11 Adjoint and Self-adjoint Matrices

In this chapter, V denotes a finite dimensional inner product space (unless stated otherwise).

11.1 Theorem (Riesz representation)
Let $f \in V^*$, i.e. f is a linear functional on V. Then there is a unique vector $w \in V$ such that

$$f(v) = \langle v, w \rangle \qquad \forall v \in V$$

Proof. Let $B = \{u_1, \ldots, u_n\}$ be an ONB in V. Then for any $v = \sum c_i u_i$ we have $f(v) = \sum c_i f(u_i)$ by linearity. Also, for any $w = \sum d_i u_i$ we have $\langle v, w \rangle = \sum c_i \bar{d}_i$. Hence, the vector $w = \sum \overline{f(u_i)} u_i$ will suffice. The uniqueness of w is obvious. □

11.2 Corollary
The identity $f \leftrightarrow w$ established in the previous theorem is "quasi-linear" in the following sense[1]: $f_1 + f_2 \leftrightarrow w_1 + w_2$ and $cf \leftrightarrow \bar{c}w$. In the real case, it is perfectly linear, though, and hence it is an isomorphism between V^* and V. This is a canonical isomorphism associated with the given (real) inner product $\langle \cdot, \cdot \rangle$

Remark. If $\dim V = \infty$, then Theorem 11.1 fails. Consider $V = C[0,1]$ (real functions) with the inner product $\langle F, G \rangle = \int_0^1 F(x)G(x)\,dx$. Pick a point $t \in [0,1]$. Let $f \in V^*$ be a linear functional defined by $f(F) = F(t)$. It does not correspond to any $G \in V$ so that $f(F) = \langle F, G \rangle$. In fact, the lack of such functions G has led to the concept of *generalized functions*: a generalized function $G_t(x)$ is "defined" by three requirements: $G_t(x) \equiv 0$ for all $x \neq t$, $G_t(t) = \infty$ and $\int_0^1 F(x)G_t(x)\,dx = F(t)$ for every $F \in C[0,1]$. Clearly, no regular function satisfies these requirements, so a "generalized function" is a purely abstract concept.

11.3 Lemma
Let $T : V \to V$ be a linear operator, $B = \{u_1, \ldots, u_n\}$ an ONB in V, and $A = [T]_B$. Then

$$A_{ij} = \langle Tu_j, u_i \rangle$$

11.4 Theorem and Definition (Adjoint Operator)
Let $T : V \to V$ be a linear operator. Then there is a unique linear operator $T^* : V \to V$ such that

$$\langle Tv, w \rangle = \langle v, T^* w \rangle \qquad \forall v, w \in V$$

[1] Recall the conjugate linearity of the complex inner product in the second argument.

T^* is called the *adjoint* of T.

Proof. Let $w \in V$. Then $f(v) := \langle Tv, w\rangle$ defines a linear functional $f \in V^*$. By the Riesz representation theorem, there is a unique $w' \in V$ such that $f(v) = \langle v, w'\rangle$. Then we define T^* by setting $T^*w = w'$. The linearity of T^* is a routine check. Note that in the complex case the conjugating bar appears twice and thus cancels. The uniqueness of T^* is obvious. □

11.5 Corollary

Let T, S be linear operators on V. Then
(i) $(T + S)^* = T^* + S^*$
(ii) $(cT)^* = \bar{c}\,T^*$ (conjugate linearity)
(iii) $(TS)^* = S^*T^*$
(iv) $(T^*)^* = T$.

11.6 Corollary

Let $T : V \to V$ be a linear operator, and B an ONB in V. Then

$$[T^*]_B = \overline{[T]_B^t}$$

Notation. For a matric $A \in \mathbb{C}^{n\times n}$ we call

$$A^* := \overline{A^t} = \bar{A}^t$$

the *adjoint matrix*. One often uses A^H instead of A^*. In the real case, we simply have $A^* = A^t$, the transposed matrix.

11.7 Theorem.

Let $T : V \to V$ be a linear operator. Then

$$\operatorname{Ker} T^* = (\operatorname{Im} T)^\perp$$

In particular, $V = \operatorname{Im} T \oplus \operatorname{Ker} T^*$.

Proof is a routine check.

11.8 Defnition (Selfadjoint Operator)

A linear operator $T : V \to V$ is said to be *selfadjoint* if $T^* = T$. A matrix A is said to be *selfadjoint* if $A^* = A$. In the real case, this is equivalent to $A^t = A$, i.e. A is a symmetric matrix. In the complex case, selfadjoint matrices are often called *Hermitean matrices*.

Note: By 11.6, an operator T is selfadjoint whenever the matrix $[T]_B$ is selfadjoint for any (and then every) ONB B.

11.9 Theorem

Let $T : V \to V$ be selfadjoint. Then
(i) All eigenvalues of T are real
(ii) If $\lambda_1 \neq \lambda_2$ are two distinct eigenvalues of T with respective eigenvectors v_1, v_2, then v_1 and v_2 are orthogonal.

Proof. If $Tv = \lambda v$, then $\lambda\langle v, v\rangle = \langle Tv, v\rangle = \langle v, Tv\rangle = \bar{\lambda}\langle v, v\rangle$, hence $\lambda = \bar{\lambda}$ (since $v \neq 0$). To prove (ii), we have $\lambda_1\langle v_1, v_2\rangle = \langle Tv_1, v_2\rangle = \langle v_1, Tv_2\rangle = \overline{\lambda_2}\langle v_1, v_2\rangle$, and in addition $\overline{\lambda_2} = \lambda_2 \neq \lambda_1$, hence $\langle v_1, v_2\rangle = 0$.

Remark. Note that in the real case Theorem 11.9 implies that the characteristic polynomial $C_T(x)$ has all real roots, i.e. $C_T(x) = \prod_i (x - \lambda_i)$ where all λ_i's are real numbers. In particular, every real symmetric matrix has at least one (real) eigenvalue.

11.10 Corollary

Any selfadjoint operator T has at least one eigenvalue.

11.11 Lemma

Let T be a selfadjoint operator and a subspace W be T-invariant, i.e. $TW \subset W$. Then $W^\perp$ is also T-invariant, i.e. $TW^\perp \subset W^\perp$.

Proof. If $v \in W^\perp$, then for any $w \in W$ we have $\langle Tv, w\rangle = \langle v, Tw\rangle = 0$, so $Tv \in W^\perp$. □

11.12 Definition (Projection)

Let V be a vector space. A linear map $T : V \to V$ is called a *projection* if $P^2 = P$.

Note: By the homework problem 3 in assignment 2 of MA631, a projection P satisfies $\operatorname{Ker} P = \operatorname{Im}(I - P)$ and $V = \operatorname{Ker} P \oplus \operatorname{Im} P$. For any vector $v \in V$ there is a unique decomposition $v = v_1 + v_2$ with $v_1 \in \operatorname{Ker} P$ and $v_2 \in \operatorname{Im} P$. Then $Pv = P(v_1 + v_2) = v_2$. We say that P is the projection on $\operatorname{Im} P$ along $\operatorname{Ker} P$.

11.13 Corollary

Let V be a vector space and $V = W_1 \oplus W_2$. Then there is a unique projection P on W_2 along W_1.

11.14 Definition (Orthogonal Projection)

Let V is an inner product vector space and $W \subset V$ a finite dimensional subspace. Then the projection on W along $W^\perp$ is called the *orthogonal projection* on W, denoted by P_W.

Remark. The assumption on W being finite dimensional is made to ensure that $V =$

$W \oplus W^\perp$, recall 9.17.

11.15 Theorem

Let V be a finite dimensional inner product space and $W \subset V$ a subspace. Then $(W^\perp)^\perp = W$ and

$$P_{W^\perp} = I - P_W$$

Remark. More generally, if $V = W_1 \oplus W_2$, then $P_1 + P_2 = I$, where P_1 is a projection on W_1 along W_2 and P_2 is a projection on W_2 along W_1.

11.16 Theorem

Let P be a projection. Then P is an orthogonal projection if and only if P is selfadjoint.

Proof. Let P be a projection on W_2 along W_1, and $V = W_1 \oplus W_2$. For any vectors $v, w \in V$ we have $v = v_1 + v_2$ and $w = w_1 + w_2$ with some $v_i, w_i \in W_i$, $i = 1, 2$. Now, if P is orthogonal, then $\langle Pv, w\rangle = \langle v_2, w\rangle = \langle v_2, w_2\rangle = \langle v, w_2\rangle = \langle v, Pw\rangle$. If P is not orthogonal, then there are $v_1 \in W_1$, $w_2 \in W_2$ so that $\langle v_1, w_2\rangle \neq 0$. Then $\langle v_1, Pw_2\rangle \neq 0 = \langle Pv_1, w_2\rangle$.

11.17 Definition (Unitary/Orthogonal Equivalence)

Two complex matrices $A, B \in \mathbb{C}^{n\times n}$ are said to be *unitary equivalent* if $B = P^{-1}AP$ for some unitary matrix P, i.e. we also have $B = P^*AP$.

Two real matrices $A, B \in \mathbb{R}^{n\times n}$ are said to be *orthogonally equivalent* if $B = P^{-1}AP$ for some orthogonal matrix P, i.e. we also have $B = P^tAP$.

11.18 Remark

A coordinate change between two ONB's is represented by a unitary (resp. orthogonal) matrix, cf. 9.22. Therefore, for any linear operator $T : V \to V$ and ONB's B, B' the matrices $[T]_B$ and $[T]_{B'}$ are unitary (resp., orthogonally) equivalent. Conversely, two matrices A, B are unitary (resp., orthogonally) equivalent iff they represent one linear operator in some two ONB's.

11.19 Remark

Any unitary matrix U is unitary equivalent to a diagonal matrix D (which is also unitary), recall 10.11.

11.20 Theorem (Spectral Theorem)

Let $T : V \to V$ be a selfadjoint operator. Then there is an ONB consisting entirely of eigenvectors of T.

Proof goes by induction on $n = \dim V$, just as the proof of 10.11. You need to use 11.10 and 11.11.

11.21 Corollary
Any complex Hermitean matrix is unitary equivalent to a diagonal matrix (with real diagonal entries). Any real symmetric matrix is orthogonally equivalent to a diagonal matrix.

11.22 Lemma
If a complex (real) matrix A is unitary (resp., orthogonally) equivalent to a diagonal matrix with real diagonal entries, then A is Hermitean (resp., symmetric).

Proof. If $A = P^{-1}DP$ with $P^{-1} = P^*$, then $A^* = P^*D^*(P^{-1})^* = A$, because $D^* = D$ (as a real diagonal matrix). □

11.23 Corollary
If a linear operator T has an ONB consisting of eigenvectors and all its eigenvalues are real, then T is selfadjoint.

11.24 Corollary
If an operator T is selfadjoint and invertible, then so is T^{-1}. If a matrix A is selfadjoint and nonsingular, then so is A^{-1}.

Proof. By the Spectral Theorem 11.20, there is an ONB B consisting of eigenvectors of T. Now T^{-1} has the same eigenvectors, and its eigenvalues are the reciprocals of those of T, hence they are real, too. Therefore, T^{-1} is selfadjoint. □

12 Bilinear Forms, Positive Definite Matrices

12.1 Definition (Bilinear Form)

A *bilinear form* on a complex vector space V is a mapping $f : V \times V \to \mathbb{C}$ such that

$$f(u_1 + u_2, v) = f(u_1, v) + f(u_2, v)$$

$$f(cu, v) = cf(u, v)$$

$$f(u, v_1 + v_2) = f(u, v_1) + f(u, v_2)$$

$$f(u, cv) = \bar{c}f(u, v)$$

for all vectors $u, v, u_i, v_i \in V$ and scalars $c \in \mathbb{C}$. In other words, f is linear in the first argument and conjugate linear in the second.

A bilinear form on a real vector space is a mapping $f : V \times V \to \mathbb{R}$ that satisfies the same properties, except c is a real scalar and so $\bar{c} = c$.

12.2 Example

If V is an inner product space and $T : V \to V$ a linear operator, then $f(u, v) := \langle Tu, v\rangle$ is a bilinear form.

12.3 Theorem

Let V be a finite dimensional inner product space. Then for every bilinear form f on V, then there is a unique linear operator $T : V \to V$ such that

$$f(u, v) = \langle Tu, v\rangle \qquad \forall u, v \in V$$

Proof. For every $v \in V$ the function $g(u) = f(u, v)$ is linear in u, so by the Riesz representation theorem 11.1 there is a vector $w \in V$ such that $f(u, v) = \langle u, w\rangle$. Define a map $S : V \to V$ by $Sv = w$. It is then a routine check that S is linear. Setting $T = S^*$ proves the existence. The uniqueness is obvious. □

12.4 Example

All bilinear forms on $\mathbb{C}^n$ are of the type $f(x, y) = \langle Ax, y\rangle$ with $A \in \mathbb{C}^{n\times n}$, i.e.

$$f(x, y) = \sum_{ij} A_{ji}x_i\bar{y}_j$$

12.5 Definition (Hermitean/Symmetric Form)

A bilinear form f on a complex (real) vector space V is called Hermitean (resp., symmetric) if

$$f(u, v) = \overline{f(v, u)} \qquad \forall u, v \in V$$

In the real case, the bar can be dropped.

For a Hermitean or symmetric form f, the function $q : V \to \mathbb{R}$ defined by $q(u) := f(u,u)$ is called the *quadratic form* associated with f. Note that $q(u) \in \mathbb{R}$ even in the complex case, because $f(u,u) = \overline{f(u,u)}$.

12.6 Theorem

A linear operator $T : V \to V$ is selfadjoint if and only if the bilinear form $f(u,v) = \langle Tu, v\rangle$ is Hermitean (symmetric, in the real case).

Proof. If T is selfadjoint, then $f(u,v) = \langle Tu, v\rangle = \langle u, Tv\rangle = \overline{\langle Tv, u\rangle} = \overline{f(v,u)}$. If f is Hermitean, then $\langle u, Tv\rangle = \overline{\langle Tv, u\rangle} = \overline{f(v,u)} = f(u,v) = \langle Tu, v\rangle = \langle u, T^*v\rangle$, hence $T = T^*$. □

12.7 Lemma

Let V be a complex inner product space and $T, S : V \to V$ a linear operator. If $\langle Tu, u\rangle = \langle Su, u\rangle$ for all $u \in V$, then $T = S$.

Note: This lemma holds only in complex spaces, it fails in real spaces.

Proof. Let B be an ONB. Then $\langle Tu, u\rangle = [u]_B^t [T]_B^t \overline{[u]_B}$, and the same holds for S. It is then enough to prove that if for some $A, B \in \mathbb{C}^{n\times n}$ we have $x^t A\bar{x} = x^t B\bar{x}$ for all $x \in \mathbb{C}^n$, then $A = B$. Equivalently, if $x^t A\bar{x} = 0$ for all $x \in \mathbb{C}^n$, then $A = 0$. This is done in the homework assignment.

12.8 Corollary

If f is a bilinear form in a complex vector space V such that $f(u,u) \in \mathbb{R}$ for all vectors $u \in V$, then f is Hermitean.

Proof. By 12.3, there is an operator $T : V \to V$ such that $f(u,v) = \langle Tu, v\rangle$. Then we have $\langle Tu, u\rangle = f(u,u) = \overline{f(u,u)} = \overline{\langle Tu, u\rangle} = \langle u, Tu\rangle = \langle T^*u, u\rangle$, hence $T = T^*$ by 12.7. Then apply 12.6. □

12.9 Definition (Positive Definite Form/Matrix)

A Hermitean (symmetric) bilinear form f on a vector space V is said to be *positive definite* if $f(u,u) > 0$ for all $u \neq 0$.

A selfadjoint operator $T : V \to V$ is said to be *positive definite* if $\langle Tu, u\rangle > 0$ for all $u \neq 0$.

A selfadjoint matrix A is said to be *positive definite* if $x^t A\bar{x} > 0$ for all $x \neq 0$.

By replacing "> 0" with "≥ 0", one gets *positive semi-definite* forms/operators/matrices.

Note: in the complex cases the Hermitean requirement can be dropped, because $f(u,u) > 0$ implies $f(u,u) \in \mathbb{R}$, see 12.8.

12.10 Theorem

Let A be a complex or real $n \times n$ matrix. The following are equivalent:
(a) A is positive definite
(b) $f(x, y) := x^t A\bar{y}$ defines an inner product in $\mathbb{C}^n$ (resp., $\mathbb{R}^n$)
(c) A is Hermitean (resp., symmetric) and all its eigenvalues are positive.

Proof. (a)$\Leftrightarrow$(b) is a routine check. To prove (a)$\Leftrightarrow$(c), find an ONB consisting of eigenvectors $u_1, \ldots, u_n$ of A (one exists by 11.20). Then for any $v = \sum c_i u_i$ we have $\langle Av, v\rangle = \sum \lambda_i |c_i|^2$. Hence, $\langle Av, v\rangle > 0\ \forall v$ if and only if $\lambda_i > 0\ \forall i$. □

Remark: Let A be a Hermitean (symmetric) matrix. Then it is positive semidefinite if and only if its eigenvalues are nonnegative.

12.11 Corollary

If a matrix A is positive definite, then so is A^{-1}. If an operator T is positive definite, then so is T^{-1}.

There is a very simple and efficient way to take square roots of positive definite matrices.

