MATH3204

Numerical Lin. Alg. & Opt.

Matthew Low

September 8, 2019

λ

Abstract

Lectured by Fred Roosta. Offered in Semester 2, 2019. These notes are not endorsed in any way by the lecturer/lecturers, and are no substitute for lecture attendance. At the heart of most modern data scientific methods in general, and machine learning in particular, lie computational techniques involving matrices as well as numerical linear algebra and optimisation algorithms. In this course, students will learn about the theory and practical aspects of many fundamental tools from matrix computations, numerical linear algebra and optimisation. In addition to classical applications, most examples will particularly focus on modern large-scale machine learning problems. Implementations will be done using MATLAB/Python. The students will also be exposed to cutting-edge developments including randomised variants of many classical deterministic methods. Students will be taught a range of analytical and algorithmic tools that are employed in research and industry, such as various matrix types, their properties and factorisations, iterative algorithms for matrix computations such as Krylov subspace methods, various eigen-solvers, elements of convex and non-convex analysis, derivative free as well as first and second-order optimisation methods, constrained and unconstrained optimisation algorithms, and introduction to non-smooth and stochastic optimisation.

Contents

Part I

Numerical linear algebra

1 Vector spaces

1.1 Preliminary definitions

Definition 1.1: Vector space

A vector space is a special collection of vectors that can be:

- added together to produce more vectors;
- scaled by a scalar to produce more vectors.

Each vector space has a corresponding field.

Definition 1.2: Field, informal

A field is essentially a set of scalar. For us, normally $\mathbb{R}$ or $\mathbb{C}$. When we don't care which one, we will use the notation $\mathbb{F}$.

Note that any mathematical statement that holds for $\mathbb{F}$ will necessarily hold for either $\mathbb{R}$ or $\mathbb{C}$. Now we can define a field formally:

Definition 1.3: Field

A field is a set $\mathbb{F}$ together with two operations, called addition $+$ and multiplication $\times$ which satisfy the field axioms, which are the following:

1. **Associativity:** $\forall a, b, c \in \mathbb{F}$:

$$a + (b + c) = (a + b) + c, \quad a \times (b \times c) = (a \times b) \times c$$

2. **Commutativity:** $\forall a, b, c \in \mathbb{F}$:

$$a + b = b + a, \quad a \times b = b \times a$$

3. **Additive and multiplicative identity:** $\exists 0 \in \mathbb{F}, 1 \in \mathbb{F}$:

$$a + 0 = a, \quad a \times 1 = a$$

4. **Additive inverses:** $\forall a \in \mathbb{F}, \exists - a \in \mathbb{F}$ such that:

$$a + (-a) = 0$$

5. **Multiplicative inverses:** $\forall a \neq 0 \in \mathbb{F}, \exists \frac{1}{a} \in \mathbb{F}$ such that:

$$a \times \frac{1}{a} = 1$$

6. **Distributivity of multipliation over addition**: $\forall a, b, c \in \mathbb{F}$:

$$a \times (b + c) = (a \times b) + (a \times c)$$

Example 1.1: Examples of fields

- $\mathbb{R}$ - real numbers;
- $\mathbb{C}$ - complex numbers;
- $\mathbb{Q}$ - rational numbers

Definition 1.4: Vector space

A vector space $\mathcal{V}$ over a field $\mathbb{F}$ is a set of objects (called vectors), together with operations of vector addition $+$ and scalar multiplication $\times$, such that the following for all $\mathbf{u}, \mathbf{v}, \mathbf{w} \in \mathcal{V}$ and scalars $a, b \in \mathbb{F}$ hold:

1. **Closure of vector addition:**
$$\mathbf{u} + \mathbf{v} \in \mathcal{V}$$
2. **Commutativity of addition:**
$$\mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$$
3. **Associativity of addition:**
$$\mathbf{u} + (\mathbf{v} + \mathbf{w}) = (\mathbf{u} + \mathbf{v}) + \mathbf{w}$$
4. **Identity of addition:**
$$\exists \mathbf{0} \in \mathcal{V} \text{ such that } \mathbf{u} + \mathbf{0} = \mathbf{u} = \mathbf{0} + \mathbf{u}$$
5. **Inverse of addition:**
$$\exists -\mathbf{u} \in \mathcal{V} \text{ such that } \mathbf{u} + (-\mathbf{u}) = \mathbf{0} = (-\mathbf{u}) + \mathbf{u}$$
6. **Closure of scalar multiplication:**
$$a \times \mathbf{u} \in \mathcal{V}$$
7. **Distributive law 1:**
$$a \times (\mathbf{u} + \mathbf{v}) = a \times \mathbf{u} + a \times \mathbf{v}$$
8. **Distributive law 2:**
$$(a + b) \times \mathbf{u} = a \times \mathbf{u} + b \times \mathbf{u}$$
9. **Associative law:**
$$(ab) \times \mathbf{u} = a \times (b \times \mathbf{u})$$
10. **Monoidal law:**
$$1 \times \mathbf{u} = \mathbf{u}$$

