 0 & -1 & \frac{16}{5} \end{pmatrix}.$$

Step 2: Next, eliminate the (2,1)-entry of $G_1 A$ using the (1,1)-entry (and we leave the third row of $G_1 A$ unchanged). To this end, we will use a Givens rotation $G_3(\theta)$ with θ such that $G(\theta)$ rotates $(-2, 5)^{\mathrm{T}}$ onto $(\sqrt{(-2)^2 + 5^2}, 0)^{\mathrm{T}} = (\sqrt{29}, 0)^{\mathrm{T}}$. We know from (3.6) what to do: we take $\theta \in [0, 2\pi)$ such that $\cos(\theta) = -\frac{2}{\sqrt{29}}$ and $\sin(\theta) = -\frac{5}{\sqrt{29}}$. Then,

$$G_3 := \begin{pmatrix} -\frac{2}{\sqrt{29}} & \frac{5}{\sqrt{29}} & 0 \\ -\frac{5}{\sqrt{29}} & -\frac{2}{\sqrt{29}} & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad G_3 G_1 A = \begin{pmatrix} \sqrt{29} & \frac{12}{\sqrt{29}} & \frac{11}{\sqrt{29}} \\ 0 & \frac{1}{\sqrt{29}} & -\frac{51}{5\sqrt{29}} \\ 0 & -1 & \frac{16}{5} \end{pmatrix}.$$

Step 3: We eliminate the (3,2)-entry of $G_3 G_1 A$ using its (2,2)-entry (and we leave the first row of $G_3 G_1 A$ unchanged, note we do not destroy our previously obtained zeros). To this end, we will use a Givens rotation $G_1(\theta)$ with θ such that $G(\theta)$ rotates $(\frac{1}{\sqrt{29}}, -1)^{\mathrm{T}}$ onto $(\sqrt{(\frac{1}{\sqrt{29}})^2 + (-1)^2}, 0)^{\mathrm{T}} = (\sqrt{\frac{30}{29}}, 0)^{\mathrm{T}}$. We know from (3.6) what to do: we take $\theta \in [0, 2\pi)$ such that $\cos(\theta) = \frac{1}{\sqrt{30}}$ and $\sin(\theta) = \sqrt{\frac{29}{30}}$. Then,

$$\tilde{G}_1 := \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{\sqrt{30}} & -\sqrt{\frac{29}{30}} \\ 0 & \sqrt{\frac{29}{30}} & \frac{1}{\sqrt{30}} \end{pmatrix}, \qquad \tilde{G}_1 G_3 G_1 A = \begin{pmatrix} \sqrt{29} & \frac{12}{\sqrt{29}} & \frac{11}{\sqrt{29}} \\ 0 & \sqrt{\frac{30}{29}} & -\frac{103}{\sqrt{870}} \\ 0 & 0 & -\frac{7}{\sqrt{30}} \end{pmatrix} =: R.$$

Noting that $G_1, G_3, \tilde{G}_1 \in \mathbb{R}^{3\times 3}$ are orthogonal, we have obtained the following QR factorization: $A = QR$ with

$$Q := G_1^{\mathrm{T}} G_3^{\mathrm{T}} \tilde{G}_1^{\mathrm{T}} = \begin{pmatrix} -\frac{2}{\sqrt{29}} & -\frac{\sqrt{5}}{\sqrt{174}} & -\frac{\sqrt{5}}{\sqrt{6}} \\ \frac{3}{\sqrt{29}} & \frac{11\sqrt{2}}{\sqrt{435}} & -\frac{\sqrt{2}}{\sqrt{15}} \\ \frac{4}{\sqrt{29}} & -\frac{19}{\sqrt{870}} & -\frac{1}{\sqrt{30}} \end{pmatrix}, \quad R := \begin{pmatrix} \sqrt{29} & \frac{12}{\sqrt{29}} & \frac{11}{\sqrt{29}} \\ 0 & \frac{\sqrt{30}}{\sqrt{29}} & -\frac{103}{\sqrt{870}} \\ 0 & 0 & -\frac{7}{\sqrt{30}} \end{pmatrix}.$$

This example concludes the short introduction to Givens rotations.

4 Linear Systems and Least Squares Problems

4.1 Gaussian elimination: LU factorization

In this section, we discuss the well-known Gaussian elimination – regarded as a matrix factorization algorithm – to solve linear systems

$$Ax = b, \qquad x \in \mathbb{R}^n$$

with given $A \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$. We are going to introduce the LU (lower-upper) factorization of a square matrix. Recalling the definition of upper-triangular matrices from Definition 3.1, we will also use the notion of lower-triangular square matrices:

Definition 4.1. A matrix $L \in \mathbb{R}^{n\times n}$ is called lower-triangular iff L^{T} is upper-triangular. Further, a matrix $L \in \mathbb{R}^{n\times n}$ is called unit lower-triangular iff L is lower-triangular and all of its diagonal entries are equal to 1.

The standard version of Gaussian elimination transforms the matrix A into an upper-triangular matrix

$$U = L_{n-1} \cdots L_2 L_1 A \in \mathbb{R}^{n\times n},$$

via left-multiplication by unit lower-triangular matrices $L_1, \dots, L_{n-1} \in \mathbb{R}^{n\times n}$ of the form

$$L_1 = \begin{pmatrix} 1 & & & & \\ * & 1 & & & \\ \vdots & & \ddots & & \\ \vdots & & & \ddots & \\ * & & & & 1 \end{pmatrix}, \; L_2 = \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & * & \ddots & & \\ & \vdots & & \ddots & \\ & * & & & 1 \end{pmatrix}, \cdots, \; L_{n-1} = \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & & * & 1 \end{pmatrix}$$

with zero-entries not shown. Assuming for the moment that the above is possible, this leads to a factorization $A = LU$ with $L := L_1^{-1} \cdots L_{n-1}^{-1} \in \mathbb{R}^{n\times n}$ lower-triangular (exercise) and $U \in \mathbb{R}^{n\times n}$ upper-triangular.

Definition 4.2. Let $n \in \mathbb{N}$ and $A \in \mathbb{R}^{n\times n}$. If there exist a lower-triangular matrix $L \in \mathbb{R}^{n\times n}$ and an upper-triangular matrix $U \in \mathbb{R}^{n\times n}$ such that there holds

$$A = LU, \tag{4.1}$$

then we call (4.1) a LU factorization of A.

Example 4.1 (Gaussian elimination). Consider the matrix

$$A = \begin{pmatrix} -2 & 2 & 1 & -1 \\ 1 & 1 & 2 & -2 \\ -1 & 4 & -1 & 1 \\ 1 & 3 & -3 & 4 \end{pmatrix} \in \mathbb{R}^{4\times 4}. \tag{4.2}$$

We illustrate Gaussian elimination.

L_1: The first step is to eliminate the sub-diagonal entries in the first column of A via adding $\frac{1}{2}/-\frac{1}{2}/\frac{1}{2}$ times row 1 to row 2/3/4:

$$L_1A = \begin{pmatrix} -2 & 2 & 1 & -1 \\ 0 & 2 & \frac{5}{2} & -\frac{5}{2} \\ 0 & 3 & -\frac{3}{2} & \frac{3}{2} \\ 0 & 4 & -\frac{5}{2} & \frac{7}{2} \end{pmatrix} \quad \text{with} \quad L_1 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ -\frac{1}{2} & 0 & 1 & 0 \\ \frac{1}{2} & 0 & 0 & 1 \end{pmatrix}.$$

L_2: The second step is to eliminate the sub-diagonal entries in the second column of L_1A via adding $-\frac{3}{2}/-2$ times row 2 to row 3/4:

$$L_2L_1A = \begin{pmatrix} -2 & 2 & 1 & -1 \\ 0 & 2 & \frac{5}{2} & -\frac{5}{2} \\ 0 & 0 & -\frac{21}{4} & \frac{21}{4} \\ 0 & 0 & -\frac{15}{2} & \frac{17}{2} \end{pmatrix} \quad \text{with} \quad L_2 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -\frac{3}{2} & 1 & 0 \\ 0 & -2 & 0 & 1 \end{pmatrix}.$$

L_3: The third step is to eliminate the sub-diagonal entries in the third column of L_2L_1A via adding $-\frac{10}{7}$ times row 3 to row 4:

$$L_3L_2L_1A = \begin{pmatrix} -2 & 2 & 1 & -1 \\ 0 & 2 & \frac{5}{2} & -\frac{5}{2} \\ 0 & 0 & -\frac{21}{4} & \frac{21}{4} \\ 0 & 0 & 0 & 1 \end{pmatrix} =: U \quad \text{with} \quad L_3 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{10}{7} & 1 \end{pmatrix}.$$

We find that $A = LU$ with U as above and L given by

$$L := L_1^{-1}L_2^{-1}L_3^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -\frac{1}{2} & 1 & 0 & 0 \\ \frac{1}{2} & 0 & 1 & 0 \\ -\frac{1}{2} & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & \frac{3}{2} & 1 & 0 \\ 0 & 2 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{10}{7} & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -\frac{1}{2} & 1 & 0 & 0 \\ \frac{1}{2} & \frac{3}{2} & 1 & 0 \\ -\frac{1}{2} & 2 & \frac{10}{7} & 1 \end{pmatrix}$$

is a LU factorization of A. Note how simple it is to compute L: the matrices L_i can be inverted by negating their sub-diagonal entries, and the matrix L can be obtained by collecting these values appropriately.

Generally, if the i-th column x_i of the matrix $L_{i-1}\cdots L_1A$ (the matrix A if $i = 1$) is the vector $x_i = (x_{1i}, \ldots, x_{ni})^{\mathrm{T}}$, then we eliminate the sub-diagonal entries in the i-th column of $L_{i-1}\cdots L_1A$ via adding $-\frac{x_{ji}}{x_{ii}}$ times row i to row j for $j = i+1, \ldots, n$:

$$L_i = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -\frac{x_{i+1,i}}{x_{ii}} & 1 & \\ & & \vdots & & \ddots & \\ & & -\frac{x_{ni}}{x_{ii}} & & & 1 \end{pmatrix} = I_n - l_i e_i^{\mathrm{T}} \in \mathbb{R}^{n\times n}, \qquad l_i := \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \frac{x_{i+1,i}}{x_{ii}} \\ \vdots \\ \frac{x_{ni}}{x_{ii}} \end{pmatrix} \in \mathbb{R}^n.$$

Now, as observed in Example 4.1, we have $L_i^{-1} = I_n + l_i e_i^{\mathrm{T}}$, i.e., the matrix L_i can be inverted by negating its sub-diagonal entries. Indeed, using that $\langle e_i, l_i\rangle = 0$, we find

$$(I_n - l_i e_i^{\mathrm{T}})(I_n + l_i e_i^{\mathrm{T}}) = I_n - l_i e_i^{\mathrm{T}} l_i e_i^{\mathrm{T}} = I_n \quad \Longrightarrow \quad (I_n - l_i e_i^{\mathrm{T}})^{-1} = I_n + l_i e_i^{\mathrm{T}}.$$

Further, the matrix L is given by

$$L = L_1^{-1} \cdots L_{n-1}^{-1} = \begin{pmatrix} 1 & & & & \\ \frac{x_{21}}{x_{11}} & 1 & & & \\ \frac{x_{31}}{x_{11}} & \frac{x_{32}}{x_{22}} & 1 & & \\ \vdots & \vdots & \ddots & \ddots & \\ \frac{x_{n1}}{x_{11}} & \frac{x_{n2}}{x_{22}} & \cdots & \frac{x_{n,n-1}}{x_{n-1,n-1}} & 1 \end{pmatrix}.$$

Indeed, looking at the product of two such matrices we find

$$L_i^{-1} L_{i+1}^{-1} = (I_n + l_i e_i^{\mathrm{T}})(I_n + l_{i+1} e_{i+1}^{\mathrm{T}}) = I_n + l_i e_i^{\mathrm{T}} + l_{i+1} e_{i+1}^{\mathrm{T}}$$

as $\langle e_i, l_{i+1} \rangle = 0$. Similarly one can compute $L = L_1^{-1} \cdots L_{n-1}^{-1}$ to obtain the above matrix.

In view of these observations, the Gauß-algorithm goes as follows:

Algorithm 4.1 (Gaussian elimination (without pivoting))**.** To obtain a LU factorization of a given matrix $A \in \mathbb{R}^{n \times n}$, do as follows:

```
$L = I_n$, $U = A$
for $i = 1, \dots, n-1$ do
  for $j = i+1, \dots, n$ do
    $l_{ji} = \frac{u_{ji}}{u_{ii}}$
    $u_{j,i:n} = u_{j,i:n} - l_{ji} u_{i,i:n}$
  end for
end for.
```

Warning: A needs to be such that no division by zero happens in the algorithm above.

Theorem 4.1. *Algorithm 4.1 requires $\sim \frac{2}{3} n^3$ flops.*

Proof. Exercise. □

Remark 4.1. Compare this with $\sim \frac{4}{3} n^3$ flops for computing a QR factorization of a $n \times n$ matrix via Householder (see Theorem 3.8). Gaussian elimination (with pivoting, see next section) is the usual method of choice to solve linear systems.

Remark 4.2 (Solving linear systems via LU factorization). For given $A \in \mathbb{R}^{n \times n}$ and $b \in \mathbb{R}^n$, consider the problem of finding $x \in \mathbb{R}^n$ such that $Ax = b$. Observe that, if there exists a LU factorization $A = LU$ with $L \in \mathbb{R}^{n \times n}$ lower-triangular and $U \in \mathbb{R}^{n \times n}$ upper-triangular, we have

$$Ax = b \quad \Longleftrightarrow \quad LUx = b \quad \Longleftrightarrow \quad \begin{cases} Ly = b, \\ Ux = y. \end{cases}$$

Therefore, once a LU factorization is computed ($\mathcal{O}(n^3)$ flops, see Theorem 4.1), we can first solve $Ly = b$ for y by forward substitution ($\mathcal{O}(n^2)$ flops) and then $Ux = y$ for x by backward substitution ($\mathcal{O}(n^2)$ flops).

Remark 4.3 (Not every matrix has a LU factorization). The matrix $A = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$ does not have a LU factorization. Indeed, if there were $L = \begin{pmatrix} l_{11} & 0 \\ l_{21} & l_{22} \end{pmatrix} \in \mathbb{R}^{2 \times 2}$ and $U = \begin{pmatrix} u_{11} & u_{12} \\ 0 & u_{22} \end{pmatrix} \in \mathbb{R}^{2 \times 2}$ such that $A = LU$, then there must hold $l_{11} u_{11} = 0$ and $l_{11} u_{12} = l_{21} u_{11} = l_{21} u_{12} + l_{22} u_{22} = 1$, which is not possible.

Algorithm 4.1 in its current form is impractical to solve general linear systems. For instance, it fails for the matrix from Remark 4.3 due to division by zero in the first step. More dramatically, the algorithm is not stable for general $n \times n$ matrices as we will see later in this course. Improvement in stability is obtained by pivoting, as we will explain in the following section.

4.2 Gaussian elimination with partial pivoting: PA=LU factorization

In the i-th step of Gaussian elimination, we add multiples of row i to rows $i+1, \dots, n$ to obtain

$$\left(\begin{array}{ccccccc}
x_{11} & x_{12} & \cdots & x_{1i} & x_{1,i+1} & \cdots & x_{1n} \\
 & x_{22} & \cdots & x_{2i} & x_{2,i+1} & \cdots & x_{2n} \\
 & & \ddots & \vdots & \vdots & & \vdots \\
\hline
 & & & x_{ii} & x_{i,i+1} & \cdots & x_{in} \\
 & & & x_{i+1,i} & x_{i+1,i+1} & \cdots & x_{i+1,n} \\
 & & & \vdots & \vdots & & \vdots \\
 & & & x_{ni} & x_{n,i+1} & \cdots & x_{nn}
\end{array}\right)
\Longrightarrow
\left(\begin{array}{ccccccc}
x_{11} & x_{12} & \cdots & x_{1i} & x_{1,i+1} & \cdots & x_{1n} \\
 & x_{22} & \cdots & x_{2i} & x_{2,i+1} & \cdots & x_{2n} \\
 & & \ddots & \vdots & \vdots & & \vdots \\
\hline
 & & & x_{ii} & x_{i,i+1} & \cdots & x_{in} \\
 & & & 0 & * & \cdots & * \\
 & & & \vdots & \vdots & & \vdots \\
 & & & 0 & * & \cdots & *
\end{array}\right)$$

and we call $x_{ii} \neq 0$ the pivot. Instead, we can also add multiples of row j with some $j \in \{i+1, \dots, n\}$ such that $x_{ji} \neq 0$ to rows $i, \dots, j-1, j+1, \dots, n$ to create zeros in the respective rows and column i:

$$\left(\begin{array}{ccccccc}
x_{11} & x_{12} & \cdots & x_{1i} & x_{1,i+1} & \cdots & x_{1n} \\
 & x_{22} & \cdots & x_{2i} & x_{2,i+1} & \cdots & x_{2n} \\
 & & \ddots & \vdots & \vdots & & \vdots \\
\hline
 & & & x_{ii} & x_{i,i+1} & \cdots & x_{in} \\
 & & & \vdots & \vdots & & \vdots \\
 & & & & & & \\
 & & & x_{ji} & x_{j,i+1} & \cdots & x_{jn} \\
 & & & \vdots & \vdots & & \vdots \\
 & & & & & & \\
 & & & x_{ni} & x_{n,i+1} & \cdots & x_{nn}
\end{array}\right)
\Longrightarrow
\left(\begin{array}{ccccccc}
x_{11} & x_{12} & \cdots & x_{1i} & x_{1,i+1} & \cdots & x_{1n} \\
 & x_{22} & \cdots & x_{2i} & x_{2,i+1} & \cdots & x_{2n} \\
 & & \ddots & \vdots & \vdots & & \vdots \\
\hline
 & & & 0 & * & \cdots & * \\
 & & & \vdots & \vdots & & \vdots \\
 & & & 0 & * & \cdots & * \\
 & & & x_{ji} & x_{j,i+1} & \cdots & x_{jn} \\
 & & & 0 & * & \cdots & * \\
 & & & \vdots & \vdots & & \vdots \\
 & & & 0 & * & \cdots & *
\end{array}\right).$$

In this case, $x_{ji} \neq 0$ is called the pivot. This procedure is thought of as follows: In the i-th step, choose a pivot $x_{ji} \neq 0$ from column i and row j (some $j \in \{i, \dots, n\}$), permute the rows of the matrix such that x_{ji} is moved to the main diagonal, and then do a standard Gaussian elimination step. For numerical stability, the pivot is chosen as the largest entry in modulus in column i and rows $i, \dots, n$. This is called Gaussian elimination with partial pivoting and leads to a LU factorization of PA for some permutation matrix P.

Definition 4.3. Let $n \in \mathbb{N}$ and $A \in \mathbb{R}^{n \times n}$. If there exist a lower-triangular matrix $L \in \mathbb{R}^{n \times n}$, an upper-triangular matrix $U \in \mathbb{R}^{n \times n}$, and a permutation matrix $P \in \mathbb{R}^{n \times n}$ (i.e., a matrix which has exactly one entry 1 in each row and column and zeros elsewhere) such that there holds

$$PA = LU, \tag{4.3}$$

then we call (4.3) a PA=LU factorization or a LU factorization with partial pivoting corresponding to A.

Remark 4.4. Observe that permutation matrices are orthogonal matrices.

We illustrate Gaussian elimination with partial pivoting at an example:

Example 4.2 (Gaussian elimination with partial pivoting). Consider the matrix $A \in \mathbb{R}^{4\times 4}$ defined in (4.2). We illustrate Gaussian elimination with partial pivoting.

P_1: As $\max\{|-2|,|1|,|-1|,|1|\} = |-2|$, we choose the $(1,1)$-entry -2 as pivot. Since this is already on the diagonal, no permutation is needed:

$$P_1 A = A \quad \text{with} \quad P_1 := I_4.$$

L_1: We eliminate the sub-diagonal entries in the first column of $P_1A = A$ via adding $\frac{1}{2}/-\frac{1}{2}/\frac{1}{2}$ times row 1 to row 2/3/4:

$$L_1P_1A = \begin{pmatrix} -2 & 2 & 1 & -1 \\ 0 & 2 & \frac{5}{2} & -\frac{5}{2} \\ 0 & 3 & -\frac{3}{2} & \frac{3}{2} \\ 0 & 4 & -\frac{5}{2} & \frac{7}{2} \end{pmatrix} \quad \text{with} \quad L_1 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ -\frac{1}{2} & 0 & 1 & 0 \\ \frac{1}{2} & 0 & 0 & 1 \end{pmatrix}.$$

P_2: As $\max\{|2|,|3|,|4|\} = |4|$, we choose the $(4,2)$-entry 4 as pivot. To this end, we permute rows 2 and 4:

$$P_2L_1P_1A = \begin{pmatrix} -2 & 2 & 1 & -1 \\ 0 & 4 & -\frac{5}{2} & \frac{7}{2} \\ 0 & 3 & -\frac{3}{2} & \frac{3}{2} \\ 0 & 2 & \frac{5}{2} & -\frac{5}{2} \end{pmatrix} \quad \text{with} \quad P_2 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

L_2: We eliminate the sub-diagonal entries in the second column of $P_2L_1P_1A$ via adding $-\frac{3}{4}/-\frac{1}{2}$ times row 2 to row 3/4:

$$L_2P_2L_1P_1A = \begin{pmatrix} -2 & 2 & 1 & -1 \\ 0 & 4 & -\frac{5}{2} & \frac{7}{2} \\ 0 & 0 & \frac{3}{8} & -\frac{9}{8} \\ 0 & 0 & \frac{15}{4} & -\frac{17}{4} \end{pmatrix} \quad \text{with} \quad L_2 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -\frac{3}{4} & 1 & 0 \\ 0 & -\frac{1}{2} & 0 & 1 \end{pmatrix}.$$

P_3: As $\max\{|\frac{3}{8}|,|\frac{15}{4}|\} = |\frac{15}{4}|$, we choose the $(4,3)$-entry $\frac{15}{4}$ as pivot. To this end, we permute rows 3 and 4:

$$P_3L_2P_2L_1P_1A = \begin{pmatrix} -2 & 2 & 1 & -1 \\ 0 & 4 & -\frac{5}{2} & \frac{7}{2} \\ 0 & 0 & \frac{15}{4} & -\frac{17}{4} \\ 0 & 0 & \frac{3}{8} & -\frac{9}{8} \end{pmatrix} \quad \text{with} \quad P_3 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

L_3: We eliminate the sub-diagonal entries in the third column of $P_3L_2P_2L_1P_1A$ via adding $-\frac{1}{10}$ times row 3 to row 4:

$$L_3P_3L_2P_2L_1P_1A = \begin{pmatrix} -2 & 2 & 1 & -1 \\ 0 & 4 & -\frac{5}{2} & \frac{7}{2} \\ 0 & 0 & \frac{15}{4} & -\frac{17}{4} \\ 0 & 0 & 0 & -\frac{7}{10} \end{pmatrix} =: U \quad \text{with} \quad L_3 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{10} & 1 \end{pmatrix}.$$

Now, setting

$$L_3' := L_3,\ L_2' := P_3 L_2 P_3^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -\frac{1}{2} & 1 & 0 \\ 0 & -\frac{3}{4} & 0 & 1 \end{pmatrix},\ L_1' := P_3 P_2 L_1 P_2^{-1} P_3^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ \frac{1}{2} & 0 & 1 & 0 \\ -\frac{1}{2} & 0 & 0 & 1 \end{pmatrix}$$

yields $L_3' L_2' L_1' P_3 P_2 P_1 A = L_3 P_3 L_2 P_2 L_1 P_1 A = U$. We find that $PA = LU$ with

$$P := P_3 P_2 P_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \qquad L := (L_3' L_2' L_1')^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -\frac{1}{2} & 1 & 0 & 0 \\ -\frac{1}{2} & \frac{1}{2} & 1 & 0 \\ \frac{1}{2} & \frac{3}{4} & \frac{1}{10} & 1 \end{pmatrix}$$

is a PA=LU factorization. Note that in contrast to the LU factorization of A from Example 4.1, all the sub-diagonal entries of L in the above PA=LU factorization are in the interval $[-1, 1]$. This is due to the choice of pivot as the largest entry in modulus among the candidates.

More generally, Gaussian elimination with partial pivoting transforms a matrix $A \in \mathbb{R}^{n\times n}$ into an upper-triangular matrix $U \in \mathbb{R}^{n\times n}$ by Gaussian elimination with an additional left-multiplication of a permutation matrix P_i at the beginning of step i:

$$L_{n-1} P_{n-1} \cdots L_2 P_2 L_1 P_1 A = U.$$

Here, $P_1, \ldots, P_{n-1} \in \mathbb{R}^{n\times n}$ are permutation matrices and $L_1, \ldots, L_{n-1} \in \mathbb{R}^{n\times n}$ are unit lower-triangular. We deduce that

$$(L_{n-1}' \cdots L_2' L_1')(P_{n-1} \cdots P_2 P_1) A = U$$

with $L_{n-1}' := L_{n-1}$ and $L_i' := P_{n-1} \cdots P_{i+1} L_i P_{i+1}^{-1} \cdots P_{n-1}^{-1}$ for $i \in \{1, \ldots, n-2\}$. Observe that the matrix L_i' has the same structure as L_i. We then obtain that $PA = LU$ is a PA=LU factorization corresponding to A with

$$L := (L_{n-1}' \cdots L_2' L_1')^{-1}, \qquad P := P_{n-1} \cdots P_2 P_1.$$

Note that P is a permutation matrix as a product of permutation matrices, and it is checked analogously to the previous section that L is well-defined and lower-triangular.

The Gauß-algorithm with partial pivoting goes as follows:

Algorithm 4.2 (Gaussian elimination with partial pivoting)**.** To obtain a PA=LU factorization of a given matrix $A \in \mathbb{R}^{n\times n}$, do as follows:

```
$P = I_n$, $L = I_n$, $U = A$
for $i = 1, \ldots, n-1$ do
  Choose $r \in \{i, \ldots, n\}$ such that $|u_{ri}| = \max_{k\in\{i,\ldots,n\}} |u_{ki}|$
  $u_{i,i:n} \leftrightarrow u_{r,i:n}$
  $l_{i,1:i-1} \leftrightarrow l_{r,1:i-1}$
  $p_{i,1:n} \leftrightarrow p_{r,1:n}$
  for $j = i+1, \ldots, n$ do
    $l_{ji} = \frac{u_{ji}}{u_{ii}}$
```

```
        $u_{j,i:n} = u_{j,i:n} - l_{ji} u_{i,i:n}$
      end for
    end for.
```

Here, "↔" denotes "interchange". Warning: A needs to be such that no division by zero happens in the algorithm above (as an exercise, think about how to obtain a PA=LU factorization if all candidates for pivots are zero at some step i).

Note that pivot selection requires $\mathcal{O}(n^2)$ operations overall. Hence, to leading order, Algorithm 4.2 requires the same amount of flops as Algorithm 4.1 (Gauß without pivoting), i.e., $\frac{2}{3}n^3$. Gaussian elimination with partial pivoting is the standard way to solve linear systems on a computer.

Remark 4.5 (Solving linear systems via PA=LU factorization). For given $A \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$, consider the problem of finding $x \in \mathbb{R}^n$ such that $Ax = b$. Observe that, if there exists a factorization $PA = LU$ with $L \in \mathbb{R}^{n\times n}$ lower-triangular, $U \in \mathbb{R}^{n\times n}$ upper-triangular, and $P \in \mathbb{R}^{n\times n}$ a permutation matrix, we have

$$Ax = b \quad \Longleftrightarrow \quad PAx = Pb \quad \Longleftrightarrow \quad LUx = Pb \quad \Longleftrightarrow \quad \begin{cases} Ly = Pb, \\ Ux = y. \end{cases}$$

Therefore, once a PA=LU factorization is computed ($\mathcal{O}(n^3)$ flops), we can first form $\tilde{b} := Pb$, then solve $Ly = \tilde{b}$ for y by forward substitution ($\mathcal{O}(n^2)$ flops) and then $Ux = y$ for x by backward substitution ($\mathcal{O}(n^2)$ flops).