12.12 Definition (Square Root)

An $n \times n$ matrix B is called a *square root* of an $n \times n$ matrix A if $A = B^2$. Notation: $B = A^{1/2}$.

12.13 Lemma

If A is a positive definite matrix, then there is a square root $A^{1/2}$ which is also positive definite.

Proof. By 11.21 and 12.10, $A = P^{-1}DP$, where D is a diagonal matrix with positive diagonal entries and P a unitary (orthogonal) matrix. If $D = \operatorname{diag}(d_1, \ldots, d_n)$, then denote $D^{1/2} = \operatorname{diag}(\sqrt{d_1}, \ldots, \sqrt{d_n})$. Now $A = B^2$ where $B = P^{-1}D^{1/2}P$, and B is selfadjoint by 11.22. □

Remark. If A is positive semidefinite, then there is a square root $A^{1/2}$ which is also positive semidefinite.

12.14 Corollary

An $n \times n$ matrix A is positive definite if and only if there is a nonsingular matrix B such that $A = B^*B$.

Proof. "$\Rightarrow$" follows from 12.13. If $A = B^*B$, then $A^* = B^*(B^*)^* = A$ and

$\langle Ax, x\rangle = \langle Bx, Bx\rangle > 0$ for any $x \neq 0$, because B is nonsingular. $\square$

Remark. An $n \times n$ matrix A is positive semi-definite if and only if there is a matrix B such that $A = B^*B$.

12.15 Lemma (Rudin)

For any matrix $A \in \mathbb{R}^{n\times n}$ there is a symmetric positive semi-definite matrix B such that
(i) $A^tA = B^2$
(ii) $||Ax|| = ||Bx||$ for all $x \in \mathbb{R}^n$
(One can think of B as a 'rectification' or 'symmetrization' of A.)

Proof. The matrix A^tA is symmetric positive semidefinite by the previous remark, so it has a symmetric square root B by 12.13. Lastly, $\langle Bx, Bx\rangle = \langle B^2x, x\rangle = \langle A^tAx, x\rangle = \langle Ax, Ax\rangle$.

12.16 Remark

Let $A \in \mathbb{R}^{n\times n}$ be symmetric. Then, by 11.20, there is an ONB $\{u_1, \ldots, u_n\}$ consisting entirely of eigenvectors of A. Denote by $\lambda_1, \ldots, \lambda_n$ the corresponding eigenvalues of A. Then

$$A = \sum_{i=1}^{n} \lambda_i u_i u_i^t$$

Proof. Just verify that $Au_i = \lambda_i u_i$ for all $1 \leq i \leq n$.

13 Cholesky Factorization

This is the continuation of Section 4 (LU Decomposition). We explore other decompositions of square matrices.

13.1 Theorem (LDM^t Decomposition)
Let A be an $n \times n$ matrix with nonsingular principal minors, i.e. $\det A_k \neq 0$ for $k = 1, \ldots, n$. Then there are unique matrices L, D, M such that L, M are unit lower triangular and D is diagonal, and

$$A = LDM^t$$

Proof. By the LU decomposition (Theorem 4.10) there are unit lower triangular matrix L and upper triangular matrix U such that $A = LU$. Let $u_{11}, \ldots, u_{nn}$ be the diagonal entries of U, and set $D = \operatorname{diag}(u_{11}, \ldots, u_{nn})$. Then the matrix $M^t := D^{-1}U$ is unit upper triangular, and $A = LDM^t$.

To establish uniqueness, let $A = LDM^t = L_1 D_1 M_1^t$. By the uniqueness of the LU decomposition 4.10, we have $L = L_1$. Hence, $(D_1^{-1}D)M^t = M_1^t$. Since both M^t and M_1^t are unit upper triangular, the diagonal matrix $D_1^{-1}D$ must be the identity matrix. Hence, $D = D_1$, and then $M = M_1$. □

13.2 Corollary
If, in addition, A is symmetric, then there exist unique unit lower triangular matrix L and diagonal D such that

$$A = LDL^t$$

Proof. By the previous theorem $A = LDM^t$. Then $A = A^t = MDL^t$, and by the uniqueness of the LDM^t decomposition we have $L = M$. □

13.3 Theorem (Sylvester's Theorem)
Let $A \in \mathbb{R}^{n \times n}$ be symmetric. Then A is positive definite if and only if $\det A_k > 0$ for all $k = 1, \ldots, n$.

Proof. Let A be positive definite. By 12.14, $\det A = (\det B)^2 > 0$. Any principal minor A_k is also a symmetric positive definite matrix, therefore by the same argument $\det A_k > 0$. Conversely, let $\det A_k > 0$. By Corollary 13.2 we have $A = LDL^t$. Denote by L_k and D_k the k-th principal minors of L and D, respectively. Then $A_k = L_k D_k L_k^t$. Note that $\det D_k = \det A_k > 0$ for all $k = 1, \ldots, n$, therefore all the diagonal entries of D are positive. Lastly, $\langle Ax, x\rangle = \langle DL^t x, L^t x\rangle = \langle Dy, y\rangle > 0$ because $y = L^t x \neq 0$ whenever $x \neq 0$. □

13.4 Corollary

Let A be a real symmetric positive definite matrix. Then $A_{ii} > 0$ for all $i = 1, \ldots, n$. Furthermore, let $1 \leq i_1 < i_2 < \cdots < i_k \leq n$, and let A' be the $k \times k$ matrix formed by the intersections of the rows and columns of A with numbers $i_1, \ldots, i_k$. Then $\det A' > 0$.

Proof. Just reorder the coordinates in $\mathbb{R}^n$ so that A' becomes a principal minor.

13.5 Theorem (Cholesky Factorization)

Let $A \in \mathbb{R}^{n \times n}$ be symmetric and positive definite. Then there exists a unique lower triangular matrix G with positive diagonal entries such that

$$A = GG^t$$

Proof. By Corollary 13.2 we have $A = LDL^t$. Let $D = \operatorname{diag}(d_1, \ldots, d_n)$. As it was shown in the proof of 13.3, all $d_i > 0$. Let $D^{1/2} = \operatorname{diag}(\sqrt{d_1}, \ldots, \sqrt{d_n})$. Then $D = D^{1/2}D^{1/2}$ and setting $G = LD^{1/2}$ gives $A = GG^t$. The diagonal entries of G are $\sqrt{d_1}, \ldots, \sqrt{d_n}$, so they are positive. To establish uniqueness, let $A = GG^t = G_1G_1^t$. Then $G_1^{-1}G = G_1^t(G^t)^{-1}$. Since this is the equality of a lower triangular matrix and an upper triangular one, then both matrices are diagonal: $G_1^{-1}G = G_1^t(G^t)^{-1} = D$. Hence, $G_1 = GD^{-1}$ and $G_1^t = DG^t$. Therefore, $D = D^{-1}$, so the diagonal entries of D are all ± 1. Note that the value -1 is not possible since the diagonal entries of both G and G_1 are positive. Hence, $D = I$, and so $G_1 = G$. $\square$

13.6 Algorithm (Cholesky Factorization).

Here we outline the algorithm of computing the matrix $G = (g_{ij})$ from the matrix $A = (a_{ij})$. Note that G is lower triangular, so $g_{ij} = 0$ for $i < j$. Hence,

$$a_{ij} = \sum_{k=1}^{\min\{i,j\}} g_{ik}g_{jk}$$

Setting $i = j = 1$ gives $a_{11} = g_{11}^2$, so

$$g_{11} = \sqrt{a_{11}}$$

(remember, g_{ii} must be positive). Next, for $2 \leq i \leq n$ we have $a_{i1} = g_{i1}g_{11}$, hence

$$g_{i1} = a_{i1}/g_{11} \qquad i = 2, \ldots, n$$

This gives the first column of G. Now, inductively, assume that we already have the first $j - 1$ columns of G. Then $a_{jj} = \sum_{k=1}^{j} g_{jk}^2$, hence

$$g_{jj} = \sqrt{a_{jj} - \sum_{k=1}^{j-1} g_{jk}^2}$$

Next, for $j+1 \le i \le n$ we have $a_{ij} = \sum_{k=1}^{j} g_{ik} g_{jk}$, hence

$$g_{ij} = \frac{1}{g_{jj}} \left(a_{ij} - \sum_{k=1}^{j-1} g_{ik} g_{jk} \right)$$

13.7 Cost of computation

The cost of computation is measured in flops, where a flop is a multiplication or division together with one addition or subtraction, recall 4.12. The above algorithm of computation of g_{ij} takes j flops for each $i = j, \ldots, n$, so the total is

$$\sum_{j=1}^{n} j(n-j) \approx n\frac{n^2}{2} - \frac{n^3}{3} = \frac{n^3}{6}$$

It also takes n square root extractions. Recall that the LU decomposition takes about $n^3/3$ flops, so the Cholesky factorization is nearly twice as efficient. It is also more stable than the LU decomposition.

13.8 Remark

The above algorithm can be used to verify that a given symmetric matrix, A, is positive definite. Whenever the square root extractions in 13.6 are all possible and non zero, i.e. whenever

$$a_{11} > 0 \quad \text{and} \quad a_{jj} - \sum_{k=1}^{j-1} g_{jk}^2 > 0 \quad \forall j \ge 2$$

the matrix A is positive definite.

14 Machine Arithmetic

This section is unusual. It discusses imprecise calculations. It may be difficult or frustrating for the students who were not much involved in computer work. But it might be easy (or trivial) for people experienced with computers. This section is necessary, since it motivates the discussion in the following section.

14.1 Binary numbers

A *bit* is a *b*inary dig*it*, it can only take two values: 0 and 1. Any natural number N can be written, in the binary system, as a sequence of binary digits:

$$N = (d_n \cdots d_1 d_0)_2 = 2^n d_n + \cdots + 2d_1 + d_0$$

For example, $5 = 101_2$, $11 = 1011_2$, $64 = 1000000_2$, etc.

14.2 More binary numbers

To represent more numbers in the binary system (not just positive integers), it is convenient to use the numbers $\pm N \times 2^{\pm M}$ where M and N are positive integers. Clearly, with these numbers we can approximate any real number arbitrarily accurately. In other words, the set of numbers $\{\pm N \times 2^{\pm M}\}$ is dense on the real line. The number $E = \pm M$ is called the *exponent* and N the *mantissa*.

Note that $N \times 2^M = (2^k N) \times 2^{M-k}$, so the same real number can be represented in many ways as $\pm N \times 2^{\pm M}$.

14.3 Floating point representation

We return to our decimal system. Any decimal number (with finitely many digits) can be written as

$$f = \pm .d_1 d_2 \ldots d_t \times 10^e$$

where d_i are decimal digits and e is an integer. For example, $18.2 = .182 \times 10^2 = .0182 \times 10^3$, etc. This is called a *floating point* representation of decimal numbers. The part $.d_1 \ldots d_t$ is called the *mantissa* and e is the *exponent*. By changing the exponent e with a fixed mantissa $, d_1 \ldots d_t$ we can move ("float") the decimal point, for example $.182 \times 10^2 = 18.2$ and $.182 \times 10^1 = 1.82$.

14.4 Normalized floating point representation

To avoid unnecessary multiple representations of the same number (as 18.2 by $.182 \times 10^2$ and $.0182 \times 10^3$ above), we require that $d_1 \neq 0$. We say the floating point representation is *normalized* if $d_1 \neq 0$. Then $.182 \times 10^2$ is the only normalized representation of the number 18.2.

For every positive real $f > 0$ there is a unique integer $e \in \mathbb{Z}$ such that $g := 10^{-e} f \in [0.1, 1)$. Then $f = g \times 10^e$ is the normalized representation of f. So, the normalized representation is unique.

14.5 Other number systems

Now suppose we are working in a number system with base $\beta \geq 2$. By analogy with 14.3, the floating point representation is

$$f = \pm .d_1 d_2 \dots d_t \times \beta^e$$

where $0 \leq d_i \leq \beta - 1$ are digits,

$$.d_1 d_2 \dots d_t = d_1 \beta^{-1} + d_2 \beta^{-2} + \cdots d_t \beta^{-t}$$

is the mantissa and $e \in \mathbb{Z}$ is the exponent. Again, we say that the above representation is normalized if $d_1 \neq 0$, this ensures uniqueness.

14.6 Machine floating point numbers

Any computer can only handle finitely many numbers. Hence, the number of digits d_i's is necessarily bounded, and the possible values of the exponent e are limited to a finite interval. Assume that the number of digits t is fixed (it characterizes the accuracy of machine numbers) and the exponent is bounded by $L \leq e \leq U$. Then the parameters β, t, L, U completely characterize the set of numbers that a particular machine system can handle. The most important parameter is t, the number of significant digits, or the length of the mantissa. (Note that the same computer can use many possible machine systems, with different values of β, t, L, U, see 14.8.)

14.7 Remark

The maximal (in absolute value) number that a machine system can handle is $M = \beta^U(1 - \beta^{-t})$. The minimal positive number is $m = \beta^{L-1}$.

14.8 Examples

Most computers use the binary system, $\beta = 2$. Many modern computers (e.g., all IBM compatible PC's) conform to the IEEE floating-point standard (ANSI/IEEE Standard 754-1985). This standard provides two systems. One is called *single precision*, it is characterized by $t = 24$, $L = -125$ and $U = 128$. The other is called *double precision*, it is characterized by $t = 53$, $L = -1021$ and $U = 1024$. The PC's equipped with the so called *numerical coprocessor* also use an internal system called *temporary format*, it is characterized by $t = 65$, $L = -16381$ and $U = 16384$.

14.9 Relative errors

Let $x > 0$ be a positive real number with the normalized floating point representation with base β

$$x = .d_1 d_2 \dots \times \beta^e$$

where the number of digits may be finite or infinite. We need to represent x in a machine system with parameters β, t, L, U. If $e > U$, then x cannot be represented (an attempt to store x in the computer memory or perform calculation that results in x should terminate

the computer program with error message OVERFLOW – a number too large). If $e < L$, the system may either represent x by 0 (quite reasonably) or terminate the program with error message UNDERFLOW – a number too small. If $e \in [L, U]$ is within the right range, then the matissa has to be reduced to t digits (if it is longer or infinite). There are two standard ways to do that reduction:
(i) just take the first t digits of the mantissa of x, i.e. $.d_1 \dots d_t$, and chop off the rest;
(ii) round off to the nearest available, i.e. take the mantissa

$$\begin{cases} .d_1 \dots d_t & \text{if } \ d_{t+1} < \beta/2 \\ .d_1 \dots d_t + .0 \dots 01 & \text{if } \ d_{t+1} \geq \beta/2 \end{cases}$$

Denote the obtained number by x_c (the computer representation of x). The relative error in this representation can be estimated as

$$\frac{x_c - x}{x} = \varepsilon \quad \text{or} \quad x_c = x(1+\varepsilon)$$

where the maximal possible value of ε is

$$\mathbf{u} = \begin{cases} \beta^{1-t} & \text{for chopped arithmetic} \\ \frac{1}{2}\beta^{1-t} & \text{for rounded arithmetic} \end{cases}$$

The number $\mathbf{u}$ is called the *unit round off* or *machine precision.*

14.10 Examples

a) For the IEEE floating-point standard with chopped arithmetic in single precision we have $\mathbf{u} = 2^{-23} \approx 1.2 \times 10^{-7}$. In other words, approximately 7 decimal digits are accurate.

b) For the IEEE floating point standard with chopped arithmetic in double precision we have $\mathbf{u} = 2^{-52} \approx 2.2 \times 10^{-16}$. In other words, approximately 16 decimal digits are accurate.

14.11 Computational errors

Let x, y be two real numbers represented in a machine system by x_c, y_c. An arithmetic operation $x * y$, where $*$ is one of $+, -, \times, \div$, is performed by a computer in the following way. The computer finds $x_c * y_c$ (first, exactly) and then represents that number by the machine system. The result is $z := (x_c * y_c)_c$. Note that, generally, z is different from $(x * y)_c$, which is the representation of the exact result $x * y$. Hence, z is not necessarily the best representation for $x * y$. In other words, the computer makes additional round off errors during computations. Assuming that $x_c = x(1+\varepsilon_1)$ and $y_c = y(1+\varepsilon_2)$ we have

$$(x_c * y_c)_c = (x_c * y_c)(1+\varepsilon_3) = [x(1+\varepsilon_1)] * [y(1+\varepsilon_2)](1+\varepsilon_3)$$

where $|\varepsilon_1|, |\varepsilon_2|, |\varepsilon_3| \leq \mathbf{u}$.

14.12 Multiplication and Division

For multiplication, we have

$$z = xy(1+\varepsilon_1)(1+\varepsilon_2)(1+\varepsilon_3) \approx xy(1+\varepsilon_1+\varepsilon_2+\varepsilon_3)$$

so the relative error is (approximately) bounded by $3\mathbf{u}$. A similar estimate can be made in the case of division.

14.13 Addition and Subtraction

For addition, we have

$$z = (x+y+x\varepsilon_1+y\varepsilon_2)(1+\varepsilon_3) = (x+y)\left(1+\frac{x\varepsilon_1+y\varepsilon_2}{x+y}\right)(1+\varepsilon_3)$$

The relative error is now small if $|x|$ and $|y|$ are not much bigger than $|x+y|$. The error, however, can be arbitrarily large if $|x+y| \ll \max\{|x|,|y|\}$. This effect is known as *catastrophic cancellation*. A similar estimate can be made in the case of subtraction $x-y$: if $|x-y|$ is not much smaller than $|x|$ or $|y|$, then the relative error is small, otherwise we may have a catastrophic cancellation.