We can simplify this for one statement defining operations on vector spaces:

Fact 1.1

For a vector space $\mathcal{V}$ over a field $\mathbb{F}$, we have:

$$a\mathbf{u} + b\mathbf{v} \in \mathcal{V}, \quad \forall \mathbf{u}, \mathbf{v} \in \mathcal{V}, \quad \forall a, b \in \mathbb{F}$$

Remark 1.1: Regarding the choice of field

Normally, it's obvious from the context. $\mathbb{F}$ is sitting in the background without being specifically mentioned.

Example 1.2: $\mathbb{F}^d$

For a given field $\mathbb{F}$ and a positive integer d, the set $\mathbb{F}^d$ (d-tuples with entries from $\mathbb{F}$) forms a vector space over $\mathbb{F}$ under element-wise addition in $\mathbb{F}^d$, defined by:

$$\alpha \begin{pmatrix} a \\ b \end{pmatrix} + \beta \begin{pmatrix} c \\ d \end{pmatrix} = \begin{pmatrix} \alpha a + \beta c \\ \alpha b + \beta d \end{pmatrix}, \quad a, b, c, d, \alpha, \beta \in \mathbb{F}$$

Example 1.3: Further examples of vector spaces

- $\mathbb{R}^d$ is a real vector space
- $\mathbb{C}^d$ is a complex vector space, and a real vector space.

Note 1.1

Elements of $\mathbb{F}^d$ are always presented as column vectors.

1.2 Subspaces, direct sums & span

Definition 1.5: Subspace

A subspace $\mathcal{W}$ of a vector space $\mathcal{V}$ over a field $\mathbb{F}$ is a subset of $\mathcal{W} \subseteq \mathcal{V}$ that is by itself a vector space of $\mathbb{F}$:

$$a\mathbf{u} + b\mathbf{v} \in \mathcal{V}, \quad \forall \mathbf{u}, \mathbf{v} \in \mathcal{W}, \quad \forall a, b \in \mathbb{F}$$

Example 1.4: No-c $\mathbb{R}^3$

$\left\{ \begin{pmatrix} a \\ b \\ 0 \end{pmatrix} \mid a, b \in \mathbb{R} \right\}$ is a subspace of $\mathbb{R}^3$.

Fact 1.2: Various facts about subspaces

The following hold:

- An intersection of subspaces $\mathcal{W} \cap \mathcal{X}$ is always a subspace.
- An union of subspaces $\mathcal{W} \cup \mathcal{X}$ **does not** need to be a subspace.

- A subspace cannot be empty, since a vector space always contains $\mathbf{0}$.

Definition 1.6: (non)Trivial subspace

The subsets $\{\mathbf{0}\}$ and $\mathcal{V}$ are always subspaces of $\mathcal{V}$. These are called **trivial subspaces**. Similarly, a subspace $\mathcal{W}$ of $\mathcal{V}$ is said to be **nontrivial** if it is not one of those.

Definition 1.7: Proper subspace

A subspace $\mathcal{W}$ of $\mathcal{V}$ is said to be a proper subspace if it is not equal to $\mathcal{V}$, eg. $\mathcal{W} \subset \mathcal{V}$.