Let us provide an existence result for the LU and PA=LU factorization without proof (see book "Matrix Analysis" by Horn and Johnson for proof).

Theorem 4.2 (Existence of LU and PA=LU factorization)**.** *The following assertions hold.*

(i) Any matrix $A \in \mathbb{R}^{n\times n}$ admits a PA=LU factorization.

(ii) Let $A \in \mathbb{R}^{n\times n}$ be invertible. Then, there exists a LU factorization of A iff there holds $\det(A_{1:i,1:i}) \neq 0$ for all $i \in \{1, \ldots, n\}$.

4.3 Gaussian elimination with full pivoting: PAQ=LU factorization

To improve numerical stability even further, one can use a strategy called full pivoting. Here, every entry of the sub-matrix $X_{i:n,i:n}$ of the working matrix X at step i is a candidate for the pivot. Let us remark that this is a procedure which is rarely used in practice due to its large computational cost. Gaussian elimination with full pivoting leads to a PAQ=LU factorization defined as follows.

Definition 4.4. Let $n \in \mathbb{N}$ and $A \in \mathbb{R}^{n\times n}$. If there exist a lower-triangular matrix $L \in \mathbb{R}^{n\times n}$, an upper-triangular matrix $U \in \mathbb{R}^{n\times n}$, and permutation matrices $P, Q \in \mathbb{R}^{n\times n}$ such that there holds

$$PAQ = LU, \tag{4.4}$$

then we call (4.4) a PAQ=LU factorization or a LU factorization with full pivoting corresponding to A.

Remark 4.6. In view of Theorem 4.2, any matrix $A \in \mathbb{R}^{n\times n}$ admits a PAQ=LU factorization with $Q = I_n$.

We illustrate Gaussian elimination with full pivoting at an example:

Example 4.3 (Gaussian elimination with full pivoting). Consider the matrix $A \in \mathbb{R}^{4\times 4}$ defined in (4.2). We illustrate Gaussian elimination with full pivoting.

P_1, Q_1: As $\max\{|-2|, |1|, |-1|, |1|, |2|, |1|, |4|, |3|, |1|, |2|, |-1|, |-3|, |-1|, |-2|, |1|, |4|\} = |4|$, we choose the $(3, 2)$-entry 4 as pivot (note we could have also chosen the $(4, 4)$-entry 4). To this end, we permute columns 1 and 2, and then rows 1 and 3:

$$P_1AQ_1 = \begin{pmatrix} 4 & -1 & -1 & 1 \\ 1 & 1 & 2 & -2 \\ 2 & -2 & 1 & -1 \\ 3 & 1 & -3 & 4 \end{pmatrix} \quad \text{with } Q_1 := \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \; P_1 := \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

L_1: We eliminate the sub-diagonal entries in the first column of P_1AQ_1 via adding $-\frac{1}{4}/-\frac{1}{2}/-\frac{3}{4}$ times row 1 to row 2/3/4:

$$L_1P_1AQ_1 = \begin{pmatrix} 4 & -1 & -1 & 1 \\ 0 & \frac{5}{4} & \frac{9}{4} & -\frac{9}{4} \\ 0 & -\frac{3}{2} & \frac{3}{2} & -\frac{3}{2} \\ 0 & \frac{7}{4} & -\frac{9}{4} & \frac{13}{4} \end{pmatrix} \quad \text{with } L_1 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ -\frac{1}{4} & 1 & 0 & 0 \\ -\frac{1}{2} & 0 & 1 & 0 \\ -\frac{3}{4} & 0 & 0 & 1 \end{pmatrix}.$$

P_2, Q_2: As $\max\{|\frac{5}{4}|, |-\frac{3}{2}|, |\frac{7}{4}|, |\frac{9}{4}|, |\frac{3}{2}|, |-\frac{9}{4}|, |-\frac{9}{4}|, |-\frac{3}{2}|, |\frac{13}{4}|\} = |\frac{13}{4}|$, we choose the $(4, 4)$-entry $\frac{13}{4}$ as pivot. To this end, we permute columns 2 and 4, and then rows 2 and 4:

$$P_2L_1P_1AQ_1Q_2 = \begin{pmatrix} 4 & 1 & -1 & -1 \\ 0 & \frac{13}{4} & -\frac{9}{4} & \frac{7}{4} \\ 0 & -\frac{3}{2} & \frac{3}{2} & -\frac{3}{2} \\ 0 & -\frac{9}{4} & \frac{9}{4} & \frac{5}{4} \end{pmatrix} \quad \text{with } Q_2 := P_2 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

L_2: We eliminate the sub-diagonal entries in the second column of $P_2L_1P_1AQ_1Q_2$ via adding $\frac{6}{13}/\frac{9}{13}$ times row 2 to row 3/4:

$$L_2P_2L_1P_1AQ_1Q_2 = \begin{pmatrix} 4 & 1 & -1 & -1 \\ 0 & \frac{13}{4} & -\frac{9}{4} & \frac{7}{4} \\ 0 & 0 & \frac{6}{13} & -\frac{9}{13} \\ 0 & 0 & \frac{9}{13} & \frac{32}{13} \end{pmatrix} \quad \text{with } L_2 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & \frac{6}{13} & 1 & 0 \\ 0 & \frac{9}{13} & 0 & 1 \end{pmatrix}.$$

P_3, Q_3: As $\max\{|\frac{6}{13}|, |\frac{9}{13}|, |-\frac{9}{13}|, |\frac{32}{13}|\} = |\frac{32}{13}|$, we choose the $(4, 4)$-entry $\frac{32}{13}$ as pivot. To this end, we permute columns 3 and 4, and then rows 3 and 4:

$$P_3L_2P_2L_1P_1AQ_1Q_2Q_3 = \begin{pmatrix} 4 & 1 & -1 & -1 \\ 0 & \frac{13}{4} & \frac{7}{4} & -\frac{9}{4} \\ 0 & 0 & \frac{32}{13} & \frac{9}{13} \\ 0 & 0 & -\frac{9}{13} & \frac{6}{13} \end{pmatrix} \quad \text{with } Q_3 := P_3 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

L_3: We eliminate the sub-diagonal entries in the third column of $P_3L_2P_2L_1P_1AQ_1Q_2Q_3$ via adding $\frac{9}{32}$ times row 3 to row 4:

$$L_3P_3L_2P_2L_1P_1AQ_1Q_2Q_3 = \begin{pmatrix} 4 & 1 & -1 & -1 \\ 0 & \frac{13}{4} & \frac{7}{4} & -\frac{9}{4} \\ 0 & 0 & \frac{32}{13} & \frac{9}{13} \\ 0 & 0 & 0 & \frac{21}{32} \end{pmatrix} =: U \text{ with } L_3 := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{9}{32} & 1 \end{pmatrix}.$$

Now, setting

$$L_3' := L_3,\ L_2' := P_3L_2P_3^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & \frac{9}{13} & 1 & 0 \\ 0 & \frac{6}{13} & 0 & 1 \end{pmatrix},\ L_1' := P_3P_2L_1P_2^{-1}P_3^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -\frac{3}{4} & 1 & 0 & 0 \\ -\frac{1}{4} & 0 & 1 & 0 \\ -\frac{1}{2} & 0 & 0 & 1 \end{pmatrix}$$

yields $L_3'L_2'L_1'P_3P_2P_1AQ_1Q_2Q_3 = L_3P_3L_2P_2L_1P_1AQ_1Q_2Q_3 = U$. We find that $PAQ = LU$ with

$$P := P_3P_2P_1 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \quad Q := Q_1Q_2Q_3 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix},$$

and

$$L := (L_3'L_2'L_1')^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{3}{4} & 1 & 0 & 0 \\ \frac{1}{4} & -\frac{9}{13} & 1 & 0 \\ \frac{1}{2} & -\frac{6}{13} & -\frac{9}{32} & 1 \end{pmatrix}, \quad U = \begin{pmatrix} 4 & 1 & -1 & -1 \\ 0 & \frac{13}{4} & \frac{7}{4} & -\frac{9}{4} \\ 0 & 0 & \frac{32}{13} & \frac{9}{13} \\ 0 & 0 & 0 & \frac{21}{32} \end{pmatrix}$$

is a PAQ $=$ LU factorization.

More generally, Gaussian elimination with full pivoting transforms a matrix $A \in \mathbb{R}^{n\times n}$ into an upper-triangular matrix $U \in \mathbb{R}^{n\times n}$ by Gaussian elimination with an additional right-multiplication of a permutation matrix Q_i and left-multiplication of a permutation matrix P_i at the beginning of step i:

$$L_{n-1}P_{n-1}\cdots L_2P_2L_1P_1AQ_1Q_2\cdots Q_{n-1} = U.$$

Here, $P_1, \dots, P_{n-1}, Q_1, \dots, Q_{n-1} \in \mathbb{R}^{n\times n}$ are permutation matrices and $L_1, \dots, L_{n-1} \in \mathbb{R}^{n\times n}$ are unit lower-triangular. We deduce that

$$(L_{n-1}'\cdots L_2'L_1')(P_{n-1}\cdots P_2P_1)A(Q_1Q_2\cdots Q_{n-1}) = U$$

with $L_{n-1}' := L_{n-1}$ and $L_i' := P_{n-1}\cdots P_{i+1}L_iP_{i+1}^{-1}\cdots P_{n-1}^{-1}$ for $i \in \{1, \dots, n-2\}$ as in the previous section. We then obtain that $PAQ = LU$ is a PAQ=LU factorization corresponding to A with

$$L := (L_{n-1}'\cdots L_2'L_1')^{-1}, \qquad P := P_{n-1}\cdots P_2P_1, \qquad Q := Q_1Q_2\cdots Q_{n-1}.$$

Note that P and Q are permutation matrices as products of permutation matrices, and that L is well-defined and lower-triangular.

Full pivoting gives a further improvement in numerical stability over partial pivoting. However, the pivot selection for full pivoting requires $\mathcal{O}(n^3)$ operations overall, which is why full pivoting is rarely used in practice. As an exercise, think about how a PAQ=LU factorization can be used to solve a linear system $Ax = b$.

4.4 Symmetric Gaussian elimination: Cholesky factorization

Let us turn our focus to symmetric positive definite matrices. We start by recalling the definition of symmetric positive/negative definite and symmetric positive/negative semidefinite matrices.

Definition 4.5. A symmetric matrix $A \in \mathbb{R}^{n\times n}$ is called

(i) positive definite, denoted $A \succ 0$, iff $\langle x, Ax\rangle > 0$ for all $x \in \mathbb{R}^n \backslash \{0\}$.

(ii) positive semidefinite, denoted $A \succeq 0$, iff $\langle x, Ax\rangle \geq 0$ for all $x \in \mathbb{R}^n$.

(iii) negative definite, denoted $A \prec 0$, iff $\langle x, Ax\rangle < 0$ for all $x \in \mathbb{R}^n \backslash \{0\}$.

(iv) negative semidefinite, denoted $A \preceq 0$, iff $\langle x, Ax\rangle \leq 0$ for all $x \in \mathbb{R}^n$.

Let us recall the spectral theorem for symmetric matrices:

Theorem 4.3 (Spectral theorem for symmetric matrices)**.** *Symmetric matrices are orthogonally diagonalizable, i.e., for any symmetric matrix $A \in \mathbb{R}^{n\times n}$ there exist an orthogonal matrix $Q \in \mathbb{R}^{n\times n}$ and a diagonal matrix $D \in \mathbb{R}^{n\times n}$ such that $A = QDQ^{\mathrm{T}}$. The diagonal entries of D are the eigenvalues of A, and the column vectors of Q are eigenvectors of A. In particular, all eigenvalues of a symmetric matrix are real.*

Proof. See previous linear algebra courses. □

Theorem 4.4. *For a symmetric matrix $A \in \mathbb{R}^{n\times n}$, we have*

(i) $A \succ 0 \iff$ all eigenvalues of A are positive,

(ii) $A \succeq 0 \iff$ all eigenvalues of A are non-negative,

(iii) $A \prec 0 \iff$ all eigenvalues of A are negative,

(iv) $A \preceq 0 \iff$ all eigenvalues of A are non-positive.

Proof. Exercise (use Theorem 4.3). □

Remark 4.7. Let $A \in \mathbb{R}^{n\times n}$ be a symmetric positive definite matrix and let $X \in \mathbb{R}^{n\times r}$ with $n \geq r$ and $\mathrm{rk}(X) = r$. Then, the matrix $X^{\mathrm{T}}AX$ is symmetric positive definite (exercise).

Without proof, let us state a useful criterion for checking positive definiteness.

Theorem 4.5 (Sylvester's criterion for positive definiteness)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix. Then,*

$$A \succ 0 \quad \iff \quad \forall i \in \{1, \dots, n\} : \det(A_{1:i,1:i}) > 0.$$

The number $\det(A_{1:i,1:i})$ is called the i-th leading principal minor of A. Therefore, a symmetric matrix is positive definite iff all of its leading principal minors are positive.

Remark 4.8. In view of Theorem 4.2, we have that any symmetric positive definite matrix admits a LU factorization.

It will turn out, that we can factorize a symmetric positive definite matrix twice as quickly into triangular factors as a general matrix. This is due to the fact that we can use symmetric Gaussian elimination which we describe in the following. This will yield a so-called Cholesky factorization.

Definition 4.6. Let $A \in \mathbb{R}^{n\times n}$ be a symmetric positive definite matrix. If there exists an upper-triangular matrix $R \in \mathbb{R}^{n\times n}$ with positive diagonal entries such that there holds

$$A = R^{\mathrm{T}} R, \tag{4.5}$$

then we call (4.5) a Cholesky factorization of A.

Let us consider a symmetric positive definite matrix $A \in \mathbb{R}^{n\times n}$. Then, we can write A as block-matrix

$$A = \left(\begin{array}{c|c} a_{11} & w^{\mathrm{T}} \\ \hline w & B \end{array}\right) \in \mathbb{R}^{n\times n}$$

with $a_{11} \in \mathbb{R}$, $w \in \mathbb{R}^{n-1}$ and a symmetric matrix $B \in \mathbb{R}^{(n-1)\times(n-1)}$. Note that, since $A \succ 0$, we have that $a_{11} = \det(A_{1:1,1:1}) > 0$ and $B \succ 0$. (The latter follows from the fact that $\langle x, Bx\rangle = \left\langle \begin{pmatrix} 0 \\ x \end{pmatrix}, A \begin{pmatrix} 0 \\ x \end{pmatrix} \right\rangle$ for $x \in \mathbb{R}^{n-1}$ and positive definiteness of A.) The first step of symmetric Gaussian elimination (compare this with classical Gauß) goes as follows:

$$L_1 A L_1^{\mathrm{T}} = \left(\begin{array}{c|c} 1 & 0_{1\times(n-1)} \\ \hline 0_{(n-1)\times 1} & B - \frac{ww^{\mathrm{T}}}{a_{11}} \end{array}\right) =: A_1 \quad \text{with} \quad L_1 := \left(\begin{array}{c|c} \frac{1}{\sqrt{a_{11}}} & 0_{1\times(n-1)} \\ \hline -\frac{w}{a_{11}} & I_{n-1} \end{array}\right),$$

which we can equivalently write as

$$A = R_1^{\mathrm{T}} A_1 R_1 \quad \text{with} \quad R_1 := (L_1^{-1})^{\mathrm{T}} = \left(\begin{array}{c|c} \sqrt{a_{11}} & \frac{w^{\mathrm{T}}}{\sqrt{a_{11}}} \\ \hline 0_{(n-1)\times 1} & I_{n-1} \end{array}\right).$$

Note that A_1 is again symmetric positive definite. Indeed, it is quickly checked that $A_1 = A_1^{\mathrm{T}}$, and that $A_1 = (L_1^{\mathrm{T}})^{\mathrm{T}} A L_1^{\mathrm{T}} \succ 0$ by Remark 4.7 since $L_1^{\mathrm{T}} \in \mathbb{R}^{n\times n}$ is of full rank. Therefore, we also have that the sub-matrix $B - \frac{ww^{\mathrm{T}}}{a_{11}} \in \mathbb{R}^{(n-1)\times(n-1)}$ is symmetric positive definite (same argument as when we deduced $B \succ 0$ from $A \succ 0$) and in particular, the $(1,1)$-entry of $B - \frac{ww^{\mathrm{T}}}{a_{11}}$ is positive. We deduce that we can factor

$$A_1 = R_2^{\mathrm{T}} A_2 R_2$$

with $R_2 \in \mathbb{R}^{n\times n}$ upper-triangular with positive diagonal entries and A_2 being of the form $A_2 = \left(\begin{array}{c|c} I_2 & 0_{2\times(n-2)} \\ \hline 0_{(n-2)\times 2} & C \end{array}\right)$, using the same procedure as before applied to $B - \frac{ww^{\mathrm{T}}}{a_{11}}$. Then, again, the sub-matrix C is symmetric positive definite, and we can continue this process until we arrive at a factorization

$$A = (R_1^{\mathrm{T}} R_2^{\mathrm{T}} \cdots R_n^{\mathrm{T}}) I_n (R_n \cdots R_2 R_1) = R^{\mathrm{T}} R$$

with $R := R_n \cdots R_2 R_1 \in \mathbb{R}^{n\times n}$ upper-triangular and having positive diagonal entries. This is a Cholesky factorization of A.

Theorem 4.6 (Existence and uniqueness of Cholesky factorization)**.** *Every symmetric positive definite matrix $A \in \mathbb{R}^{n\times n}$ admits a unique Cholesky factorization.*

Proof. Symmetric Gaussian elimination as discussed above provides existence of a Cholesky factorization (argument can be made rigorous via induction). For uniqueness, suppose that $R, M \in \mathbb{R}^{n\times n}$ are two upper-triangular matrices with positive diagonal entries such that there holds $A = R^{\mathrm{T}}R = M^{\mathrm{T}}M$. Note that $D := MR^{-1}$ is an upper-triangular matrix, but also, since

$$D = MR^{-1} = (M^{\mathrm{T}})^{-1}R^{\mathrm{T}} = (D^{-1})^{\mathrm{T}},$$

it must be lower-triangular as well, hence diagonal. Noting that $I_n = D^{\mathrm{T}}D = D^2$, the diagonal entries of D are all ± 1. Finally, since $DR = M$ and the diagonal entries of R and M are positive, we must have that $R = M$. □

Example 4.4 (Symmetric Gaussian elimination). We consider the symmetric positive definite matrix

$$A := \begin{pmatrix} 16 & -8 & 12 \\ -8 & 5 & -9 \\ 12 & -9 & 22 \end{pmatrix} \in \mathbb{R}^{3\times 3}.$$

We illustrate symmetric Gaussian elimination for finding the unique Cholesky factorization of A.

L_1: We eliminate the sub-diagonal entries in the first column of A by adding $\frac{1}{2}$/-$\frac{3}{4}$ times row 1 to row 2/3, and multiply the first row by $\frac{1}{\sqrt{a_{11}}} = \frac{1}{4}$:

$$L_1A = \begin{pmatrix} 4 & -2 & 3 \\ 0 & 1 & -3 \\ 0 & -3 & 13 \end{pmatrix} \quad \text{with} \quad L_1 := \begin{pmatrix} \frac{1}{4} & 0 & 0 \\ \frac{1}{2} & 1 & 0 \\ -\frac{3}{4} & 0 & 1 \end{pmatrix}.$$

Next, we right-multiply L_1A with L_1^{T} which creates a 1 in the $(1,1)$ entry and zeros in the $(1,2)$ and $(1,3)$ entries:

$$L_1AL_1^{\mathrm{T}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -3 \\ 0 & -3 & 13 \end{pmatrix}.$$

L_2: We eliminate the sub-diagonal entry in the second column of $L_1AL_1^{\mathrm{T}}$ by adding 3 times row 2 to row 3 (and multiply the second row by $\frac{1}{\sqrt{1}} = 1$):

$$L_2L_1AL_1^{\mathrm{T}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -3 \\ 0 & 0 & 4 \end{pmatrix} \quad \text{with} \quad L_2 := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{pmatrix}.$$

Next, we right-multiply $L_2L_1AL_1^{\mathrm{T}}$ with L_2^{T} which creates a zero in the $(2,3)$ entry:

$$L_2L_1AL_1^{\mathrm{T}}L_2^{\mathrm{T}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 4 \end{pmatrix}.$$

L_3: We multiply the third row of $L_2L_1AL_1^{\mathrm{T}}L_2^{\mathrm{T}}$ by $\frac{1}{\sqrt{4}} = \frac{1}{2}$:

$$L_3L_2L_1AL_1^{\mathrm{T}}L_2^{\mathrm{T}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix} \quad \text{with} \quad L_3 := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{2} \end{pmatrix}.$$

Finally, we right-multiply $L_3L_2L_1AL_1^{\mathrm{T}}L_2^{\mathrm{T}}$ by L_3^{T} which creates a 1 in the $(3,3)$ entry:

$$L_3L_2L_1AL_1^{\mathrm{T}}L_2^{\mathrm{T}}L_3^{\mathrm{T}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = I_3.$$

We find that $A = R^{\mathrm{T}}R$ with

$$R := [L_1^{-1}L_2^{-1}L_3^{-1}]^{\mathrm{T}} = \left[\begin{pmatrix} 4 & 0 & 0 \\ -2 & 1 & 0 \\ 3 & 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -3 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}\right]^{\mathrm{T}} = \begin{pmatrix} 4 & -2 & 3 \\ 0 & 1 & -3 \\ 0 & 0 & 2 \end{pmatrix}$$

is the (unique) Cholesky factorization of A.

An efficient algorithm to obtain the Cholesky factorization to a given symmetric positive definite matrix is given below.

Algorithm 4.3 (Cholesky factorization)**.** To obtain the Cholesky factorization $A = R^{\mathrm{T}}R$ of a given symmetric positive definite matrix $A \in \mathbb{R}^{n \times n}$, do as follows:

```
$R = A$
for $i = 1, \ldots, n$ do
    for $j = i+1, \ldots, n$ do
        $R_{j,j:n} = R_{j,j:n} - \frac{R_{i,j:n}R_{ij}}{R_{ii}}$
    end for
    $R_{i,i:n} = \frac{R_{i,i:n}}{\sqrt{R_{ii}}}$
end for.
```

Theorem 4.7. *Algorithm 4.3 requires $\sim \frac{1}{3}n^3$ flops.*

Proof. Exercise. □

Remark 4.9. This is only half the cost of Gaussian elimination.

Remark 4.10 (Solving linear systems via Cholesky factorization)*.* For a given symmetric positive definite matrix $A \in \mathbb{R}^{n \times n}$ and a vector $b \in \mathbb{R}^n$, consider the problem of finding $x \in \mathbb{R}^n$ such that $Ax = b$. The standard way to solve the system in this case is by Cholesky factorization: If $A = R^{\mathrm{T}}R$ is the Cholesky factorization of A, we have

$$Ax = b \quad \Longleftrightarrow \quad R^{\mathrm{T}}Rx = b \quad \Longleftrightarrow \quad \begin{cases} R^{\mathrm{T}}y = b, \\ Rx = y. \end{cases}$$

Therefore, once the Cholesky factorization is computed ($\mathcal{O}(n^3)$ flops), we can first solve $R^{\mathrm{T}}y = b$ for y by forward substitution ($\mathcal{O}(n^2)$ flops) and then $Rx = y$ for x by backward substitution ($\mathcal{O}(n^2)$ flops).

4.5 Least squares problems

Let us consider an over-determined linear system (more equations than unknowns): Given a matrix $A = (a_{ij}) \in \mathbb{R}^{m\times n}$ with $m > n$ and a vector $b = (b_1, \dots, b_m)^{\mathrm{T}} \in \mathbb{R}^m$, find a vector $x = (x_1, \dots, x_n)^{\mathrm{T}} \in \mathbb{R}^n$ such that

$$Ax = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nn} \\ a_{n+1,1} & \cdots & a_{n+1,n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ \vdots \\ b_n \\ b_{n+1} \\ \vdots \\ b_m \end{pmatrix} = b. \tag{4.6}$$

Clearly, such a problem does not admit a solution in general.

Remark 4.11. Let $m, n \in \mathbb{N}$ with $m > n$. Then, given $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$, there exists a solution $x \in \mathbb{R}^n$ to $Ax = b$ iff $b \in \mathcal{R}(A)$. Noting that $\dim(\mathcal{R}(A)) \leq n < m = \dim(\mathbb{R}^m)$, such an over-determined system $Ax = b$ is, in general, only solvable for special choices of $b \in \mathbb{R}^m$.

As we cannot expect a general over-determined system (4.6) to admit a solution, we pose the following problem instead: Find $x \in \mathbb{R}^n$ such that the *residual* $r := Ax - b$ is as small as possible. To measure the size of r, we use the Euclidean norm.

Definition 4.7. Given $A \in \mathbb{R}^{m\times n}$, $m \geq n$, and $b \in \mathbb{R}^m$, we call the following problem the least squares problem corresponding to the matrix A and the vector b:

$$\text{Minimize } \|Av - b\|_2 \text{ over } v \in \mathbb{R}^n. \tag{4.7}$$

If there exists a minimizer, i.e., a vector $x \in \mathbb{R}^n$ such that

$$\|Ax - b\|_2 = \inf_{v\in\mathbb{R}^n} \|Av - b\|_2,$$

then we call this minimizer x a solution to the least squares problem.

Before we discuss existence and uniqueness of solutions to least squares problems, we provide some more motivation.

Example 4.5 (Polynomial interpolation vs. least squares fitting). Suppose we are given data points $(t_1, y_1), \dots, (t_n, y_n)$ with $t_1, \dots, t_n \in \mathbb{R}$ distinct and $y_1, \dots, y_n \in \mathbb{R}$.

(i) Polynomial interpolation: There exists a unique polynomial $p(t) = \sum_{k=0}^{n-1} p_k t^k$ of degree $n - 1$ such that $p(t_i) = y_i$ for all $i \in \{1, \dots, n\}$. This polynomial p is called the polynomial interpolant corresponding to the given data points. The coefficients $p_0, \dots, p_{n-1} \in \mathbb{R}$ of the polynomial interpolant are uniquely determined from the linear system

$$[V(t_1, \dots, t_n)] \begin{pmatrix} p_0 \\ \vdots \\ p_{n-1} \end{pmatrix} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \qquad V(t_1, \dots, t_n) = \begin{pmatrix} 1 & t_1 & t_1^2 & \cdots & t_1^{n-1} \\ 1 & t_2 & t_2^2 & \cdots & t_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & t_n & t_n^2 & \cdots & t_n^{n-1} \end{pmatrix}.$$

Note that the so-called *Vandermonde matrix* $V(t_1, \dots, t_n)$ is invertible since the values $\{t_i\}$ are distinct (exercise). A typical behavior of polynomial interpolation is the appearance of large oscillations near the ends of the interval $[t_1, t_n]$.