14.14 Quantitative estimation

Assume that $z = x+y$ is the exact sum of x and y. Let $x_c = x+\Delta x$ and $y_c = y+\Delta y$ be the machine representations of x and y. Let $z+\Delta z = x_c+y_c$. We want to estimate the relative error $|\Delta z|/|z|$ in terms of the relative errors $|\Delta x|/|x|$ and $|\Delta y|/|y|$:

$$\frac{|\Delta z|}{|z|} \le \frac{|x|}{|x+y|}\frac{|\Delta x|}{|x|} + \frac{|y|}{|x+y|}\frac{|\Delta y|}{|y|}$$

In particular, assume that $|\Delta x|/|x| \le \mathbf{u}$ and $|\Delta y|/|y| \le \mathbf{u}$, so that x_c and y_c are the best possible machine representations of x and y. Then

$$\frac{|\Delta z|}{|z|} \le \frac{|x|+|y|}{|x+y|}\mathbf{u}$$

so that the value $q := (|x|+|y|)/|x+y|$ characterizes the accuracy of the machine addition of x and y. More precisely, if $q \ll \mathbf{u}^{-1}$, then the result is fairly accurate. On the contrary, if $q \approx \mathbf{u}^{-1}$ or higher, then a catastrophic cancellation occurs.

14.15 Example

Consider the system of equations

$$\begin{pmatrix} 0.01 & 2 \\ 1 & 3 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 2 \\ 4 \end{pmatrix}$$

The exact solution is

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 200/197 \\ 196/197 \end{pmatrix} = \begin{pmatrix} 1.015\ldots \\ 0.995\ldots \end{pmatrix}$$

One can solve the system with the chopped arithmetic with base $\beta = 10$ and $t = 2$ (i.e. working with a two digit mantissa) by the LU decomposition without pivoting. One gets $(x_c, y_c) = (0, 1)$. Increasing the accuracy to $t = 3$ gives $(x_c, y_c) = (2, 0.994)$, not much of improvement, since x_c is still far off.

By applying partial pivoting one gets $(x_c, y_c) = (1, 1)$ for $t = 2$ and $(x_c, y_c) = (1.02, 0, 994)$ for $t = 3$, almost perfect accuracy!

One can see that solving linear systems in machine arithmetic is different from solving them exactly. We 'pretend' that we are computers subject to the strict rules of machine arithmetic, in particular we are limited to t significant digits. As we notice, these limitations may lead to unexpectedly large errors in the end.

15 Sensitivity and Stability

One needs to avoid catastrophic cancellations. At the very least, one wants to have means to determine whether catastrophic cancellations can occur. Here we develop such means for the problem of solving systems of linear equations $Ax = b$.

15.1 Convention

Let $||\cdot||$ be a norm in $\mathbb{R}^n$. If $x \in \mathbb{R}^n$ is a vector and $x + \Delta x \in \mathbb{R}^n$ is a nearby vector (say, a machine representation of x), we consider $||\Delta x||/||x||$ as the relative error (in x). If $A \in \mathbb{R}^{n\times n}$ is a matrix and $A + \Delta A \in \mathbb{R}^{n\times n}$ a nearby matrix, we consider $||\Delta A||/||A||$ as the relative error (in A). Here $||\cdot||$ is the matrix norm induced by the norm $||\cdot||$ in $\mathbb{R}^n$.

15.2 Definition (Condition Number)

For a nonsingular matrix A, the condition number with respect to the given matrix norm $||\cdot||$ is

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

We denote by $\kappa_1(A)$, $\kappa_2(A)$, $\kappa_\infty(A)$ the condition numbers with respect to the 1-norm, 2-norm and ∞-norm, respectively.

15.3 Theorem

Suppose we have

$$\begin{aligned} Ax &= b \\ (A + \Delta A)(x + \Delta x) &= b + \Delta b \end{aligned}$$

with a nonsingular matrix A. Assume that $||\Delta A||$ is small so that $||\Delta A||\,||A^{-1}|| < 1$. Then

$$\frac{||\Delta x||}{||x||} \le \frac{\kappa(A)}{1 - \kappa(A)\frac{||\Delta A||}{||A||}} \left(\frac{||\Delta A||}{||A||} + \frac{||\Delta b||}{||b||} \right)$$

Proof. Expanding out the second equation, subtracting the first one and multiplying by A^{-1} gives

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

Taking norms and using the triangular inequality and 8.10 gives

$$||\Delta x|| \le ||A^{-1}||\,||\Delta A||\big(||x|| + ||\Delta x||\big) + ||A^{-1}||\,||\Delta b||$$

Using $||b|| \le ||A||\,||x||$, this rearranges to

$$\big(1 - ||A^{-1}||\,||\Delta A||\big)||\Delta x|| \le \left(||A^{-1}||\,||\Delta A|| + ||A^{-1}||\,||\Delta A|| \frac{||\Delta b||}{||b||} \right) ||x||$$

Recall that $||\Delta A||\,||A^{-1}|| < 1$, so the first factor above is positive. The theorem follows immediately. □

Interpretation. Let $Ax = b$ be a given system of linear equations to be solved numerically. A computer represents A by $A_c = A + \Delta A$ and b by $b_c = b + \Delta b$ and then solves the system $A_c x = b_c$. Assume now (ideally) that the computer finds an exact solution $x_c = x + \Delta x$, i.e., $A_c x_c = b_c$. Then the relative error $||\Delta x||/||x||$ can be estimated by 15.3. The smaller the condition number $\kappa(A)$, the tighter (better) estimate on $||\Delta x||/||x||$ we get. The value of $\kappa(A)$ thus characterizes the *sensitivity* of the solution of the linear system $Ax = b$ to small errors in A and b.

15.4 Corollary

Assume that in Theorem 15.3 we have $||\Delta A|| \leq \mathbf{u}||A||$ and $||\Delta b|| \leq \mathbf{u}||b||$, i.e. the matrix A and the vector b are represented with the best possible machine accuracy. Then

$$\frac{||\Delta x||}{||x||} \leq \frac{2\mathbf{u}\kappa(A)}{1 - \mathbf{u}\kappa(A)}$$

Interpretation. If $\mathbf{u}\kappa(A) \ll 1$, then the numerical solution of $Ax = b$ is quite accurate in the ideal case (when the computer solves the system $A_c x = b_c$ exactly). If, however, $\mathbf{u}\kappa(A) \approx 1$ or > 1, then the numerical solution is completely unreliable, no matter how accurately the computer works. Comparing this to 14.14, we can interpret $\kappa(A)$ as a quantitative indicator of the possibility of catastrophic cancellations.

Linear systems $Ax = b$ with $\kappa(A) \ll \mathbf{u}^{-1}$ are often called *well-conditioned*. Those with $\kappa(A)$ of order $\mathbf{u}^{-1}$ or higher are called *ill-conditioned*. In practical applications, when one runs (or may run) into an ill-conditioned linear system, one needs to reformulate the underlying problem rather than trying to use numerical tricks to deal with the ill-conditioning. We will face this problem in Section 16.

15.5 Example.

Assume that $\mathbf{u} \approx 10^{-l}$, i.e. the machine system provides l accurate digits. Then if $\kappa(A) \approx 10^k$ with $k < l$, then $||\Delta x||/||x|| \leq 10^{l-k}$, i.e. even an ideal numerical solution only provides $l - k$ accurate digits.

15.6 Proposition

1. $\kappa(\lambda A) = \kappa(A)$ for $\lambda \neq 0$
2. $$\kappa(A) = \frac{\max_{||x||=1} ||Ax||}{\min_{||x||=1} ||Ax||}$$
3. If a_j denotes the j-th column of A, then $\kappa(A) \geq ||a_j||/||a_i||$
4. $\kappa_2(A) = \kappa_2(A^t)$
5. $\kappa(I) = 1$

6. $\kappa(A) \geq 1$
7. For any orthogonal matrix Q,

$$\kappa_2(QA) = \kappa_2(AQ) = \kappa_2(A)$$

8. If $D = \operatorname{diag}(d_1, \ldots, d_n)$ then

$$\kappa_2(D) = \kappa_1(D) = \kappa_\infty(D) = \frac{\max_{1 \leq i \leq n} |d_i|}{\min_{1 \leq i \leq n} |d_i|}$$

9. If λ_M is the largest eigenvalue of A^tA and λ_m is its smallest eigenvalue, then

$$\kappa_2(A^tA) = \kappa_2(AA^t) = (\kappa_2(A))^2 = \lambda_M/\lambda_m$$

10. $\kappa_2(A) = 1$ if and only if A is a multiple of an orthogonal matrix.

15.7 Example

Let $A = \begin{pmatrix} 1 & 0 \\ 0 & \varepsilon \end{pmatrix}$. Then $\kappa_2(A) = 1/\varepsilon$ by 15.6 (8), and so the system $Ax = b$ is ill-conditioned. It is, however, easy to 'correct' the system by multiplying the second equation by $1/\varepsilon$. Then the new system $A'x' = b'$ will be perfectly conditioned, since $\kappa_2(A') = 1$ by 15.6 (5). The trick of multiplying rows (and columns) by scalars is called the *scaling* of the matrix A. Sometimes, scaling allows to significantly reduce the condition number of A. Such *scalable* matrices are, however, not very often in practice. Besides, no satisfactory method exists for detecting scalable matrices.

15.8 Remark

Another way to look at the condition number $\kappa(A)$ is the following. Since in practice the exact solution x of the system $Ax = b$ is rarely known, one can try to verify whether the so called *residual vector* $r = b - Ax_c$ is small relative to b. Since $Ax_c = b + r$, Theorem 15.3 with $\Delta A = 0$ implies that

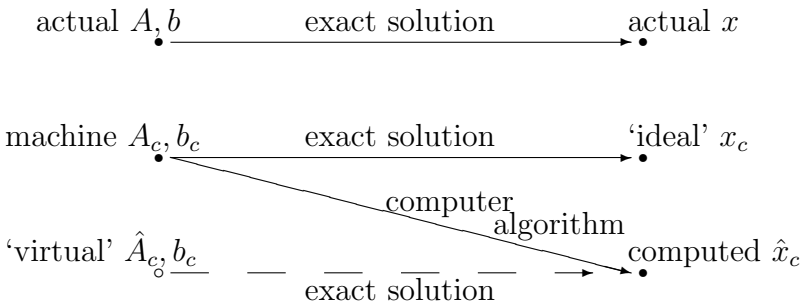
$$\frac{||x_c - x||}{||x||} \leq \kappa(A) \frac{||r||}{||b||}$$

If A is well conditioned, the smallness of $||r||/||b||$ ensures the smallness of the relative error $||x_c - x||/||x||$. If A is ill-conditioned, this does not work: one can find x_c far from x for which r is still small.

15.9 Remark

In 15.3–15.8, we assumed that x_c was an exact solution of the system $A_cx = b_c$. We now get back to reality and consider the numerical solution $\hat{x}_c$ of the system $A_cx = b_c$ obtained by a computer. Because of computational errors (see 14.11–14.13) the vector $\hat{x}_c$ will not be an exact solution, i.e. we have $A_c\hat{x}_c \neq b_c$. Hence, we have

$$\hat{x}_c - x = (\hat{x}_c - x_c) + (x_c - x) = \hat{\Delta}x + \Delta x$$



The error Δx is solely due to inaccurate representation of the input data, A and b, in the computer memory (by A_c and b_c). The error Δx is estimated in Theorem 15.3 with the help of the condition number $\kappa(A)$. If $\kappa(A)$ is too large, the error Δx may be too big, and one should not even attempt to solve the system numerically.

15.10 Round-off error analysis

Assume now that $\kappa(A)$ is small enough, so that we need not worry about Δ_c. Now we need to estimate $\hat{\Delta}_c = \hat{x}_c - x_c$. This results from computational errors, see 14.11–14.13. Note that even small errors may accumulate to a large error in the end.

In order to estimate $\hat{x}_c - x_c$, a typical approach is to find another matrix, $\hat{A}_c = A_c + \delta A$ such that $(A_c + \delta A)\hat{x}_c = b_c$. We call $A_c + \delta A$ a *virtual* matrix, since it is neither given nor computed numerically. Moreover, it is only specified by the fact that it takes the vector $\hat{x}_c$ to b_c, hence it far from being uniquely defined. One wants to find a virtual matrix as close to A_c as possible, to make δA small. Then one can use Theorem 15.3 to estimate $\hat{\Delta}_c$:

$$\frac{||\hat{x}_c - x_c||}{||x_c||} \leq \frac{\kappa(A_c)}{1 - \kappa(A_c)\frac{||\delta A||}{||A_c||}} \frac{||\delta A||}{||A_c||}$$

Clearly, the numerical solution $\hat{x}_c$, and then the virtual matrix $A_c + \delta A$, will depend on the method (algorithm) used. For example, one can use the LU decomposition method with or without pivoting. For a symmetric positive definite matrix A, one can use the Cholesky factorization.

We will say that the numerical method is (algebraically) *stable* if one can find a virtual matrix so that $||\delta A||/||A||$ is small, and *unstable* otherwise. The rule of thumb is then that for stable methods, computational errors do not accumulate too much and the numerical solution $\hat{x}_c$ is close to the ideal solution x_c. For unstable methods, computational errors may accumulate too much and force $\hat{x}_c$ to be far from x_c, this making the numerical solution $\hat{x}_c$ unreliable.

15.11 Wilkinson's analysis for the LU decomposition

Here we provide, without proof, an explicit estimate on the matrix δA for the LU

decomposition method described in Section 4. Denote by $L_c = (l^c_{ij})$ and $U_c = (u^c_{ij})$ the computed matrices L and U by the LU decomposition algorithm. Then

$$|\delta A_{ij}| \le n\mathbf{u}\left(3|a_{ij}| + 5\sum_{k=1}^{n} |l^c_{ik}|\,|u^c_{kj}|\right) + O(\mathbf{u}^2)$$

If the values of l^c_{ik} and u^c_{kj} are not too large compared to those of a_{ij}, this provides a good upper bound on $||\delta A||/||A||$, it is of order $n^2\mathbf{u}$. In this case the LU decomposition method is stable (assuming that n is not too big). But it may become unstable if $\sum |l^c_{ik}|\,|u^c_{kj}|$ is large compared to $|a_{ij}|$.

One can improve the stability by using partial pivoting. In this case $|l^c_{ij}| \le 1$. So, the trouble can only occur if one gets large elements of U_c. Practically, however, it is observed that $||U_c||_\infty \approx ||A||_\infty$, so that the partial pivoting algorithm is usually stable.

The LU decomposition with complete pivoting is always stable. In this case it can be proved that

$$\frac{||U_c||_\infty}{||A||_\infty} \le n^{1/2}(2^1 3^{1/2} 4^{1/3} \cdots n^{1/(n-1)})^{1/2}$$

This bound grows slowly with n and ensures the stability. (Exercise: show that this bound grows approximately like $n^{0.25\ln n}$.)

15.12 Remark

The Cholesky factorization $A = GG^t$ of a symmetric positive definite matrix A, see 13.5, is a particular form of the LU decomposition, so the above analysis applies. In this case, however, we know that

$$a_{ii} = \sum_{j=1}^{i} g_{ij}^2$$

see 13.6. Hence, the elements of G cannot be large compared to the elements of A. This proves that the Cholesky factorization is always stable.

15.13 Remark (Iterative Improvement)

In some practical applications, a special technique can be used to improve the numerical solution $\hat{x}_c$ of a system $Ax = b$. First one solves the system $A_c x = b_c$ numerically and then finds the residual vector $r = b_c - A_c\hat{x}_c$. Then one solves the system $A_c z = r_c$ for z and replaces $\hat{x}_c$ by $\hat{x}_c + z_c$. This is one iteration of the so called *iterative improvement* or *iterative refinement*. Note that solving the system $A_c z = r_c$ is relatively cheap, since the LU decomposition of A is obtained (and stored) already. Sometimes one uses the $2t$-digit arithmetic to compute r, if the solution $\hat{x}_c$ is computed in the t-digit arithmetic (one must have it available, though, which is not always the case).

It is known that by repeating the iterations k times with machine precision $\mathbf{u} = 10^{-d}$ and $\kappa_\infty = 10^q$ one expects to obtain approximately $\min(d, k(d-q))$ correct digits.

If one uses the LU decomposition with partial pivoting, then just one iteration of the above improvement makes the solution algebraically stable.

16 Overdetermined Linear Systems

16.1 Definition (Overdetermined Linear System)

A system of linear equations $Ax = b$ with $A \in \mathbb{C}^{m\times n}$, $x \in \mathbb{C}^n$ and $b \in \mathbb{C}^m$, is said to be *overdetermined* if $m > n$. Since there are more equations than unknowns, the system usually has no solutions. We will be concerned here exactly with that "no solution" situation.

Note that the matrix A defines a linear transformation $T_A : \mathbb{C}^n \to \mathbb{C}^m$. Then $Ax = b$ has a solution if and only if $b \in \operatorname{Im} T_A$. In this case the solution is unique if and only if $\operatorname{Ker} T_A = \{0\}$, i.e. the columns of A are linearly independent.