Definition 1.8: Span

Let $\mathcal{V}$ be a vector space over $\mathbb{F}$ and $\mathcal{S} \subseteq \mathcal{V}$. The span $\mathsf{Span}(\mathcal{S})$ is the intersection of all subspaces that contain $\mathcal{S}$. If $\mathcal{S}$ is non-empty, then $\mathsf{Span}(\mathcal{S})$ is all of the linear combinations of all finitely many vectors in $\mathcal{S}$.

$$\mathsf{Span}(\mathcal{S}) = \begin{cases} \sum_{i=1}^{k} a_i \mathbf{v}_i \mid \mathbf{v}_1, \ldots, \mathbf{v}_k \in \mathcal{S}, a_1, \ldots, a_k \in \mathbb{F}, k \in \mathbb{N} & \text{non-empty} \\ \{\mathbf{0}\} & \text{empty} \end{cases}$$

We can work with sums of subspaces (instead of unions), defining the following sum:

Definition 1.9: Sum of two subspaces

Let $\mathcal{S}_1$ and $\mathcal{S}_2$ be subspaces of a vector space $\mathcal{V}$ over a field $\mathbb{F}$. Then the **sum** of $\mathcal{S}_1$ and $\mathcal{S}_2$ is defined as:

$$\mathcal{S}_1 + \mathcal{S}_2 = \mathsf{Span}(\mathcal{S}_1 \cup \mathcal{S}_2) = \{\mathbf{u} + \mathbf{v} \mid \mathbf{u} \in \mathcal{S}_1, \mathbf{v} \in \mathcal{S}_2\}$$

Example 1.5

$$\mathcal{S}_1 = \left\{ \begin{pmatrix} x \\ x \\ y \\ y \end{pmatrix} \mid x, y \in \mathbb{F} \right\}, \mathcal{S}_2 = \left\{ \begin{pmatrix} x \\ x \\ x \\ y \end{pmatrix} \mid x, y \in \mathbb{F} \right\}$$

$$\implies \mathcal{S}_1 + \mathcal{S}_2 = \left\{ \begin{pmatrix} x \\ x \\ y \\ z \end{pmatrix} \mid x, y, z \in \mathbb{F} \right\}$$

Theorem 1.1: Sum of subspaces is smallest subspace

Let $\mathcal{S}_1$ and $\mathcal{S}_2$ be subspaces of a vector space $\mathcal{V}$ over a field $\mathbb{F}$. Then, $\mathcal{S}_1 + \mathcal{S}_2$ is the **smallest subspace** containing $\mathcal{S}_1$ and $\mathcal{S}_2$.

Proof. $\mathcal{S}_1 + \mathcal{S}_2$ is trivially a subspace. $\mathcal{S}_1, \mathcal{S}_2 \subseteq \mathcal{S}_1 + \mathcal{S}_2$. Conversely, every subspace containing $\mathcal{S}_1, \mathcal{S}_2$ must contain $\mathcal{S}_1 + \mathcal{S}_2$. Hence, $\mathcal{S}_1 + \mathcal{S}_2$ is the smallest subspace that contains $\mathcal{S}_1$ and $\mathcal{S}_2$. □

Definition 1.10: Direct sum

If $\mathcal{S}_1 \cap \mathcal{S}_2 = \{\mathbf{0}\}$, then $\mathcal{S}_1 + \mathcal{S}_2$ is referred to as **direct sum**, and is denoted by $\oplus$.

Example 1.6: $\mathbb{R}^3$ composed of two subspaces

$$\mathbb{R}^3 = \left\{ \begin{pmatrix} a \\ b \\ 0 \end{pmatrix} \mid a, b \in \mathbb{R} \right\} \oplus \left\{ \begin{pmatrix} 0 \\ 0 \\ c \end{pmatrix} \mid c \in \mathbb{R} \right\}$$

Theorem 1.2: Uniquely represented as sum

Any $\mathbf{w} \in \mathcal{S}_1 \oplus \mathcal{S}_2$ can be **uniquely** represented as:

$$\mathbf{w} = \mathbf{u} + \mathbf{v}, \quad \mathbf{u} \in \mathcal{S}_1, \quad \mathbf{v} \in \mathcal{S}_2$$

Proof. Proof by contradiction. By the definition of subspace sum, any vector in $\mathcal{S}_1 \oplus \mathcal{S}_2$ can be written as

$$\mathbf{w} = \mathbf{u}_1 + \mathbf{v}_1, \quad \mathbf{u}_1 \in \mathcal{S}_1, \mathbf{v}_1 \in \mathcal{S}_2$$

Suppose we also write

$$\mathbf{w} = \mathbf{u}_2 + \mathbf{v}_2, \quad \mathbf{u}_2 \in \mathcal{S}_1, \mathbf{v}_2 \in \mathcal{S}_2$$