(ii) Polynomial least squares fitting: Let us now try to fit the data points by a lower-degree polynomial $p(t) = \sum_{k=0}^{N-1} p_k t^k$ with $N < n$. The condition $p(t_i) = y_i$ for $i \in \{1, \ldots, n\}$ leads to the over-determined system

$$Ap_{\text{coeff}} = b \quad \textit{with} \quad A := \begin{pmatrix} 1 & t_1 & t_1^2 & \cdots & t_1^{N-1} \\ 1 & t_2 & t_2^2 & \cdots & t_2^{N-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & t_n & t_n^2 & \cdots & t_n^{N-1} \end{pmatrix}, \; p_{\text{coeff}} := \begin{pmatrix} p_0 \\ \vdots \\ p_{N-1} \end{pmatrix}, \; b := \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}$$

which may not have a solution. Instead, we choose the coefficient vector $p_{\text{coeff}} = (p_0, \ldots, p_{N-1})^{\mathrm{T}} \in \mathbb{R}^N$ such that it solves the corresponding least squares problem

$$\|Ap_{\text{coeff}} - b\|_2 = \inf_{v \in \mathbb{R}^N} \|Av - b\|_2$$

(assume for the moment that such a minimizer exists). Observe that the corresponding least squares fit $p(t) = \sum_{k=0}^{N-1} p_k t^k$ minimizes the quantity $\sqrt{\sum_{i=1}^{n} |p(t_i) - y_i|^2}$ among polynomials of degree at most $N - 1$.

We are going to discuss later how to obtain such a solution. The least squares solution does not interpolate the given data points, but it often describes the overall behavior better than the interpolant (do experiments with MATLAB as an exercise).

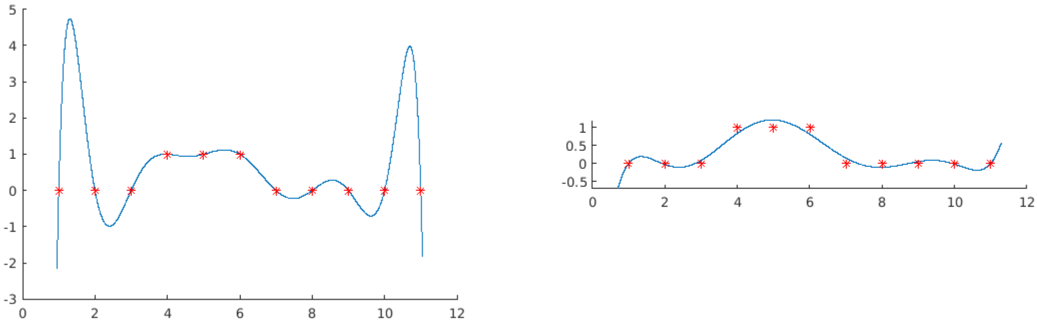

Figure 5: Polynomial interpolant of degree 10 (left) and least squares fit of degree 7 (right) to the data points $(1, 0), (2, 0), (3, 0), (4, 1), (5, 1), (6, 1), (7, 0), (8, 0), (9, 0), (10, 0), (11, 0)$.

Existence and uniqueness

Let us turn to the question of existence of solutions to least squares problems. First of all, let us introduce the minimization problem

$$\text{Minimize } \|w - b\|_2 \text{ over } w \in \mathcal{R}(A). \tag{4.8}$$

If there exists a minimizer $y \in \mathcal{R}(A)$ such that $\|y - b\|_2 = \inf_{w \in \mathcal{R}(A)} \|w - b\|_2$, then we call this minimizer y a solution to (4.8).

Remark 4.12. We observe the following relation between the least squares problem (4.7) and the minimization problem (4.8).

(i) If there exists a solution $x \in \mathbb{R}^n$ to the least squares problem (4.7), then $y = Ax \in \mathcal{R}(A)$ is a solution to the minimization problem (4.8).

(ii) If there exists a solution $y \in \mathcal{R}(A)$ to the minimization problem (4.8), then any $x \in \mathbb{R}^n$ satisfying $Ax = y$ is a solution to the least squares problem (4.7).

(iii) There holds $\inf_{v\in\mathbb{R}^n} \|Av - b\|_2 = \inf_{w\in\mathcal{R}(A)} \|w - b\|_2$.

Geometrically, a solution $y \in \mathcal{R}(A)$ to the minimization problem (4.8) is the closest point in $\mathcal{R}(A)$ to b (with distance measured in the Euclidean distance). We expect that the solution to this problem should be given by $y = Pb$ where $P \in \mathbb{R}^{m\times m}$ is the orthogonal projector onto $\mathcal{R}(A)$. Let us now make this idea rigorous and start with the following central result.

Theorem 4.8 (Existence of solutions to the normal equation)**.** *Let $A \in \mathbb{R}^{m\times n}$. Then, for any $b \in \mathbb{R}^m$ there exists a vector $x \in \mathbb{R}^n$ satisfying the equation*

$$A^{\mathrm{T}}Ax = A^{\mathrm{T}}b. \tag{4.9}$$

We call an equation of the form (4.9) *normal equation.*

Proof. We need to show that $A^{\mathrm{T}}b \in \mathcal{R}(A^{\mathrm{T}}A)$ for any $b \in \mathbb{R}^m$. We are going to show that $\mathcal{R}(A^{\mathrm{T}}) = \mathcal{R}(A^{\mathrm{T}}A)$. This can be shown as follows:

$$\mathcal{R}(A^{\mathrm{T}}) = [\mathcal{N}(A)]^{\perp} = [\mathcal{N}(A^{\mathrm{T}}A)]^{\perp} = \mathcal{R}((A^{\mathrm{T}}A)^{\mathrm{T}}) = \mathcal{R}(A^{\mathrm{T}}A), \tag{4.10}$$

where we have used that $\mathcal{N}(A) = \mathcal{N}(A^{\mathrm{T}}A)$ and the fact that $[\mathcal{N}(M)]^{\perp} = \mathcal{R}(M^{\mathrm{T}})$ for any matrix M (exercise). □

The main tool is the orthogonal projector onto the range of a given matrix.

Theorem 4.9 (Orthogonal projector onto range of matrix)**.** *Let $A \in \mathbb{R}^{m\times n}$. Then, we have the following assertions.*

(i) $\mathcal{R}(A)$ and $\mathcal{N}(A^{\mathrm{T}})$ are complementary subspaces of $\mathbb{R}^m$,

(ii) $\mathcal{R}(A) \perp \mathcal{N}(A^{\mathrm{T}})$.

In particular, there exists a unique projector $P \in \mathbb{R}^{m\times m}$ such that $\mathcal{R}(P) = \mathcal{R}(A)$ and $\mathcal{N}(P) = \mathcal{N}(A^{\mathrm{T}})$ (the projector onto $\mathcal{R}(A)$ along $\mathcal{N}(A^{\mathrm{T}})$), and this projector is the unique orthogonal projector onto $\mathcal{R}(A)$.

Proof. Exercise. □

We can now prove the main result.

Theorem 4.10 (Existence and uniqueness result for least squares problems)**.** *Let $A \in \mathbb{R}^{m\times n}$, $m \geq n$, and $b \in \mathbb{R}^m$. Let $P \in \mathbb{R}^{m\times m}$ be the orthogonal projector onto $\mathcal{R}(A)$ given by Theorem 4.9. Then, we have the following results.*

(i) There exists a unique solution to the minimization problem (4.8)*, i.e., a unique vector $y \in \mathcal{R}(A)$ satisfying $\|y - b\|_2 = \inf_{w\in\mathcal{R}(A)} \|w - b\|_2$. This solution is given by*

$$y = Pb.$$

(ii) There exists a solution to the least squares problem (4.7)*, i.e., a vector $x \in \mathbb{R}^n$ satisfying $\|Ax - b\|_2 = \inf_{v\in\mathbb{R}^n} \|Av - b\|_2$. Moreover, $x \in \mathbb{R}^n$ is a solution to* (4.7) *iff*

$$Ax = Pb, \text{ or equivalently, } A^{\mathrm{T}}Ax = A^{\mathrm{T}}b.$$

(iii) The least squares problem (4.7) *has a unique solution iff A is of full rank.*

Proof. (i) We have $Pb \in \mathcal{R}(P) = \mathcal{R}(A)$ and

$$\begin{aligned}\|w - b\|_2 &= \sqrt{\|(w - Pb) + (Pb - b)\|_2^2} \\ &= \sqrt{\|w - Pb\|_2^2 + \|Pb - b\|_2^2} > \|Pb - b\|_2 \qquad \forall w \in \mathcal{R}(A)\backslash\{Pb\},\end{aligned}$$

where we have used that $\langle w - Pb, Pb - b\rangle = 0$ for all $w \in \mathcal{R}(A)$ (note that $w - Pb \in \mathcal{R}(P)$ for $w \in \mathcal{R}(A) = \mathcal{R}(P)$, that $Pb - b \in \mathcal{N}(P)$, and $\mathcal{R}(P) \perp \mathcal{N}(P)$). It follows that $y = Pb$ is the unique element in $\mathcal{R}(A)$ satisfying $\|y - b\|_2 = \inf_{w\in\mathcal{R}(A)} \|w - b\|_2$.

(ii) By (i) and in view of Remark 4.12(ii), any $x \in \mathbb{R}^n$ satisfying $Ax = Pb$ is a solution to (4.7). Conversely, in view of Remark 4.12(i), if $x \in \mathbb{R}^n$ is a solution to (4.7), then Ax is a solution to (4.8) and consequently, using (i), we must have $Ax = Pb$. It remains to show that for $x \in \mathbb{R}^n$ there holds $Ax = Pb \iff A^{\mathrm{T}}Ax = A^{\mathrm{T}}b$. If $x \in \mathbb{R}^n$ is such that there holds $Ax = Pb$, then $Ax - b = Pb - b \in \mathcal{N}(P) = \mathcal{N}(A^{\mathrm{T}})$, i.e., $A^{\mathrm{T}}Ax = A^{\mathrm{T}}b$. Conversely, if $x \in \mathbb{R}^n$ is such that there holds $A^{\mathrm{T}}Ax = A^{\mathrm{T}}b$, then $Ax - b \in \mathcal{N}(A^{\mathrm{T}}) = \mathcal{N}(P)$ and hence, $Ax - Pb = (I_m - P)Ax + P(Ax - b) = 0$, where we have used that $Ax \in \mathcal{R}(A) = \mathcal{R}(P) = \mathcal{N}(I_m - P)$.

(iii) In view of (ii), the least squares problem has a unique solution iff the matrix $A^{\mathrm{T}}A \in \mathbb{R}^{n\times n}$ is invertible, i.e., iff $\mathrm{rk}(A^{\mathrm{T}}A) = n$. Noting that $\mathrm{rk}(A^{\mathrm{T}}A) = \mathrm{rk}(A)$ (note from (4.10) that $\mathrm{rk}(A^{\mathrm{T}}A) = \mathrm{rk}(A^{\mathrm{T}})$ and recall $\mathrm{rk}(A^{\mathrm{T}}) = \mathrm{rk}(A)$), we find that (4.7) has a unique solution iff $\mathrm{rk}(A) = n$, i.e., iff A is of full rank (note $m \geq n$). □

Remark 4.13. Let $A \in \mathbb{R}^{m\times n}$, $m \geq n$, and assume that $\mathrm{rk}(A) = n$. As we have already observed, this implies that $A^{\mathrm{T}}A \in \mathbb{R}^{n\times n}$ is invertible. Consequently, the unique solution to the least squares problem (4.7) is given by

$$x = A^{\dagger}b \in \mathbb{R}^n, \quad \text{where} \quad A^{\dagger} := (A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}} \in \mathbb{R}^{n\times m}.$$

The matrix $A^{\dagger}$ is called the Moore–Penrose inverse (or pseudoinverse) of A. The Moore–Penrose inverse is a generalization of the matrix inverse and is being discussed extensively on the problem sheets.

Solution algorithms

We present three well-known algorithms for solving least squares problems. The first solution algorithm is via the normal equation $A^{\mathrm{T}}Ax = A^{\mathrm{T}}b$. Suppose that A is of full rank and observe the following:

Remark 4.14. Let $A \in \mathbb{R}^{m\times n}$, $m \geq n$, $b \in \mathbb{R}^m$ and assume $\mathrm{rk}(A) = n$. Then, the matrix $A^{\mathrm{T}}A \in \mathbb{R}^{n\times n}$ is symmetric positive definite. Indeed, we have $(A^{\mathrm{T}}A)^{\mathrm{T}} = A^{\mathrm{T}}A$ and

$$\langle x, A^{\mathrm{T}}Ax\rangle = \langle Ax, Ax\rangle = \|Ax\|_2^2 > 0 \qquad \forall x \in \mathbb{R}^n\backslash\{0\}.$$

Here, we have used that $Ax \in \mathbb{R}^m \backslash \{0\}$ for $x \in \mathbb{R}^n \backslash \{0\}$ since $\mathrm{rk}(A) = n$ (recall that $\mathrm{nullity}(A) = n - \mathrm{rk}(A)$ from Theorem 1.2(ii)). Therefore, by Theorem 4.6, $A^{\mathrm{T}}A$ admits a unique Cholesky factorization $A^{\mathrm{T}}A = R^{\mathrm{T}}R$ with $R \in \mathbb{R}^{n \times n}$ upper-triangular with positive diagonal entries, and the normal equation turns into

$$A^{\mathrm{T}}Ax = A^{\mathrm{T}}b \quad \Longleftrightarrow \quad R^{\mathrm{T}}Rx = A^{\mathrm{T}}b.$$

This leads to the following algorithm:

Algorithm 4.4 (Solution of least squares problems via normal equation)**.** Given $m, n \in \mathbb{N}$ with $m \geq n$, a matrix $A \in \mathbb{R}^{m \times n}$ with $\mathrm{rk}(A) = n$, and a vector $b \in \mathbb{R}^m$, the unique solution $x \in \mathbb{R}^n$ to the least squares problem (4.7) can be obtained as follows:

Step 1) Compute the matrix $\tilde{A} := A^{\mathrm{T}}A \in \mathbb{R}^{n \times n}$ and the vector $\tilde{b} := A^{\mathrm{T}}b \in \mathbb{R}^n$.

Step 2) Compute the Cholesky factorization $\tilde{A} = R^{\mathrm{T}}R$ of $\tilde{A}$.

Step 3) Solve the lower-triangular system $R^{\mathrm{T}}z = \tilde{b}$ for $z \in \mathbb{R}^n$.

Step 4) Solve the upper-triangular system $Rx = z$ for $x \in \mathbb{R}^n$.

The work for Algorithm 4.4 is dominated by the computation of $\tilde{A} = A^{\mathrm{T}}A$ ($\sim mn^2$ flops, using symmetry of $\tilde{A}$) and the computation of its Cholesky factorization ($\sim \frac{1}{3}n^3$ flops via Algorithm 4.3).

Theorem 4.11. *Algorithm 4.4 requires $\sim mn^2 + \frac{1}{3}n^3$ flops.*

The second algorithm we present is via QR factorization, and is based on the following observation.

Remark 4.15. Let $A \in \mathbb{R}^{m \times n}$, $m \geq n$, $b \in \mathbb{R}^m$, and assume that we have found a reduced QR factorization $A = \hat{Q}\hat{R}$ of A. Then, $x \in \mathbb{R}^n$ is a solution to the least squares problem (4.7) iff $A^{\mathrm{T}}Ax = A^{\mathrm{T}}b$, or equivalently, $\hat{R}^{\mathrm{T}}\hat{Q}^{\mathrm{T}}\hat{Q}\hat{R}x = \hat{R}^{\mathrm{T}}\hat{Q}^{\mathrm{T}}b$ which can be simplified to $\hat{R}^{\mathrm{T}}\hat{R}x = \hat{R}^{\mathrm{T}}\hat{Q}^{\mathrm{T}}b$. Observe that if A is of full rank, then $\hat{R}$ is invertible (see proof of Theorem 3.2) in which case the unique solution $x \in \mathbb{R}^n$ to the least squares problem is determined from

$$\hat{R}x = \hat{Q}^{\mathrm{T}}b.$$

This leads to the following algorithm:

Algorithm 4.5 (Solution of least squares problems via QR)**.** Given $m, n \in \mathbb{N}$ with $m \geq n$, a matrix $A \in \mathbb{R}^{m \times n}$ with $\mathrm{rk}(A) = n$, and a vector $b \in \mathbb{R}^m$, the unique solution $x \in \mathbb{R}^n$ to the least squares problem (4.7) can be obtained as follows:

Step 1) Compute a reduced QR factorization $A = \hat{Q}\hat{R}$ of A.

Step 2) Compute $\tilde{b} = \hat{Q}^{\mathrm{T}}b \in \mathbb{R}^n$.

Step 3) Solve the upper-triangular system $\hat{R}x = \tilde{b}$ for $x \in \mathbb{R}^n$.

The work for Algorithm 4.5 is dominated by the computation of a reduced QR factorization ($\sim 2mn^2 - \frac{2}{3}n^3$ flops via Householder, see Algorithm 3.4).

Theorem 4.12. *Algorithm 4.5 requires* $\sim 2mn^2 - \frac{2}{3}n^3$ *flops.*

The third algorithm we present is via the SVD, and is based on the following observation.

Remark 4.16. Let $A \in \mathbb{R}^{m\times n}$, $m \geq n$, $b \in \mathbb{R}^m$, and assume that we have found a reduced SVD $A = \hat{U}\hat{\Sigma}V^{\mathrm{T}}$ of A. Then, $x \in \mathbb{R}^n$ is a solution to the least squares problem (4.7) iff $A^{\mathrm{T}}Ax = A^{\mathrm{T}}b$, or equivalently, $V\hat{\Sigma}^{\mathrm{T}}\hat{U}^{\mathrm{T}}\hat{U}\hat{\Sigma}V^{\mathrm{T}}x = V\hat{\Sigma}^{\mathrm{T}}\hat{U}^{\mathrm{T}}b$ which can be simplified to $V\hat{\Sigma}^{\mathrm{T}}\hat{\Sigma}V^{\mathrm{T}}x = V\hat{\Sigma}^{\mathrm{T}}\hat{U}^{\mathrm{T}}b$. Observe that if A is of full rank, then $V\hat{\Sigma}^{\mathrm{T}} \in \mathbb{R}^{n\times n}$ is invertible in which case the unique solution $x \in \mathbb{R}^n$ to the least squares problem is determined from

$$\hat{\Sigma}V^{\mathrm{T}}x = \hat{U}^{\mathrm{T}}b.$$

This leads to the following algorithm:

Algorithm 4.6 (Solution of least squares problems via SVD)**.** Given $m, n \in \mathbb{N}$ with $m \geq n$, a matrix $A \in \mathbb{R}^{m\times n}$ with $\mathrm{rk}(A) = n$, and a vector $b \in \mathbb{R}^m$, the unique solution $x \in \mathbb{R}^n$ to the least squares problem (4.7) can be obtained as follows:

Step 1) Compute a reduced SVD $A = \hat{U}\hat{\Sigma}V^{\mathrm{T}}$ of A.

Step 2) Compute $\tilde{b} = \hat{U}^{\mathrm{T}}b \in \mathbb{R}^n$.

Step 3) Solve the diagonal system $\hat{\Sigma}z = \tilde{b}$ for $z \in \mathbb{R}^n$.

Step 4) Compute $x = Vz \in \mathbb{R}^n$.

The work for Algorithm 4.6 is dominated by the computation of a reduced SVD (requires $\sim 2mn^2 + 11n^3$ flops, see Trefethen, Bau).

Theorem 4.13. *Algorithm 4.5 requires* $\sim 2mn^2 + 11n^3$ *flops.*

Let us compare the three algorithms. In view of speed, the first algorithm (Algorithm 4.4) seems to be the best. However, the second algorithm (Algorithm 4.5) is superior with regards to numerical stability and is indeed the standard method to solve least squares problems in practice. The third algorithm 4.6 is rarely used due to its computational cost, but it comes in handy when A is close to being rank-deficient.

5 Conditioning and Stability

5.1 Conditioning of mathematical problems

In this section, we study the perturbation behavior of mathematical problems, which is referred to as *conditioning.* We regard a problem as a function

$$f : X \to Y$$

with normed vector spaces X (the data space) and Y (the solution space). A problem f, together with a particular data point $x \in X$ (a pair (f, x) is called problem instance or simply problem as well), is called well-conditioned if small changes in x only lead to small changes in $f(x)$. Otherwise, i.e., if a small change in x can lead to a large change in $f(x)$, we call the problem (instance) ill-conditioned.

The condition number defined below is a measure for the perturbation behavior of a problem.

Definition 5.1. Let $(X, \|\cdot\|_X)$ and $(Y, \|\cdot\|_Y)$ be normed vector spaces. For a problem $f : X \to Y$ and a given data point $x \in X$, we define

(i) the absolute condition number $\hat{\kappa} = \hat{\kappa}(x)$ by

$$\hat{\kappa} := \lim_{\delta \to 0} \sup_{\substack{\Delta x \in X \\ 0 < \|\Delta x\|_X \leq \delta}} \frac{\|f(x + \Delta x) - f(x)\|_Y}{\|\Delta x\|_X},$$

(ii) and, if $x \in X \backslash \{0\}$ and $f(x) \in Y \backslash \{0\}$, the relative condition number $\kappa = \kappa(x)$ by

$$\kappa := \lim_{\delta \to 0} \sup_{\substack{\Delta x \in X \\ 0 < \|\Delta x\|_X \leq \delta}} \frac{\frac{\|f(x+\Delta x) - f(x)\|_Y}{\|f(x)\|_Y}}{\frac{\|\Delta x\|_X}{\|x\|_X}}. \tag{5.1}$$

We will choose the relative condition number to decide whether a problem is well-conditioned (κ is more important than $\hat{\kappa}$ due to floating point arithmetic used by computers, see next section). If κ is small (e.g., $1, 10, 100$), the problem is called well-conditioned, and if κ is large (e.g., $10^6, 10^{12}$), the problem is called ill-conditioned.

Remark 5.1. Let $X = \mathbb{R}^n$ and $Y = \mathbb{R}^m$ with chosen norms $\|\cdot\|_{(n)}$ on $\mathbb{R}^n$ and $\|\cdot\|_{(m)}$ on $\mathbb{R}^m$. Consider a problem $f : X \to Y$, a given data point $x \in \mathbb{R}^n$, and assume that f is differentiable at x. Then, we have

$$\hat{\kappa} = \|J_f(x)\|_{(m,n)}, \qquad \kappa = \frac{\|J_f(x)\|_{(m,n)} \|x\|_{(n)}}{\|f(x)\|_{(m)}}$$

where $J_f(x) \in \mathbb{R}^{m \times n}$ denotes the Jacobian of f at x whose entries are given by $(J_f(x))_{ij} = \partial_j f_i$, and $\|\cdot\|_{(m,n)}$ denotes the matrix norm on $\mathbb{R}^{m \times n}$ induced by the norms $\|\cdot\|_{(n)}$ on $\mathbb{R}^n$ and $\|\cdot\|_{(m)}$ on $\mathbb{R}^m$ (recall Definition 1.9).

Example 5.1 (Some first examples on conditioning). (i) Constant multiple of a real number: For $X = Y = \mathbb{R}$ with norm $\|\cdot\|_{(1)} := |\cdot|$ on $\mathbb{R}$, consider the problem $f : \mathbb{R} \to \mathbb{R}$, $x \mapsto 7x$, i.e., the problem of obtaining $7x$ from $x \in \mathbb{R}$. Note that f is differentiable on $\mathbb{R}$ and we have $J_f(x) = f'(x) = 7$ for all $x \in \mathbb{R}$. Hence,

$$\kappa = \frac{\|J_f(x)\|_{(1,1)} \|x\|_{(1)}}{\|f(x)\|_{(1)}} = \frac{|7||x|}{|7x|} = 1.$$

The problem is well-conditioned.

(ii) Addition of two real numbers: For $X = \mathbb{R}^2$ with norm $\|\cdot\|_{(2)} := \|\cdot\|_2$ on $\mathbb{R}^2$, and $Y = \mathbb{R}$ with norm $\|\cdot\|_{(1)} := |\cdot|$ on $\mathbb{R}$, consider the problem $f : \mathbb{R}^2 \to \mathbb{R}$, $(x_1, x_2) \mapsto x_1 + x_2$, i.e., the problem of finding the sum of two real values. Note that f is differentiable on $\mathbb{R}^2$ and we have $J_f(x) = \begin{pmatrix} \partial_1 f & \partial_2 f \end{pmatrix} = \begin{pmatrix} 1 & 1 \end{pmatrix} \in \mathbb{R}^{1\times 2}$. Hence,

$$\kappa = \frac{\|x\|_{(2)}}{\|f(x)\|_{(1)}} \|J_f(x)\|_{(1,2)} = \frac{\sqrt{x_1^2 + x_2^2}}{|x_1 + x_2|} \sup_{\substack{z \in \mathbb{R}^2 \\ \|z\|_2 = 1}} \left|\begin{pmatrix} 1 & 1 \end{pmatrix} z\right| = \sqrt{2} \frac{\sqrt{x_1^2 + x_2^2}}{|x_1 + x_2|}.$$

Note that when $x_2 \approx -x_1$ and $x_1 \neq 0$ we have that κ is large and the problem is ill-conditioned. This effect is referred to as *cancellation error*.

(iii) Polynomial root-finding: Consider the polynomial

$$p_1(t) := t^2 - 2t + 1$$

with a double root at $t = 1$. We are interested in the perturbation behavior in the roots with respect to changes in the coefficients – say we keep the coefficients of t^2 and t fixed, and consider the polynomial

$$p_x(t) := t^2 - 2t + x$$

for $x \leq 1$. Note that the roots of p_x are at $t = 1 \pm \sqrt{1 - x}$ for $x \leq 1$.

To bring it into our setting, we set $X = Y = \mathbb{R}$ with norm $|\cdot|$ on $\mathbb{R}$ and define the problem $f : \mathbb{R} \to \mathbb{R}$, $x \mapsto f(x)$ by setting $f(x)$ to be the largest root of p_x if $x \leq 1$, and set $f(x) := f(1) = 1$ for all $x > 1$ (note this doesn't introduce perturbation errors to the right of $x = 1$ as $f(1 + \Delta x) - f(1) = 0$ for $\Delta x > 0$).

Let us show that the condition number of the problem at $x = 1$ is $\kappa(1) = \infty$, i.e., the problem is severely ill-conditioned. Observe that $f(1) = 1$. If we perturb $x = 1$ by some $\Delta x < 0$, we find a change in $f(x)$ of size $|f(1 + \Delta x) - f(1)| = \sqrt{-\Delta x}$. (If we perturb $x = 1$ by some $\Delta x > 0$, we find no change in $f(x)$ by construction). Hence, for any $\delta > 0$ we have

$$\sup_{\Delta x \in [-\delta, \delta] \setminus \{0\}} \frac{|f(1 + \Delta x) - f(1)|}{|\Delta x|} \frac{|1|}{|f(1)|} = \sup_{\Delta x \in [-\delta, 0)} \frac{\sqrt{-\Delta x}}{-\Delta x} = \sup_{\Delta x \in [-\delta, 0)} \frac{1}{\sqrt{-\Delta x}} = \infty,$$

and thus, $\kappa(1) = \infty$.

We proceed with the conditioning of matrix-vector multiplication and the conditioning of the solution of linear systems, leading to the two central conditioning theorems in numerical linear algebra.