16.2 Definition (Least Squares Fit)

Let (x_i, y_i), $1 \le i \le m$, be given points on the real xy plane. For any straight line $y = a + bx$ one defines the "distance" of that line from the given points by

$$E(a, b) = \sum_{i=1}^{m} (a + bx_i - y_i)^2$$

This quantity is called the *residual sum of squares (RSS)*. Suppose that the line $y = \hat{a}+\hat{b}x$ minimizes the function $E(a, b)$, i.e.

$$E(\hat{a}, \hat{b}) = \min_{a,b} E(a, b)$$

Then $y = \hat{a} + \hat{b}x$ is called the *least squares approximation* to the given points (x_i, y_i). Finding the least squares approximation is called the *least squares fit* of a straight line to the given points. It is widely used in statistics.

16.3 Example

Let $\{x_i\}$ be the heights of some m people and $\{y_i\}$ their weights. One can expect an approximately linear dependence $y = a + bx$ of the weight from the height. The least squares fit of a line $y = a + bx$ to the experimental data (x_i, y_i), $1 \le i \le m$, gives the numerical estimates of the coefficients a and b in the formula $y = a + bx$.

Note: in many statistical applications the least squares estimates are the best possible (most accurate). There are, however, just as many exceptions, these questions are beyond the scope of this course.

16.4 Remark

For the least squares fit in 16.2, let

$$A = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_m \end{pmatrix} \quad \mathbf{x} = \begin{pmatrix} a \\ b \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}$$

Then

$$E(a, b) = ||\mathbf{b} - A\mathbf{x}||_2^2$$

Note that $E(a, b) \geq 0$. Also, $E(a, b) = 0$ if and only if $\mathbf{x} = \begin{pmatrix} a \\ b \end{pmatrix}$ is an exact solution of the system $A\mathbf{x} = \mathbf{b}$. The system $A\mathbf{x} = \mathbf{b}$ is overdetermined whenever $m > 2$, so exact solutions rarely exist.

16.5 Definition (Least Squares Solution)

Let $Ax = b$ be an overdetermined linear system. A vector $x \in \mathbb{C}^n$ that minimizes the function

$$E(x) = ||b - Ax||_2^2$$

is called a *least squares solution* of $Ax = b$.

16.6 Remark

If A, b and x are real, then one can easily differentiate the function $E(x)$ with respect to the components of the vector x and equate the derivatives to zero. Then one arrives at the linear system $A^tAx = A^tx$.

16.7 Definition (Normal Equations)

Let $Ax = b$ be an overdetermined linear system. Then the linear system

$$A^*Ax = A^*b$$

is called the *system of normal equations* associated with the overdetermined system $Ax = b$. In the real case, the system of normal equations is

$$A^tAx = A^tb$$

Note: For a matrix $A \in \mathbb{C}^{m \times n}$, its adjoint is defined by $A^* = \overline{A^t} = \bar{A}^t \in \mathbb{C}^{n \times m}$.

16.8 Lemma

The matrix A^*A is a square $n \times n$ Hermitean and positive semidefinite. If A has full rank, then A^*A is positive definite.

16.9 Theorem

Let $Ax = b$ be an overdetermined linear system.
(a) A vector x minimizes $E(x) = ||b - Ax||^2$ if and only if it is an exact solution of the system $Ax = \hat{b}$, where $\hat{b}$ is the orthogonal projection of b onto $\operatorname{Im} T_A$.
(b) A vector x minimizing $E(x)$ always exists. It is unique if and only if A has full rank, i.e. the columns of A are linearly independent.
(c) A vector x minimizes $E(x)$ if and only if it is a solution of the system of normal equations $A^*Ax = A^*b$.

Proof. The matrix A^* defines a linear transformation $T_{A^*} : \mathbb{C}^m \to \mathbb{C}^n$. Note that $(\operatorname{Im} T_A)^\perp = \operatorname{Ker} T_{A^*}$, which generalizes 11.7. The proof is a routine check, just as that of 11.7. So, we have an orthogonal decomposition $\mathbb{C}^m = \operatorname{Im} T_A \oplus \operatorname{Ker} T_{A^*}$. Then we can write $b = \hat{b} + r$ uniquely, where $\hat{b} \in \operatorname{Im} T_A$ and $r \in \operatorname{Ker} T_{A^*}$. Since $\langle \hat{b}, r \rangle = 0$, it follows from Theorem of Pythagoras that

$$||b - Ax||^2 = ||\hat{b} - Ax||^2 + ||r||^2 \geq ||r||^2$$

Hence, $\min_x E(x) = ||r||^2$ is attained whenever $Ax = \hat{b}$. Note that $\hat{b} \in \operatorname{Im} T_A$, so there is always an $x \in \mathbb{C}^n$ such that $Ax = \hat{b}$. The vector x is unique whenever $\operatorname{Ker} T_A = \{0\}$, i.e. $\dim \operatorname{Im} T_A = n$, which occurs precisely when the columns of A are linearly independent, i.e. A has full rank ($\operatorname{rank} A = n$). This proves (a) and (b).

To prove (c), observe that if x minimizes $E(x)$, then by (a) we have $b - Ax = r \in \operatorname{Ker} T_{A^*}$, and thus $A^*(b - Ax) = 0$. Conversely, if $A^*(b - Ax) = 0$, then $b - Ax \in \operatorname{Ker} T_{A^*}$, and by the uniqueness of the decomposition $b = \hat{b} + r$ we have $Ax = \hat{b}$. □

16.10 Normal Equations, Pro and Con

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

a) By Lemma 16.8, the matrix A^tA is symmetric positive definite, provided A has full rank. In this case the solution of the system of normal equations $A^tAx = A^tb$ can be effectively found by Cholesky factorization. Furthermore, in many applications $n \ll m$, so the system $A^tAx = A^tb$ is much smaller than $Ax = b$,

b) It may happen, though, that the matrix A is somewhat ill-conditioned (i.e. almost causes catastrophic cancellations). In that case the condition of the matrix A^tA will be much worse than that of A, compare this to 15.6 (9). Solving the normal equations can be disasterous. For example, let

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

then

$$A^tA = \begin{pmatrix} 1 + \varepsilon^2 & 1 \\ 1 & 1 + \varepsilon^2 \end{pmatrix}$$

If ε is so small that $\varepsilon^2 < \mathbf{u}$ (for example, $\varepsilon = 10^{-4}$ in single precision), then the matrix A^tA will be stored as a singular matrix, and the numerical solution of the normal equations

will crash. Still, it is possible to find a numerical least squares solution of the original system $Ax = b$, if one uses more elaborate methods.

c) One can define a condition number of an $m \times n$ rectangular matrix A with $m > n$ by

$$\kappa(A) := \frac{\max_{||x||=1} ||Ax||}{\min_{||x||=1} ||Ax||}$$

Then, in the above example,

$$\kappa_2(A) = \varepsilon^{-1}\sqrt{2+\varepsilon^2} \quad \text{and} \quad \kappa_2(A^tA) = \varepsilon^{-2}(2+\varepsilon^2)$$

Hence, $\kappa_2(A^tA) = [\kappa_2(A)]^2$.

d) Clearly, $\kappa_2(A) = \kappa_2(QA)$ for any orthogonal $m \times m$ matrix Q. Hence, one can safely (without increasing the 2-condition number of A) multiply A by orthogonal $m \times m$ matrices. We will try to find an orthogonal matrix Q so that QA is upper triangular (i.e., perform *orthogonal triangularization*).

16.11 Definition (Hyperplane, Reflection)

Let V be a finite dimensional inner product space (real or complex). A subspace $W \subset V$ is called a *hyperplane* if $\dim W = \dim V - 1$. Note that in this case $\dim W^\perp = 1$.

Let $W \subset V$ be a hyperplane. For any vector $v \in V$ we have a unique decomposition $v = w + w'$, where $w \in W$ and $w' \in W^\perp$. The linear operator P on V defined by $Pv = w - w'$ is called a *reflection* (or *reflector*) across the hyperplane W. It is identical on W and negates vectors orthogonal to W.

16.12 Definition (Reflection Matrix)

Let $x \neq 0$ be a vector in $\mathbb{R}^n$ or $\mathbb{C}^n$. The matrix

$$P = I - 2\frac{x\bar{x}^t}{x^t\bar{x}}$$

is called a *reflection matrix* (or a *reflector matrix*). Obviously, P is unchanged if x is replaced by cx for any $c \neq 0$. Note also that $x^t\bar{x} = ||x||^2$

16.13 Theorem

Let P be a reflection matrix corresponding to a vector $x \neq 0$. Then
(a) $Px = -x$
(b) $Py = y$ whenever $\langle y, x\rangle = 0$
(c) P is Hermitean (in the real case it is symmetric)
(d) P is unitary (in the real case it is orthogonal)
(e) P is involution, i.e. $P^2 = I$
Note that (a)+(b) mean that P is a reflection of $\mathbb{R}^n$ or $\mathbb{C}^n$ across the hyperplane orthogonal to the vector x.

Proof. (a), (b) and (c) are proved by direct calculation. To prove (e), write

$$P^2 = I - 4\frac{x\bar{x}^t}{x^t\bar{x}} + 4\frac{x\bar{x}^t x\bar{x}^t}{(x^t\bar{x})^2}$$

Then (e) follows from the fact that $\bar{x}^t x = x^t\bar{x} = ||x||^2$ is a scalar quantity. Next, (d) follows from (c) and (e).

16.14 Theorem

Let y be a vector in $\mathbb{R}^n$ or $\mathbb{C}^n$. Choose a scalar σ so that $|\sigma| = ||y||$ and $\sigma \cdot \langle e_1, y\rangle \in \mathbb{R}$. Suppose that $x = y + \sigma e_1 \neq 0$. Let $P = I - 2x\bar{x}^t/||x||^2$ be the reflection matrix defined in 16.12. Then $Py = -\sigma e_1$.

Proof. Notice that $\langle y - \sigma e_1, y + \sigma e_1\rangle = ||y||^2 - \sigma\langle e_1, y\rangle + \bar{\sigma}\langle y, e_1\rangle - |\sigma|^2 = 0$. Hence, $P(y - \sigma e_1) = y - \sigma e_1$ by 16.13 (b). Besides, $P(y + \sigma e_1) = -y - \sigma e_1$ by 16.13 (a). Adding these two equations gives the theorem.

16.15 Remark

(a) To choose σ in 16.14, write a polar representation for $\langle e_1, y\rangle = \bar{y}_1 = re^{i\theta}$ and then set $\sigma = \pm||y||e^{-i\theta}$.

(b) In the real case, we have $y_1 \in \mathbb{R}$, and one can just set

$$\sigma = \pm||y||$$

Note: It is geometrically obvious that for any two unit vectors $x, y \in \mathbb{R}^n$ there is a reflection P that takes x to y. In the complex space $\mathbb{C}^n$, this is not true: for generic unit vectors x, y one can only find a reflection that takes x to cy with some scalar $c \in \mathbb{C}$.

16.16 Corollary

For any vector y in $\mathbb{R}^n$ or $\mathbb{C}^n$ there is a scalar σ (defined in 16.14) and a matrix P, which is either a reflection or an identity, such that $Py = -\sigma e_1$.

Proof. Apply Theorem 16.14 in the case $y + \sigma e_1 \neq 0$ and set $P = I$ otherwise. □

16.17 Theorem (QR Decomposition)

Let A be an $m \times n$ complex or real matrix with $m \geq n$. Then there is a unitary (resp., orthogonal) $m \times m$ matrix Q and an upper triangular $m \times n$ matrix R (i.e., $R_{ij} = 0$ for $i > j$) such that

$$A = QR$$

Furthermore, Q may be found as a product of at most n reflection matrices.

Proof. We use induction on n. Let $n = 1$, so that A is a column m-vector. By 16.16 there is a matrix P (a reflection or an identity) such that $PA = -\sigma e_1$ for a scalar σ. Hence, $A = PR$ where $R = -\sigma e_1$ is upper triangular. Now, let $n \geq 1$ and $\mathbf{a}_1$ the first column of A. Again, by 16.16 there is a (reflection or identity) matrix P such that $P\mathbf{a}_1 = -\sigma e_1$. Hence,

$$PA = \begin{pmatrix} -\sigma & w^t \\ 0 & B \end{pmatrix}$$

where w is an $(n-1)$ vector and B an $(m-1) \times (n-1)$ matrix. By the inductive assumption, there is an $(m-1) \times (m-1)$ unitary matrix Q' and an $(m-1) \times (n-1)$ upper triangular matrix R' such that $B = Q'R'$. Consider the unitary $m \times m$ matrix

$$Q_1 = \begin{pmatrix} 1 & 0 \\ 0 & Q' \end{pmatrix}$$

(by 10.6, Q_1 is unitary whenever Q' is). One can easily check that $PA = Q_1 R$ where

$$R = \begin{pmatrix} -\sigma & w^t \\ 0 & R' \end{pmatrix}$$

is an upper triangular matrix. Hence, $A = QR$ with $Q = PQ_1$. □

16.18 Remark

In the above theorem, denote by $\mathbf{a}_j \in \mathbb{C}^m$, $1 \leq j \leq n$, the columns of A, by $\mathbf{q}_j \in \mathbb{C}^m$, $1 \leq j \leq m$, the columns of Q, and put $R = (r_{ij})$. Then $A = QR$ may be written as

$$\begin{aligned} \mathbf{a}_1 &= r_{11}\mathbf{q}_1 \\ \mathbf{a}_2 &= r_{12}\mathbf{q}_1 + r_{22}\mathbf{q}_2 \\ &\cdots \\ \mathbf{a}_n &= r_{1n}\mathbf{q}_1 + r_{2n}\mathbf{q}_2 + \cdots + r_{nn}\mathbf{q}_n \end{aligned}$$

16.19 Corollary

If, in addition, A has full rank ($\operatorname{rank} A = n$), then
(a) $\operatorname{span}\{\mathbf{a}_1, \ldots, \mathbf{a}_k\} = \operatorname{span}\{\mathbf{q}_1, \ldots, \mathbf{q}_k\}$ for every $1 \leq k \leq n$.
(b) one can find a unitary Q so that the diagonal entries of R will be real and positive ($r_{ii} > 0$ for $1 \leq i \leq n$).

Proof. (a) follows from 16.18. To prove (b), let $r_{jj} = |r_{jj}|e^{i\theta_j}$ be the polar representation of r_{jj}. Then multiplying every column $\mathbf{q}_j$, $1 \leq j \leq n$, by $e^{i\theta_j}$ gives (b). The new matrix Q will be still unitary by 10.6. □

The matrix A depends only on the first n columns of Q. This suggests a reduced QR decomposition.

16.20 Theorem ("Skinny" QR decomposition)
Let A be an $m \times n$ real or complex matrix with $m \geq n$. Then there is an $m \times n$ matrix $\hat{Q}$ with orthonormal columns and an upper triangular $m \times m$ matrix $\hat{R}$ such that

$$A = \hat{Q}\hat{R}$$

Proof. Apply Theorem 16.17. Let $\hat{Q}$ be the left $m \times n$ rectangular block of Q (the first n columns of Q). Let $\hat{R}$ be the top $n \times n$ square block of R (note that the remainder of R is zero). Then $A = \hat{Q}\hat{R}$. □

16.21 Corollary
If, in addition, A has full rank ($\operatorname{rank} A = n$), then
(a) The columns of $\hat{Q}$ make an ONB in the column space of A (=$\operatorname{Im} T_A$)
(b) one can find $\hat{Q}$ so that the diagonal entries of $\hat{R}$ will be real and positive ($r_{ii} > 0$). Such $\hat{Q}$ and $\hat{R}$ are unique.

Proof. This follows from 16.19, except the uniqueness. For simplicity, we prove it only in the real case. Let $A = \hat{Q}\hat{R} = \hat{Q}_1\hat{R}_1$. Then $A^tA = \hat{R}^t\hat{R} = \hat{R}_1^t\hat{R}_1$. Since A^tA is positive definite, we can use the uniqueness of Cholesky factorization and obtain $\hat{R} = \hat{R}_1$. Then also $\hat{Q}_1 = \hat{Q}\hat{R}\hat{R}_1^{-1} = \hat{Q}$. □

In the remainder of this section, we only consider real matrices and vectors.

16.22 Algorithm
Let $Ax = b$ be a real overdetermined system with matrix A of full rank. One finds the decomposition $A = QR$ with reflectors, as shown in the proof of Theorem 16.17. Then $Q^tQ = I$ and so $Rx = Q^tb$. Denote $Q^tb = \begin{pmatrix} c \\ d \end{pmatrix}$ where $c \in \mathbb{R}^n$ is the vector of top n components of Q^tb and $d \in \mathbb{R}^{m-n}$ its bottom part. Next, one finds x by using backward substitution to solve the system $\hat{R}x = c$, where $\hat{R}$ is the top $n \times n$ square block of R. Lastly, one finds the value of $E(x) = ||r - Ax||^2$ by computing $||d||^2$. Indeed,

$$||b - Ax||^2 = ||Q^tb - Q^tAx||^2 = ||(c,d)^t - Rx||^2 = ||d||^2$$

The above algorithm is the most suitable for computer implementation. It does not worsen the condition of the given system $Ax = b$, see also numerical hints in 16.26.