Combining these statements gives:

$$\mathbf{0} = (\mathbf{u}_1 - \mathbf{u}_2) + (\mathbf{v}_1 - \mathbf{v}_2)$$

Clearly, $\mathbf{u}_1 \neq \mathbf{u}_2 \implies \mathbf{v}_1 \neq \mathbf{v}_2$ and vice versa. This implies that:

$$\mathbf{u}_1 - \mathbf{u}_2 = \mathbf{v}_2 - \mathbf{v}_1 \implies \mathcal{S}_1 \cap \mathcal{S}_2 \neq \{\mathbf{0}\}$$

This is a contradiction from the fact that we are doing a direct sum, since the intersection must be zero for a direct sum. Therefore, $\mathbf{u}_1 = \mathbf{u}_2$ and $\mathbf{v}_1 = \mathbf{v}_2$. □

Note 1.2

One can show that considering whether $\mathbf{0}$ can be uniquely written as an appropriate sum is actually enough to decide whether a sum is a direct sum.

1.3 Linear independence, basis, dimension, ortho-

Let's consider we have a set of d vectors $\mathbf{v}_1, \ldots, \mathbf{v}_d$. We define a vector space $\mathcal{V} = \text{Span}\{\mathbf{v}_1, \ldots, \mathbf{v}_d\}$. We take any $\mathbf{v} \in \mathcal{V}$. By the definition of span (see above), there are scalars $\alpha_1, \ldots, \alpha_d$ such that:

$$\mathbf{v} = \sum_{i=1}^{d} \alpha_i \mathbf{v}_1$$

We define linear independence as whether this representation is unique. Consider that there is some $\beta_1, \ldots, \beta_d$ such that:

$$\mathbf{v} = \sum_{i=1}^{d} \beta_i \mathbf{v}_1$$

which implies:

$$\mathbf{0} = \sum_{i=1}^{d} (\alpha_i - \beta_i)\mathbf{v}_1$$

If $\mathbf{0}$ can only be written as the most obvious way ($\alpha_i = \beta_i$) then linearly independent.

Definition 1.11: Linear dependence & independence

- A finite set of vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ in a vector space $\mathcal{V}$ over a field $\mathbb{F}$ is **linearly dependent** if and only if there are scalars $a_1, \ldots, a_k \in \mathbb{F}$, **not all zero**, such that $\sum_{i=1}^{k} a_i\mathbf{v}_i = 0$.
- A finite set of vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ is **linearly independent** if they are not linearly dependent, i.e. if $\sum_{i=1}^{k} a_i\mathbf{v}_i = 0$ then we must have $a_1 = \ldots = a_k = 0$.

Note 1.3

An alternative explanation is that a set of vectors is linearly dependent **if and only if** some nontrivial linear combination of the vectors is the zero vectors (at least one of the vectors is a linear combination of some of the others). We can actually throw out that vector without changing the span of the original list.

Note 1.4

Linear dependence can be extended to include a set of countably infinite vectors.

- The set is linearly dependent over $\mathbb{F}$ if there exists a finite subset that is linearly dependent;
- The set is linearly independent over $\mathbb{F}$ if every finite subset is linearly independent.

Definition 1.12: Basis

A set of vectors that is linearly independent and spans some vector space forms a **basis** for that vector space. A set $\mathcal{B}$ (which could be countably infinite) is a basis for the vector space $\mathcal{V}$ if and only if:

- $\text{Span}(\mathcal{B}) = \mathcal{V}$;
- $\mathcal{B}$ is linearly independent.

Fact 1.3

Note the following two facts:

- Any $\mathbf{v} \in \mathcal{V}$ can be represented uniquely in terms of elements in $\mathcal{B}$. There is only one and only one way to choose $\mathbf{b}_1, \ldots, \mathbf{b}_k \in \mathcal{B}$ and $\alpha_1, \ldots, \alpha_k \in \mathbb{F}$ such that $\mathbf{v} = \sum_{i=1}^{k} \alpha_i\mathbf{b}_i$.
- Any linearly independent subset of list of $\mathcal{V}$ can be extended, perhaps in may ways, to form a basis of $\mathcal{V}$.

Definition 1.13: Finite-dimensional

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

Definition 1.14: Dimension

The dimension of a vector space $\mathcal{V}$, written as $\dim(\mathcal{V})$, over $\mathbb{F}$ is the number of vectors of any basis of $\mathcal{V}$ over $\mathbb{F}$.