Conditioning of matrix-vector multiplication

Let $X = \mathbb{R}^n$ and $Y = \mathbb{R}^m$ with chosen norms $\|\cdot\|_{(n)}$ on $\mathbb{R}^n$ and $\|\cdot\|_{(m)}$ on $\mathbb{R}^m$, and consider a matrix $A \in \mathbb{R}^{m\times n}$. We are now looking at the problem

$$f : \mathbb{R}^n \to \mathbb{R}^m, \quad x \mapsto Ax,$$

i.e., the problem of computing the matrix-vector product $Ax \in \mathbb{R}^m$ from $x \in \mathbb{R}^n$. Noting that f is differentiable and $J_f(x) = A$ for all $x \in \mathbb{R}^n$, we have by Remark 5.1 that

$$\kappa = \frac{\|J_f(x)\|_{(m,n)}\|x\|_{(n)}}{\|f(x)\|_{(m)}} = \frac{\|A\|_{(m,n)}\|x\|_{(n)}}{\|Ax\|_{(m)}},$$

where $\|\cdot\|_{(m,n)}$ denotes the matrix norm on $\mathbb{R}^{m\times n}$ induced by the norms $\|\cdot\|_{(n)}$ on $\mathbb{R}^n$ and $\|\cdot\|_{(m)}$ on $\mathbb{R}^m$. If $m = n$, $\|\cdot\|_{(m)} = \|\cdot\|_{(n)}$, and A is invertible, then

$$\kappa = \|A\|_{(n,n)}\frac{\|A^{-1}Ax\|_{(n)}}{\|Ax\|_{(n)}} \leq \|A\|_{(n,n)}\|A^{-1}\|_{(n,n)}. \tag{5.2}$$

This upper bound is attained for certain choices of x.

Definition 5.2. Let $A \in \mathbb{R}^{n\times n}$ be invertible and let $\|\cdot\|$ be a norm on $\mathbb{R}^{n\times n}$. Then, we define the condition number of A with respect to the norm $\|\cdot\|$ to be $\kappa_{\|\cdot\|}(A) := \|A\|\,\|A^{-1}\|$. If this quantity is small, we call A well-conditioned. Otherwise, we call A ill-conditioned.

The condition number of a singular square matrix is typically defined to be ∞.

Theorem 5.1. *Let $A \in \mathbb{R}^{n\times n}$ be invertible. Consider the vector space $\mathbb{R}^n$ with a chosen norm $\|\cdot\|_{(n)}$ on $\mathbb{R}^n$, and let $\|\cdot\|_{(n,n)}$ denote the matrix norm on $\mathbb{R}^{n\times n}$ induced by the vector norm $\|\cdot\|_{(n)}$. Then, we have the following:*

(i) For the problem $f : \mathbb{R}^n \to \mathbb{R}^n$, $x \mapsto Ax$, i.e., the problem of finding $b = Ax$ from $x \in \mathbb{R}^n$, the condition number $\kappa = \kappa(x)$ is given by

$$\kappa = \|A\|_{(n,n)}\frac{\|x\|_{(n)}}{\|b\|_{(n)}} \leq \kappa_{\|\cdot\|_{(n,n)}}(A). \tag{5.3}$$

If $\|\cdot\|_{(n)} = \|\cdot\|_2$ is the vector 2-norm (and hence, $\|\cdot\|_{(n,n)} = \|\cdot\|_2$ the spectral norm), we have equality in (5.3) *if x is a multiple of a right singular vector of A corresponding to the smallest singular value σ_n.*

(ii) For the problem $f : \mathbb{R}^n \to \mathbb{R}^n$, $b \mapsto A^{-1}b$, i.e., the problem of finding the solution $x \in \mathbb{R}^n$ to $Ax = b$ from the right-hand side $b \in \mathbb{R}^n$, the condition number $\kappa = \kappa(b)$ is given by

$$\kappa = \|A^{-1}\|_{(n,n)}\frac{\|b\|_{(n)}}{\|x\|_{(n)}} \leq \kappa_{\|\cdot\|_{(n,n)}}(A). \tag{5.4}$$

If $\|\cdot\|_{(n)} = \|\cdot\|_2$ is the vector 2-norm (and hence, $\|\cdot\|_{(n,n)} = \|\cdot\|_2$ the spectral norm), we have equality in (5.4) *if b is a multiple of a left singular vector of A corresponding to the largest singular value σ_1.*

Proof. Observe that (5.3) has already been shown in (5.2), and that (5.4) follows from (5.2) with A replaced by A^{-1} and x replaced by b. We leave the remaining parts as an exercise. □

Remark 5.2. Let us revisit the problem for $A \in \mathbb{R}^{m\times n}$ being a rectangular matrix with $m \geq n$ and $\mathrm{rk}(A) = n$. Then, observing that $A^\dagger A = I_n$, i.e., the Moore-Penrose inverse $A^\dagger \in \mathbb{R}^{n\times m}$ is a left-inverse, we find that

$$\kappa = \|A\|_{(m,n)} \frac{\|A^\dagger Ax\|_{(n)}}{\|Ax\|_{(m)}} \leq \|A\|_{(m,n)} \|A^\dagger\|_{(n,m)},$$

where $\|\cdot\|_{(m,n)}$ is the induced matrix norm on $\mathbb{R}^{m\times n}$, and $\|\cdot\|_{(n,m)}$ is the induced matrix norm on $\mathbb{R}^{n\times m}$ (induced by the vector norms $\|\cdot\|_{(n)}$ on $\mathbb{R}^n$, $\|\cdot\|_{(m)}$ on $\mathbb{R}^m$). We define the condition number of A to be $\kappa_{\|\cdot\|_{(m,n)},\|\cdot\|_{(n,m)}}(A) := \|A\|_{(m,n)}\|A^\dagger\|_{(n,m)}$.

Next, let us discuss the conditioning of the solution of linear systems $Ax = b$ with respect to perturbations in the system matrix A.

Conditioning of linear systems

Let $X = \mathbb{R}^{n\times n}$ and $Y = \mathbb{R}^n$ with a chosen norm $\|\cdot\|_{(n)}$ on $\mathbb{R}^n$ and induced matrix norm $\|\cdot\|_{(n,n)}$ on $\mathbb{R}^{n\times n}$. Let $b \in \mathbb{R}^n$ be fixed. Consider the problem

$$f : A \mapsto A^{-1}b \in \mathbb{R}^n \quad \text{for } A \in \mathbb{R}^{n\times n} \text{ invertible},$$

i.e., the problem of finding the solution $x \in \mathbb{R}^n$ to $Ax = b$. Although the space of invertible $n \times n$ matrices is not a vector space, we can still study the perturbation behavior of f since a perturbed invertible matrix is still invertible if the perturbation is sufficiently small: the following result is often referred to as the perturbation lemma.

Lemma 5.1 (Perturbation lemma)**.** *Let $A \in \mathbb{R}^{n\times n}$ be invertible, and let $\|\cdot\|$ be a submultiplicative norm on $\mathbb{R}^{n\times n}$ (i.e., a norm satisfying $\|M_1M_2\| \leq \|M_1\|\,\|M_2\|$ for any $M_1, M_2 \in \mathbb{R}^{n\times n}$). Then, for any $\Delta A \in \mathbb{R}^{n\times n}$ with $\|\Delta A\| < \|A^{-1}\|^{-1}$, the perturbed matrix $A + \Delta A \in \mathbb{R}^{n\times n}$ is invertible and there holds*

$$\|(A + \Delta A)^{-1}\| \leq \frac{\|A^{-1}\|}{1 - \|\Delta A\|\,\|A^{-1}\|}.$$

Proof. Lemma 2.1 in “Applied Numerical Linear Algebra” by J.W. Demmel (SIAM, 1997) shows that for any $X \in \mathbb{R}^{n\times n}$ with $\|X\| < 1$, we have that $I_n - X$ is invertible and there holds $(I_n - X)^{-1} = \sum_{i=0}^\infty X^i$ (Neumann series) and $\|(I_n - X)^{-1}\| \leq \frac{1}{1-\|X\|}$ (we omit the proof of this fact). Now let $A \in \mathbb{R}^{n\times n}$ be invertible and $\Delta A \in \mathbb{R}^{n\times n}$ be such that $\|\Delta A\| < \|A^{-1}\|^{-1}$. Observe that we can write

$$A + \Delta A = (I_n - X)A \quad \text{with} \quad X := -(\Delta A)A^{-1} \in \mathbb{R}^{n\times n}$$

and, using submultiplicativity of $\|\cdot\|$, that $\|X\| = \|(\Delta A)A^{-1}\| \leq \|\Delta A\|\,\|A^{-1}\| < 1$. Hence, we find that $I_n - X$ is invertible as a product of invertible matrices, and we find that

$$\begin{aligned}\|(A + \Delta A)^{-1}\| &= \|A^{-1}(I_n - X)^{-1}\| \\ &\leq \|A^{-1}\|\|(I_n - X)^{-1}\| \leq \frac{\|A^{-1}\|}{1 - \|X\|} \leq \frac{\|A^{-1}\|}{1 - \|\Delta A\|\,\|A^{-1}\|},\end{aligned}$$

where we have used submultiplicativity of $\|\cdot\|$, the bound $\|(I_n - X)^{-1}\| \leq \frac{1}{1-\|X\|}$ and the bound $\|X\| = \|(\Delta A)A^{-1}\| \leq \|\Delta A\|\,\|A^{-1}\|$. □

Recall from Remark 1.11 that the induced norm $\|\cdot\|_{(n,n)}$ is submultiplicative and hence the perturbation lemma can be applied. Let $A \in \mathbb{R}^{n\times n}$ be invertible and let $\Delta A \in \mathbb{R}^{n\times n}$ be such that $\|\Delta A\|_{(n,n)} < \|A^{-1}\|_{(n,n)}^{-1}$ (so that the perturbation lemma applies to the perturbed matrix $A + \Delta A$). We are interested in the quantity

$$q(\Delta A) := \frac{\|f(A+\Delta A) - f(A)\|_{(n)}}{\|\Delta A\|_{(n,n)}} \frac{\|A\|_{(n,n)}}{\|f(A)\|_{(n)}} = \frac{\|(A+\Delta A)^{-1}b - A^{-1}b\|_{(n)}}{\|\Delta A\|_{(n,n)}} \frac{\|A\|_{(n,n)}}{\|A^{-1}b\|_{(n)}}.$$

Note that writing $(A+\Delta A)^{-1}b = x + \Delta x$ for some $\Delta x \in \mathbb{R}^n$ where $x := A^{-1}b$, the vectors x and $x + \Delta x$ are the solutions to

$$Ax = b, \qquad (A+\Delta A)(x+\Delta x) = b.$$

This yields $(\Delta A)x + (A+\Delta A)\Delta x = 0$, i.e., $\Delta x = -(A+\Delta A)^{-1}(\Delta A)x$ and hence,

$$(A+\Delta A)^{-1}b - A^{-1}b = \Delta x = -(A+\Delta A)^{-1}(\Delta A)A^{-1}b.$$

We find that

$$\begin{aligned} q(\Delta A) = \frac{\|(A+\Delta A)^{-1}(\Delta A)A^{-1}b\|_{(n)}\|A\|_{(n,n)}}{\|\Delta A\|_{(n,n)}\|A^{-1}b\|_{(n)}} &\leq \|(A+\Delta A)^{-1}\|_{(n,n)}\|A\|_{(n,n)} \\ &\leq \frac{\|A\|_{(n,n)}\|A^{-1}\|_{(n,n)}}{1 - \|\Delta A\|_{(n,n)}\|A^{-1}\|_{(n,n)}} \\ &= \frac{\kappa_{\|\cdot\|_{(n,n)}}(A)}{1 - \frac{\|\Delta A\|_{(n,n)}}{\|A\|_{(n,n)}}\,\kappa_{\|\cdot\|_{(n,n)}}(A)}, \end{aligned}$$

and it follows that the condition number for the problem f at the matrix A is bounded by the condition number of the matrix A:

$$\kappa = \lim_{\delta\to 0} \sup_{\substack{\Delta A \in \mathbb{R}^{n\times n} \\ 0 < \|\Delta A\|_{(n,n)} \leq \delta}} q(\Delta A) \leq \kappa_{\|\cdot\|_{(n,n)}}(A). \tag{5.5}$$

It can actually be shown that there holds equality in the above estimate (we omit the proof) and we arrive at the following important theorem:

Theorem 5.2. *Consider the vector space $\mathbb{R}^n$ with a chosen norm $\|\cdot\|_{(n)}$ on $\mathbb{R}^n$, and let $\|\cdot\|_{(n,n)}$ denote the matrix norm on $\mathbb{R}^{n\times n}$ induced by the vector norm $\|\cdot\|_{(n)}$. Then, for a fixed $b \in \mathbb{R}^n$, the condition number for the problem of finding the solution $x \in \mathbb{R}^n$ of $Ax = b$ from $A \in \{M \in \mathbb{R}^{n\times n} : M$ invertible$\}$ is given by*

$$\kappa = \kappa_{\|\cdot\|_{(n,n)}}(A).$$

Conditioning of least squares problems

Given $A \in \mathbb{R}^{m\times n}$, $m \geq n$, with $\operatorname{rk}(A) = n$ and $b \in \mathbb{R}^m$, we consider the least squares problem

$$\text{Minimize } \|Av - b\|_2 \text{ over } v \in \mathbb{R}^n.$$

Recall that in this situation we have

- $x = A^\dagger b$ is the unique solution to the least squares problem, i.e., the unique vector $x \in \mathbb{R}^n$ satisfying $\|Ax - b\|_2 = \inf_{v\in\mathbb{R}^n} \|Av - b\|_2$,
- $y = Ax = AA^\dagger b = Pb$ is the unique point in $\mathcal{R}(A)$ closest to b in the Euclidean distance, i.e., the unique vector $y \in \mathcal{R}(A)$ satisfying $\|y - b\|_2 = \inf_{w\in\mathcal{R}(A)} \|w - b\|_2$,

where $A^\dagger = (A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}} \in \mathbb{R}^{n\times m}$ is the Moore–Penrose inverse of the matrix A, and $P = AA^\dagger \in \mathbb{R}^{m\times m}$ is the orthogonal projector onto $\mathcal{R}(A)$ (see Remark 3.11).

We consider the following mathematical problems:

(i) obtain y from b for fixed A, i.e., $f_{b\mapsto y} : \mathbb{R}^m \to \mathbb{R}^m$, $b \mapsto AA^\dagger b$,

(ii) obtain x from b for fixed A, i.e., $f_{b\mapsto x} : \mathbb{R}^m \to \mathbb{R}^n$, $b \mapsto A^\dagger b$,

(iii) obtain y from A for fixed b, i.e., $f_{A\mapsto y} : A \mapsto AA^\dagger b \in \mathbb{R}^m$ for $A \in \mathbb{R}^{m\times n}$, $\operatorname{rk}(A) = n$,

(iv) obtain x from A for fixed b, i.e., $f_{A\mapsto x} : A \mapsto A^\dagger b \in \mathbb{R}^n$ for $A \in \mathbb{R}^{m\times n}$, $\operatorname{rk}(A) = n$,

and we consider the 2-norm on $\mathbb{R}^m$ and $\mathbb{R}^n$, and the spectral norm on $\mathbb{R}^{m\times n}$ and $\mathbb{R}^{n\times m}$.

Theorem 5.3 (Conditioning of least squares problems)**.** *In this situation, there holds*

$$\kappa_{b\mapsto y} = \frac{1}{\cos(\theta)}, \quad \kappa_{b\mapsto x} = \frac{\kappa(A)}{\eta\cos(\theta)}, \quad \kappa_{A\mapsto y} \leq \frac{\kappa(A)}{\cos(\theta)}, \quad \kappa_{A\mapsto x} \leq \kappa(A) + \frac{(\kappa(A))^2\tan(\theta)}{\eta},$$

where $\kappa_{i\mapsto j}$ ($i \in \{b, A\}$, $j \in \{x, y\}$) denotes the condition number for $f_{i\mapsto j}$, and

$$\kappa(A) := \|A\|_2\|A^\dagger\|_2 \geq 1,\ \ \theta := \cos^{-1}\left(\frac{\|AA^\dagger b\|_2}{\|b\|_2}\right) \in \left[0, \frac{\pi}{2}\right],\ \ \eta := \frac{\|A\|_2\|A^\dagger b\|_2}{\|AA^\dagger b\|_2} \in [1, \kappa(A)].$$

Before we prove the theorem, let us make some observations.

Remark 5.3. For $A \in \mathbb{R}^{m\times n}$, $m \geq n$, $\operatorname{rk}(A) = n$, the condition number in the spectral norm is given by $\kappa(A) = \|A\|_2\|A^\dagger\|_2 = \frac{\sigma_1}{\sigma_n} \in [1, \infty)$ with σ_1 denoting the largest and σ_n the smallest singular value of A.

Remark 5.4. The angle θ is a measure for the closeness of the projection $Pb = AA^\dagger b$ to b.

Remark 5.5. If $m = n$, we have $A^\dagger = A^{-1}$ and hence $\theta = 0$. In particular, we find $\kappa_{b\mapsto x} = \frac{\kappa(A)}{\eta} = \frac{\|A^{-1}\|_2\|b\|_2}{\|A^{-1}b\|_2}$ and $\kappa_{A\mapsto x} \leq \kappa(A) = \|A\|_2\|A^{-1}\|_2$, i.e., we recover the previous results (5.4) and (5.5) on the conditioning of square linear systems.

Proof of Theorem 5.3. (i) Let $A \in \mathbb{R}^{m\times n}$, $m \geq n$, with $\operatorname{rk}(A) = n$ be fixed, and consider the problem $f_{b\mapsto y} : \mathbb{R}^m \to \mathbb{R}^m$, $b \mapsto AA^\dagger b$. We take a SVD of A: Let $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ be orthogonal matrices and $\Sigma = \operatorname{diag}_{m\times n}(\sigma_1, \dots, \sigma_n) \in \mathbb{R}^{m\times n}$ with $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n > 0$ (all positive as $\operatorname{rk}(A) = n$) be such that $A = U\Sigma V^{\mathrm{T}}$. Then, $\Sigma^{\mathrm{T}}\Sigma \in \mathbb{R}^{n\times n}$ is invertible and we find that

$$\begin{aligned} A^\dagger = (A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}} &= (V\Sigma^{\mathrm{T}}U^{\mathrm{T}}U\Sigma V^{\mathrm{T}})^{-1}V\Sigma^{\mathrm{T}}U^{\mathrm{T}} \\ &= (V\Sigma^{\mathrm{T}}\Sigma V^{\mathrm{T}})^{-1}V\Sigma^{\mathrm{T}}U^{\mathrm{T}} = V(\Sigma^{\mathrm{T}}\Sigma)^{-1}\Sigma^{\mathrm{T}}U^{\mathrm{T}} = V\Sigma^\dagger U^{\mathrm{T}}. \end{aligned}$$

Hence, we have

$$AA^\dagger = U\Sigma V^{\mathrm{T}}V\Sigma^\dagger U^{\mathrm{T}} = U\Sigma\Sigma^\dagger U^{\mathrm{T}} = U \left(\begin{array}{c|c} I_n & 0_{n\times(m-n)} \\ \hline 0_{(m-n)\times n} & 0_{(m-n)\times(m-n)} \end{array}\right) U^{\mathrm{T}}.$$

Note that this is a SVD of $AA^\dagger$ and we see that $\|AA^\dagger\|_2 = 1$. We find that the condition number $\kappa_{b\mapsto y} = \kappa_{b\mapsto y}(b)$ of $f_{b\mapsto y}$ is given by

$$\kappa_{b\mapsto y} = \frac{\|J_{f_{b\mapsto y}}(b)\|_2\|b\|_2}{\|f_{b\mapsto y}(b)\|_2} = \frac{\|AA^\dagger\|_2\|b\|_2}{\|AA^\dagger b\|_2} = \frac{\|b\|_2}{\|AA^\dagger b\|_2} = \frac{1}{\cos(\theta)},$$

as required.

(ii) Let $A \in \mathbb{R}^{m\times n}$, $m \geq n$, with $\operatorname{rk}(A) = n$ be fixed, and consider the problem $f_{b\mapsto x} : \mathbb{R}^m \to \mathbb{R}^n$, $b \mapsto A^\dagger b$. Then, the condition number $\kappa_{b\mapsto x} = \kappa_{b\mapsto x}(b)$ of $f_{b\mapsto x}$ is given by

$$\kappa_{b\mapsto x} = \frac{\|J_{f_{b\mapsto x}}(b)\|_2\|b\|_2}{\|f_{b\mapsto x}(b)\|_2} = \frac{\|A^\dagger\|_2\|b\|_2}{\|A^\dagger b\|_2} = \|A\|_2\|A^\dagger\|_2\frac{\|AA^\dagger b\|_2}{\|A\|_2\|A^\dagger b\|_2}\frac{\|b\|_2}{\|AA^\dagger b\|_2} = \frac{\kappa(A)}{\eta\cos(\theta)},$$

as required.

(iii), (iv) We omit the proof for the two remaining problems $f_{A\mapsto y}$ and $f_{A\mapsto x}$. □

5.2 Floating point numbers and floating point arithmetic

Before we start to study stability of numerical algorithms, we need to have an understanding of the representation of real numbers on a computer. As a first observation, we note that computers use a finite number of bits to represent a real number and hence,

- there must be a largest represented number $x^+_{\text{max}} > 0$, a smallest represented number $x^-_{\text{min}} < 0$, a smallest positive represented number $x^+_{\text{min}} > 0$, and a largest negative represented number $x^-_{\text{max}} < 0$, i.e., the set of all represented numbers is a finite subset of $[x^-_{\text{min}}, x^-_{\text{max}}] \cup \{0\} \cup [x^+_{\text{min}}, x^+_{\text{max}}]$.
- there must be gaps between represented numbers.

Definition 5.3. Given $\beta \in \mathbb{N}$ with $\beta \geq 2$ (the base, usually taken to be 2), $t \in \mathbb{N}$ (the precision), and $e_{\text{min}}, e_{\text{max}} \in \mathbb{Z}$ (minimal and maximal exponent), we define the floating point system $F = F(\beta, t, e_{\text{min}}, e_{\text{max}}) \subseteq \mathbb{R}$ to be the set of real numbers that can be written as

$$x = (-1)^s \cdot (m_1\beta^{-1} + \dots + m_t\beta^{-t}) \cdot \beta^e =: (-1)^s \cdot [0.m_1 \dots m_t]_\beta \cdot \beta^e$$

for some $m_1, \dots, m_t \in \{0, 1, \dots, \beta - 1\}$, $e \in \mathbb{Z} \cap [e_{\text{min}}, e_{\text{max}}]$ and $s \in \{0, 1\}$. We call the number $[0.m_1 \dots m_t]_\beta \in [0, 1)$ the mantissa of x, and the number $e \in \mathbb{Z}$ the exponent of x. By requiring $m_1 \neq 0$ if $x \neq 0$ and setting $m_1 = 0$ if $x = 0$, the representation is unique.

Remark 5.6. In a floating point system $F = F(\beta, t, e_{\text{min}}, e_{\text{max}})$, the largest represented number is

$$x^+_{\text{max}} = (\beta - 1)\left(\sum_{i=1}^{t} \beta^{-i}\right) \cdot \beta^{e_{\text{max}}} = (1 - \beta^{-t})\beta^{e_{\text{max}}},$$

the smallest represented number is $x^-_{\text{min}} = -(1-\beta^{-t})\beta^{e_{\text{max}}}$, the smallest positive represented number is

$$x^+_{\text{min}} = \beta^{-1} \cdot \beta^{e_{\text{min}}} = \beta^{e_{\text{min}}-1},$$

and the largest negative represented number is $x^-_{\text{max}} = -\beta^{e_{\text{min}}-1}$. Therefore, we have

$$F(\beta, t, e_{\text{min}}, e_{\text{max}}) \subseteq [-(1-\beta^{-t})\beta^{e_{\text{max}}}, -\beta^{e_{\text{min}}-1}] \cup \{0\} \cup [\beta^{e_{\text{min}}-1}, (1-\beta^{-t})\beta^{e_{\text{max}}}].$$

Example 5.2. In the widely used IEEE double precision arithmetic, one uses $\beta = 2$, $t = 53$, and the represented numbers are of the form

$$\begin{aligned} x &= (-1)^s \cdot (m_1 2^{-1} + \cdots + m_{53} 2^{-53}) \cdot 2^{(c_{10}2^{10}+\cdots+c_0 2^0)-1022} \\ &= (-1)^s \cdot [0.m_1 \ldots m_{53}]_2 \cdot 2^{[c_{10}\ldots c_0]_2 - 1022} \end{aligned} \tag{5.6}$$

with $s, c_0, \ldots, c_{10}, m_1, \ldots, m_{53} \in \{0, 1\}$, biased exponent $[c_{10} \ldots c_0]_2 \in \{1, \ldots, 2046\}$, and $m_1 = 1$ for normalization purposes. The excluded numbers $[c_{10} \ldots c_0]_2 \in \{0, 2047\}$ are used for representing 0 and "NaN". The number x from (5.6) is equivalent to

$$x = (-1)^s \cdot (1 + [0.m_2 \ldots m_{53}]_2) \cdot 2^{[c_{10}\ldots c_0]_2 - 1023} = (-1)^s \cdot [1.m_2 \ldots m_{53}]_2 \cdot 2^{[c_{10}\ldots c_0]_2 - 1023}$$

and is stored as the binary number

$$|\underbrace{\;s\;}_{\text{1 bit}}|\underbrace{c_{10}|c_9|c_8|\ldots|c_2|c_1|c_0}_{\text{11 bits}}|\underbrace{m_2|m_3|m_4|\ldots|m_{51}|m_{52}|m_{53}}_{\text{52 bits}}|.$$

Note that we have $x^+_{\text{max}} = (1-2^{-53})2^{1024} \approx 1.8 \cdot 10^{308}$, $x^+_{\text{min}} = 2^{-1022} \approx 2.2 \cdot 10^{-308}$, and $x^-_{\text{min}} = -(1-2^{-53})2^{1024} \approx -1.8 \cdot 10^{308}$, $x^-_{\text{max}} = -2^{-1022} \approx -2.2 \cdot 10^{-308}$.

Observe that, in IEEE double precision arithmetic, the represented numbers

- in the interval $[1, 2]$ are $\{1 + j \cdot 2^{-52} \,|\, j \in \{0, 1, \ldots, 2^{52}\}\}$,
- in the interval $[2, 4]$ are $\{2 + j \cdot 2^{-51} \,|\, j \in \{0, 1, \ldots, 2^{52}\}\}$,
- in the interval $[2^k, 2^{k+1}]$ are $\{2^k + j \cdot 2^{k-52} \,|\, j \in \{0, 1, \ldots, 2^{52}\}\}$. Hence, the distance between adjacent numbers in a relative sense is at most $2^{-52} \approx 2.2 \cdot 10^{-16}$.

(Note that the represented numbers in $[2^{52}, 2^{53}]$ are precisely the integers $\mathbb{N} \cap [2^{52}, 2^{53}]$).

Remark 5.7. The gaps between adjacent numbers in a floating point system scale in proportion to their size. In contrast, in a fixed point system, one would have that the gaps between any two adjacent numbers are of the same size.

We proceed with the definition of the machine epsilon corresponding to a floating point system F, that is, a number measuring the resolution of F.

Definition 5.4. To a floating point system $F = F(\beta, t, e_{\min}, e_{\max})$, we associate the number

$$\varepsilon_{\text{machine}} = \frac{1}{2}\beta^{1-t},$$

called the machine epsilon.

Remark 5.8. For any $x \in [x^-_{\min}, x^-_{\max}] \cup [x^+_{\min}, x^+_{\max}]$ there exists a represented number $x' \in F$ satisfying

$$\frac{|x - x'|}{|x|} \leq \varepsilon_{\text{machine}}, \tag{5.7}$$

i.e., the distance between x and x' in a relative sense is at most $\varepsilon_{\text{machine}}$. Indeed, if we define a rounding operator $\text{fl} : [x^-_{\min}, x^-_{\max}] \cup \{0\} \cup [x^+_{\min}, x^+_{\max}] \to F$ with the property $|x - \text{fl}(x)| = \inf_{y \in F}|x - y|$ for all $x \in [x^-_{\min}, x^-_{\max}] \cup \{0\} \cup [x^+_{\min}, x^+_{\max}]$, then $x' = \text{fl}(x)$ satisfies (5.7). In particular,

$$\forall x \in [x^-_{\min}, x^-_{\max}] \cup \{0\} \cup [x^+_{\min}, x^+_{\max}] \quad \exists\, \varepsilon \in [-\varepsilon_{\text{machine}}, \varepsilon_{\text{machine}}] \quad \text{s.t.} \quad \text{fl}(x) = x(1 + \varepsilon). \tag{5.8}$$

Remark 5.9. The machine epsilon in IEEE double precision arithmetic is given by

$$\varepsilon_{\text{machine}} = \frac{2^{1-53}}{2} = 2^{-53} \approx 1.1 \cdot 10^{-16}.$$

An example for a rounding operator fl is the natural rounding defined via $\text{fl}(x) = \text{sign}(x)[0.m_1 \dots m_{53}]_2 2^e$ if $m_{54} = 0$ and $\text{fl}(x) = \text{sign}(x)([0.m_1 \dots m_{53}]_2 + 2^{-53})2^e$ if $m_{54} = 1$.