On the other hand, for calculations 'by hands' (such as on tests!), the following reduced version of this algorithm is more convenient.

16.23 Algorithm

Let $Ax = b$ be a real overdetermined system with matrix A of full rank. One can obtain a "skinny" QR decomposition $A = \hat{Q}\hat{R}$ by using 16.18 and employing a version of the Gram-Schmidt orthogonalization method:

$$r_{11} = ||\mathbf{a}_1|| \quad \text{and} \quad \mathbf{q}_1 = \mathbf{a}_1/r_{11}$$

$$r_{12} = \langle \mathbf{a}_2, \mathbf{q}_1 \rangle, \quad r_{22} = ||\mathbf{a}_2 - r_{12}\mathbf{q}_1|| \quad \text{and} \quad \mathbf{q}_2 = (\mathbf{a}_2 - r_{12}\mathbf{q}_1)/r_{22}$$

$$\dots$$

$$r_{in} = \langle \mathbf{a}_n, \mathbf{q}_i \rangle \ \ (i < n), \quad r_{nn} = ||\mathbf{a}_n - \sum r_{in}\mathbf{q}_i|| \quad \text{and} \quad \mathbf{q}_n = (\mathbf{a}_n - \sum r_{in}\mathbf{q}_i)/r_{nn}$$

Then one solves the triangular system $\hat{R}x = \hat{Q}^t b$ by backward substitution. This method does not give the value of $||b - Ax||^2$, one has to compute it directly, if necessary.

Below we outline an alternative approach to finding a QR decomposition.

16.24 Definition (Rotation Matrix)

Let $1 \le p < q \le m$ and $\theta \in [0, 2\pi)$. The matrix $G = G_{p,q,\theta} = (g_{ij})$ defined by $g_{pp} = \cos\theta$, $g_{pq} = \sin\theta$, $g_{qp} = -\sin\theta$, $g_{qq} = \cos\theta$ and $g_{ij} = \delta_{ij}$ otherwise is called a *Givens rotation matrix*. It defines a rotation through the angle θ of the $x_p x_q$ coordinate plane in $\mathbb{R}^m$ with all the other coordinates fixed. Obviously, G is orthogonal.

16.25 Algorithm

Let $Ax = b$ be a real overdetermined system with matrix A of full rank. Let $\mathbf{a}_j$ be the leftmost column of A that contains a nonzero entry below the main diagonal, $a_{ij} \neq 0$ with some $i > j$. Consider the matrix $A' = GA$ where $G = G_{j,i,\theta}$ is a Givens rotation matrix. One easily checks that
(a) the first $j - 1$ columns of A' are zero below the main diagonal;
(b) in the j-th column, only the elements a'_{jj} and a'_{ij} will be different from the corresponding elements of A, and moreover

$$a'_{ij} = -a_{jj}\sin\theta + a_{ij}\cos\theta$$

Now we find $\sin\theta$ and $\cos\theta$ so that $a'_{ij} = 0$. For example, let

$$\cos\theta = \frac{a_{jj}}{\sqrt{a_{jj}^2 + a_{ij}^2}} \quad \text{and} \quad \sin\theta = \frac{a_{ij}}{\sqrt{a_{jj}^2 + a_{ij}^2}}$$

Note that one never actually evaluates the angle θ, since $G_{j,i,\theta}$ only contains $\cos\theta$ and $\sin\theta$.

In this way one zeroes out one nonzero element a_{ij} below the main diagonal. Working from left to right, one can convert A into an upper triangular matrix $\tilde{G}A = R$ where $\tilde{G}$ is a product of Givens's rotation matrices. Each nonzero element of A below the main diagonal requires one multiplication by a rotation matrix. Then we get $A = QR$ with an orthogonal matrix $Q = \tilde{G}^{-1}$. This algorithm is generally more expensive than the QR decomposition with reflectors. But it works very efficiently if the matrix A is *sparse*, i.e. contains just a few nonzero elements below the main diagonal.

16.26 Remarks (Numerical Hints)

The most accurate numerical algorithm for solving a generic overdetermined linear system $Ax = b$ is based on the QR decomposition with reflectors, then it proceeds as in 16.22. In the process of computing the reflector matrices, two rules should be followed: (i) The sign of σ in 16.15(b) must coincide with the sign of $y_1 = \langle y, e_1 \rangle$ every time one uses Theorem 16.14. This helps to avoid catastrophic cancellations when calculating $x = y + \sigma e_1$ (note that the "skinny" algorithm in 16.22 does not provide this flexibility) (ii) In the calculation of

$$||y|| = \sqrt{y_1^2 + \cdots + y_m^2}$$

in Theorem 16.14, there is a danger of overflow (if one of y_i's is too large) or underflow (if all y_i's are too small, so that y_i^2 is a machine zero, then so will be $||y||$). To avoid this, find first

$$y_{\max} = \max\{|y_1|, \ldots, |y_m|\}$$

and then compute

$$||y|| = y_{\max} \cdot \sqrt{|y_1/y_{\max}|^2 + \cdots + |y_m/y_{\max}|^2}$$

The same trick must be used when computing the rotation matrices in 16.25.

16.27 Polynomial Least Squares Fit

Generalizing 16.2, one can fit a set of data points (x_i, y_i), $1 \leq i \leq m$, by a polynomial $y = p(x) = a_0 + a_1 x + \cdots + a_n x^n$ with $n + 1 \leq m$. The least squares fit is based on minimizing the function

$$E(a_0, \ldots, a_n) = \sum_{i=1}^{m} \Big(a_0 + a_1 x_i + \cdots + a_n x_i^n - y_i\Big)^2$$

This leads to an overdetermined linear system generalizing 16.4:

$$a_0 + a_1 x_i + \cdots + a_n x_i^n = y_i \qquad 1 \leq i \leq m$$

16.28 Continuous Least Squares Fit

Instead of fitting a discrete data set (x_i, y_i) one can fit a continuous function $y = f(x)$ on $[0, 1]$ by a polynomial $y = p(x) \in P_n(\mathbb{R})$. The least squares fit is the one minimizing

$$E(a_0, \ldots, a_n) = \int_0^1 |f(x) - p(x)|^2 \, dx$$

The solution of this problem is the orthogonal projection of $f(x)$ onto $P_n(\mathbb{R})$, recall the homework problem #1 in Assignment 1.

To find the solution, consider a basis $\{1, x, \ldots, x^n\}$ in $P_n(\mathbb{R})$. Then $a_0, \ldots, a_n$ can be found from the system of normal equations

$$\sum_{j=0}^{n} a_j \langle x^j, x^i \rangle = \langle f, x^i \rangle \qquad 1 \le i \le n$$

The matrix A of coefficients here is

$$a_{ij} = \langle x^j, x^i \rangle = \int_0^1 x^{i+j} \, dx = \frac{1}{1 + i + j}$$

This is the infamous Hilbert matrix, it is ill-conditioned even for moderately large n, since $\kappa_2(A) \approx 10^{1.5n}$. One can chose an orthonormal basis in $P_n(\mathbb{R})$ (Legendre polynomials, see 9.12) to improve the condition of the problem.

17 Other Decomposition Theorems

17.1 Theorem (Schur Decomposition)
If $A \in \mathbb{C}^{n\times n}$, then there exists a unitary matrix Q such that

$$Q^*AQ = T$$

where T is upper triangular, i.e. A is unitary equivalent to an upper triangular matrix. Moreover, the matrix Q can be chosen so that the eigenvalues of A appear in any order on the diagonal of T.

The columns of the matrix Q are called *Schur vectors*.

Proof. We use induction on n. The theorem is clearly true for $n = 1$. Assume that it holds for matrices of order less than n. Let λ be an eigenvalue of A and let x be a unit eigenvector for λ. Let R be a unitary matrix whose first column is x (note: such a matrix exists, because there is an ONB in $\mathbb{C}^n$ whose first vector is x, by 9.13, and then R can be constructed by using the vectors of that ONB as columns of R). Note that $Re_1 = x$, and hence $R^*x = e_1$, since $R^{-1} = R^*$. Hence, we have,

$$R^*ARe_1 = R^*Ax = \lambda R^*x = \lambda e_1$$

so e_1 is an eigenvector of the matrix R^*AR for the eigenvalue λ. Thus,

$$R^*AR = \begin{pmatrix} \lambda & w^t \\ 0 & B \end{pmatrix}$$

with some $w \in \mathbb{C}^{n-1}$ and $B \in \mathbb{C}^{(n-1)\times(n-1)}$. By the induction assumption, there is a unitary matrix $Q_1 \in \mathbb{C}^{(n-1)\times(n-1)}$ such that $Q_1^*BQ_1 = T_1$, where T_1 is upper triangular. Let

$$Q = R \begin{pmatrix} 1 & 0 \\ 0 & Q_1 \end{pmatrix}$$

Note: by 10.6, the second factor is a unitary matrix because so is Q_1, and then note that the product of two unitary matrices is a unitary matrix Q. Next,

$$Q^*AQ = \begin{pmatrix} 1 & 0 \\ 0 & Q_1^* \end{pmatrix} \begin{pmatrix} \lambda & w^t \\ 0 & B \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & Q_1 \end{pmatrix} = \begin{pmatrix} \lambda & w^tQ_1 \\ 0 & Q_1^*BQ_1 \end{pmatrix} = \begin{pmatrix} \lambda & w^tQ_1 \\ 0 & T_1 \end{pmatrix}$$

which is upper triangular, as required. □

17.2 Definition (Normal matrix)
A matrix $A \in \mathbb{C}^{n\times n}$ is said to be *normal* if $AA^* = A^*A$.

Note: unitary and Hermitean matrices are normal.

17.3 Lemma

If A is normal and Q unitary, then $B = Q^*AQ$ is also normal (i.e., the class of normal matrices is closed under unitary equivalence).

Proof. Note that $B^* = Q^*A^*Q$ and $BB^* = Q^*AA^*Q = Q^*A^*AQ = B^*B$. □

17.4 Lemma

If A is normal and upper triangular, then A is diagonal.

Proof. We use induction on n. For $n = 1$ the theorem is trivially true. Assume that it holds for matrices of order less than n. Compute the top left element of the matrix $AA^* = A^*A$. On the one hand, it is

$$\sum_{i=1}^{n} a_{1i}\bar{a}_{1i} = \sum_{i=1}^{n} |a_{1i}|^2$$

On the other hand, it is just $|a_{11}|^2$. Hence, $a_{12} = \cdots = a_{1n} = 0$, and

$$A = \begin{pmatrix} a_{11} & 0 \\ 0 & B \end{pmatrix}$$

One can easily check that $AA^* = A^*A$ implies $BB^* = B^*B$. By the induction assumption, B is diagonal. □

Note: Any diagonal matrix is normal.

17.5 Theorem

(i) A matrix $A \in \mathbb{C}^{n\times n}$ is normal if and only if it is unitary equivalent to a diagonal matrix. In that case the Schur decomposition takes form

$$Q^*AQ = D$$

where D is a diagonal matrix.

(ii) If A is normal, the columns of Q (Schur vectors) are eigenvectors of A (specifically, the jth column of Q is an eigenvector of A for the eigenvalue in the jth position on the diagonal of D).

Proof. The first claim follows from 17.1–17.4. The second is verified by direct inspection. □

Note: Theorem 17.5 applies to unitary and Hermitean matrices.

17.6 Remark

Three classes of complex matrices have the same property, that their matrices are unitary equivalent to a diagonal matrix (i.e., admit an ONB consisting of eigenvectors). The difference between those classes lies in restrictions on eigenvalues: unitary matrices have eigenvalues on the unit circle ($|\lambda| = 1$), Hermitean matrices have real eigenvalues ($\lambda \in \mathbb{R}$), and now normal matrices may have arbitrary complex eigenvalues.

Schur decomposition established in 17.1 is mainly of theoretical interest. As most matrices of practical interest have real entries the following variation of the Schur decomposition is of some practical value.

17.7 Theorem (Real Schur Decomposition)

If $A \in \mathbb{R}^{n\times n}$ then there exists an orthogonal matrix Q such that

$$Q^t A Q = \begin{pmatrix} R_{11} & R_{12} & \cdots & R_{1m} \\ 0 & R_{22} & \cdots & R_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R_{mm} \end{pmatrix}$$

where each R_{ii} is either a 1×1 matrix or a 2×2 matrix having complex conjugate eigenvalues.

Proof. We use the induction on n. If the matrix A has a real eigenvalue, the we can reduce the dimension and use induction just like in the proof of Schur Theorem 17.1. Assume that A has no real eigenvalues. Since the characteristic polynomial of A has real coefficients, its roots come in conjugate pairs. Let $\lambda_1 = \alpha + i\beta$ and $\lambda_2 = \alpha - i\beta$ be a pair of roots of $C_A(x)$, with $\beta \neq 0$. The root λ_1 is a complex eigenvalue of A, considered as a complex matrix, with a complex eigenvector $x + iy$, where $x, y \in \mathbb{R}^n$. The equation

$$A(x + iy) = (\alpha + i\beta)(x + iy)$$

can be written as

$$\begin{aligned} Ax &= \alpha x - \beta y \\ Ay &= \beta x + \alpha y \end{aligned}$$

or in the matrix form

$$A \begin{pmatrix} x & y \end{pmatrix} = \begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} \alpha & \beta \\ -\beta & \alpha \end{pmatrix}$$

where $\begin{pmatrix} x & y \end{pmatrix}$ is an $n \times 2$ matrix with columns x and y. Observe that x and y must be lnearly independent, because if not, one can easily show that $\beta = 0$, contrary to the assumption. Let

$$\begin{pmatrix} x & y \end{pmatrix} = P \begin{pmatrix} R \\ 0 \end{pmatrix}$$

be a QR decomposition of the matrix $\begin{pmatrix} x & y \end{pmatrix}$, where $R \in \mathbb{R}^{2\times 2}$ is an upper triangular matrix and $P \in \mathbb{R}^{n\times n}$ orthogonal. Note that R is nonsingular, because $\begin{pmatrix} x & y \end{pmatrix}$ has full rank. Denote

$$P^t AP = \begin{pmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{pmatrix}$$

where T_{11} is a 2×2 matrix, T_{12} is a $2 \times (n-2)$ matrix, T_{21} is a $(n-2) \times 2$ matrix, and T_{22} is a $(n-2) \times (n-2)$ matrix. Now, combining the above equations gives

$$\begin{pmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{pmatrix} \begin{pmatrix} R \\ 0 \end{pmatrix} = \begin{pmatrix} R \\ 0 \end{pmatrix} \begin{pmatrix} \alpha & \beta \\ -\beta & \alpha \end{pmatrix}$$

This implies

$$T_{11}R = R \begin{pmatrix} \alpha & \beta \\ -\beta & \alpha \end{pmatrix}$$

and

$$T_{21}R = 0$$

Since R is nonsingular, the latter equation implies $T_{21} = 0$, and the former equation shows that the matrix T_{11} is similar to $\begin{pmatrix} \alpha & \beta \\ -\beta & \alpha \end{pmatrix}$. That last matrix has eigenvalues $\lambda_{1,2} = \alpha \pm i\beta$, hence so does T_{11}, by similarity, therefore it satisfies the requirements of the theorem. Now we can apply the induction assumption to the $(n-2)\times(n-2)$ matrix T_{22}. $\square$

In the next theorem we consider a matrix $A \in \mathbb{R}^{m\times n}$. It defines a linear transformation $T_A : \mathbb{R}^n \to \mathbb{R}^m$. Let B be an ONB in $\mathbb{R}^n$ and B' be an ONB in $\mathbb{R}^m$. Then the transformation T_A is represented in the bases B and B' by the matrix $U^t AV$, where U and V are orthogonal matrices of size $m \times m$ and $n \times n$, respectively. The following theorem shows that one can always find bases B and B' so that the matrix $U^t AV$ will be diagonal.

Note: $D \in \mathbb{R}^{m\times n}$ is said to be diagonal if $D_{ij} = 0$ for $i \neq j$. It has exactly $p = \min\{m, n\}$ diagonal elements and can be denoted by $D = \operatorname{diag}(d_1, \ldots, d_p)$.

17.8 Theorem (Singular Value Decomposition (SVD))

Let $A \in \mathbb{R}^{m\times n}$ have rank r and let $p = \min\{m, n\}$. Then there are orthogonal matrices $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ and a diagonal matrix $D = \operatorname{diag}(\sigma_1, \ldots, \sigma_p)$ such that

$$U^t AV = D$$

. In the diagonal of D, exactly r elements are nonzero. If additionally we require that $\sigma_1 \geq \cdots \geq \sigma_r > 0$ and $\sigma_{r+1} = \cdots = \sigma_p = 0$, then the matrix D is unique.