Fact 1.4

More fun facts about dimensions:

- If $\mathcal{W}$ is a subspace of $\mathcal{V}$, then $\dim(\mathcal{W}) \leq \dim(\mathcal{V})$.
- If $\mathcal{W}$ is a subspace of $\mathcal{V}$ and $\dim(\mathcal{W}) = \dim(\mathcal{V})$, then $\mathcal{W} = \mathcal{V}$.
- If $\dim(\mathcal{V}) = d$, then every system of linearly independent vectors of $\mathcal{V}$ has at most d elements, and any basis of $\mathcal{V}$ has exactly d elements (this is called the Dimension Theorem).
- The only vector space with dimension 0 is $\{0\}$.

Example 1.7: Examples of bases

- $\mathcal{B} = \{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ is the canonical/standard basis for $\mathbb{R}^3$.
- $\left\{ \begin{pmatrix} a \\ b \\ 0 \end{pmatrix} \mid a, b \in \mathbb{R} \right\}$ is a 2-dimensional subspace of $\mathbb{R}^3$.
- The vector space $\mathbb{C}^d$ has dimension d over the field $\mathbb{C}$, but dimension $2d$ over the field $\mathbb{R}$ (think about real and imaginary parts in $a + bi$):
  - $\mathbb{F} = \mathbb{C} \implies \mathcal{B}_{\mathbb{C}} = \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}$
  - $\mathbb{F} = \mathbb{R} \implies \mathcal{B}_{\mathbb{R}} = \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} i \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ i \end{pmatrix} \right\}$
- $\mathcal{B} = \{1, t, t^2, t^3, \ldots\}$ is a basis for the real vector space of all polynomials with real coefficients. It is easy to see that each polynomial is a unique linear combination of (finitely many) elements in the basis.

Definition 1.15: Orthogonal/orthonormal vectors

A list of vectors $\mathbf{v}_1, \ldots, \mathbf{v}_m \in \mathbb{C}^n$ is orthogonal if:

$$\langle \mathbf{v}_i, \mathbf{v}_j \rangle = \mathbf{v}_i^* \mathbf{v}_j = \mathbf{v}_j^* \mathbf{v}_i = 0, \quad \forall i, j \in \{1, \ldots, m\}$$

Furthermore, the list is orthonormal if:

$$\|\mathbf{v}_i\|^1 = \langle \mathbf{v}_i, \mathbf{v}_i \rangle = 1, \quad \forall i \in \{1, \ldots, m\}$$

Fact 1.5: More facts about orthogonal/normal

- Every orthonormal list of vectors in $\mathbb{C}^n$ is linearly independent;

$$\begin{aligned} \mathbf{0} &= \sum_{i=1}^{n} \alpha_i \mathbf{v}_i \\ \implies 0 &= \left\langle \sum_{i=1}^{n} \alpha_i \mathbf{v}_i, \sum_{i=1}^{n} \alpha_i \mathbf{v}_i \right\rangle \\ &= \sum_{i=1}^{n} |\alpha_i|^2 \\ \implies \alpha_i &= 0, \quad \forall i \end{aligned}$$

- For a list of m orthonormal vectors in $\mathbb{C}^n$, we must have $m \leq n$;
- Any list of m orthonormal vectors in $\mathbb{C}^n$ form a basis for their span as an m-dimensional subspace of $\mathbb{C}^n$.

Lemma 1.1: Subspace Intersection Lemma

$$\dim(\mathcal{S}_1 \cap \mathcal{S}_2) + \dim(\mathcal{S}_1 + \mathcal{S}_2) = \dim(\mathcal{S}_1) + \dim(\mathcal{S}_2)$$

Recall the Inclusion-Exclusion Principle! Note the direct sum $\dim(\mathcal{S}_1 \oplus \mathcal{S}_2) = \dim(\mathcal{S}_1) + \dim(\mathcal{S}_2)$. Since $\dim(\mathcal{S}_1 + \mathcal{S}_2) \leq \dim(\mathcal{V})$, if $\dim(\mathcal{S}_1) + \dim(\mathcal{S}_2) - \dim(\mathcal{V}) \geq 1$, then $\mathcal{S}_1 \cap \mathcal{S}_2$ contains a non-zero vector.