We can now present the analogue of the elementary operations (addition, subtraction, multiplication, and division of two real numbers) for two numbers of a floating point system.

Definition 5.5. Let F be a floating point system. We then define the floating point operations $\oplus, \ominus, \otimes, \oslash$ on F by

$$x \circledast y := \text{fl}(x * y), \qquad (x, y \in F)$$

for $\circledast \in \{\oplus, \ominus, \otimes, \oslash\}$.

In view of (5.8), we have the following result.

Theorem 5.4 (Fundamental axiom of floating point arithmetic)**.** *Let F be a floating point system and $\circledast \in \{\oplus, \ominus, \otimes, \oslash\}$. Then, for all $x, y \in F$ ($y \neq 0$ if $\circledast = \oslash$) there exists $\varepsilon \in [-\varepsilon_{\text{machine}}, \varepsilon_{\text{machine}}]$ such that there holds*

$$x \circledast y = (x * y)(1 + \varepsilon). \tag{5.9}$$

In particular, there holds $|x \circledast y - x * y| \leq \varepsilon_{\text{machine}}\, |x * y|$ for all $x, y \in F$.

5.3 Stability of numerical algorithms

Remark 5.10. From now on, for simplicity, we consider an idealized floating point system $F = F(\beta, t)$ ignoring overflow and underflow (all integer exponents $e \in \mathbb{Z}$ allowed).

Let us start by discussing the mathematical definition of an algorithm for "solving" a mathematical problem $f : X \to Y$ (with X, Y normed vector spaces). Suppose we have a computer with floating point system satisfying (5.9). We regard an algorithm for the problem as a map

$$\tilde{f} : X \to Y,$$

where for $x \in X$, $\tilde{f}(x)$ is defined as follows: First, round x to a floating point number $\mathrm{fl}(x)$ (with a rounding operator fl satisfying (5.8)), then run the (fixed) implementation/program of the algorithm with input $\mathrm{fl}(x)$, and define $\tilde{f}(x)$ to be the output (this is going to be a collection of floating point numbers in Y).

As we are going to frequently use the Landau symbol $\mathcal{O}$, let us briefly recall its definition:

Definition 5.6. For real-valued functions $u = u(t)$ and $v = v(t)$ of a variable $t \in \mathbb{R}_{>0}$, we define

$$u(t) = \mathcal{O}(v(t)) \text{ as } t \searrow 0 \quad :\Longleftrightarrow \quad \exists t_0, C > 0 : |u(t)| \leq Cv(t) \quad \forall t \in (0, t_0),$$

and

$$u(t) = \mathcal{O}(v(t)) \text{ as } t \to \infty \quad :\Longleftrightarrow \quad \exists t_0, C > 0 : |u(t)| \leq Cv(t) \quad \forall t \in (t_0, \infty).$$

Let us now define what we mean by an algorithm being accurate and by an algorithm being stable.

Definition 5.7. Let X and Y be normed vector spaces with norms $\|\cdot\|_X$ and $\|\cdot\|_Y$. Let $f : X \to Y$ be a problem and $\tilde{f} : X \to Y$ be an algorithm for f. Then, we make the following definitions.

(i) $\tilde{f}$ is called accurate iff for each $x \in X$ there holds

$$\frac{\|\tilde{f}(x) - f(x)\|_Y}{\|f(x)\|_Y} = \mathcal{O}(\varepsilon_{\text{machine}}). \tag{5.10}$$

(ii) $\tilde{f}$ is called stable iff for each $x \in X$ there holds

$$\frac{\|\tilde{f}(x) - f(\tilde{x})\|_Y}{\|f(\tilde{x})\|_Y} = \mathcal{O}(\varepsilon_{\text{machine}}) \quad \text{for some } \tilde{x} \in X \text{ with } \frac{\|\tilde{x} - x\|_X}{\|x\|_X} = \mathcal{O}(\varepsilon_{\text{machine}}). \tag{5.11}$$

Remark 5.11. The above statements of the form $\frac{\|p(x,\varepsilon_{\text{machine}})\|}{\|q(x,\varepsilon_{\text{machine}})\|} = \mathcal{O}(\varepsilon_{\text{machine}})$ (note the quantities inside the norm on the left-hand sides of (5.10) and (5.11) do indeed implicitly depend on $\varepsilon_{\text{machine}}$) are meant in the sense

$$\frac{\|p(x, \varepsilon_{\text{machine}})\|}{\|q(x, \varepsilon_{\text{machine}})\|} = \mathcal{O}(\varepsilon_{\text{machine}}) \text{ as } \varepsilon_{\text{machine}} \searrow 0, \quad \text{uniformly in } x,$$

which is to be understood as

$$\exists\, \varepsilon_0, C > 0: \ \|p(x, \varepsilon_{\text{machine}})\| \leq C\varepsilon_{\text{machine}}\|q(x, \varepsilon_{\text{machine}})\| \quad \forall \varepsilon_{\text{machine}} \in (0, \varepsilon_0), \ x \in X.$$

(Note that this makes (5.10) and (5.11) well defined also when the denominator is zero). The limit process $\varepsilon_{\text{machine}} \searrow 0$ can be thought of as running the algorithm on a family of computers satisfying (5.8) and (5.9) with corresponding values of $\varepsilon_{\text{machine}}$ tending to zero.

Remark 5.12. If the problem f is ill-conditioned, there is little hope to construct an accurate algorithm $\tilde{f}$. Even if the only error would stem from rounding the input data (and say everything else is performed exactly), this small perturbation can already lead to large changes in the result if f is ill-conditioned. This is why the appropriate goal in constructing algorithms is stability, which we think of as follows: if the algorithm $\tilde{f}$ is stable, it gives the almost right answer to an almost right question.

Often, one encounters algorithms which are backward stable, that is, they satisfy a stronger condition than stability. Namely, backward stable algorithms give the exact answer to an almost right question:

Definition 5.8. Let X and Y be normed vector spaces with norms $\|\cdot\|_X$ and $\|\cdot\|_Y$. Let $f: X \to Y$ be a problem and $\tilde{f}: X \to Y$ be an algorithm for f. Then, $\tilde{f}$ is called backward stable iff for each $x \in X$ there holds

$$\tilde{f}(x) = f(\tilde{x}) \quad \text{for some } \tilde{x} \in X \text{ with } \frac{\|\tilde{x} - x\|_X}{\|x\|_X} = \mathcal{O}(\varepsilon_{\text{machine}}). \tag{5.12}$$

Remark 5.13. Any backward stable algorithm is stable.

Let us make the following observation:

Theorem 5.5 (Independence of norm)**.** *If X, Y are finite-dimensional, the definitions of accuracy, stability, and backward stability are independent of the choice of norms in X and Y in the sense that the corresponding conditions either all hold or fail independently of the choice of norms.*

Proof. Exercise. □

Theorem 5.6 (Accuracy of backward stable algorithms)**.** *Let X and Y be normed vector spaces with norms $\|\cdot\|_X$ and $\|\cdot\|_Y$. Consider a problem $f: X \to Y$ with condition number κ given by* (5.1)*, and a backward stable algorithm $\tilde{f}: X \to Y$ for f. Then, there holds*

$$\frac{\|\tilde{f}(x) - f(x)\|_Y}{\|f(x)\|_Y} = \mathcal{O}(\kappa(x)\, \varepsilon_{\text{machine}}).$$

In particular, if $\kappa(x) = \mathcal{O}(1)$, then $\tilde{f}$ is accurate.

Proof. Using the definition (5.12) of backward stability and the definition (5.1) of $\kappa = \kappa(x)$, we find (with $\tilde{x}$ from (5.12))

$$\frac{\|\tilde{f}(x) - f(x)\|_Y}{\|f(x)\|_Y} = \frac{\|f(\tilde{x}) - f(x)\|_Y}{\|f(x)\|_Y} \leq (\kappa(x) + o(1))\frac{\|\tilde{x} - x\|_X}{\|x\|_X} = \mathcal{O}(\kappa(x)\, \varepsilon_{\text{machine}}).$$

(Here, the Landau notation $o(1)$ denotes a quantity converging to 0 as $\varepsilon_{\text{machine}} \searrow 0$.) □

Let us discuss some examples.

Example 5.3 (Stability of floating point arithmetic). The floating point operations $\oplus, \ominus, \otimes, \oslash$ are all backward stable. We prove this for $\oplus$ and leave the remaining operations as an exercise. The key for the stability analysis are (5.8) and (5.9). Let us consider the problem

$$f : \mathbb{R}^2 \to \mathbb{R}, \quad f(x_1, x_2) := x_1 + x_2,$$

and the algorithm

$$\tilde{f} : \mathbb{R}^2 \to \mathbb{R}, \quad \tilde{f}(x_1, x_2) := \mathrm{fl}(x_1) \oplus \mathrm{fl}(x_2).$$

We choose the 1-norm $\|\cdot\|_1$ on $\mathbb{R}^2$ and the absolute value $|\cdot|$ as norm on $\mathbb{R}$ (any other choices are fine as well by Theorem 5.5). Let $x = (x_1, x_2)^{\mathrm{T}} \in \mathbb{R}^2$. Then, by (5.8), we have $\mathrm{fl}(x_1) = x_1(1 + \varepsilon_1)$ and $\mathrm{fl}(x_2) = x_2(1 + \varepsilon_2)$ for some $\varepsilon_1, \varepsilon_2 \in [-\varepsilon_{\text{machine}}, \varepsilon_{\text{machine}}]$, and by (5.9), we have $\mathrm{fl}(x_1) \oplus \mathrm{fl}(x_2) = (\mathrm{fl}(x_1) + \mathrm{fl}(x_2))(1 + \varepsilon_3)$ for some $\varepsilon_3 \in [-\varepsilon_{\text{machine}}, \varepsilon_{\text{machine}}]$. Therefore, we find

$$\begin{aligned}\tilde{f}(x) = \mathrm{fl}(x_1) \oplus \mathrm{fl}(x_2) &= (\mathrm{fl}(x_1) + \mathrm{fl}(x_2))(1 + \varepsilon_3) \\ &= (x_1(1 + \varepsilon_1) + x_2(1 + \varepsilon_2))(1 + \varepsilon_3) \\ &= x_1(1 + \varepsilon_1)(1 + \varepsilon_3) + x_2(1 + \varepsilon_2)(1 + \varepsilon_3) = \tilde{x}_1 + \tilde{x}_2 = f(\tilde{x})\end{aligned}$$

with $\tilde{x}_1 = x_1(1 + \varepsilon_1)(1 + \varepsilon_3)$, $\tilde{x}_2 = x_2(1 + \varepsilon_2)(1 + \varepsilon_3)$ and $\tilde{x} = (\tilde{x}_1, \tilde{x}_2)^{\mathrm{T}}$. We have

$$\begin{aligned}|\tilde{x}_1 - x_1| &= |\varepsilon_1 + \varepsilon_3 + \varepsilon_1\varepsilon_3|\,|x_1| \le (|\varepsilon_1| + |\varepsilon_3| + |\varepsilon_1|\,|\varepsilon_3|)|x_1| \le C(\varepsilon_{\text{machine}})|x_1|, \\ |\tilde{x}_2 - x_2| &= |\varepsilon_2 + \varepsilon_3 + \varepsilon_2\varepsilon_3|\,|x_2| \le (|\varepsilon_2| + |\varepsilon_3| + |\varepsilon_2|\,|\varepsilon_3|)|x_2| \le C(\varepsilon_{\text{machine}})|x_2|,\end{aligned}$$

with $C(\varepsilon_{\text{machine}}) := 2\varepsilon_{\text{machine}} + \varepsilon_{\text{machine}}^2$, and hence,

$$\|\tilde{x} - x\|_1 = |\tilde{x}_1 - x_1| + |\tilde{x}_2 - x_2| \le C(\varepsilon_{\text{machine}})(|x_1| + |x_2|) = C(\varepsilon_{\text{machine}})\|x\|_1.$$

Since $C(\varepsilon_{\text{machine}}) = 2\varepsilon_{\text{machine}} + \varepsilon_{\text{machine}}^2 = \mathcal{O}(\varepsilon_{\text{machine}})$, it follows that $\tilde{f}$ is backward stable.

Example 5.4 (Stability of adding 1). Let us consider the problem $f : \mathbb{R} \to \mathbb{R}$, $f(x) := x + 1$, and the algorithm $\tilde{f} : \mathbb{R} \to \mathbb{R}$, $\tilde{f}(x) := \mathrm{fl}(x) \oplus 1$. Then, $\tilde{f}$ is stable but not backward stable. Stability can be shown as follows: We choose the absolute value $|\cdot|$ as norm on $\mathbb{R}$. For $x \in \mathbb{R}$ set $\tilde{x} = \mathrm{fl}(x)$ so that we have $|\tilde{x} - x| \le \varepsilon_{\text{machine}}|x|$ and

$$\begin{aligned}|\tilde{f}(x) - f(\tilde{x})| = |(\mathrm{fl}(x) \oplus 1) - (\tilde{x} + 1)| &= |(\tilde{x} \oplus 1) - (\tilde{x} + 1)| \\ &\le \varepsilon_{\text{machine}}|\tilde{x} + 1| = \varepsilon_{\text{machine}}|f(\tilde{x})|.\end{aligned}$$

It follows that $\tilde{f}$ is stable. We leave it as an exercise to show that $\tilde{f}$ is not backward stable (hint: note that $x \oplus 1 = 1$ for all $x \in F$ with $|x| \le \frac{1}{\beta}\varepsilon_{\text{machine}}$).

Example 5.5 (Stability of computing inner and outer product). Examples without proof:

(i) Inner product: Consider the problem $f : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, $f(x, y) := x^{\mathrm{T}}y$. Then, the algorithm

$$\begin{aligned}&\tilde{f} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}, \quad \tilde{f}(x, y) := \\ &= [[[(\mathrm{fl}(x_1) \otimes \mathrm{fl}(y_1)) \oplus (\mathrm{fl}(x_2) \otimes \mathrm{fl}(y_2))] \oplus (\mathrm{fl}(x_3) \otimes \mathrm{fl}(y_3))] \oplus \ldots] \oplus (\mathrm{fl}(x_n) \otimes \mathrm{fl}(y_n))\end{aligned}$$

is backward stable.

(ii) Outer product: Consider the problem $f : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^{m\times n}$, $f(x, y) := xy^{\mathrm{T}}$. Then, the algorithm

$$\tilde{f} : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^{m\times n}, \quad \tilde{f}(x, y) := \begin{pmatrix} \mathrm{fl}(x_1) \otimes \mathrm{fl}(y_1) & \cdots & \mathrm{fl}(x_1) \otimes \mathrm{fl}(y_n) \\ \vdots & & \vdots \\ \mathrm{fl}(x_m) \otimes \mathrm{fl}(y_1) & \cdots & \mathrm{fl}(x_m) \otimes \mathrm{fl}(y_n) \end{pmatrix}$$

is stable, but not backward stable.

Example 5.6 (Unstable algorithm for computing eigenvalues)**.** Consider the following algorithm for computing eigenvalues of a matrix $A \in \mathbb{R}^{n\times n}$: First, find the coefficients of the characteristic polynomial (i.e., $\lambda \mapsto \det(\lambda I_n - A)$) and then, find its roots. This algorithm is unstable (hence, we do not use this algorithm in practice). Note that for e.g. $A = I_2 \in \mathbb{R}^{2\times 2}$ we have the characteristic polynomial p_1 from Example 5.1(iii). When we compute the characteristic polynomial, we will have errors of order $\mathcal{O}(\varepsilon_{\text{machine}})$, leading to errors in the roots of order $\mathcal{O}(\sqrt{\varepsilon_{\text{machine}}})$. In IEEE double precision arithmetic, this means a loss of eight digits of accuracy.

5.4 Stability of solution algorithms for linear systems

We discuss the stability of several solution algorithms for linear systems.

Solving linear systems via QR obtained from Householder triangularization

Let us analyze the following solution algorithm for linear systems in view of numerical stability:

Algorithm 5.1 (Solving linear systems via QR factorization)**.** Given an invertible matrix $A \in \mathbb{R}^{n\times n}$ and a vector $b \in \mathbb{R}^n$, do the following to obtain the solution $x \in \mathbb{R}^n$ to $Ax = b$.

Step 1) Use Algorithm 3.4 to obtain the factor $R \in \mathbb{R}^{n\times n}$ of a QR factorization $A = QR$, and the reflection vectors $v_1, \dots, v_n \in \mathbb{R}^n$ (the matrix Q is not explicitly formed).

Step 2) Use Algorithm 3.5 to compute $y := Q^{\mathrm{T}} b \in \mathbb{R}^n$ from the vectors $v_1, \dots, v_n$ and b.

Step 3) Solve the upper-triangular system $Rx = y$ by backward substitution.

The main result is the following:

Theorem 5.7 (Backward stability of Algorithm 5.1)**.** *Algorithm 5.1 is backward stable in the sense that*

$$(A + \Delta A)\tilde{x} = b \quad \textit{for some } \Delta A \in \mathbb{R}^{n\times n} \textit{ with } \frac{\|\Delta A\|}{\|A\|} = \mathcal{O}(\varepsilon_{\text{machine}})$$

for all matrix norms $\|\cdot\|$ on $\mathbb{R}^{n\times n}$, where $\tilde{x} \in \mathbb{R}^n$ is the solution computed by Algorithm 5.1. In particular, in view of Theorem 5.6 and Theorem 5.2, we have

$$\frac{\|\tilde{x} - x\|_{(n)}}{\|x\|_{(n)}} = \mathcal{O}(\kappa_{\|\cdot\|_{(n,n)}}(A)\, \varepsilon_{\text{machine}})$$

for any norm $\|\cdot\|_{(n)}$ on $\mathbb{R}^n$ with corresponding induced matrix norm $\|\cdot\|_{(n,n)}$ on $\mathbb{R}^{n\times n}$, where $x = A^{-1}b \in \mathbb{R}^n$ denotes the exact solution to $Ax = b$.

Remark 5.14. The vector $b \in \mathbb{R}^n$ is considered fixed and the problem is $f : A \mapsto A^{-1}b$ for $A \in \mathbb{R}^{n\times n}$ invertible, i.e., obtain the solution x to $Ax = b$ from A. The algorithm is $\tilde{f} : A \mapsto \tilde{f}(A)$ for $A \in \mathbb{R}^{n\times n}$ invertible with $\tilde{x} = \tilde{f}(A)$ being the output of Algorithm 5.1 with input A.

It can be shown that each step of Algorithm 5.1 is backward stable (we only state the results and omit the proofs): For Step 1, we have the following result.

Theorem 5.8 (Backward stability of QR via Householder)**.** *Suppose we apply Algorithm 3.4 to an invertible matrix $A \in \mathbb{R}^{n\times n}$, leading to outputs $\tilde{R} \in \mathbb{R}^{n\times n}$ and $\tilde{v}_1, \dots, \tilde{v}_n \in \mathbb{R}^n$ (the computed factor R and reflection vectors v_i in floating point computation). Writing $\tilde{Q} := \tilde{Q}_1\tilde{Q}_2 \dots \tilde{Q}_n$ with $\tilde{Q}_i$ denoting the orthogonal matrix from Section 3.5 corresponding to the reflection vector $\tilde{v}_i$, there holds*

$$\tilde{Q}\tilde{R} = A + \Delta A \quad \textit{for some } \Delta A \in \mathbb{R}^{n\times n} \textit{ with } \frac{\|\Delta A\|}{\|A\|} = \mathcal{O}(\varepsilon_{\text{machine}}) \tag{5.13}$$

for all matrix norms $\|\cdot\|$ on $\mathbb{R}^{n\times n}$.

For Step 2, i.e., the computation of $y = \tilde{Q}^{\mathrm{T}}b = \tilde{Q}^{-1}b$, we have that the computed result $\tilde{y}$ satisfies

$$(\tilde{Q} + \Delta Q)\tilde{y} = b \quad \text{for some } \Delta Q \in \mathbb{R}^{n\times n} \text{ with } \|\Delta Q\| = \mathcal{O}(\varepsilon_{\text{machine}}) \tag{5.14}$$

for all matrix norms $\|\cdot\|$ on $\mathbb{R}^{n\times n}$. For Step 3, i.e., the solution of the upper-triangular system $\tilde{R}x = \tilde{y}$ by backward substitution, we have that the computed result $\tilde{x}$ satisfies

$$(\tilde{R} + \Delta R)\tilde{x} = \tilde{y} \quad \text{for some } \Delta R \in \mathbb{R}^{n\times n} \text{ with } \frac{\|\Delta R\|}{\|\tilde{R}\|} = \mathcal{O}(\varepsilon_{\text{machine}}) \tag{5.15}$$

for all matrix norms $\|\cdot\|$ on $\mathbb{R}^{n\times n}$. Now, we can prove Theorem 5.7.

Proof of Theorem 5.7. Exercise. Use (5.13), (5.14) and (5.15). □

Solving linear systems via Gaussian elimination

Let us first consider the solution of non-singular linear systems $Ax = b$ via LU factorization (Gaussian elimination without pivoting; see Algorithm 4.1 and Remark 4.2) and via PA=LU factorization (Gaussian elimination with partial pivoting; see Algorithm 4.2 and Remark 4.5).

Theorem 5.9. *We have the following results.*

(i) Gaussian elimination without pivoting: Suppose a LU factorization $A = LU$ of an invertible matrix $A \in \mathbb{R}^{n\times n}$, for which there exists a LU factorization, is computed by Algorithm 4.1. Then, for sufficiently small values of $\varepsilon_{\text{machine}}$, no zero-pivots arise and the algorithm completes successfully in floating point arithmetic, and for the computed $\tilde{L}$ and $\tilde{U}$ there holds

$$\tilde{L}\tilde{U} = A + \Delta A \quad \textit{for some } \Delta A \in \mathbb{R}^{n\times n} \textit{ with } \frac{\|\Delta A\|}{\|L\|\|U\|} = \mathcal{O}(\varepsilon_{\text{machine}}) \tag{5.16}$$

for all matrix norms $\|\cdot\|$ on $\mathbb{R}^{n\times n}$.

(ii) Gaussian elimination with partial pivoting: Suppose a PA=LU factorization $PA = LU$ of an invertible matrix $A \in \mathbb{R}^{n\times n}$ is computed by Algorithm 4.2. Then, for the computed $\tilde{P}$, $\tilde{L}$, and $\tilde{U}$ there holds

$$\tilde{L}\tilde{U} = \tilde{P}A + \Delta A \quad \textit{for some } \Delta A \in \mathbb{R}^{n\times n} \textit{ with } \frac{\|\Delta A\|}{\|A\|} = \mathcal{O}(\rho\, \varepsilon_{\text{machine}})$$

for all matrix norms $\|\cdot\|$ on $\mathbb{R}^{n\times n}$, where ρ denotes the growth factor of A defined by

$$\rho := \frac{\max_{i,j\in\{1,\dots,n\}}|u_{ij}|}{\max_{i,j\in\{1,\dots,n\}}|a_{ij}|}.$$

Further, if $|l_{ij}| < 1$ for all $i > j$, then $\tilde{P} = P$ for $\varepsilon_{\text{machine}}$ sufficiently small.

We omit the proof, but discuss the implications.

Remark 5.15. Let us discuss the result (i) of Theorem 5.9. Although it looks similar to other stability results, this is very different in that the quantity $\|L\|\|U\|$ appears instead of $\|A\|$ in the denominator of (5.16). Hence, we will have backward stability if $\|L\|\|U\| = \mathcal{O}(\|A\|)$. Otherwise, backward instability is to be expected. It is known that both L and U can be unboundedly large and that Gaussian elimination without pivoting is unstable, and hence, should not be used in general. We give a simple example illustrating the problem: Consider $A := \begin{pmatrix} 10^{-20} & 1 \\ 1 & 1 \end{pmatrix}$, for which Gaussian elimination performed exactly gives

$$A = LU, \qquad L := \begin{pmatrix} 1 & 0 \\ 10^{20} & 1 \end{pmatrix}, \quad U := \begin{pmatrix} 10^{-20} & 1 \\ 0 & 1 - 10^{20} \end{pmatrix}.$$

In IEEE double precision arithmetic, the computed result would be

$$\tilde{L} := \begin{pmatrix} 1 & 0 \\ 10^{20} & 1 \end{pmatrix}, \quad \tilde{U} := \begin{pmatrix} 10^{-20} & 1 \\ 0 & -10^{20} \end{pmatrix}$$

and we note that $\tilde{L}\tilde{U} = \begin{pmatrix} 10^{-20} & 1 \\ 1 & 0 \end{pmatrix}$ which is drastically different from $LU = A$ in the (2,2) entry. Considering $Ax = b := (1,0)^{\mathrm{T}}$ with exact solution $x \approx (-1,1)^{\mathrm{T}}$, we find from $\tilde{L}\tilde{U}\tilde{x} = b$ that $\tilde{x} = (0,1)^{\mathrm{T}}$ which is very different from the exact solution.

Remark 5.16. Let us discuss the result (ii) of Theorem 5.9. It can be shown that the growth factor ρ satisfies the bound $\rho \leq 2^{n-1}$ and that this is sharp. It is attained by the matrix $A = (a_{ij}) \in \mathbb{R}^{n\times n}$ with $a_{ii} = a_{in} = 1$ for all $1 \leq i \leq n$, $a_{ij} = -1$ for all $i > j$, and $a_{ij} = 0$ otherwise (exercise). A growth factor of 2^n means a loss of around n bits of precision, which is a huge problem for high-dimensional problems (as they arise in practice). Still, according to our definition, Gaussian elimination with partial pivoting is backward stable (as dependence of the constant on the dimension is allowed). However, we should rather think of it as stable for most problems, but very unstable for certain matrices. In practice, for problems with real applications studied in the past centuries, Gaussian elimination with partial pivoting performed in a stable way.

Solving linear systems via Cholesky factorization

Cholesky factorization is the method of choice for linear systems with a symmetric positive definite system matrix as it is always stable. To be precise, let us consider the following algorithm:

Algorithm 5.2 (Solving linear systems via Cholesky factorization)**.** Given a symmetric positive definite matrix $A \in \mathbb{R}^{n\times n}$ and a vector $b \in \mathbb{R}^n$, do the following to obtain the solution $x \in \mathbb{R}^n$ to $Ax = b$.

Step 1) Use Algorithm 4.3 to obtain the factor $R \in \mathbb{R}^{n\times n}$ of the Cholesky factorization $A = R^{\mathrm{T}}R$.

Step 2) Solve the lower-triangular system $R^{\mathrm{T}}y = b$ for $y \in \mathbb{R}^n$ by forward substitution.

Step 3) Solve the upper-triangular system $Rx = y$ for $x \in \mathbb{R}^n$ by backward substitution.