Proof. The matrix A^t defines a linear transformation $\mathbb{R}^m \to \mathbb{R}^n$. Recall that the matrices $A^tA \in \mathbb{R}^{n\times n}$ and $AA^t \in \mathbb{R}^{m\times m}$ are symmetric and positive semi-definite. Note that

$$\operatorname{Ker} A = \operatorname{Ker} A^tA \quad \text{and} \quad \operatorname{Ker} A^t = \operatorname{Ker} AA^t$$

(indeed, if $A^tAx = 0$, then $0 = \langle A^tAx, x\rangle = \langle Ax, Ax\rangle$, so $Ax = 0$; the second identity is proved similarly). Hence,

$$\operatorname{rank} A^tA = \operatorname{rank} AA^t = \operatorname{rank} A = r$$

Indeed,

$$\operatorname{rank} A^tA = n - \dim \operatorname{Ker} A^tA = n - \dim \operatorname{Ker} A = \operatorname{rank} A = r$$

and similarly $\operatorname{rank} AA^t = \operatorname{rank} A^t = \operatorname{rank} A = r$. Next, by 11.21 and the remark after 12.10, there is an orthogonal matrix $V \in \mathbb{R}^{n\times n}$ such that $A^tA = VSV^t$ where $S = \operatorname{diag}(s_1, \dots, s_n) \in \mathbb{R}^{n\times n}$ and $s_i \geq 0$ are the eigenvalues of A^tA. Note that $\operatorname{rank} S = \operatorname{rank} A^tA = r$, so exactly r diagonal elements of S are positive, and we can reorder them (e.g., according to 17.1 and 17.5) so that $s_1 \geq \cdots \geq s_r > 0$ and $s_{r+1} = \cdots = s_n = 0$. Let $v_1, \dots, v_n$ be the columns of V, which are the eigenvectors of A^tA, according to 17.5, i.e. $A^tAv_i = s_iv_i$. Define vectors $u_1, \dots, u_m$ as follows. Let

$$u_i = \frac{1}{\sqrt{s_i}} Av_i \qquad \text{for} \qquad 1 \leq i \leq r$$

For $r + 1 \leq i \leq m$, let $\{u_i\}$ be an arbitrary ONB of the subspace $\operatorname{Ker} AA^t$, whose dimension is $m - \operatorname{rank} AA^t = m - r$. Note that

$$AA^tu_i = \frac{1}{\sqrt{s_i}} A(A^tA)v_i = \sqrt{s_i}Av_i = s_iu_i$$

for $i = 1, \dots, r$, and $AA^tu_i = 0$ for $i \geq r + 1$. Therefore, $u_1, \dots, u_m$ are the eigenvectors of AA^t corresponding to the eigenvalues $s_1, \dots, s_r, 0, \dots, 0$. Next, we claim that $\{u_i\}$ make an ONB in $\mathbb{R}^m$, i.e. $\langle u_i, u_j\rangle = \delta_{ij}$. This is true for $i, j \geq r + 1$ by our choice of u_i. It is true for $i \leq r < j$ and $j \leq r < i$ because then u_i, u_j are eigenvectors of the symmetric matrix AA^t corresponding to distinct eigenvalues, so they are orthogonal by 11.9 (ii). Lastly, for $i, j \leq r$

$$\langle u_i, u_j\rangle = \frac{1}{\sqrt{s_is_j}}\langle Av_i, Av_j\rangle = \frac{1}{\sqrt{s_is_j}}\langle A^tAv_i, v_j\rangle = \frac{\sqrt{s_i}}{\sqrt{s_j}}\langle v_i, v_j\rangle = \delta_{ij}$$

Now let $U \in \mathbb{R}^{m\times m}$ be the matrix whose columns are $u_1, \dots, u_m$. Then U is orthogonal, and $U^tA^tAU = S' \in \mathbb{R}^{m\times m}$ where $S' = \operatorname{diag}(s_1, \dots, s_r, 0, \dots, 0)$. Let $\sigma_i = \sqrt{s_i}$ for $1 \leq i \leq p$ and observe that $Av_i = \sigma_iu_i$. Consequently, if $D \in \mathbb{R}^{m\times n}$ is defined by $D = \operatorname{diag}(\sigma_1, \dots, \sigma_p)$, then $AV = UD$ (the i-th column of the left matrix is Av_i, and that of the right matrix is σ_iu_i). Hence, $U^tAV = D$ as required. It remains to prove the uniqueness of D. Let $A = \hat{U}\hat{D}\hat{V}^t$ be another SVD. Then $A^tA = \hat{V}\hat{D}^2\hat{V}^t$, so the diagonal elements of $\hat{D}^2$ are the eigenvalues of A^tA, hence they are $s_1, \dots, s_r, 0, \dots, 0$. Since the diagonal elements of $\hat{D}$ are nonnegative, they are $\sqrt{s_1}, \dots, \sqrt{s_r}, 0, \dots, 0$. □

17.9 Definition (Singular Values and Vectors)

The positive numbers $\sigma_1, \ldots, \sigma_r$ are called the *singular values* of A. The columns $v_1, \ldots, v_n$ of the matrix V are called the *right singular vectors* for A, and the columns $u_1, \ldots, u_m$ of the matrix U are called the *left singular vectors* for A.

17.10 Remark

For $1 \leq i \leq r$ we have

$$Av_i = \sigma_i u_i \qquad \text{and} \qquad A^t u_i = \sigma_i v_i$$

We also have

$$\operatorname{Ker} A = \operatorname{span}\{v_{r+1}, \ldots, v_n\} \quad \text{and} \quad \operatorname{Ker} A^t = \operatorname{span}\{u_{r+1}, \ldots, u_m\}$$

and

$$\operatorname{Im} A = \operatorname{span}\{u_1, \ldots, u_r\} \quad \text{and} \quad \operatorname{Im} A^t = \operatorname{span}\{v_1, \ldots, v_r\}$$

Illustration to Remark 17.10:

$$\begin{array}{cccccc} & & A & & A^t & \\ v_1 & \xrightarrow{\sigma_1} & u_1 & \xrightarrow{\sigma_1} & v_1 \\ v_2 & \xrightarrow{\sigma_2} & u_2 & \xrightarrow{\sigma_2} & v_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ v_r & \xrightarrow{\sigma_r} & u_r & \xrightarrow{\sigma_r} & v_r \\ v_{r+1} & \to 0 & u_{r+1} & \to 0 & \\ \vdots & \vdots & \vdots & \vdots & \\ v_n & \to 0 & u_m & \to 0 & \end{array}$$

17.11 Remark

We have the following *SVD expansion*:

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^t$$

Proof. It is enough to observe that for every v_j

$$\left(\sum_{i=1}^{r} \sigma_i u_i v_i^t \right) v_j = \sigma_j u_j = A v_j$$

because $v_i^t v_j = \delta_{ij}$. □

17.12 Remark

The positive numbers $\sigma_1^2, \ldots, \sigma_r^2$ are all non-zero eigenvalues of both A^tA and AA^t. This can be used as a practical method to compute the singular values $\sigma_1, \ldots, \sigma_r$. The multiplicity of $\sigma_i > 0$ as a singular value of A equals the (algebraic and geometric) multiplicity of σ_i^2 as an eigenvalue of both A^tA and AA^t.

Much useful information about the matrix A is revealed by the SVD.

17.13 Remark

Assume that $m = n$, i.e. A is a square matrix. Then $||A||_2 = ||A^t||_2 = ||D||_2 = \sigma_1$. If A is nonsingular, then similarly we have $||A^{-1}||_2 = ||D^{-1}||_2 = \sigma_n^{-1}$. Therefore, $\kappa_2(A) = \kappa_2(D) = \sigma_1/\sigma_n$.

SVD is helpful in the study of the matrix rank.

17.14 Definition

A matrix $A \in \mathbb{R}^{m\times n}$ is said to have *full rank* if its rank equals $p = \min\{m, n\}$. Otherwise, A is said to be *rank deficient*.

Note: If $A = UDV^t$ is an SVD, then $\operatorname{rank} A = \operatorname{rank} D$ is the number of positive singular values. If A has full rank, then all its singular values are positive. Rank deficient matrices have at least one zero singular value.

17.15 Definition

The *2-norm* of a rectangular matrix $A \in \mathbb{R}^{m\times n}$ is defined by

$$||A||_2 = \max_{||x||_2=1} ||Ax||_2$$

where $||x||_2$ is the 2-norm in $\mathbb{R}^n$ and $||Ax||_2$ is the 2-norm in $\mathbb{R}^m$.

Note: $||A||_2 = ||A^t||_2 = ||D||_2 = \sigma_1$, generalizing Remark 17.13.

The linear space $\mathbb{R}^{m\times n}$ with the distance between matrices given by $||A-B||_2$ is a *metric space*. Then topological notions, like open sets, dense sets, etc., apply.

17.16 Theorem

Full rank matrices make an open and dense subset of $\mathbb{R}^{m\times n}$.

Openness means that for any full rank matrix A there is an $\varepsilon > 0$ such that $A + E$ has full rank whenever $||E||_2 \le \varepsilon$. Denseness means that if A is rank deficient, then for any $\varepsilon > 0$ there is E, $||E||_2 \le \varepsilon$, such that $A + E$ has full rank.

Proof. To prove denseness, let A be a rank deficient matrix and $A = UDV^t$ its SVD. For $\varepsilon > 0$, put $D_\varepsilon = \varepsilon I$ and $E = UD_\varepsilon V^t$. Then $||E||_2 = ||D_\varepsilon||_2 = \varepsilon$ and

$$\operatorname{rank}(A+E) = \operatorname{rank}(U(D+D_\varepsilon)V^t) = \operatorname{rank}(D+D_\varepsilon) = p$$

For the openness, see homework assignment (one can use the method of the proof of 17.18 below).

One can see that slight perturbation of a rank deficient matrix (e.g., caused by round-off errors) can produce a full rank matrix. On the other hand, a full rank matrix may be perturbed by round-off errors and become rank deficient, a very undesirable event. To describe such situations, we introduce the following

17.17 Definition (Numerical Rank)
The *numerical rank* of $A \in \mathbb{R}^{m\times n}$ with *tolerance* ε is

$$\operatorname{rank}(A,\varepsilon) = \min_{||E||_2 \le \varepsilon} \operatorname{rank}(A+E)$$

Note: $\operatorname{rank}(A,\varepsilon) \le \operatorname{rank} A$. The numerical rank gives the minimum rank of A under perturbations of norm $\le \varepsilon$. If A has full rank $p = \min\{m,n\}$ but $\operatorname{rank}(A,\varepsilon) < p$, then A is 'nearly rank deficient', which indicates dangerous situation in numerical calculations.

17.18 Theorem
The numerical rank of a matrix A, i.e. $\operatorname{rank}(A,\varepsilon)$, equals the number of singular values of A (counted with multiplicity) that are greater than ε.

Proof. Let $A = UDV^t$ be a SVD and $\sigma_1 \ge \cdots \ge \sigma_p$ the singular values of A. Let k be defined so that $\sigma_k > \varepsilon \ge \sigma_{k+1}$. We show that $\operatorname{rank}(A,\varepsilon) = k$. Let $||E||_2 \le \varepsilon$ and $E' = U^tEV$. Observe that $||E'||_2 = ||E||_2 \le \varepsilon$ and

$$\operatorname{rank}(A+E) = \operatorname{rank}(U(D+E')V^t) = \operatorname{rank}(D+E')$$

Hence,

$$\operatorname{rank}(A,\varepsilon) = \min_{||E'||_2 \le \varepsilon} \operatorname{rank}(D+E')$$

Now, for any $x = \sum_1^k x_i e_i$ we have

$$||(D+E')x||_2 \ge ||Dx||_2 - ||E'x||_2 \ge (\sigma_k - \varepsilon)||x||_2$$

Hence, $(D+E')x \ne 0$ for $x \ne 0$, so $\operatorname{rank}(D+E') \ge k$. On the other hand, setting $E' = \operatorname{diag}(0,\dots,0,-\sigma_{k+1},\dots,-\sigma_p)$ implies that $||E'||_2 \le \varepsilon$ and $\operatorname{rank}(D+E') = k$. □

17.19 Corollary
For every $\varepsilon > 0$, $\operatorname{rank}(A^t, \varepsilon) = \operatorname{rank}(A, \varepsilon)$.

Proof. A and A^t have the same singular values. □

17.20 Theorem (Distance to the nearest singular matrix)
Let $A \in \mathbb{R}^{n\times n}$ be a nonsingular matrix. Then

$$\min\left\{\frac{||A - A_s||_2}{||A||_2} : \ A_s \text{ is singular}\right\} = \frac{1}{\kappa_2(A)}$$

Proof. Follows from 17.13 and 17.18. □

17.21 Example
Let $A = \begin{pmatrix} 3 \\ 4 \end{pmatrix}$. Then $A^tA = (25)$ is a 1×1 matrix, it has the only eigenvalue $\lambda = 25$ with eigenvector $v_1 = e_1 = (1)$. Hence, $\sigma_1 = \sqrt{\lambda} = 5$. The matrix $AA^t = \begin{pmatrix} 9 & 12 \\ 12 & 16 \end{pmatrix}$ has eigenvalues $\lambda_1 = 25$ and $\lambda_2 = 0$ with corresponding eigenvectors $u_1 = (3/5\ 4/5)^t$ and $u_2 = (4/5\ \ -3/5)^t$. The SVD takes form

$$\begin{pmatrix} 3/4 & 4/5 \\ 4/5 & -3/5 \end{pmatrix} \begin{pmatrix} 3 \\ 4 \end{pmatrix} (1) = \begin{pmatrix} 5 \\ 0 \end{pmatrix}$$

Observe that $Av_1 = 5u_1$ and $A^tu_1 = 5v_1$ in accordance with the SVD theorem.

18 Eigenvalues and eigenvectors: sensitivity

18.1 Definition (Left Eigenvector)
Let $A \in \mathbb{C}^{n\times n}$. A nonzero vector $x \in \mathbb{C}^n$ is called a *left eigenvector* of A corresponding to an eigenvalue λ if

$$x^*A = \lambda x^*$$

Note that this is equivalent to $A^*x = \bar{\lambda}x$, i.e. x being an ordinary (right) eigenvector of A^* corresponding to the eigenvalue $\bar{\lambda}$.

18.2 Lemma
A matrix A has a left eigenvector corresponding to λ if and only if λ is an eigenvalue of A (a root of the characteristic polynomial of A).

Proof. $x^*A = \lambda x^*$ for an $x \neq 0$ is equivalent to $(A^* - \bar{\lambda}I)x = 0$, which means that $\det(A^* - \bar{\lambda}I) = 0$, or equivalently, $\det(A - \lambda I) = 0$, i.e. $C_A(\lambda) = 0$. □

This explains why we do not introduce a notion of a *left* eigenvalue: the eigenvalues for left eigenvectors and those for ordinary (right) eigenvectors are the same.

18.3 Lemma
For any eigenvalue λ of A the dimension of the ordinary (right) eigenspace equals the dimension of the left eigenspace (i.e., the geometric multiplicity of λ is the same).

Proof. $\mathrm{Ker}(A - \lambda I) = n - \mathrm{rank}(A - \lambda I) = n - \mathrm{rank}(A^* - \bar{\lambda}I) = \mathrm{Ker}(A^* - \bar{\lambda}I)$. □

18.4 Definition (Rayleigh Quotient)
Let $A \in \mathbb{C}^{n\times n}$ and $x \in \mathbb{C}^n$ a nonzero vector. We call

$$\frac{x^*Ax}{x^*x} = \frac{\langle Ax, x\rangle}{\langle x, x\rangle}$$

the *Rayleigh quotient* for A. One can consider it as a function of x.

Note: The Rayleigh quotient takes the same value for all scalar multiples of a given vector x, i.e. it is constant on the line spanned by x (with the zero vector removed). Recall that any nonzero vector is a scalar multiple of a unit vector. Hence, Rayleigh quotient as a function of x is completely defined by its values on the unit vectors (on the unit sphere). For unit vectors x, the Rayleigh quotient can be simply defined by

$$x^*Ax = \langle Ax, x\rangle$$

18.5 Theorem

Let $A \in \mathbb{C}^{n\times n}$ and x a unit vector in $\mathbb{C}^n$. Then

$$||Ax - (x^*Ax)x||_2 = \min_{\mu\in C} ||Ax - \mu x||_2$$

That is, the vector $(x^*Ax)x$ is the orthogonal projection of the vector Ax on the line spanned by x.

Proof. From the homework problem #1 in the first assignment we know that

$$\min_{\mu\in C} ||Ax - \mu x||_2$$

is attained when $Ax - \mu x$ is orthogonal to x. This gives the value $\mu = x^*Ax$. Alternatively, consider an overdetermined $(n \times 1)$ system $x\mu = Ax$ where μ is (the only) unknown. Then the result follows from Theorem 16.9. □

Note: If x is a unit eigenvector for an eigenvalue λ, then $x^*Ax = \lambda$. Suppose that x is not an eigenvector. If one wants to regard x as an eigenvector, then the Rayleigh quotient for x is the best choice that one could make for the associated eigenvalue in the sense that this value of μ comes closest (in the 2-norm) to achieving $Ax - \mu x = 0$. So, one could regard x^*Ax as a "quasi-eigenvalue" for x.