The main result is the following: (proof omitted)

Theorem 5.10 (Backward stability of Cholesky factorization and of Algorithm 5.2)**.** *We have the following results:*

(i) Backward stability of Cholesky factorization: Suppose we apply Algorithm 4.3 to a symmetric positive definite matrix $A \in \mathbb{R}^{n\times n}$, leading to an output $\tilde{R} \in \mathbb{R}^{n\times n}$ (the computed factor R in floating point computation). Then, there holds

$$\tilde{R}^{\mathrm{T}}\tilde{R} = A + \Delta A \quad \textit{for some } \Delta A \in \mathbb{R}^{n\times n} \textit{ with } \frac{\|\Delta A\|}{\|A\|} = \mathcal{O}(\varepsilon_{\text{machine}})$$

for all matrix norms $\|\cdot\|$ on $\mathbb{R}^{n\times n}$.

(ii) Backward stability of Algorithm 5.2: Algorithm 5.2 is backward stable in the sense that

$$(A + \Delta A)\tilde{x} = b \quad \textit{for some } \Delta A \in \mathbb{R}^{n\times n} \textit{ with } \frac{\|\Delta A\|}{\|A\|} = \mathcal{O}(\varepsilon_{\text{machine}})$$

for all matrix norms $\|\cdot\|$ on $\mathbb{R}^{n\times n}$, where $\tilde{x} \in \mathbb{R}^n$ is the solution computed by Algorithm 5.2.

Remark 5.17. An intuitive reason for the stability of Cholesky factorization, compared to LU factorization, is that the factor R in the Cholesky factorization $A = R^{\mathrm{T}}R$ cannot become very large compared to A (e.g., we have $\|R\|_2 = \|R^{\mathrm{T}}\|_2 = \sqrt{\|A\|_2}$ (exercise)).

5.5 Stability of solution algorithms for least squares problems

Omitted. (If you are interested in this, see Trefethen–Bau Chapter 3.)

6 Eigenvalue Problems

6.1 The eigenvalue problem: the basics

We study the eigenvalue problem corresponding to a square matrix $A \in \mathbb{C}^{n\times n}$:

$$\text{Find } x \in \mathbb{C}^n\backslash\{0\} \text{ and } \lambda \in \mathbb{C} \text{ such that } Ax = \lambda x.$$

We write $\mathbb{C}^{m\times n}$ for the set of complex $m \times n$ matrices, and $\mathbb{C}^m := \mathbb{C}^{m\times 1}$ for the set of complex column m-vectors. For $A = (a_{ij}) \in \mathbb{C}^{m\times n}$ write $\bar{A} := (\overline{a_{ij}}) \in \mathbb{C}^{m\times n}$ (complex conjugate each entry), and denote the *adjoint* (*conjugate transpose*) by $A^* := \overline{A^{\mathrm{T}}} \in \mathbb{C}^{n\times m}$. We introduce three important classes of square matrices:

$A \in \mathbb{C}^{n\times n}$ is called *hermitian* iff $A^* = A$, (if A real: *hermitian* $\Leftrightarrow$ *symmetric*)

$A \in \mathbb{C}^{n\times n}$ is called *normal* iff $A^*A = AA^*$,

$A \in \mathbb{C}^{n\times n}$ is called *unitary* iff $A^*A = AA^* = I_n$. (if A real: *unitary* $\Leftrightarrow$ *orthogonal*)

We recall the basics for eigenvalue problems: For a square matrix $A \in \mathbb{C}^{n\times n}$,

- $\lambda \in \mathbb{C}$ is called an *eigenvalue* of A iff there holds $Ax = \lambda x$ for some $x \in \mathbb{C}^n\backslash\{0\}$. Then, any $x \in \mathbb{C}^n\backslash\{0\}$ with $Ax = \lambda x$ is called an *eigenvector* of A corresponding to the eigenvalue λ.

  Remark 6.1. If A is hermitian, then all of its eigenvalues are real. (exercise)

- its *characteristic polynomial* p_A is defined as $p_A : \mathbb{C} \to \mathbb{C}$, $z \mapsto \det(zI_n - A)$.

- its *spectrum* $\Lambda(A) \subseteq \mathbb{C}$ is defined by $\Lambda(A) := \{\lambda \in \mathbb{C} : \lambda \text{ is an eigenvalue of } A\}$, and its *spectral radius* $\rho(A) \in [0, \infty)$ is defined by $\rho(A) := \max\{|\lambda| : \lambda \in \Lambda(A)\}$.

  Remark 6.2. There holds $\Lambda(A) = \{\lambda \in \mathbb{C} : p_A(\lambda) = 0\}$. Indeed, $\lambda \in \mathbb{C}$ satisfies $\lambda \in \Lambda(A)$ iff $(\exists x \in \mathbb{C}^n\backslash\{0\} : (\lambda I_n - A)x = 0)$ iff $(\lambda I_n - A$ is singular$)$ iff $\det(\lambda I_n - A) = 0$.

  Remark 6.3. Note that $p_A(z) = \sum_{k=0}^n p_k z^k$ for some $p_0, \dots, p_{n-1} \in \mathbb{C}$ and $p_n = 1$, i.e., p_A is a monic polynomial. Hence, by the fundamental theorem of algebra, there exist $\lambda_1, \dots, \lambda_n \in \mathbb{C}$ such that $p_A(z) = \prod_{i=1}^n (z - \lambda_i)$, and thus $\Lambda(A) = \{\lambda_1, \dots, \lambda_n\}$. Note that $\det(A) = (-1)^n p_A(0) = \prod_{i=1}^n \lambda_i$ and, comparing the coefficient of z^{n-1} in $\det(zI_n - A) = \prod_{i=1}^n (z - \lambda_i)$, that $\mathrm{tr}(A) = \sum_{i=1}^n \lambda_i$.

- the *algebraic multiplicity* $\mu_A(\lambda) \in \{1, \dots, n\}$ of an eigenvalue $\lambda \in \Lambda(A)$ is the multiplicity of λ as a root of p_A. We call $\lambda \in \Lambda(A)$ with $\mu_A(\lambda) = 1$ a *simple eigenvalue.*

- the *eigenspace* $E_\lambda \subseteq \mathbb{C}^n$ of an eigenvalue $\lambda \in \Lambda(A)$ is defined to be $E_\lambda := \mathcal{N}(\lambda I_n - A)$. We call $\gamma_A(\lambda) := \dim(E_\lambda) \in \{1, \dots, n\}$ the *geometric multiplicity* of $\lambda \in \Lambda(A)$.

  Remark 6.4. There holds $\gamma_A(\lambda) \le \mu_A(\lambda)$ for any $\lambda \in \Lambda(A)$. We omit the proof; see undergraduate linear algebra.

- an eigenvalue $\lambda \in \Lambda(A)$ is called *defective* iff $\gamma_A(\lambda) < \mu_A(\lambda)$. A matrix $A \in \mathbb{C}^{n\times n}$ is called defective iff it has a defective eigenvalue.

- For an invertible matrix $X \in \mathbb{C}^{n\times n}$, the map $S_X : \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}$, $A \mapsto X^{-1}AX$ is called a *similarity transformation* of A. Further, a matrix $B \in \mathbb{C}^{n\times n}$ is called *similar* to $A \in \mathbb{C}^{n\times n}$ iff $\exists\, X \in \mathbb{C}^{n\times n}$ invertible s.t. $B = X^{-1}AX$.

Remark 6.5. If $B \in \mathbb{C}^{n\times n}$ is similar to $A \in \mathbb{C}^{n\times n}$, then $p_A = p_B$, $\Lambda(A) = \Lambda(B)$, and there holds $\mu_A(\lambda) = \mu_B(\lambda)$ and $\gamma_A(\lambda) = \gamma_B(\lambda)$ for all $\lambda \in \Lambda(A) = \Lambda(B)$.

A very useful result for estimating the location of eigenvalues in the complex plane is Gerschgorin's theorem. We denote the closed disc in the complex plane around a point $a \in \mathbb{C}$ with radius $r > 0$ by $D(a, r) := \{z \in \mathbb{C} : |z - a| \leq r\} \subseteq \mathbb{C}$.

Theorem 6.1 (Gerschgorin's theorem)**.** *Let $A = (a_{ij})_{1\leq i,j\leq n} \in \mathbb{C}^{n\times n}$. Define the numbers $r_1, \dots, r_n \geq 0$ given by*

$$r_i := \sum_{j\in\{1,\dots,n\}\backslash\{i\}} |a_{ij}|, \qquad i \in \{1, \dots, n\}.$$

Then, there holds

$$\Lambda(A) \subseteq \bigcup_{i=1}^{n} D(a_{ii}, r_i),$$

i.e., every eigenvalue of A lies in at least one of the n so-called Gerschgorin discs $D(a_{11}, r_1), \dots, D(a_{nn}, r_n)$. Moreover, if there are $1 \leq k \leq n$ Gerschgorin discs such that their union U is a connected set which is disjoint from the union of the remaining $n - k$ Gerschgorin discs, then U contains exactly k eigenvalues of A.

Proof. Let $\lambda \in \Lambda(A)$. We can find an eigenvector $x = (x_1, \dots, x_n)^{\mathrm{T}} \in \mathbb{C}^n\backslash\{0\}$ satisfying $Ax = \lambda x$ and $\|x\|_\infty = \max_{k\in\{1,\dots,n\}}|x_k| = 1$. Let $i \in \{1, \dots, n\}$ be such that $|x_i| = 1$. Then,

$$\begin{aligned} |\lambda - a_{ii}| = |(\lambda - a_{ii})x_i| &= |(Ax)_i - a_{ii}x_i| \\ &= \left|\sum_{j=1}^{n} a_{ij}x_j - a_{ii}x_i\right| = \left|\sum_{j\in\{1,\dots,n\}\backslash\{i\}} a_{ij}x_j\right| \leq r_i\|x\|_\infty = r_i, \end{aligned}$$

i.e., $\lambda \in D(a_{ii}, r_i)$. We conclude that every eigenvalue of A lies in at least one of the n Gerschgorin discs. We omit the proof of the second part of the theorem. □

Remark 6.6. Noting that $\Lambda(A) = \Lambda(A^{\mathrm{T}})$ for any $A \in \mathbb{C}^{n\times n}$ (observe $p_A = p_{A^{\mathrm{T}}}$), we can obtain additional information on $\Lambda(A)$ by applying Gerschgorin's theorem to A^{T} as well.

6.2 Eigenvalue-revealing factorizations

We start by discussing eigenvalue-revealing factorizations, i.e., factorizations of a given matrix from which we can directly read off its eigenvalues.

Diagonalization

Let us first discuss the eigenvalue decomposition, which, as the name suggests, is an eigenvalue-revealing decomposition.

Definition 6.1 (Eigenvalue decomposition)**.** Let $A \in \mathbb{C}^{n\times n}$. If there exists an invertible matrix $X \in \mathbb{C}^{n\times n}$ and a diagonal matrix $D \in \mathbb{C}^{n\times n}$ such that

$$A = XDX^{-1}, \tag{6.1}$$

then we call (6.1) an eigenvalue decomposition of A.

Definition 6.2. For a matrix $A \in \mathbb{C}^{n\times n}$,

(i) we say A is diagonalizable iff there exists an eigenvalue decomposition of A.

(ii) we say A is unitary diagonalizable iff there exists an eigenvalue decomposition (6.1) of A with X unitary, i.e., iff $\exists\, X \in \mathbb{C}^{n\times n}$ unitary, $D \in \mathbb{C}^{n\times n}$ diagonal: $A = XDX^*$.

Remark 6.7. Note that (6.1) is equivalent to $AX = XD$. Writing $X = (x_1|\dots|x_n)$ and $D = \operatorname{diag}_{n\times n}(\lambda_1, \dots, \lambda_n)$, this yields $Ax_i = \lambda_i x_i$ for $i \in \{1, \dots, n\}$, i.e., x_i is an eigenvector with corresponding eigenvalue λ_i. So, the eigenvalue decomposition is a eigenvalue-revealing decomposition as we can directly read off the eigenvalues from the diagonal of D.

Theorem 6.2 (Characterization of diagonalizable matrices)**.** *A matrix $A \in \mathbb{C}^{n\times n}$ is diagonalizable iff it is non-defective, i.e., iff $\gamma_A(\lambda) = \mu_A(\lambda)$ for all $\lambda \in \Lambda(A)$.*

Proof. First, suppose $A \in \mathbb{C}^{n\times n}$ has an eigenvalue decomposition $A = XDX^{-1}$ with some invertible matrix $X \in \mathbb{C}^{n\times n}$ and a diagonal matrix $D \in \mathbb{C}^{n\times n}$. Then, A is similar to D and hence, $\Lambda(A) = \Lambda(D) =: \Lambda$, and there holds $\mu_A(\lambda) = \mu_D(\lambda)$ and $\gamma_A(\lambda) = \gamma_D(\lambda)$ for all $\lambda \in \Lambda$. Since D is diagonal, we have $\gamma_D(\lambda) = \mu_D(\lambda)$ for all $\lambda \in \Lambda$ and hence, $\gamma_A(\lambda) = \gamma_D(\lambda) = \mu_D(\lambda) = \mu_A(\lambda)$ for all $\lambda \in \Lambda$, i.e., A is non-defective.

Conversely, suppose that $A \in \mathbb{C}^{n\times n}$ is non-defective. Denote its distinct eigenvalues by $\lambda_1, \dots, \lambda_k \in \Lambda(A)$, $k \le n$. Then, to each λ_i we can find $\gamma_A(\lambda_i)$ many linear independent eigenvectors of A. Noting that eigenvectors to distinct eigenvalues are linearly independent (exercise), we can find a total of $\sum_{i=1}^k \gamma_A(\lambda_i) = \sum_{i=1}^k \mu_A(\lambda_i) = n$ (first equality uses A non-defective) linearly independent eigenvectors $x_1, \dots, x_n \in \mathbb{C}^n\backslash\{0\}$ for A. Then, the matrix $X := (x_1|\dots|x_n) \in \mathbb{C}^{n\times n}$ is invertible and, setting $D := \operatorname{diag}_{n\times n}(d_1, \dots, d_n)$ with $d_1, \dots, d_n \in \mathbb{C}$ satisfying $Ax_i = d_i x_i$, there holds $AX = XD$ and hence $A = XDX^{-1}$. □

Theorem 6.3 (Characterization of unitary diagonalizable matrices)**.** *A matrix $A \in \mathbb{C}^{n\times n}$ is unitary diagonalizable iff it is normal, i.e., iff $A^*A = AA^*$. In particular, every hermitian matrix is unitary diagonalizable.*

Proof. Omitted. □

Remark 6.8. If $A \in \mathbb{R}^{n\times n}$ is symmetric, then there exists a real eigenvalue decomposition $A = XDX^{-1} = XDX^{\mathrm{T}}$ with $X \in \mathbb{R}^{n\times n}$ orthogonal and $D \in \mathbb{R}^{n\times n}$ diagonal. We thus call real symmetric matrices orthogonally diagonalizable. We omit the proof.

In the sense of the following definition, we thus have that any symmetric matrix is orthogonally equivalent to a diagonal matrix.

Definition 6.3. Two matrices $A, B \in \mathbb{R}^{n\times n}$ are called orthogonally equivalent iff there exists an orthogonal matrix $Q \in \mathbb{R}^{n\times n}$ such that $A = QBQ^{\mathrm{T}}$.

Schur factorization

The drawback of the eigenvalue decomposition is that it only exists for a certain class of matrices (non-defective matrices). We now introduce the most useful eigenvalue-revealing decomposition in numerical analysis.

Definition 6.4 (Schur factorization)**.** Let $A \in \mathbb{C}^{n\times n}$. If there exists a unitary matrix $Q \in \mathbb{C}^{n\times n}$ and an upper-triangular matrix $T \in \mathbb{C}^{n\times n}$ such that

$$A = QTQ^*, \tag{6.2}$$

then we call (6.2) a Schur factorization of A.

Remark 6.9. Suppose A has a Schur factorization $A = QTQ^*$. Then, A is similar to T and hence $\Lambda(A) = \Lambda(T)$. Hence, since the eigenvalues of the upper-triangular matrix T are its diagonal entries, we can read off the eigenvalues of A from the diagonal of T.

Theorem 6.4 (Existence of Schur factorization)**.** *Every matrix $A \in \mathbb{C}^{n\times n}$ has a Schur factorization.*

Proof. We use induction on $n \in \mathbb{N}$. For the case $n = 1$, i.e., $A = (a) \in \mathbb{C}^{1\times 1}$, we have that $A = (a) = (1)(a)(1) = I_1 A I_1^*$ is a Schur factorization of A. As induction hypothesis suppose the claim is true for some $n \in \mathbb{N}$.

For the induction step, let $A \in \mathbb{C}^{(n+1)\times(n+1)}$ and our goal is to construct a Schur factorization of A. Let $\lambda \in \Lambda(A)$ and $x \in \mathbb{C}^{n+1}\backslash\{0\}$ be a corresponding normalized eigenvector with $x^*x = 1$ and $Ax = \lambda x$. We can now find a unitary matrix $U = (u_1|\dots|u_n|u_{n+1}) \in \mathbb{C}^{(n+1)\times(n+1)}$ with first column $u_1 = x$. Then,

$$U^*AU = \left(\begin{array}{c|c} \lambda & w^* \\ \hline 0_{n\times 1} & B \end{array}\right) \in \mathbb{C}^{(n+1)\times(n+1)}$$

for some $w \in \mathbb{C}^n$ and $B \in \mathbb{C}^{n\times n}$. By the hypothesis there exists a Schur factorization of B, i.e., a unitary matrix $V \in \mathbb{C}^{n\times n}$ and an upper-triangular matrix $R \in \mathbb{C}^{n\times n}$ such that $B = VRV^*$. Then, we compute

$$\begin{aligned}\left[U\left(\begin{array}{c|c} 1 & 0_{1\times n} \\ \hline 0_{n\times 1} & V \end{array}\right)\right]^* A \left[U\left(\begin{array}{c|c} 1 & 0_{1\times n} \\ \hline 0_{n\times 1} & V \end{array}\right)\right] &= \left(\begin{array}{c|c} 1 & 0_{1\times n} \\ \hline 0_{n\times 1} & V^* \end{array}\right)\left(\begin{array}{c|c} \lambda & w^* \\ \hline 0_{n\times 1} & B \end{array}\right)\left(\begin{array}{c|c} 1 & 0_{1\times n} \\ \hline 0_{n\times 1} & V \end{array}\right) \\ &= \left(\begin{array}{c|c} \lambda & w^*V \\ \hline 0_{n\times 1} & R \end{array}\right) =: T \in \mathbb{C}^{(n+1)\times(n+1)},\end{aligned}$$

and we find that

$$A = QTQ^* \quad \text{with} \quad Q := U\left(\begin{array}{c|c} 1 & 0_{1\times n} \\ \hline 0_{n\times 1} & V \end{array}\right) \in \mathbb{C}^{(n+1)\times(n+1)}.$$

Noting that Q is unitary and T is upper-triangular, this is a Schur factorization of A. □

Remark 6.10. Note that if $A \in \mathbb{C}^{n\times n}$ is normal and $A = QTQ^*$ is a Schur factorization of A, then T must be diagonal. (exercise)

6.3 Transformation into upper-Hessenberg form

We now turn our attention to the construction of algorithms for computing the eigenvalues of a given matrix. Unfortunately, there does not exist an algorithm which can compute the eigenvalues of an arbitrary matrix in a finite number of steps and thus, any eigenvalue solver must be iterative. This can be seen as follows:

Remark 6.11 (Eigenvalue solvers must be iterative). Let $a := (a_0, \dots, a_{n-1})^{\mathrm{T}} \in \mathbb{C}^n$ and let $e_1, \dots, e_n \in \mathbb{R}^n$ denote the canonical basis vectors in $\mathbb{R}^n$. Observe that the problem of finding the roots of the monic polynomial $p : \mathbb{C} \to \mathbb{C}$, $p(z) = z^n + \sum_{i=0}^{n-1} a_i z^i$ is equivalent to finding the eigenvalues of the matrix

$$A := \big(B \,|\, -a\big) \in \mathbb{C}^{n\times n}, \quad \text{where} \quad B := (e_2|e_3|\cdots|e_n) \in \mathbb{R}^{n\times(n-1)}.$$

Indeed, denoting the roots of p by $z_1, \dots, z_n \in \mathbb{C}$, the vector $(1, z_i, z_i^2, \dots, z_i^{n-1})^{\mathrm{T}} \in \mathbb{C}^n$ is an eigenvector of A^{T} with eigenvalue z_i for $i \in \{1, \dots, n\}$. Hence, since $\Lambda(A) = \Lambda(A^{\mathrm{T}})$ (see Remark 6.6), we find that $\Lambda(A) = \{z_1, \dots, z_n\}$.

We deduce that, if there were an algorithm which can compute the exact eigenvalues of an arbitrary matrix in finite steps, we would have a formula for computing the roots of any arbitrary polynomial. However, this is impossible since it is known that no such formula exists for polynomials of degree greater than or equal to 5.

In view of this result, we will aim for algorithms that yield sequences converging to the eigenvalues (desirably as rapidly as possible). Although we cannot find a Schur factorization in a finite number of steps (i.e., we cannot transform a given matrix into an upper-triangular matrix via unitary similarity transformations), we can transform a given matrix into an "almost" triangular matrix (a so-called Hessenberg matrix) via unitary similarity transformations in a finite number of steps: (illustration for $n = 6$)

$$A = \begin{pmatrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \end{pmatrix} \quad \Longrightarrow \quad H = Q^*AQ = \begin{pmatrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & 0 & * & * & * & * \\ 0 & 0 & 0 & * & * & * \\ 0 & 0 & 0 & 0 & * & * \end{pmatrix}.$$

Definition 6.5 (upper-Hessenberg matrix)**.** A square matrix $A = (a_{ij}) \in \mathbb{C}^{n\times n}$ is called an upper-Hessenberg matrix iff $a_{ij} = 0$ whenever $i > j + 1$.

Definition 6.6 (Hessenberg decomposition)**.** Let $A \in \mathbb{C}^{n\times n}$. If there exist a unitary matrix $Q \in \mathbb{C}^{n\times n}$ and an upper-Hessenberg matrix $H \in \mathbb{C}^{n\times n}$ such that there holds

$$A = QHQ^*, \tag{6.3}$$

then we call (6.3) a Hessenberg decomposition of A.

Theorem 6.5 (Existence of Hessenberg decomposition)**.** *Any square matrix $A \in \mathbb{C}^{n\times n}$ has a Hessenberg decomposition. Moreover, if $A \in \mathbb{R}^{n\times n}$ is real, then there exists a Hessenberg decomposition $A = QHQ^{\mathrm{T}}$ with $Q \in \mathbb{R}^{n\times n}$ orthogonal and $H \in \mathbb{R}^{n\times n}$ upper-Hessenberg.*

Transformation into upper-Hessenberg form via unitary similarity transformations is typically the first phase of any eigenvalue algorithm. Let us explain how to obtain such a Hessenberg decomposition by looking at an explicit example.

Example 6.1. Consider the matrix $A := \begin{pmatrix} 1 & 1 & 0 & -1 & 0 \\ -2 & -1 & 1 & 1 & 0 \\ 1 & 1 & -1 & 1 & 0 \\ 2 & 1 & 1 & -1 & 0 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix}$. We explain how to find an orthogonal matrix $Q \in \mathbb{R}^{5\times 5}$ such that $Q^{\mathrm{T}}AQ$ is an upper-Hessenberg matrix.

Step 1: We want to find an orthogonal matrix $Q_1 \in \mathbb{R}^{5\times 5}$ such that $A_1 := Q_1^{\mathrm{T}} A Q_1$ is of the form $A_1 = \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & * & * & * & * \\ 0 & * & * & * & * \end{pmatrix}$. We take $Q_1^{\mathrm{T}} = Q_1$ to be a Householder reflector that leaves the first row unchanged and introduces the desired zeros. Set $x_1 := (-2, 1, 2, 0)^{\mathrm{T}}$ and $v_1 := \operatorname{sign}(\langle x_1, e_1\rangle)\|x_1\|_2 e_1 + x_1 = (-5, 1, 2, 0)^{\mathrm{T}}$, and take

$$Q_1 := \left(\begin{array}{c|c} 1 & 0_{1\times 4} \\ \hline 0_{4\times 1} & I_4 - 2\frac{v_1 v_1^{\mathrm{T}}}{\|v_1\|_2^2} \end{array}\right) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & -\frac{2}{3} & \frac{1}{3} & \frac{2}{3} & 0 \\ 0 & \frac{1}{3} & \frac{14}{15} & -\frac{2}{15} & 0 \\ 0 & \frac{2}{3} & -\frac{2}{15} & \frac{11}{15} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Then, $Q_1 A = Q_1^{\mathrm{T}} A$ has the desired zero-entries in its first column, and so does $Q_1^{\mathrm{T}} A Q_1$ (right-multiplication by Q_1 leaves first column unchanged). Indeed, we have

$$A_1 := Q_1^{\mathrm{T}} A Q_1 = \begin{pmatrix} 1 & -\frac{4}{3} & * & * & * \\ 3 & -\frac{17}{9} & * & * & * \\ 0 & \frac{17}{45} & * & * & * \\ 0 & \frac{19}{45} & * & * & * \\ 0 & \frac{1}{3} & * & * & * \end{pmatrix}.$$

Step 2: We want to find an orthogonal matrix $Q_2 \in \mathbb{R}^{5\times 5}$ such that $A_2 := Q_2^{\mathrm{T}} A_1 Q_2$ is of the form $A_2 = \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & * & * & * \end{pmatrix}$. We take $Q_2^{\mathrm{T}} = Q_2$ to be a Householder reflector that leaves the first two rows unchanged and introduces the desired zeros. Set $x_2 := (\frac{17}{45}, \frac{19}{45}, \frac{1}{3})^{\mathrm{T}}$ and $v_2 := \operatorname{sign}(\langle x_2, e_1\rangle)\|x_2\|_2 e_1 + x_2 = \frac{1}{45}(17 + 5\sqrt{35}, 19, 15)^{\mathrm{T}}$, and take

$$Q_2 := \left(\begin{array}{c|c} I_2 & 0_{2\times 3} \\ \hline 0_{3\times 2} & I_3 - 2\frac{v_2 v_2^{\mathrm{T}}}{\|v_2\|_2^2} \end{array}\right) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\frac{17}{5\sqrt{35}} & -\frac{19}{5\sqrt{35}} & -\frac{3}{\sqrt{35}} \\ 0 & 0 & -\frac{19}{5\sqrt{35}} & \frac{39375+6137\sqrt{35}}{102550} & -\frac{9975-969\sqrt{35}}{20510} \\ 0 & 0 & -\frac{3}{\sqrt{35}} & -\frac{9975-969\sqrt{35}}{20510} & \frac{2527+153\sqrt{35}}{4102} \end{pmatrix}.$$

Then, $Q_2 A_1 = Q_2^{\mathrm{T}} A_1$ has the desired zero-entries in its second column, and so does $Q_2^{\mathrm{T}} A_1 Q_2$:

$$A_2 := Q_2^{\mathrm{T}} A_1 Q_2 = \begin{pmatrix} 1 & -\frac{4}{3} & -\frac{4}{3\sqrt{35}} & * & * \\ 3 & -\frac{17}{9} & -\frac{26}{9\sqrt{35}} & * & * \\ 0 & -\frac{\sqrt{35}}{9} & \frac{523}{315} & * & * \\ 0 & 0 & \frac{2565\sqrt{35}-8721}{20510} & * & * \\ 0 & 0 & -\frac{6885+3249\sqrt{35}}{20510} & * & * \end{pmatrix}.$$

Step 3: We want to find an orthogonal matrix $Q_3 \in \mathbb{R}^{5\times 5}$ such that $A_3 := Q_3^{\mathrm{T}} A_2 Q_3$ is of the form $A_3 = \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix}$. We take $Q_3^{\mathrm{T}} = Q_3$ to be a Householder reflector that leaves the first three rows unchanged and introduces the desired zeros. Set $x_3 := (\frac{2565\sqrt{35}-8721}{20510}, -\frac{6885+3249\sqrt{35}}{20510})^{\mathrm{T}}$ and $v_3 := \operatorname{sign}(\langle x_3, e_1\rangle)\|x_3\|_2 e_1 + x_3$, and take

$$Q_3 := \left(\begin{array}{c|c} I_3 & 0_{3\times 2} \\ \hline 0_{2\times 3} & I_2 - 2\frac{v_3 v_3^{\mathrm{T}}}{\|v_3\|_2^2} \end{array}\right) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -\frac{285\sqrt{910}-969\sqrt{26}}{15236} & \frac{765\sqrt{26}+361\sqrt{910}}{15236} \\ 0 & 0 & 0 & \frac{765\sqrt{26}+361\sqrt{910}}{15236} & \frac{285\sqrt{910}-969\sqrt{26}}{15236} \end{pmatrix}.$$

Then, $Q_3 A_2 = Q_3^{\mathrm{T}} A_2$ has the desired zero-entry in its third column, and so does $Q_3^{\mathrm{T}} A_2 Q_3$:

$$Q_3^{\mathrm{T}} A_2 Q_3 = \begin{pmatrix} 1 & -\frac{4}{3} & -\frac{4}{3\sqrt{35}} & -\frac{4}{\sqrt{910}} & -\frac{2}{\sqrt{26}} \\ 3 & -\frac{17}{9} & -\frac{26}{9\sqrt{35}} & -\frac{\sqrt{910}}{105} & 0 \\ 0 & -\frac{\sqrt{35}}{9} & \frac{523}{315} & \frac{8\sqrt{26}}{105} & 0 \\ 0 & 0 & -\frac{9\sqrt{26}}{35} & \frac{8}{35} & 0 \\ 0 & 0 & 0 & 0 & -2 \end{pmatrix} =: H.$$

This is in upper-Hessenberg form. We find that $A = QHQ^{\mathrm{T}}$ with H as above and

$$Q := Q_1 Q_2 Q_3 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & -\frac{2}{3} & -\frac{11}{3\sqrt{35}} & -\frac{11}{\sqrt{910}} & \frac{1}{\sqrt{26}} \\ 0 & \frac{1}{3} & -\frac{8}{3\sqrt{35}} & -\frac{8}{\sqrt{910}} & -\frac{4}{\sqrt{26}} \\ 0 & \frac{2}{3} & -\frac{7}{3\sqrt{35}} & -\frac{7}{\sqrt{910}} & \frac{3}{\sqrt{26}} \\ 0 & 0 & -\frac{3}{\sqrt{35}} & \frac{26}{\sqrt{910}} & 0 \end{pmatrix}$$

is a Hessenberg decomposition of A (note Q is orthogonal as a product of orthogonal matrices).

Using this methodology, any arbitrary square matrix $A \in \mathbb{C}^{n\times n}$ can be transformed into upper-Hessenberg form via unitary similarity transformations in (at most) $n-2$ steps. We are now able to find a Hessenberg decomposition to any given square matrix.

Remark 6.12 (Non-uniqueness of Hessenberg decomposition). The Hessenberg decomposition is not unique. Consider, e.g., a 2×2 matrix $A \in \mathbb{C}^{2\times 2}$. Then, for any unitary $Q \in \mathbb{C}^{2\times 2}$, we have that $A = Q(Q^*AQ)Q^*$ is a Hessenberg decomposition of A (note $Q^*AQ \in \mathbb{C}^{2\times 2}$ is upper-Hessenberg as any 2×2 matrix is upper-Hessenberg).

Remark 6.13 (Hessenberg decomposition of hermitian matrices). Let $A \in \mathbb{C}^{n\times n}$ be hermitian, and let $A = QHQ^*$ be a Hessenberg decomposition of A. Then, $H^* = (Q^*AQ)^* = Q^*A^*Q = Q^*AQ = H$, i.e., H is a hermitian matrix in upper-Hessenberg form and thus, H must be tridiagonal. Therefore, we can transform any hermitian matrix via unitary similarity transformations into a hermitian tridiagonal matrix, and any real symmetric matrix via orthogonal similarity transformations into a symmetric tridiagonal matrix: (illustration for $n = 6$)

$$A = \begin{pmatrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \end{pmatrix}, \quad A^* = A \quad \Longrightarrow \quad H = Q^*AQ = \begin{pmatrix} * & * & 0 & 0 & 0 & 0 \\ * & * & * & 0 & 0 & 0 \\ 0 & * & * & * & 0 & 0 \\ 0 & 0 & * & * & * & 0 \\ 0 & 0 & 0 & * & * & * \\ 0 & 0 & 0 & 0 & * & * \end{pmatrix}.$$

To summarize, we are now able to transform any square matrix $A \in \mathbb{C}^{n\times n}$ via unitary similarity transformations into upper-Hessenberg form $H = Q^*AQ$, and if A is hermitian, the resulting Hessenberg matrix is actually a hermitian tridiagonal matrix. Recall that similarity transformations do not change the spectrum of the matrix and hence, $\Lambda(A) = \Lambda(H)$. This "reduction" to upper-Hessenberg form is typically the first step of eigenvalue algorithms. Let us provide the following algorithm:

Algorithm 6.1 (Transformation into upper-Hessenberg form)**.** Let $A \in \mathbb{R}^{n\times n}$. To obtain the factor H of a Hessenberg decomposition $A = QHQ^{\mathrm{T}}$, do as follows:

```
for $i = 1, \dots, n-2$ do
    $x = A_{i+1:n,i}$
    $v_i = \mathrm{sign}(\langle x, e_1\rangle)\|x\|_2 e_1 + x$
    $v_i = \frac{v_i}{\|v_i\|_2}$
    $A_{i+1:n,i:n} = A_{i+1:n,i:n} - 2v_i\left(v_i^{\mathrm{T}} A_{i+1:n,i:n}\right)$
    $A_{1:n,i+1:n} = A_{1:n,i+1:n} - 2\left(A_{1:n,i+1:n}v_i\right)v_i^{\mathrm{T}}$
end for.
```

The algorithm stores the result H in place of A. Note that Q is not explicitly formed, but can be obtained from the vectors $v_1, \dots, v_{n-2}$, if desired, analogously to Section 3.5.

Remark 6.14. The above algorithm works for complex matrices as well. Note $\mathrm{sign}(z) := \frac{z}{|z|}$ for $z \in \mathbb{C}$, $\langle x, y\rangle := y^*x$ for $x, y \in \mathbb{C}^n$, and $\|x\|_2 := \sqrt{x^*x}$ for $x \in \mathbb{C}^n$.

Theorem 6.6. *Algorithm 6.1 requires* $\sim \frac{10}{3}n^3$ *flops.*

Proof. Omitted. □

Remark 6.15. If $A \in \mathbb{R}^{n\times n}$ is symmetric, clever modifications of Algorithm 6.1 are used in practice to transform into tridiagonal form (recall Remark 6.13) using only $\sim \frac{4}{3}n^3$ flops.

Theorem 6.7 (Backward stability of Hessenberg via Householder)**.** *Suppose we apply Algorithm 6.1 to a matrix* $A \in \mathbb{R}^{n\times n}$*, leading to outputs* $\tilde{H} \in \mathbb{R}^{n\times n}$ *and* $\tilde{v}_1, \dots, \tilde{v}_n \in \mathbb{R}^n$ *(the computed factor* H *and reflection vectors* v_i *in floating point computation). Writing* $\tilde{Q} := \tilde{Q}_1\tilde{Q}_2 \dots \tilde{Q}_{n-2}$ *with* $\tilde{Q}_i$ *denoting the orthogonal matrix corresponding to the reflection vector* $\tilde{v}_i$*, there holds*

$$\tilde{Q}\tilde{H}\tilde{Q}^{\mathrm{T}} = A + \Delta A \quad \textit{for some } \Delta A \in \mathbb{R}^{n\times n} \textit{ with } \frac{\|\Delta A\|}{\|A\|} = \mathcal{O}(\varepsilon_{\mathrm{machine}})$$

for all matrix norms $\|\cdot\|$ *on* $\mathbb{R}^{n\times n}$*.*

6.4 Some classical algorithms

Restriction: For simplicity, we will assume from now on that $A = A^{\mathrm{T}} \in \mathbb{R}^{n\times n}$, i.e., that A is a real symmetric matrix. Then, there exist an orthogonal matrix $Q \in \mathbb{R}^{n\times n}$ and a diagonal matrix $D = \operatorname{diag}_{n\times n}(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^{n\times n}$ with $\{\lambda_1, \dots, \lambda_n\} = \Lambda(A) \subseteq \mathbb{R}$ such that $A = QDQ^{\mathrm{T}}$. (Note the i-th column of Q is an eigenvector to the eigenvalue λ_i.)

The Rayleigh quotient

The Rayleigh quotient plays an important role in the numerical computation of eigenvalues and is defined as follows:

Definition 6.7 (Rayleigh quotient)**.** Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix. We define the map

$$R_A : \mathbb{R}^n\backslash\{0\} \to \mathbb{R}, \qquad x \mapsto \frac{x^{\mathrm{T}}Ax}{x^{\mathrm{T}}x} = \frac{\langle Ax, x\rangle}{\|x\|_2^2} = \left\langle A\frac{x}{\|x\|_2}, \frac{x}{\|x\|_2}\right\rangle.$$

For $x \in \mathbb{R}^n\backslash\{0\}$, we call the value $R_A(x) \in \mathbb{R}$ the Rayleigh quotient of x (corresponding to the matrix A).

Theorem 6.8 (Properties of the Rayleigh quotient)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix. Then, we have the following:*

(i) If $x \in \mathbb{R}^n\backslash\{0\}$ is an eigenvector of A, then $R_A(x)$ is its corresponding eigenvalue.

(ii) R_A is differentiable on $\mathbb{R}^n\backslash\{0\}$ with gradient

$$\nabla R_A : \mathbb{R}^n\backslash\{0\} \to \mathbb{R}^n, \qquad x \mapsto 2\,\frac{Ax - (R_A(x))x}{\|x\|_2^2}.$$

For $x \in \mathbb{R}^n\backslash\{0\}$, there holds $\nabla R_A(x) = 0$ iff x is an eigenvector of A (i.e., the stationary points of R_A are the eigenvectors of A).

(iii) If $q \in \mathbb{R}^n\backslash\{0\}$ is an eigenvector of A, then $|R_A(x) - R_A(q)| = \mathcal{O}(\|x - q\|_2^2)$ as $x \to q$.

Proof. (i) Let $x \in \mathbb{R}^n\backslash\{0\}$ be an eigenvector of A and let $\lambda \in \mathbb{R}$ be its corresponding eigenvalue, i.e., $Ax = \lambda x$. Then, $R_A(x) = \frac{\langle Ax,x\rangle}{\|x\|_2^2} = \frac{\langle \lambda x,x\rangle}{\|x\|_2^2} = \lambda\frac{\langle x,x\rangle}{\|x\|_2^2} = \lambda$.

(ii) Let us define the maps $f, g : \mathbb{R}^n \to \mathbb{R}$ given by $f(x) := x^{\mathrm{T}}Ax$ and $g(x) := x^{\mathrm{T}}x$, i.e., writing $x = (x_1, \dots, x_n)^{\mathrm{T}}$:

$$f(x) = \sum_{i,j=1}^{n} a_{ij}x_ix_j, \qquad g(x) = \sum_{i=1}^{n} x_i^2.$$

Note that $R_A(x) = \frac{f(x)}{g(x)}$ for any $x \in \mathbb{R}^n\backslash\{0\}$. We compute

$$\begin{aligned}\nabla f(x) = \sum_{i,j=1}^{n} a_{ij}\,(x_je_i + x_ie_j) &= \sum_{i,j=1}^{n} a_{ij}x_je_i + \sum_{i,j=1}^{n} a_{ij}x_ie_j \\ &= 2\sum_{i,j=1}^{n} a_{ij}x_je_i = 2\sum_{i=1}^{n}(Ax)_ie_i = 2Ax\end{aligned}$$

and $\nabla g(x) = 2x$ (follows from previous calculation with $A = I_n$ since $g(x) = x^{\mathrm{T}} I_n x$). Therefore, for any $x \in \mathbb{R}^n \backslash \{0\}$, we have

$$\nabla R_A(x) = \left(\frac{g\, \nabla f - f\, \nabla g}{g^2} \right)(x) = \frac{2\|x\|_2^2 Ax - 2(x^{\mathrm{T}}Ax)x}{\|x\|_2^4} = 2\, \frac{Ax - (R_A(x))x}{\|x\|_2^2}.$$

We now show the second part of (ii). If $x \in \mathbb{R}^n \backslash \{0\}$ is such that $\nabla R_A(x) = 0 \in \mathbb{R}^n$, then $Ax = (R_A(x))x$ and thus, x is an eigenvector of A (corresponding to the eigenvalue $R_A(x)$). Conversely, suppose $x \in \mathbb{R}^n \backslash \{0\}$ is an eigenvector of A and denote the corresponding eigenvalue by λ, i.e., $Ax = \lambda x$. We know from (i) that $\lambda = R_A(x)$ and hence, $Ax = (R_A(x))x$. It follows that $\nabla R_A(x) = 0 \in \mathbb{R}^n$.

(iii) Let $q \in \mathbb{R}^n \backslash \{0\}$ be an eigenvector of A. Since R_A is a smooth function, we have by Taylor's theorem that $R_A(x) = R_A(q) + (\nabla R_A(q))^{\mathrm{T}} x + \mathcal{O}(\|x - q\|_2^2)$ as $x \to q$. In view of (ii), we have that $\nabla R_A(q) = 0 \in \mathbb{R}^n$ and the result follows. □

Power iteration (Von Mises iteration): A method for the largest eigenvalue

The following algorithm computes the largest (in absolute value) eigenvalue and a corresponding normalized eigenvector of a given matrix (under suitable assumptions):

Algorithm 6.2 (Power iteration)**.** Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix. Choose a vector $v^{(0)} \in \mathbb{R}^n$ with $\|v^{(0)}\|_2 = 1$, and do the following:

```
for $k = 1, 2, 3, \ldots$ do
    $w = Av^{(k-1)}$
    $v^{(k)} = \frac{w}{\|w\|_2}$
    $\lambda^{(k)} = \langle Av^{(k)}, v^{(k)} \rangle$
end for
```

Remark 6.16. In practice, a suitable stopping criterion is necessary, an issue which we neglect in this course.

Remark 6.17. The algorithm produces a sequence $(v^{(k)})_{k\in\mathbb{N}}$ of vectors in $\mathbb{R}^n$ given by the relation $v^{(k)} = \frac{Av^{(k-1)}}{\|Av^{(k-1)}\|_2}$ $\forall k \in \mathbb{N}$, i.e., $v^{(k)} = \frac{A^k v^{(0)}}{\|A^k v^{(0)}\|_2}$ $\forall k \in \mathbb{N}$, and a sequence $(\lambda^{(k)})_{k\in\mathbb{N}}$ of real numbers given by $\lambda^{(k)} = R_A(v^{(k)})$ (note $\|v^{(k)}\|_2 = 1$ for all k).

Theorem 6.9 (Convergence of power iteration)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix with an eigenvalue decomposition $A = QDQ^{\mathrm{T}}$ with $Q = (q_1|\cdots|q_n) \in \mathbb{R}^{n\times n}$ orthogonal and $D = \mathrm{diag}_{n\times n}(\lambda_1, \ldots, \lambda_n) \in \mathbb{R}^{n\times n}$ with $\{\lambda_1, \ldots, \lambda_n\} = \Lambda(A)$ and $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$. Let $v^{(0)} \in \mathbb{R}^n$ with $\|v^{(0)}\|_2 = 1$, and let $(v^{(k)}) \subseteq \mathbb{R}^n$ and $(\lambda^{(k)})_{k\in\mathbb{N}}$ be the sequences produced by Algorithm 6.2. If $|\lambda_1| > |\lambda_2|$ and $\langle v^{(0)}, q_1 \rangle \neq 0$, then there holds*

$$\lambda^{(k)} \to \lambda_1 \quad \textit{with convergence rate} \quad |\lambda^{(k)} - \lambda_1| = \mathcal{O}\left(\left| \frac{\lambda_2}{\lambda_1} \right|^{2k} \right) \quad \textit{as} \quad k \to \infty, \tag{6.4}$$

and there holds

$$\|v^{(k)} - s_k q_1\|_2 = \mathcal{O}\left(\left| \frac{\lambda_2}{\lambda_1} \right|^{k} \right) \quad \textit{as} \quad k \to \infty \tag{6.5}$$

for some $(s_k)_{k\in\mathbb{N}} \subseteq \{-1, 1\}$. (We may say $\mathrm{span}(v^{(k)})$ *converges to* $\mathrm{span}(q_1)$ *as $k \to \infty$.)*

Proof. Suppose $|\lambda_1| > |\lambda_2|$ and $\langle v^{(0)}, q_1\rangle \neq 0$. Let us write $v^{(0)} = \sum_{i=1}^n c_i q_i$ with $c_1, \ldots, c_n \in \mathbb{R}$ given by $c_i = \langle v^{(0)}, q_i\rangle$ for $i \in \{1, \ldots, n\}$. Note that $c_1 \neq 0$. Then, we have that

$$v^{(k)} = \frac{A^k v^{(0)}}{\|A^k v^{(0)}\|_2} = \frac{QD^kQ^{\mathrm{T}} v^{(0)}}{\|A^k v^{(0)}\|_2} = \frac{\sum_{i=1}^n c_i \lambda_i^k q_i}{\|\sum_{i=1}^n c_i \lambda_i^k q_i\|_2} = \frac{c_1 \lambda_1^k}{|c_1 \lambda_1^k|} \frac{q_1 + \sum_{i=2}^n \frac{c_i}{c_1} \left(\frac{\lambda_i}{\lambda_1}\right)^k q_i}{\left\| q_1 + \sum_{i=2}^n \frac{c_i}{c_1} \left(\frac{\lambda_i}{\lambda_1}\right)^k q_i \right\|_2}.$$

If $\lambda_1 > 0$, we find that $v^{(k)} \to \operatorname{sign}(c_1) q_1$ as $k \to \infty$ with the desired rate (i.e., (6.5) holds with $s_k := \operatorname{sign}(c_1)$ for all k). If $\lambda_1 < 0$, we find (6.5) holds with $s_k := (-1)^k \operatorname{sign}(\mathrm{c}_1)$. Convergence of the sequence $(\lambda^{(k)})$ to λ_1 as claimed in (6.4) now follows from Theorem 6.8(iii) (recall $\lambda^{(k)} = R_A(v^{(k)})$ from Remark 6.17). □

Remark 6.18 (Drawbacks of power iteration). The power iteration has the following drawbacks:

(i) It only computes the normalized eigenvector for the largest eigenvalue (and it computes only this largest eigenvalue).

(ii) The rate of convergence for $\operatorname{span}(v^{(k)})$ to $\operatorname{span}(q_1)$ is only linear, i.e., the error in each step is reduced by a constant factor ($\approx |\frac{\lambda_1}{\lambda_2}|$).

(iii) If $|\lambda_1| > |\lambda_2|$, but $|\lambda_1|$ is close to $|\lambda_2|$, then the convergence is very slow (as $|\frac{\lambda_2}{\lambda_1}|$ is only slightly below 1).

Inverse iteration: Power iteration for $(A - \mu I_n)^{-1}$

Let us explain how to resolve the drawbacks (i) and (iii) from Remark 6.18 of power iteration. The key observation is the following:

Remark 6.19. Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix with an eigenvalue decomposition $A = QDQ^{\mathrm{T}}$ with $Q = (q_1|\cdots|q_n) \in \mathbb{R}^{n\times n}$ orthogonal and $D = \operatorname{diag}_{n\times n}(\lambda_1, \ldots, \lambda_n) \in \mathbb{R}^{n\times n}$ with $\{\lambda_1, \ldots, \lambda_n\} = \Lambda(A)$. Let $\mu \in \mathbb{R}\backslash\Lambda(A)$. Then, the matrix $A - \mu I_n \in \mathbb{R}^{n\times n}$ is invertible and we have that

$$\Lambda\left((A - \mu I_n)^{-1}\right) = \left\{(\lambda_1 - \mu)^{-1}, \ldots, (\lambda_n - \mu)^{-1}\right\}.$$

Indeed, for $i \in \{1, \ldots, n\}$, we have that $(A - \mu I_n)^{-1} q_i = (\lambda_i - \mu)^{-1} q_i$ since

$$(A - \mu I_n)\left((\lambda_i - \mu)^{-1} q_i\right) = (\lambda_i - \mu)^{-1}(A q_i - \mu q_i) = (\lambda_i - \mu)^{-1}(\lambda_i - \mu) q_i = q_i,$$

i.e., q_i is an eigenvector to $(A - \mu I_n)^{-1}$ corresponding to the eigenvalue $(\lambda_i - \mu)^{-1}$. (Note that the eigenvectors of $(A - \mu I_n)^{-1}$ are the same as the eigenvectors of A.)

We observe that the eigenvalue of $(A - \mu I_n)^{-1}$ with the largest absolute value is $(\lambda_j - \mu)^{-1}$, where λ_j is the eigenvalue of A closest to μ.

In view of this observation, we can apply the power iteration to $(A - \mu I_n)^{-1}$ to find the eigenvalue of A which is closest to μ (and a corresponding normalized eigenvector).

Algorithm 6.3 (Inverse iteration). Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix and $\mu \in \mathbb{R}\backslash\Lambda(A)$. Choose a vector $v^{(0)} \in \mathbb{R}^n$ with $\|v^{(0)}\|_2 = 1$, and do the following:

```
for $k = 1, 2, 3, \dots$ do
    Solve the linear system $(A - \mu I_n)w = v^{(k-1)}$        $(\iff w = (A - \mu I_n)^{-1} v^{(k-1)})$
    $v^{(k)} = \frac{w}{\|w\|_2}$
    $\lambda^{(k)} = \langle A v^{(k)}, v^{(k)} \rangle$
end for
```

Remark 6.20. Without going into detail, let us mention that possible ill-conditioning of $(A - \mu I_n)^{-1}$ when μ is close to an eigenvalue of A does not pose a problem here.

Theorem 6.10 (Convergence of inverse iteration)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix with an eigenvalue decomposition $A = QDQ^{\mathrm{T}}$ with $Q = (q_1|\cdots|q_n) \in \mathbb{R}^{n\times n}$ orthogonal and $D = \operatorname{diag}_{n\times n}(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^{n\times n}$ with $\{\lambda_1, \dots, \lambda_n\} = \Lambda(A)$. Let $\mu \in \mathbb{R}\backslash\Lambda(A)$, let $v^{(0)} \in \mathbb{R}^n$ with $\|v^{(0)}\|_2 = 1$, and let $(v^{(k)}) \subseteq \mathbb{R}^n$ and $(\lambda^{(k)})_{k\in\mathbb{N}}$ be the sequences produced by Algorithm 6.3. Suppose that $\lambda_j, \lambda_k \in \Lambda(A)$ are such that $|\mu - \lambda_j| < |\mu - \lambda_k| \leq |\mu - \lambda_i|$ $\forall i \in \{1, \dots, n\}\backslash\{j\}$ (i.e., λ_j is the closest and λ_k the second closest eigenvalue of A to μ) and that $\langle v^{(0)}, q_j\rangle \neq 0$. Then, there holds*

$$|\lambda^{(k)} - \lambda_j| = \mathcal{O}\left(\left|\frac{\lambda_j - \mu}{\lambda_k - \mu}\right|^{2k}\right), \qquad \|v^{(k)} - s_k q_j\|_2 = \mathcal{O}\left(\left|\frac{\lambda_j - \mu}{\lambda_k - \mu}\right|^{k}\right) \quad \text{as} \quad k \to \infty$$

for some $(s_k)_{k\in\mathbb{N}} \subseteq \{-1, 1\}$.

Proof. This result follows from Theorem 6.9 applied to the matrix $\tilde{A} := (A - \mu I_n)^{-1}$ upon noting that $\tilde{\lambda}_1 := (\lambda_j - \mu)^{-1}$ is the eigenvalue of $\tilde{A}$ with the largest absolute value, $\tilde{\lambda}_2 := (\lambda_k - \mu)^{-1}$ is the eigenvalue of $\tilde{A}$ with the second largest absolute value, and that $\tilde{q}_1 := q_j$ is the eigenvector of $\tilde{A}$ corresponding to the eigenvalue $\tilde{\lambda}_1$. □

Remark 6.21. If we have a good estimate for a certain eigenvalue of A, we can now apply inverse iteration to produce this eigenvalue and a corresponding normalized eigenvector. In particular, inverse iteration is the go-to method if one wants to find eigenvectors to eigenvalues which are already known. The drawback of inverse iteration is the slow speed of convergence (linear convergence, same as for power iteration).

Rayleigh quotient iteration: combining inverse iteration and Rayleigh quotient

The key idea of the Rayleigh quotient iteration is to combine the Rayleigh quotient (a way to find an eigenvalue from an eigenvector) with inverse iteration (a way to find an eigenvector from an eigenvalue).

Algorithm 6.4 (Rayleigh quotient iteration)**.** Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix. Choose a vector $v^{(0)} \in \mathbb{R}^n$ with $\|v^{(0)}\|_2 = 1$, set $\lambda^{(0)} := \langle A v^{(0)}, v^{(0)}\rangle$ and do the following:

```
for $k = 1, 2, 3, \dots$ do
    Solve the linear system $(A - \lambda^{(k-1)} I_n)w = v^{(k-1)}$
    $v^{(k)} = \frac{w}{\|w\|_2}$
    $\lambda^{(k)} = \langle A v^{(k)}, v^{(k)} \rangle$
end for
```

Theorem 6.11 (Convergence of Rayleigh quotient iteration)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix with an eigenvalue decomposition $A = QDQ^{\mathrm{T}}$ with $Q = (q_1|\cdots|q_n) \in \mathbb{R}^{n\times n}$*

orthogonal and $D = \operatorname{diag}_{n\times n}(\lambda_1,\dots,\lambda_n) \in \mathbb{R}^{n\times n}$ *with* $\{\lambda_1,\dots,\lambda_n\} = \Lambda(A)$. *Then, for almost all (all except for a set of measure zero)* $v^{(0)} \in \mathbb{R}^n$ *with* $\|v^{(0)}\|_2 = 1$, *the sequences* $(v^{(k)}) \subseteq \mathbb{R}^n$ *and* $(\lambda^{(k)}) \subseteq \mathbb{R}$ *produced by Algorithm 6.4 converge to an eigenvector and eigenvalue of* A. *Further, in this case and if* $\lambda_j \in \Lambda(A)$ *is such that* $v^{(0)}$ *is sufficiently close to* q_j, *then there holds*

$$|\lambda^{(k+1)} - \lambda_j| = \mathcal{O}\left(|\lambda^{(k)} - \lambda_j|^3\right), \quad \|v^{(k+1)} - s_{k+1}q_j\|_2 = \mathcal{O}\left(\|v^{(k)} - s_k q_j\|_2^3\right) \quad \textit{as} \quad k \to \infty$$

for some $(s_k)_{k\in\mathbb{N}} \subseteq \{-1, 1\}$.