18.6 Theorem

Let $A \in \mathbb{C}^{n\times n}$ and x a unit eigenvector of A corresponding to eigenvalue λ. Let y be another unit vector and $\rho = y^*Ay$. Then

$$|\lambda - \rho| \leq 2\,||A||_2\,||x - y||_2$$

Moreover, if A is a Hermitean matrix, then there is a constant $C = C(A) > 0$ such that

$$|\lambda - \rho| \leq C\,||x - y||_2^2$$

Proof. To prove the first part, put

$$\lambda - \rho = x^*A(x - y) + (x - y)^*Ay$$

and then using the triangle inequality and Cauchy-Schwarz inequality gives the result. Now, assume that A is Hermitean. Then there is an ONB of eigenvectors, and we can assume that x is one of them. Denote that ONB by $\{x, x_2, \ldots, x_n\}$ and the corresponding eigenvalues by $\lambda, \lambda_2, \ldots, \lambda_n$. Let $y = cx + c_2x_2 + \cdots + c_nx_n$. Then

$$||y - x||^2 = |c - 1|^2 + \sum_{i=2}^{n} |c_i|^2 \geq \sum_{i=2}^{n} |c_i|^2$$

On the other hand, $||y|| = 1$, so

$$\lambda = \lambda|c|^2 + \sum_{i=2}^{n} \lambda|c_i|^2$$

Now, $Ay = c\lambda x + \sum_{i=2}^{n} c_i\lambda_i x_i$, so

$$\rho = \langle Ay, y\rangle = \lambda|c|^2 + \sum_{i=2}^{n} \lambda_i|c_i|^2$$

Therefore,

$$\lambda - \rho = \sum_{i=2}^{n} (\lambda - \lambda_i)|c_i|^2$$

The result now follows with

$$C = \max_{2\le i\le n} |\lambda - \lambda_i|$$

The theorem is proved. □.

Note: The Rayleigh quotient function x^*Ax is obviously continuous on the unit sphere, because it is a polynomial of the coordinates of x. Since at every eigenvector x the value of this function equals the eigenvalue λ for x, then for y close to x its values are close to λ. Suppose that $||y - x|| = \varepsilon$. The first part of the theorem gives a specific estimate on how close y^*Ay to λ is, the difference between them is $O(\varepsilon)$. The second part says that for Hermitean matrices y^*Ay is *very* close to λ, the difference is now $O(\varepsilon^2)$.

Note: If A is Hermitean, then x^*Ax is a real number for any vector $x \in \mathbb{C}^n$, i.e. the Rayleigh quotient is a real valued function.

18.7 Theorem

Let $A \in \mathbb{C}^{n\times n}$ be Hermitean with eigenvalues $\lambda_1 \le \cdots \le \lambda_n$. Then for any $||x|| = 1$

$$\lambda_1 \le x^*Ax \le \lambda_n$$

Proof. Let $\{x_1, \ldots, x_n\}$ be an ONB consisting of eigenvectors of A. Let $x = c_1x_1 + \cdots + c_nx_n$. Then $x^*Ax = \lambda_1|c_1|^2 + \cdots + \lambda_n|c_n|^2$. The result now follows easily. □

18.8 Lemma

Let L and G be subspaces of $\mathbb{C}^n$ and $\dim G > \dim L$. Then there is a nonzero vector in G orthogonal to L.

Proof. By way of contradiction, if $G \cap L^\perp = \{0\}$, then $G \oplus L^\perp$ is a subspace of $\mathbb{C}^n$ with dimension $\dim G + n - \dim L$, which is $> n$, which is impossible. □

18.9 Theorem (Courant-Fisher Minimax Theorem)

Let $A \in \mathbb{C}^{n\times n}$ be Hermitean with eigenvalues $\lambda_1 \le \cdots \le \lambda_n$. Then for every $i = 1, \ldots, n$

$$\lambda_i = \min_{\dim L = i} \max_{x \in L\setminus\{0\}} \frac{x^*Ax}{x^*x}$$

where L stands for a subspace of $\mathbb{C}^n$.

Proof. Let $\{u_1, \ldots, u_n\}$ be an ONB of eigenvectors of A corresponding to the eigenvalues $\lambda_1, \ldots, \lambda_n$. If $\dim L = i$, then by Lemma 18.8 there is a nonzero vector $x \in L$ orthogonal to the space $\operatorname{span}\{u_1, \ldots, u_{i-1}\}$. Hence, the first i coordinates of x are zero and $x = \sum_{j=i}^n c_j u_j$, so

$$\frac{x^*Ax}{x^*x} = \frac{\sum_{j=i}^n |c_j|^2 \lambda_j}{\sum_{j=i}^n |c_j|^2} \ge \lambda_i$$

Therefore,

$$\max_{x\in L\setminus\{0\}} \frac{x^*Ax}{x^*x} \ge \lambda_i$$

Now, take the subspace $L = \operatorname{span}\{u_1, \ldots, u_i\}$. Obviously, $\dim L = i$ and for every nonzero vector $x \in L$ we have $x = \sum_{j=1}^i c_j u_j$, so

$$\frac{x^*Ax}{x^*x} = \frac{\sum_{j=1}^i |c_j|^2 \lambda_j}{\sum_{j=1}^i |c_j|^2} \le \lambda_i$$

The theorem is proved. □

18.10 Theorem

Let A and ΔA be Hermitean matrices. Let $\alpha_1 \le \cdots \le \alpha_n$ be the eigenvalues of A, $\delta_{\min}$ and $\delta_{\max}$ the smallest and the largest eigenvalues of ΔA. Denote the eigenvalues of the matrix $B = A + \Delta A$ by $\beta_1 \le \cdots \le \beta_n$. Then for each $i = 1, \ldots, n$

$$\alpha_i + \delta_{\min} \le \beta_i \le \alpha_i + \delta_{\max}$$

Proof. Let $\{u_1, \ldots, u_n\}$ be an ONB consisting of eigenvectors of A corresponding to the eigenvalues $\alpha_1, \ldots, \alpha_n$. Let $L = \operatorname{span}\{u_1, \ldots, u_i\}$. Then, by 18.9,

$$\begin{aligned}
\beta_i &\le \max_{x\in L\setminus\{0\}} \frac{x^*Bx}{x^*x} \\
&\le \max_{x\in L\setminus\{0\}} \frac{x^*Ax}{x^*x} + \max_{x\in L\setminus\{0\}} \frac{x^*\Delta Ax}{x^*x} \\
&\le \alpha_i + \max_{x\in C^n\setminus\{0\}} \frac{x^*\Delta Ax}{x^*x} \\
&= \alpha_i + \delta_{\max}
\end{aligned}$$

which is the right inequality. Now apply this theorem to the matrices B, $-\Delta A$ and $A = B + (-\Delta A)$. Then its right inequality, just proved, will read $\beta_i \leq \alpha_i - \delta_{\min}$ (note that the largest eigenvalue of $-\Delta A$ is $-\delta_{\min}$). The theorem is completely proved. $\square$

Theorem 18.10 shows that if we perturb a Hermitean matrix A by a small Hermitean matrice ΔA, the eigenvalues of A will not change much. This is clearly related to numerical calculations of eigenvalues.

18.11 Remark.

A similar situation occurs when one knows an approximate eigenvalue λ and an approximate eigenvector x of a matrix A. We can assume that $||x|| = 1$. To estimate the closeness of λ to the actual but unknown eigenvalue of A, one can compute the residual $r = Ax - \lambda x$. Assume that r is small and define the matrix $\Delta A = -rx^*$. Then $||\Delta A||_2 = ||r||_2$ (see the homework assignment) and

$$(A + \Delta A)x = Ax - rx^*x = \lambda x$$

Therefore, (λ, x) are an exact eigenpair of a perturbed matrix $A + \Delta A$, and the norm $||\Delta A||$ is known. One could then apply Theorem 18.10 to estimate the closeness of λ to the actual eigenvalue of A, if the matrices A and ΔA were Hermitean. Since this is not always the case, we need to study how eigenvalues of a generic matrix change under small perturbations of the matrix. This is the issue of *eigenvalue sensitivity*.

18.12 Theorem (Bauer-Fike)

Let $A \in \mathbb{C}^{n\times n}$ and suppose that

$$Q^{-1}AQ = D = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$$

If μ is an eigenvalue of a perturbed matrix $A + \Delta A$, then

$$\min_{1\leq i\leq n} |\lambda_i - \mu| \leq \kappa_p(Q)||\Delta A||_p$$

where $||\cdot||_p$ stands for any p-norm.

Proof. If μ is an eigenvalue of A, the claim is trivial. If not, the matrix $D - \mu I$ is invertible. Observe that

$$\begin{aligned} Q^{-1}(A + \Delta A - \mu I)Q &= D + Q^{-1}\Delta AQ - \mu I \\ &= (D - \mu I)(I + (D - \mu I)^{-1}(Q^{-1}\Delta AQ)) \end{aligned}$$

Since the matrix $A + \Delta A - \mu I$ is singular, so is the matrix $I + (D - \mu I)^{-1}(Q^{-1}\Delta AQ)$. Then the Neumann lemma implies

$$1 \leq ||(D - \mu I)^{-1}(Q^{-1}\Delta AQ)||_p \leq ||(D - \mu I)^{-1}||_p||Q^{-1}||_p||\Delta A||_p||Q||_p$$

Lastly, observe that $(D-\mu I)^{-1}$ is diagonal, so

$$||(D-\mu I)^{-1}||_p = \max_{1\le i\le n} \frac{1}{|\lambda_i - \mu|} = \frac{1}{\min_{1\le i\le n} |\lambda_i - \mu|}$$

The theorem now follows. □

This theorem answers the question raised in 18.11, it gives an estimate on the error in the eigenvalue in terms of $||\Delta A||$ and $\kappa(Q)$. However, this answer is not good enough – it gives one estimate for all eigenvalues. In practice, some eigenvalues can be estimated much better than others. It is important then to develop finer estimates for individual eigenvalues.

18.13 Lemma

Let $A \in \mathbb{C}^{n\times n}$.

(i) If λ is an eigenvalue with a right eigenvector x, and $\mu \neq \lambda$ is another eigenvalue with a left eigenvector y, then $y^*x = 0$.

(ii) If λ is a simple eigenvalue (of algebraic multiplicity one) with right and left eigenvectors x and y, respectively, then $y^*x \neq 0$.

Proof. To prove (i), observe that $\langle Ax, y\rangle = \lambda\langle x, y\rangle$ and, by a remark after 18.1, $\langle x, A^*y\rangle = \langle x, \bar{\mu}y\rangle = \mu\langle x, y\rangle$. Hence, $\lambda\langle x, y\rangle = \mu\langle x, y\rangle$, which proves (i), since $\lambda \neq \mu$.

To prove (ii), assume that $||x|| = 1$. By the Schur decomposition theorem, there is a unitary matrix R with first column x such that

$$R^*AR = \begin{pmatrix} \lambda & h^* \\ 0 & C \end{pmatrix}$$

with some $h \in \mathbb{C}^{n-1}$ and $C \in \mathbb{C}^{(n-1)\times(n-1)}$. Note also that $Re_1 = x$. Since λ is a simple eigenvalue of A, it is not an eigenvalue of C, so the matrix $\lambda I - C$ is invertible, and so is $\bar{\lambda}I - C^*$. Let $z = (\bar{\lambda}I - C^*)^{-1}h$. Then $\bar{\lambda}z - C^*z = h$, hence

$$h^* + z^*C = \lambda z^*$$

One can immediately verify that

$$(1, z^*)R^*AR = \lambda(1, z^*)$$

Put $w^* = (1, z^*)R^*$. The above equation can be rewritten as

$$w^*A = \lambda w^*$$

Hence w is a left eigenvector of A. By the simplicity of λ, the vector w is a nonzero multiple of y. However, observe that

$$w^*x = (1, z^*)R^*Re_1 = 1$$

which proves the lemma. □

18.14 Theorem

Let $A \in \mathbb{C}^{n\times n}$ have a simple eigenvalue λ with right and left unit eigenvectors x and y, respectively. Let $E \in \mathbb{C}^{n\times n}$ such that $||E||_2 = 1$. For small ε, denote by $\lambda(\varepsilon)$ and $x(\varepsilon)$, $y(\varepsilon)$ the eigenvalue and unit eigenvectors of the matrix $A + \varepsilon E$ obtained from λ and x, y. Then

$$|\lambda'(0)| \leq \frac{1}{|y^*x|}$$

Proof. Write the equation

$$(A + \varepsilon E)\, x(\varepsilon) = \lambda(\varepsilon)\, x(\varepsilon)$$

and differentiate it in ε, set $\varepsilon = 0$, and get

$$Ax'(0) + Ex = \lambda'(0)x + \lambda\, x'(0)$$

Then multiply this equation through on the left by the vector y^*, use the fact that $y^*A = \lambda y^*$ and get

$$y^*Ex = \lambda'(0)\, y^*x$$

Now the result follows since $|y^*Ex| = |\langle y, Ex\rangle| \leq ||y||\,||Ex|| \leq 1$. Note that $y^*x \neq 0$ by 18.13. □

Note: The matrix $A+\varepsilon E$ is the perturbation of A in the direction of E. If the perturbation matrix E is known, one has exactly

$$\lambda'(0) = \frac{y^*Ex}{y^*x}$$

and so

$$\lambda(\varepsilon) = \lambda + \frac{y^*Ex}{y^*x}\varepsilon + O(\varepsilon^2)$$

by Taylor expansion, a fairly precise estimate on $\lambda(\varepsilon)$. In practice, however, the matrix E is absolutely unknown, so one has to use the bound in 18.14 to estimate the sensitivity of λ to small perturbations of A.

18.15 Definition (Condition Number of An Eigenvalue)

Let λ be a simple eigenvalue (of algebraic multiplicity one) of a matrix $A \in \mathbb{C}^{n\times n}$ and x, y the corresponding right and left unit eigenvectors. The *condition number* of λ is

$$K(\lambda) = \frac{1}{|y^*x|}$$

The condition number $K(\lambda)$ describes the sensitivity of a (simple) eigenvalue to small perturbations of the matrix. Large $K(\lambda)$ signifies an *ill-conditioned* eigenvalue.

18.16 Simple properties of $K(\lambda)$
(a) Obviously, $K(\lambda) \geq 1$.
(b) If a matrix A is normal, then $K(\lambda) = 1$ for all its eigenvalues.
(c) Conversely, if a matrix A has all simple eigenvalues with $K(\lambda) = 1$, then it is normal.

Normal matrices are characterized by the fact that the Shur decomposition $Q^*AQ = T$ results in a diagonal matrix T. One can expect that if the matrix T is nearly diagonal (its off-diagonal elements are small), then the eigenvalues of A are well-conditioned. On the contrary, if some off-diagonal elements of T are substantial, then at least some eigenvalues of A are ill-conditioned.

18.17 Remark

It remains to discuss the case of multiple eigenvalues (of algebraic multiplicity ≥ 2). If λ is a multiple eigenvalue, the left and right eigenvectors may be orthogonal even if the geometric multiplicity of λ equals one. Example: $A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$, the right and left eigenvectors are $x = e_1$ and $y = e_2$, respectively. Moreover, if the geometric multiplicity is ≥ 2, then for any right eigenvector x there is a left eigenvector y such that $y^*x = 0$. Hence, the definition 18.17 gives an infinite value of $K(\lambda)$.

This does not necessarily mean that a multiple eigenvalue is always ill-conditioned. It does mean, however, that an ill-conditioned simple eigenvalue is 'nearly multiple'. Precisely, if λ is a simple eigenvalue of A with $K(\lambda) > 1$, then there is a matrix E such that

$$\frac{||E||_2}{||A||_2} \leq \frac{1}{\sqrt{K(\lambda)^2 - 1}}$$

and λ is a multiple eigenvalue of $A + E$. We leave out the proof. We will not further discuss the sensitivity of multiple eigenvalues.

18.18 Theorem (Gershgorin)

Let $A \in \mathbb{C}^{n\times n}$ be 'almost diagonal'. Precisely, let $A = D+E$, where $D = \mathrm{diag}(d_1, \ldots, d_n)$ and $E = (e_{ij})$ is small. Then every eigenvalue of A lies in at least one of the circular disks

$$D_i = \{z : |z - d_i| \leq \sum_{j=1}^{n} |e_{ij}|\}$$

Note: D_i are called *Gershgorin disks*.

Proof. Let λ be an eigenvalue of A with eigenvector x. Let $|x_r| = \max_i\{|x_1|, \ldots, |x_n|\}$ be the maximal (in absolute value) component of x. We can normalize x so that $x_r = 1$

and $|x_i| \le 1$ for all i. On equating the r-th components in $Ax = \lambda x$ we obtain

$$(Ax)_r = d_r x_r + \sum_{j=1}^{n} e_{rj} x_j = d_r + \sum_{j=1}^{n} e_{rj} x_j = \lambda x_r = \lambda$$

Hence

$$|\lambda - d_r| \le \sum_{j=1}^{n} |e_{rj}|\,|x_j| \le \sum_{j=1}^{n} |e_{rj}|$$

The theorem is proved. □

18.19 Theorem

If k of the Gershgorin disks D_i form a connected region which is isolated from the other disks, then there are precisely k eigenvalues of A (counting multiplicity) in this connected region.

We call connected components of the union $\cup D_i$ *clusters*.