Proof. Omitted. □

Remark 6.22. Let us emphasize that we have cubic convergence! (extremely quick)

Example 6.2. Consider the symmetric matrix $A = \begin{pmatrix} -1 & 2 & 2 \\ 2 & 1 & 2 \\ 2 & 2 & -1 \end{pmatrix}$. We perform Rayleigh quotient iteration with $v^{(0)} := \frac{1}{3}(1,-2,2)^{\mathrm{T}}$. To illustrate the speed of convergence, we have colored the correct digits in red.
Step 0: Compute

$$\lambda^{(0)} := \langle Av^{(0)}, v^{(0)} \rangle = -\frac{17}{9} = -1.8888\dots$$

Step 1: ($k = 1$.) Solve $(A - \lambda^{(0)} I_3)w^{(1)} = v^{(0)}$. We find $w^{(1)} = \frac{3}{70}(191, -265, 184)^{\mathrm{T}}$. Compute

$$v^{(1)} := \frac{w^{(1)}}{\|w^{(1)}\|_2} = \begin{pmatrix} \frac{191}{3\sqrt{15618}} \\ \frac{-265}{3\sqrt{15618}} \\ \frac{184}{3\sqrt{15618}} \end{pmatrix} = \begin{pmatrix} 0.5094\dots \\ -0.7068\dots \\ 0.4907\dots \end{pmatrix}, \quad \lambda^{(1)} := \langle Av^{(1)}, v^{(1)} \rangle = -\frac{128518}{70281} = -1.8286\dots$$

Step 2: ($k = 2$.) Solve $(A - \lambda^{(1)} I_3)w^{(2)} = v^{(1)}$ and compute

$$v^{(2)} := \frac{w^{(2)}}{\|w^{(2)}\|_2} = \begin{pmatrix} 0.49999838\dots \\ -0.70710677\dots \\ 0.50000162\dots \end{pmatrix}, \qquad \lambda^{(2)} := \langle Av^{(2)}, v^{(2)} \rangle = -1.82842712475\dots$$

(Remark: $(\lambda^{(k)})$ converges to $1 - 2\sqrt{2}$, and $\operatorname{span}(v^{(k)})$ converges to $\operatorname{span}((\frac{1}{2}, -\frac{1}{\sqrt{2}}, \frac{1}{2})^{\mathrm{T}})$.)

Stopping at $k = 2$, we see that our approximation $\lambda^{(2)}$ to the exact eigenvalue is already accurate to 11 digits. If the algorithm would be preformed in exact arithmetic, we would expect at $k = 3$ accuracy to around 33 digits and, e.g., at $k = 5$ accuracy to around 297 digits).

6.5 The QR algorithm

Restriction: As in the previous section, we assume that $A = A^{\mathrm{T}} \in \mathbb{R}^{n\times n}$, i.e., that A is a real symmetric matrix. Then, there exist an orthogonal matrix $Q \in \mathbb{R}^{n\times n}$ and a diagonal matrix $D = \operatorname{diag}_{n\times n}(\lambda_1,\dots,\lambda_n) \in \mathbb{R}^{n\times n}$ with $\{\lambda_1,\dots,\lambda_n\} = \Lambda(A) \subseteq \mathbb{R}$ and $A = QDQ^{\mathrm{T}}$.

Let us recall that if $A \in \mathbb{R}^{n\times n}$ is symmetric, we can transform A into a symmetric tridiagonal matrix via orthogonal similarity transforms, i.e., we can find a Hessenberg decomposition $A = QHQ^{\mathrm{T}}$ with $Q \in \mathbb{R}^{n\times n}$ orthogonal and $H \in \mathbb{R}^{n\times n}$ symmetric and tridiagonal (see Remark 6.13). This is what we do as the first step of the so-called QR algorithm: we reduce A to tridiagonal form in the aforementioned way and work with H instead of A.

QR algorithm

Let us discuss the following algorithm:

Algorithm 6.5 (QR algorithm)**.** Let $A \in \mathbb{R}^{n\times n}$ be a symmetric tridiagonal matrix. Set $A^{(0)} := A$ and do the following:

```
for $k = 1, 2, 3, \dots$ do
  Compute a QR factorization $A^{(k-1)} = Q^{(k)}R^{(k)}$ of $A^{(k-1)}$
  $A^{(k)} = R^{(k)}Q^{(k)}$
end for
```

Remark 6.23. Note that the iterates in Algorithm 6.5 satisfy $A^{(k)} = (Q^{(k)})^{\mathrm{T}}A^{(k-1)}Q^{(k)}$, i.e., the QR algorithm consists of orthogonal similarity transformations.

We are going to see that the sequence $(A^{(k)})_{k\in\mathbb{N}}$ produced by Algorithm 6.5 converges under suitable assumptions to a Schur form of A (i.e., in view of Remark 6.10, to a diagonal matrix containing the eigenvalues of A on the diagonal). Let us introduce a second method, the simultaneous iteration, which will actually turn out to be equivalent to the QR algorithm.

Simultaneous iteration (block power iteration)

Suppose we are given a symmetric tridiagonal matrix $A \in \mathbb{R}^{n\times n}$ (i.e., Hessenberg reduction has already been performed). Consider the following natural approach. Take linearly independent vectors $v_1^{(0)}, \dots, v_n^{(0)} \in \mathbb{R}^n$ and apply the power iteration to these vectors simultaneously in the following sense: Setting $V^{(0)} := (v_1^{(0)}|\cdots|v_n^{(0)})$, compute the matrix $V^{(k)} := A^kV^{(0)}$ and write $(v_1^{(k)}|\cdots|v_n^{(k)}) = V^{(k)} = (A^kv_1^{(0)}|\cdots|A^kv_n^{(0)})$, and finally orthogonalize $V^{(k)}$ in the sense of computing a QR factorization $V^{(k)} = Q^{(k)}R^{(k)}$. Then, under suitable assumptions, the span of the first l columns of $Q^{(k)}$ will converge to the span of the eigenvectors corresponding to the l largest (in absolute value) eigenvalues of A.

In practice, in view of numerical stability, the following normalized version of simultaneous iteration is used (orthonormalize at each step):

Algorithm 6.6 (Simultaneous iteration)**.** Let $A \in \mathbb{R}^{n\times n}$ be a symmetric tridiagonal matrix. Choose an orthogonal matrix $Q^{(0)} \in \mathbb{R}^{n\times n}$. Do the following:

```
for $k = 1, 2, 3, \dots$ do
  $Z = AQ^{(k-1)}$
  Compute a QR factorization $Z = Q^{(k)}R^{(k)}$ of $Z$
  $A^{(k)} = (Q^{(k)})^{\mathrm{T}}AQ^{(k)}$
end for
```

Theorem 6.12 (Convergence of simultaneous iteration)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric tridiagonal matrix with an eigenvalue decomposition $A = QDQ^{\mathrm{T}}$ with $Q = (q_1|\cdots|q_n) \in$*

$\mathbb{R}^{n\times n}$ *orthogonal and* $D = \operatorname{diag}_{n\times n}(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^{n\times n}$ *with* $\{\lambda_1, \dots, \lambda_n\} = \Lambda(A)$*. Suppose*

$$|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n|.$$

Then, if Algorithm 6.6 is performed with an initial choice $Q^{(0)} \in \mathbb{R}^{n\times n}$ *satisfying*

$$\det(M_{1:i,1:i}) \neq 0 \quad \forall i \in \{1, \dots, n\}, \text{ where } \quad M := Q^{\mathrm{T}} Q^{(0)},$$

and writing $Q^{(k)} = (q_1^{(k)}|\dots|q_n^{(k)})$*, we have for any* $j \in \{1, \dots, n\}$ *that for some* $(s_k)_{k\in\mathbb{N}} \subseteq \{-1, 1\}$ *there holds*

$$\|q_j^{(k)} - s_k q_j\|_2 = \mathcal{O}\left(\left(\max_{i\in\{1,\dots,n-1\}} \left|\frac{\lambda_{i+1}}{\lambda_i}\right|\right)^k\right).$$

Theorem 6.13 (Equivalence of QR algorithm and simultaneous iteration)**.** *Algorithm 6.5 and Algorithm 6.6 with* $Q^{(0)} := I_n$ *produce the same sequences* $(A^{(k)})_{k\in\mathbb{N}}$*. Further, we have that*

$$\begin{aligned} Q_{\mathrm{sIt}}^{(k)} &= Q_{\mathrm{QR}}^{(1)} Q_{\mathrm{QR}}^{(2)} \cdots Q_{\mathrm{QR}}^{(k)} =: \tilde{Q}_{\mathrm{QR}}^{(k)}, \\ \tilde{R}_{\mathrm{sIt}}^{(k)} := R_{\mathrm{sIt}}^{(k)} \cdots R_{\mathrm{sIt}}^{(2)} R_{\mathrm{sIt}}^{(1)} &= R_{\mathrm{QR}}^{(k)} \cdots R_{\mathrm{QR}}^{(2)} R_{\mathrm{QR}}^{(1)} =: \tilde{R}_{\mathrm{QR}}^{(k)} \end{aligned}$$

for any $k \in \mathbb{N}$*, and there holds*

$$\begin{aligned} A^{(k)} &= (\tilde{Q}_{\mathrm{QR}}^{(k)})^{\mathrm{T}} A\, \tilde{Q}_{\mathrm{QR}}^{(k)}, \\ A^k &= \tilde{Q}_{\mathrm{QR}}^{(k)} \tilde{R}_{\mathrm{QR}}^{(k)}. \end{aligned}$$

(Here, the subscript sIt *refers to the iterates from simultaneous iteration and the subscript* QR *refers to the iterates from the QR algorithm.)*

Theorem 6.14 (Convergence of QR algorithm)**.** *Let* $A \in \mathbb{R}^{n\times n}$ *be a symmetric tridiagonal matrix with an eigenvalue decomposition* $A = QDQ^{\mathrm{T}}$ *with* $Q = (q_1|\cdots|q_n) \in \mathbb{R}^{n\times n}$ *orthogonal and* $D = \operatorname{diag}_{n\times n}(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^{n\times n}$ *with* $\{\lambda_1, \dots, \lambda_n\} = \Lambda(A)$*. Suppose*

$$|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n|$$

and that

$$\det(Q_{1:i,1:i}) \neq 0 \quad \forall i \in \{1, \dots, n\}.$$

Let $(A^{(k)})_{k\in\mathbb{N}}$ *and* $(Q^{(k)})_{k\in\mathbb{N}}$ *be the sequences produced by Algorithm 6.5 applied to* A*, and let* $(\tilde{Q}^{(k)})_{k\in\mathbb{N}}$ *be the sequence with* $\tilde{Q}^{(k)} := (\tilde{q}_1^{(k)}|\cdots|\tilde{q}_n^{(k)}) := Q^{(1)}Q^{(2)}\cdots Q^{(k)}$ *for* $k \in \mathbb{N}$*. Then, as* $k \to \infty$*, there holds* $A^{(k)} \to D$*, and for any* $j \in \{1, \dots, n\}$ *we have for some* $(s_k)_{k\in\mathbb{N}} \subseteq \{-1, 1\}$ *that* $\tilde{q}_j^{(k)} - s_k q_j \to 0$*. The speed of convergence is linear with constant* $\max_{i\in\{1,\dots,n-1\}} \left|\frac{\lambda_{i+1}}{\lambda_i}\right|$*.*

Remark 6.24. Note that the diagonal entries of the iterates $A^{(k)}$ produced by the QR algorithm are Rayleigh quotients: Writing $A^{(k)} = (a_{ij}^{(k)})$, we have

$$a_{ii}^{(k)} = \langle e_i, A^{(k)} e_i \rangle = \langle e_i, (\tilde{Q}^{(k)})^{\mathrm{T}} A \tilde{Q}^{(k)} e_i \rangle = \langle \tilde{Q}^{(k)} e_i, A \tilde{Q}^{(k)} e_i \rangle = \langle \tilde{q}_i^{(k)}, A \tilde{q}_i^{(k)} \rangle = R_A(\tilde{q}_i^{(k)})$$

for any $i \in \{1, \dots, n\}$.

Example 6.3 (QR algorithm). Let us consider the matrix $A := \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$. The matrix A has the eigenvalue decomposition $A = QDQ^{\mathrm{T}}$ with

$$D := \operatorname{diag}_{3\times 3}(\lambda_1, \lambda_2, \lambda_3) := \begin{pmatrix} 1+\sqrt{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1-\sqrt{2} \end{pmatrix} = \begin{pmatrix} 2.414\ldots & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -0.414\ldots \end{pmatrix},$$

$$Q := (q_1|q_2|q_3) := \begin{pmatrix} -\frac{1}{2} & \frac{1}{\sqrt{2}} & -\frac{1}{2} \\ \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} \\ \frac{1}{2} & \frac{1}{\sqrt{2}} & \frac{1}{2} \end{pmatrix} = \begin{pmatrix} -0.5 & 0.707\ldots & -0.5 \\ 0.707\ldots & 0 & -0.707\ldots \\ 0.5 & 0.707\ldots & 0.5 \end{pmatrix}.$$

Note that the assumptions of Theorem 6.14 are satisfied. Let us perform the QR algorithm:

$k = 1$: We need to compute a QR factorization of $A^{(0)} := A$. We omit the details and take the QR factorization $A^{(0)} = Q^{(1)}R^{(1)}$ with

$$Q^{(1)} := \begin{pmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 1 & 0 \end{pmatrix}, \qquad R^{(1)} := \begin{pmatrix} \sqrt{2} & -\sqrt{2} & -\frac{1}{\sqrt{2}} \\ 0 & 1 & 1 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{pmatrix}.$$

We compute

$$A^{(1)} := R^{(1)}Q^{(1)} = \begin{pmatrix} 2 & -\frac{1}{\sqrt{2}} & 0 \\ -\frac{1}{\sqrt{2}} & 1 & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & 0 \end{pmatrix} = \begin{pmatrix} 2 & -0.707\ldots & 0 \\ -0.707\ldots & 1 & 0.707\ldots \\ 0 & 0.707\ldots & 0 \end{pmatrix},$$

$$\tilde{Q}^{(1)} := Q^{(1)} = \begin{pmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0.707\ldots & 0 & 0.707\ldots \\ -0.707\ldots & 0 & 0.707\ldots \\ 0 & 1 & 0 \end{pmatrix}.$$

$k = 2$: We need to compute a QR factorization of $A^{(1)}$. We omit the details and take the QR factorization $A^{(1)} = Q^{(2)}R^{(2)}$ with

$$Q^{(2)} := \begin{pmatrix} \frac{2\sqrt{2}}{3} & \frac{1}{3\sqrt{2}} & \frac{1}{3\sqrt{2}} \\ -\frac{1}{3} & \frac{2}{3} & \frac{2}{3} \\ 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}, \qquad R^{(2)} := \begin{pmatrix} \frac{3}{\sqrt{2}} & -1 & -\frac{1}{3\sqrt{2}} \\ 0 & 1 & \frac{\sqrt{2}}{3} \\ 0 & 0 & \frac{\sqrt{2}}{3} \end{pmatrix}.$$

We compute

$$A^{(2)} := R^{(2)}Q^{(2)} = \begin{pmatrix} \frac{7}{3} & -\frac{1}{3} & 0 \\ -\frac{1}{3} & 1 & \frac{1}{3} \\ 0 & \frac{1}{3} & -\frac{1}{3} \end{pmatrix} = \begin{pmatrix} 2.333\ldots & -0.333\ldots & 0 \\ -0.333\ldots & 1 & 0.333\ldots \\ 0 & 0.333\ldots & -0.333\ldots \end{pmatrix},$$

$$\tilde{Q}^{(2)} := \tilde{Q}^{(1)}Q^{(2)} = \begin{pmatrix} \frac{2}{3} & \frac{2}{3} & -\frac{1}{3} \\ -\frac{2}{3} & \frac{1}{3} & -\frac{2}{3} \\ -\frac{1}{3} & \frac{2}{3} & \frac{2}{3} \end{pmatrix} = \begin{pmatrix} 0.666\ldots & 0.666\ldots & -0.333\ldots \\ -0.666\ldots & 0.333\ldots & -0.666\ldots \\ -0.333\ldots & 0.666\ldots & 0.666\ldots \end{pmatrix}.$$

$k = 3$: We need to compute a QR factorization of $A^{(2)}$. We omit the details and take the QR factorization $A^{(2)} = Q^{(3)}R^{(3)}$ with

$$Q^{(3)} := \begin{pmatrix} \frac{7}{5\sqrt{2}} & \frac{2}{15} & \frac{1}{15\sqrt{2}} \\ -\frac{1}{5\sqrt{2}} & \frac{14}{15} & \frac{7}{15\sqrt{2}} \\ 0 & \frac{1}{3} & -\frac{2\sqrt{2}}{3} \end{pmatrix}, \qquad R^{(3)} := \begin{pmatrix} \frac{5\sqrt{2}}{3} & -\frac{\sqrt{2}}{3} & -\frac{1}{15\sqrt{2}} \\ 0 & 1 & \frac{1}{5} \\ 0 & 0 & \frac{3}{5\sqrt{2}} \end{pmatrix}.$$

We compute

$$A^{(3)} := R^{(3)}Q^{(3)} = \begin{pmatrix} \frac{12}{5} & -\frac{1}{5\sqrt{2}} & 0 \\ -\frac{1}{5\sqrt{2}} & 1 & \frac{1}{5\sqrt{2}} \\ 0 & \frac{1}{5\sqrt{2}} & -\frac{2}{5} \end{pmatrix} = \begin{pmatrix} 2.4 & -0.141\ldots & 0 \\ -0.141\ldots & 1 & 0.141\ldots \\ 0 & 0.141\ldots & -0.4 \end{pmatrix},$$

$$\tilde{Q}^{(3)} := \tilde{Q}^{(2)}Q^{(3)} = \begin{pmatrix} \frac{4}{5\sqrt{2}} & \frac{3}{5} & \frac{4}{5\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ -\frac{3}{5\sqrt{2}} & \frac{4}{5} & -\frac{3}{5\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 0.565\ldots & 0.6 & 0.565\ldots \\ -0.707\ldots & 0 & 0.707\ldots \\ -0.424\ldots & 0.8 & -0.424\ldots \end{pmatrix}.$$

$k = 4$: We need to compute a QR factorization of $A^{(3)}$. We omit the details and take the QR factorization $A^{(3)} = Q^{(4)}R^{(4)}$ with

$$Q^{(4)} := \begin{pmatrix} \frac{12\sqrt{2}}{17} & \frac{7}{85\sqrt{2}} & \frac{1}{85\sqrt{2}} \\ -\frac{1}{17} & \frac{84}{85} & \frac{12}{85} \\ 0 & \frac{1}{5\sqrt{2}} & -\frac{7}{5\sqrt{2}} \end{pmatrix}, \qquad R^{(4)} := \begin{pmatrix} \frac{17}{5\sqrt{2}} & -\frac{1}{5} & -\frac{1}{85\sqrt{2}} \\ 0 & 1 & \frac{\sqrt{2}}{17} \\ 0 & 0 & \frac{5\sqrt{2}}{17} \end{pmatrix}.$$

We compute

$$A^{(4)} := R^{(4)}Q^{(4)} = \begin{pmatrix} \frac{41}{17} & -\frac{1}{17} & 0 \\ -\frac{1}{17} & 1 & \frac{1}{17} \\ 0 & \frac{1}{17} & -\frac{7}{17} \end{pmatrix} = \begin{pmatrix} 2.411\ldots & -0.058\ldots & 0 \\ -0.058\ldots & 1 & 0.058\ldots \\ 0 & 0.058\ldots & -0.411\ldots \end{pmatrix},$$

$$\tilde{Q}^{(4)} := \tilde{Q}^{(3)}Q^{(4)} = \begin{pmatrix} \frac{9}{17} & \frac{12}{17} & -\frac{8}{17} \\ -\frac{12}{17} & \frac{1}{17} & -\frac{12}{17} \\ -\frac{8}{17} & \frac{12}{17} & \frac{9}{17} \end{pmatrix} = \begin{pmatrix} 0.529\ldots & 0.705\ldots & -0.470\ldots \\ -0.705\ldots & 0.058\ldots & -0.705\ldots \\ -0.470\ldots & 0.705\ldots & 0.529\ldots \end{pmatrix}.$$

We see that after 4 steps of the QR algorithm, we have obtained the following approximations to the eigenvalues:

$$\begin{aligned} &\lambda_1 \approx \tfrac{41}{17} = 2.411\ldots, && (\text{recall } \lambda_1 = 1 + \sqrt{2} = 2.414\ldots) \\ &\lambda_2 \approx 1, && (\text{recall } \lambda_2 = 1) \\ &\lambda_3 \approx -\tfrac{7}{17} = -0.411\ldots, && (\text{recall } \lambda_3 = 1 - \sqrt{2} = -0.414\ldots) \end{aligned}$$

and the following approximations to the (subspaces spanned by the) eigenvectors:

$$\operatorname{span}(q_1) \approx \operatorname{span}(\begin{pmatrix} \frac{9}{17} \\ -\frac{12}{17} \\ -\frac{8}{17} \end{pmatrix}) = \operatorname{span}(\begin{pmatrix} 0.529\ldots \\ -0.705\ldots \\ -0.470\ldots \end{pmatrix}), \quad (\text{recall } q_1 = \begin{pmatrix} -\frac{1}{2} \\ \frac{1}{\sqrt{2}} \\ \frac{1}{2} \end{pmatrix})$$

$$\operatorname{span}(q_2) \approx \operatorname{span}(\begin{pmatrix} \frac{12}{17} \\ \frac{1}{17} \\ \frac{12}{17} \end{pmatrix}) = \operatorname{span}(\begin{pmatrix} 0.705\ldots \\ 0.058\ldots \\ 0.705\ldots \end{pmatrix}), \quad (\text{recall } q_2 = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ \frac{1}{\sqrt{2}} \end{pmatrix})$$

$$\operatorname{span}(q_3) \approx \operatorname{span}(\begin{pmatrix} -\frac{8}{17} \\ -\frac{12}{17} \\ \frac{9}{17} \end{pmatrix}) = \operatorname{span}(\begin{pmatrix} -0.470\ldots \\ -0.705\ldots \\ 0.529\ldots \end{pmatrix}) \quad (\text{recall } q_3 = \begin{pmatrix} -\frac{1}{2} \\ -\frac{1}{\sqrt{2}} \\ \frac{1}{2} \end{pmatrix}).$$

$\vdots$

Exercise: do a few more iterations (you may use MATLAB). Further, perform simultaneous iteration applied to A and verify at this example the results from Theorem 6.13.

QR algorithm with Rayleigh quotient shift

Algorithm 6.7 (QR algorithm with Rayleigh quotient shift)**.** Let $A \in \mathbb{R}^{n\times n}$ be a symmetric tridiagonal matrix. Set $A^{(0)} := A$ and do the following:

```
**for** $k = 1, 2, 3, \dots$ **do**
    $\mu^{(k)} = A_{nn}^{(k-1)}$          [here, $A_{nn}^{(k-1)}$ is the (n,n)-entry of $A^{(k-1)}$]
    Compute a QR factorization $A^{(k-1)} - \mu^{(k)} I_n = Q^{(k)} R^{(k)}$ of the matrix $A^{(k-1)} - \mu^{(k)} I_n$
    $A^{(k)} = R^{(k)} Q^{(k)} + \mu^{(k)} I_n$
**end for**
```

Remark 6.25. Let us make some observations.

(i) For $k \in \mathbb{N}$ define $\tilde{Q}^{(k)} := Q^{(1)} Q^{(2)} \cdots Q^{(k)}$ and $\tilde{R}^{(k)} := R^{(k)} \cdots R^{(1)}$. Then, for any $k \in \mathbb{N}$ we have

$$A^{(k)} = (\tilde{Q}^{(k)})^{\mathrm{T}} A \tilde{Q}^{(k)}, \qquad (A - \mu^{(k)} I_n)(A - \mu^{(k-1)} I_n) \cdots (A - \mu^{(1)} I_n) = \tilde{Q}^{(k)} \tilde{R}^{(k)}.$$

The first result follows from the fact that

$$\begin{aligned} A^{(k)} &= (Q^{(k)})^{\mathrm{T}} Q^{(k)} (R^{(k)} Q^{(k)} + \mu^{(k)} I_n) = (Q^{(k)})^{\mathrm{T}} \left((Q^{(k)} R^{(k)}) Q^{(k)} + \mu^{(k)} Q^{(k)} \right) \\ &= (Q^{(k)})^{\mathrm{T}} \left((A^{(k-1)} - \mu^{(k)} I_n) Q^{(k)} + \mu^{(k)} Q^{(k)} \right) = (Q^{(k)})^{\mathrm{T}} A^{(k-1)} Q^{(k)} \end{aligned}$$

for any $k \in \mathbb{N}$. The proof of the second result is omitted.

(ii) The first column of $\tilde{Q}^{(k)}$ is the result of applying k steps of shifted power iteration to e_1 with shifts $\mu^{(1)}, \dots, \mu^{(k)}$, and the last column of $\tilde{Q}^{(k)}$ is the result of applying k steps of shifted inverse iteration to e_n with shifts $\mu^{(1)}, \dots, \mu^{(k)}$. To see the latter, define $P := (e_n | \cdots | e_2 | e_1) \in \mathbb{R}^{n\times n}$ and note that

$$\begin{aligned} (A - \mu^{(k)} I_n)^{-1} (A - \mu^{(k-1)} I_n)^{-1} \cdots (A - \mu^{(1)} I_n)^{-1} P &= ((\tilde{Q}^{(k)} \tilde{R}^{(k)})^{-1})^{\mathrm{T}} P \\ &= \left((\tilde{R}^{(k)})^{-1} (\tilde{Q}^{(k)})^{\mathrm{T}} \right)^{\mathrm{T}} P = (\tilde{Q}^{(k)} P)(P((\tilde{R}^{(k)})^{-1})^{\mathrm{T}} P) \end{aligned}$$

is a QR factorization of the left-hand side.

(iii) For any $k \in \mathbb{N}$, we have

$$\begin{aligned} A_{nn}^{(k)} = \langle e_n, A^{(k)} e_n \rangle = \langle e_n, (\tilde{Q}^{(k)})^{\mathrm{T}} A \tilde{Q}^{(k)} e_n \rangle = \langle \tilde{Q}^{(k)} e_n, A \tilde{Q}^{(k)} e_n \rangle &= \langle \tilde{q}_n^{(k)}, A \tilde{q}_n^{(k)} \rangle \\ &= R_A(\tilde{q}_n^{(k)}), \end{aligned}$$

where $\tilde{q}_n^{(k)} := \tilde{Q}^{(k)} e_n$ denotes the last column of $\tilde{Q}^{(k)}$.

(iv) The approximation $\mu^{(k)}$ to the eigenvalue corresponding to the eigenvector approximated by $\tilde{q}_n^{(k)}$, and the approximated eigenvector $\tilde{q}_n^{(k)}$, are the result of Rayleigh quotient iteration applied to e_n. It follows that we have cubic convergence for the convergence of $\operatorname{span}(\tilde{q}_n^{(k)})$ to the span of an eigenvector.

QR algorithm in practice

In practice, a technique called *deflation* is used:

Algorithm 6.8 (QR algorithm in practice)**.** Let A be a real symmetric tridiagonal square matrix. Set $A^{(0)} := A$ and do the following:

```
for $k = 1, 2, 3, \dots$ do
    Choose a shift $\mu^{(k)}$, e.g., the final diagonal entry of $A^{(k-1)}$
    Compute a QR factorization $A^{(k-1)} - \mu^{(k)} I_n = Q^{(k)} R^{(k)}$ of the matrix $A^{(k-1)} - \mu^{(k)} I_n$
    $A^{(k)} = R^{(k)} Q^{(k)} + \mu^{(k)} I_n$
    If an off-diagonal element $A^{(k)}_{i,i+1}$ is sufficiently close to 0, set $A^{(k)}_{i,i+1} := 0$, $A^{(k)}_{i+1,i} := 0$
    so that $A^{(k)} = \left(\begin{array}{c|c} A_1 & 0 \\ \hline 0 & A_2 \end{array}\right)$ is block-diagonal and apply the algorithm to $A_1$ and $A_2$.
end for
```