Proof. For brevity, denote

$$h_i = \sum_{j=1}^{n} |e_{ij}|$$

Consider a family of matrices $A(s) = D + sE$ for $0 \le s \le 1$. Clearly, the eigenvalues of $A(s)$ depend continuously on s. The Gershgorin disks $D_i(s)$ for the matrix $A(s)$ are centered at d_i and have radii sh_i. As s increases, the disks $D_i(s)$ grow concentrically, until they reach the size of the Gershgorin disks D_i of Theorem 18.18 at $s = 1$. When $s = 0$, each Gershgorin disk $D_i(0)$ is just a point, d_i, which is an eigenvalue of the matrix $A(0) = D$. So, if d_i is an eigenvalue of multiplicity $m \ge 1$, then exactly m disks $D_j(0)$ coincide with the point d_i, which make a cluster of m disks containing m eigenvalues (more precisely, one eigenvalue of multiplicity m). As the disks grow with s, the eigenvalues cannot jump from one cluster to another (by continuity), unless two cluster overlap and then make one cluster. When two clusters overlap (merge) at some s, they will be in one cluster for all larger values of s, including $s = 1$. This proves the theorem. □

Note: If the Gershgorin disks are disjoint, then each contains exactly one eigenvalue of A.

19 Eigenvalues and eigenvectors: computation

Eigenvalues of a matrix $A \in \mathbb{C}^{n\times n}$ are the roots of its characteristic polynomial, $C_A(x)$. It is a consequence of the famous Galois group theory (Abel's theorem) that there is no finite algorithm for calculation of the roots of a generic polynomial of degree > 4. Therefore, all the methods of computing eigenvalues of matrices larger than 4×4 are necessarily iterative, they only provide successive approximations to the eigenvalues.

If an eigenvalue λ of a matrix A is known, an eigenvector x can be found by solving the linear system $(A - \lambda I)x = 0$, which can be done by a finite algorithm (say, LU decomposition). But since the eigenvalues can only be obtained approximately, by iterative procedures, the same goes for eigenvectors. It is then reasonable to define a procedure that gives approximations for both eigenvalues and eigenvectors, in parallel. Also, knowing an approximate eigenvector x one can approximate the corresponding eigenvalue λ by the Rayleigh quotient x^*Ax/x^*x.

To simplify the matter, we always assume that the matrix A is diagonalizable, i.e. it has a complete set of eigenvectors $x_1, \ldots, x_n$ with eigenvalues $\lambda_1, \ldots, \lambda_n$, which are ordered in absolute value:

$$|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$$

19.1 Definition (Dominant Eigenvalue/Eigenvector)

Assume that $|\lambda_1| > |\lambda_2|$, i.e. largest eigenvalue is simple. We call λ_1 the *dominant eigenvalue* and x_1 a *dominant eigenvector*.

19.2 Power method: the idea

Let λ_1 be the dominant eigenvalue of A and

$$q = c_1x_1 + \cdots + c_nx_n$$

an arbitrary vector such that $c_1 \neq 0$. Then

$$\begin{aligned} A^kq &= c_1\lambda_1^kx_1 + \cdots + c_n\lambda_n^kx_n \\ &= \lambda_1^k[c_1x_1 + c_2(\lambda_2/\lambda_1)^kx_2 + \cdots + c_n(\lambda_n/\lambda_1)^kx_n] \end{aligned}$$

Denote

$$q^{(k)} = A^kq/\lambda_1^k = c_1x_1 + \underbrace{c_2(\lambda_2/\lambda_1)^kx_2 + \cdots + c_n(\lambda_n/\lambda_1)^kx_n}_{\Delta_k}$$

19.3 Lemma

The vector $q^{(k)}$ converges to c_1x_1. Moreover,

$$||\Delta_k|| = ||q^{(k)} - c_1x_1|| \leq \text{ const} \cdot r^k$$

where $r = |\lambda_2/\lambda_1| < 1$.

Therefore, the vectors $A^k q$ (obtained by the powers of A) will align in the direction of the dominant eigenvector x_1 as $k \to \infty$. The number r characterizes the speed of alignment, i.e. the speed of convergence $||\Delta_k|| \to 0$. Note that if $c_2 \neq 0$, then

$$||q^{(k+1)} - c_1 x_1||/||q^{(k)} - c_1 x_1|| \to r$$

The number r is called the *convergence ratio* or the *contraction number*.

19.4 Definition (Linear/Quadratic Convergence)
We say that the convergence $a_k \to a$ is *linear* if

$$|a_{k+1} - a| \leq r|a_k - a|$$

for some $r < 1$ and all sufficiently large k. If

$$|a_{k+1} - a| \leq C|a_k - a|^2$$

with some $C > 0$, then the convergence is said to be quadratic. It is much faster than linear.

19.5 Remark
In practice, the vector $q^{(k)} = A^k q/\lambda_1^k$ is inaccessible because we do not know λ_1 in advance. On the other hand, it is impractical to work with $A^k q$, because $||A^k q|| \to \infty$ if $|\lambda_1| > 1$ and $||A^k q|| \to 0$ if $|\lambda_1| < 1$. In order to avoid the danger of overflow or underflow, we must somehow normalize, or scale, the vector $A^k q$.

19.6 Power method: algorithm
Pick an initial vector q_0. For $k \geq 1$, define

$$q_k = Aq_{k-1}/\sigma_k$$

where σ_k is a properly chosen scaling factor. A common choice is $\sigma_k = ||Aq_{k-1}||$, so that $||q_k|| = 1$. Then one can approximate the eigenvalue λ_1 by the Rayleigh quotient

$$\lambda_1^{(k)} = q_k^* A q_k$$

Note that

$$q_k = A^k q_0/(\sigma_1 \cdots \sigma_k) = A^k q_0/||A^k q_0||$$

To estimate how close the unit vector q_k is to the one-dimensional eigenspace $\text{span}\{x_1\}$, denote by p_k the orthogonal projection of q_k on $\text{span}\{x_1\}$ and by $d_k = q_k - p_k$ the orthogonal component. Then $||d_k||$ measures the distance from q_k to $\text{span}\{x_1\}$.

19.7 Theorem (Convergence of Power Method)

Assume that λ_1 is the dominant eigenvalue, and $q_0 = \sum c_i x_i$ is chosen so that $c_1 \neq 1$. Then the distance from q_k to the eigenspace $\text{span}\{x_1\}$ converges to zero and $\lambda_1^{(k)}$ converges to λ_1. Furthermore,

$$||d_k|| \leq \text{const} \cdot r^k \qquad |\lambda_1^{(k)} - \lambda_1| \leq \text{const} \cdot r^k$$

Note: The sequence of unit vectors q_k need not have a limit, see examples.

Proof. It is a direct calculation, based on the representation $A^k q_0 = \lambda_1^k (c_1 x_1 + \Delta_k)$ of 19.2 and Lemma 19.3. □

19.8 Remark

Another popular choice for σ_k in 19.6 is the largest (in absolute value) component of the vector Aq_{k-1}. This ensures that $||q_k||_\infty = 1$. Assume that the vector x has one component of the largest absolute values. In that case Theorem 19.7 applies, and moreover, $\sigma_k \to \lambda_1$ so that

$$|\sigma_k - \lambda_1| \leq \text{const} \cdot r^k$$

Furthermore, the vector q_k will now converge to a dominant eigenvector.

19.9 Examples

(a) Let $A = \begin{pmatrix} 3 & 2 \\ 1 & 1 \end{pmatrix}$. Pick $q_0 = (1,1)^t$ and chose σ_k as in 19.8. Then $\sigma_1 = 5$ and $q_1 = (1, 0.4)^t$, $\sigma_2 = 3.8$ and $q_2 = (1, 0.368)^t$, $\sigma_3 = 3.736$ etc. Here σ_k converges to the dominant eigenvalue $\lambda_1 = 2 + \sqrt{3} = 3.732$ and q_k converges to a dominant eigenvector $(1, \sqrt{3}/2 - 1/2)^t = (1, 0.366)^t$.

(b) Let $A = \begin{pmatrix} -1 & 0 \\ 0 & 0 \end{pmatrix}$. Pick $q_0 = (1,1)^t$ and chose σ_k as in 19.6. Then $q_k = ((-1)^k, 0)$ does not have a limit, it oscillates. With σ_k chosen as in 19.8, we have $q_k = (1,0)$ and $\sigma_k = -1 = \lambda_1$ for all $k \geq 1$.

19.10 Remark

The choice of the initial vector q_0 only has to fulfill the requirement $c_1 \neq 0$. Since the vectors with $c_1 = 0$ form a hyperplane in $\mathbb{C}^n$, one hopes that a vector q_0 picked "at random" will not lie in that hyperplane "with probability one". Furthermore, even if $c_1 = 0$, round-off errors will most likely pull the numerical vectors q_k away from that hyperplane. If that does not seem to be enough, one can carry out the power method for n different initial vectors that make a basis, say $e_1, \ldots, e_n$. One of these vectors surely lies away from that hyperplane.

19.11 Inverse Power Method

Assume that A is invertible. Then $\lambda_1^{-1}, \ldots, \lambda_n^{-1}$ are the eigenvalues of A^{-1}, with the same eigenvectors $x_1, \ldots, x_n$. Note that $|\lambda_1^{-1}| \leq \cdots \leq |\lambda_n^{-1}|$.

Assume that $|\lambda_n^{-1}| > |\lambda_{n-1}^{-1}|$. Then λ_n^{-1} is the dominant eigenvalue of A^{-1} and x_n a dominant eigenvector. One can apply the power method to A^{-1} and find λ_n^{-1} and x_n. The rate of convergence of iterations will be characterized by the ratio $r = |\lambda_n/\lambda_{n-1}|$. This is called the *inverse power method*.

Now we know how to compute the largest and the smallest eigenvalues. The following trick allows us to compute any simple eigenvalue.

19.12 Inverse Power Method with Shift.

Recall that if λ is an eigenvalue of A with eigenvector x, then $\lambda - \rho$ is an eigenvalue of $A - \rho I$ with the same eigenvector x.

Assume that ρ is a good approximation to a simple eigenvalue λ_i of A, so that $|\lambda_i - \rho| < |\lambda_j - \rho|$ for all $j \neq i$. Then the matrix $A - \rho I$ will have the smallest eigenvalue $\lambda_i - \rho$ with an eigenvector x_i.

The inverse power method can now be applied to $A - \rho I$ to find $\lambda_i - \rho$ and x_i. The convergence of iterations will be linear with ratio

$$r = \frac{|\lambda_i - \rho|}{\min_{j \neq i} |\lambda_j - \rho|}$$

Hence, the better ρ approximates λ_i, the faster convergence is guaranteed.

By subtracting ρ from all the eigenvalues of A we shift the entire spectrum of A by ρ. The number ρ is called the *shift*. The above algorithm for computing λ_i and x_i is called the *inverse power method with shift*.

19.13 Rayleigh Quotient Iterations with Shift

This is an improvement of the algorithm 19.12. Since at each iteration of the inverse power method we obtain a better approximation to the eigenvalue λ_i, we can use it as the shift ρ for the next iteration. So, the shift ρ will be updated at each iteration. This will ensure a faster convergence. The algorithm goes as follows: one chooses an initial vector q_0 and an initial approximation ρ_0, and for $k \geq 1$ computes

$$q_k = \frac{(A - \rho_{k-1} I)^{-1} q_{k-1}}{\sigma_k}$$

and

$$\rho_k = \frac{q_k^* A q_k}{q_k^* q_k}$$

where σ_k a convenient scaling factor, for example, $\sigma_k = ||(A - \rho_{k-1} I)^{-1} q_{k-1}||$. The convergence of the Rayleigh quotient iterations is, generally, quadratic (better than linear).

The power method and its variations described above are classic. In recent years, however, the most widely used algorithm for calculating the complete set of eigenvalues of a matrix has been the QR algorithm.

19.14 The QR Algorithm

Let $A \in \mathbb{C}^{n\times n}$ be nonsingular. The algorithm starts with $A_0 = A$ and generates a sequence of matrices A_k defined as follows:

$$A_{k-1} = Q_k R_k \qquad R_k Q_k = A_k$$

That is, a QR factorizarion of A_{k-1} is computed and then its factors are reversed to produce A_k. One iteration of the QR algorithm is called a QR step.

19.15 Lemma

All matrices A_k in the QR algorithm are unitary equivalent, in particular they have the same eigenvalues.

Proof. Indeed, $A_{k+1} = Q_k^* A_k Q_k$. □

19.16 Theorem (Convergence of the QR Algorithm)

Let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of A satisfying

$$|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n| > 0$$

Under one technical assumption, see below, the matrix $A_k = (a_{ij}^{(k)})$ will converge to an upper triangular form, so that
(a) $a_{ij}^{(k)} \to 0$ as $k \to \infty$ for all $i > j$.
(b) $a_{ii}^{(k)} \to \lambda_i$ as $k \to \infty$ for all i.

This theorem is given without proof. The technical assumption here is that the matrix Y whose i-th row is a left eigenvector of A corresponding to λ_i for all i, must have an LU decomposition.

The QR algorithm described above is very reliable but quite expensive – each iteration takes too much flops and the convergence is rather slow. These problems can be solved with the help of Hessenberg matrices.

19.17 Definition (Upper Hessenberg Matrix)

A is called an *upper Hessenberg matrix* if $a_{ij} = 0$ for all $i > j + 1$, i.e. it has the form

$$\begin{pmatrix} \times & \cdots & & & \times \\ \times & \times & & & \\ 0 & \times & \times & & \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \times & \times \end{pmatrix}$$

19.18 Lemma
Every matrix $A \in \mathbb{C}^{n\times n}$ is unitary equivalent to an upper Hessenberg matrix, i.e.

$$A = Q^* A_0 Q$$

where A_0 is an upper Hessenberg matrix and Q is a unitary matrix. There is an explicit and inexpensive algorithm to compute such A_0 and Q.

This lemma is given without proof. It shows that one can first transform A to an upper Hessenberg matrix A_0, which has the same eigenvalues as A by similarity, and then start the QR algorithm with A_0.

19.19 Lemma
If the matrix A_0 is upper Hessenberg, then all the matrices A_k generated by the QR algorithm are upper Hessenberg.

Proof. By induction, let A_{k-1} be upper Hessenberg. Then $A_{k-1} = Q_k R_k$ and so $Q_k = A_{k-1} R_k^{-1}$. Since this is a product of an upper Hessenberg matrix and an upper triangular matrix, it is verified by direct inspection that Q_k is upper Hessenberg. Then, similarly, $A_k = R_k Q_k$ is a product of an upper triangular and upper Hessenberg matrices, so it is upper Hessenberg. □

For upper Hessenberg matrices, the QR algorithm can be implemented with the help of rotation matrices at a relatively low cost, cf. 16.25. This trick provides an inexpensive modification of the QR algorithm.

There is a further improvement of the QR algorithm that accelerates the convergence in Theorem 19.16.

19.20 Lemma
Assume that A_0, and hence A_k for all $k \geq 1$, are upper Hessenberg matrices. Then the convergence $a_{i+1,i}^{(k)} \to 0$ as $k \to \infty$ in Theorem 19.16 is linear with ratio $r = |\lambda_{i+1}/\lambda_i|$.

This lemma is given without proof. Note that $|\lambda_{i+1}/\lambda_i| < 1$ for all i. Clearly, the smaller this ratio the faster the convergence. Next, the faster the subdiagonal entries $a_{i+1,i}^{(k)}$ converge to zero the faster the diagonal entries $a_{ii}^{(k)}$ converge to the eigenvalues of A.

One can modify the matrix A to decrease the ratio $|\lambda_n/\lambda_{n-1}|$ and thus accelerate the convergence of $a_{nn}^{(k)} \to \lambda_n$, with the help of shifting, as in 19.12. One applies the QR steps to the matrix $A - \rho I$ where ρ is an appropriate approximation to λ_n. Then the

convergence $a^{(k)}_{n,n-1} \to 0$ will be linear with ratio $r = |\lambda_n - \rho|/|\lambda_{n-1} - \rho|$. The better ρ approximates λ_n the faster the convergence.

19.21 The QR Algorithm with Shift

Further improving the convergence, one can adjust the shift $\rho = \rho_k$ at each iteration. The *QR algorithm with shift* then goes as follows:

$$A_{k-1} - \rho_{k-1} I = Q_k R_k \qquad R_k Q_k + \rho_{k-1} I = A_k$$

where the shift ρ_k is updated at each step k. A standard choice for the shift ρ_k is the Rayleigh quotient of the vector e_n:

$$\rho_k = e_n^* A_k e_n = a^{(k)}_{nn}$$

(regarding e_n as the approximate eigenvector corresponding to λ_n). This is called the *Rayleigh quotient shift*. The convergence of $a^{(k)}_{n,n-1} \to 0$ and $a^{(k)}_{nn} \to \lambda_n$ is, generally, quadratic.

However, the other subdiagonal entries, $a^{(k)}_{i+1,i}$, $1 \le i \le n-2$, move to zero slowly (linearly). To speed them up, one uses the following trick. After making $a^{(k)}_{n,n-1}$ practically zero, one ensures that $a^{(k)}_{nn}$ is practically λ_n. Then one can partition the matrix A_k as

$$A_k = \begin{pmatrix} \hat{A}_k & b_k \\ 0 & \lambda_n \end{pmatrix}$$

where $\hat{A}_k$ is an $(n-1)\times(n-1)$ upper Hessenberg matrix, whose eigenvalues are (obviously) $\lambda_1, \dots, \lambda_{n-1}$. Then one can apply further steps of the QR algorithm with shift to the matrix $\hat{A}_k$ instead of A_k. This quickly produces its smallest eigenvalue, λ_{n-1}, which can be split off as above, etc. This procedure is called the *deflation* of the matrix A.

In practice, each eigenvalue of A requires 3-5 iterations (QR steps), on the average. The algorithm is rather fast and very accurate.