Example 1.8

Let $\dim(\mathcal{S}_1) = 2, \dim(\mathcal{S}_2) = 2, \dim(\mathcal{S}_3) = 3$ which gives us:

$$\dim(\mathcal{S}_1 \cap \mathcal{S}_2) = 2 + 2 - 3 = 1$$

2 Linear maps & matrices

2.1 Mappings, operators & isomorphisms

Definition 2.1: Linear map

Let $\mathcal{U}$ and $\mathcal{V}$ be vector spaces over the same field $\mathbb{F}$. The mapping $\mathbf{f} : \mathcal{U} \to \mathcal{V}$ is called **linear** if:

$$\mathbf{f}(\alpha\mathbf{x} + \beta\mathbf{y}) = \alpha\mathbf{f}(\mathbf{x}) + \beta\mathbf{f}(\mathbf{y}), \quad \forall \mathbf{x}, \mathbf{y} \in \mathcal{U}, \quad \forall \alpha, \beta \in \mathbb{F}$$

Example 2.1: Examples of linear maps

- Differentiation: $f : \mathcal{P}(\mathbb{R}) \to \mathcal{P}(\mathbb{R})$ defined as $f(p) = p'$.
- Integration: $f : \mathcal{P}(\mathbb{R}) \to \mathcal{P}(\mathbb{R})$ defined as $f(p) = \int p(x)dx$.

- Multiplication by x: $f : \mathcal{P}(\mathbb{R}) \to \mathcal{P}(\mathbb{R})$ defined as $(fp)(x) = xp(x)$.

Definition 2.2: Invertible map

Let $\mathcal{U}, \mathcal{V}$ be vector spaces over the same field $\mathbb{F}$. The mapping $\mathbf{f} : \mathcal{U} \to \mathcal{V}$ is called invertible if $\exists! \mathbf{g} : \mathcal{V} \to \mathcal{U}$ such that:

1. $\mathbf{g} \circ \mathbf{f} : \mathcal{U} \to \mathcal{U}, \quad \mathbf{g} \circ \mathbf{f}(\mathbf{u}) = \mathbf{u}, \quad \forall \mathbf{u} \in \mathcal{U}$
2. $\mathbf{f} \circ \mathbf{g} : \mathcal{V} \to \mathcal{V}, \quad \mathbf{f} \circ \mathbf{g}(\mathbf{v}) = \mathbf{v}, \quad \forall \mathbf{v} \in \mathcal{V}$

$\mathbf{f}$ is invertible if it is a bijection.

Note 2.1

The inverse is often denoted by $\mathbf{f}^{-1}$.

Theorem 2.1: Inverse of a mapping is unique

An invertible map has a unique inverse.

Proof. Suppose $\mathbf{f}$ is invertible with inverses $\mathbf{g}_1$ and $\mathbf{g}_2$, so we have:

$$\mathbf{g}_1 = \mathbf{g}_1 \circ \mathbf{I} = \mathbf{g}_1 \circ (\mathbf{f} \circ \mathbf{g}_2) = (\mathbf{g}_1 \circ \mathbf{f}) \circ \mathbf{g}_2 = \mathbf{I} \circ \mathbf{g}_2 = \mathbf{g}_2$$

therefore the inverses $\mathbf{g}_1 = \mathbf{g}_2$ and it is unique. □

Generally, we need a map to be injective and surjective to be invertible. However, with linear operators (mappings from a vector space to itself on fin-dim spaces) this is not necessary:

Theorem 2.2: Invertibility of linear operators

Suppose $\mathcal{V}$ is finite dimensional and $\mathbf{f} : \mathcal{V} \to \mathcal{V}$ is a linear map. Then the following are equivalent:

- $\mathbf{f}$ is invertible;
- $\mathbf{f}$ is injective;
- $\mathbf{f}$ is surjective.

Proof. Later; see Rank-Nullity Theorem. □

Definition 2.3: Isomorphism for vector spaces

Let $\mathcal{U}, \mathcal{V}$ be vector spaces over the same field $\mathbb{F}$ with the same dimension. The mapping $\mathbf{f} : \mathcal{U} \to \mathcal{V}$ is called an isomoprhism if it is both linear and invertible. In this case, we say that $\mathcal{U}$ and $\mathcal{V}$ are isomorphic.

Note 2.2

One can think of an isomorphism $\mathbf{f} : \mathcal{U} \to \mathcal{V}$ as relabelling $\mathbf{u} \in \mathcal{U}$ as $\mathbf{f}(\mathbf{u}) \in \mathcal{V}$.

Fact 2.1

Note the following:

- Two finite-dimensional vector spaces over the same field are isomorphic if and only if they have the same dimension.
- Any d-dimensional vector space over $\mathbb{F}$ is isomorphic to $\mathbb{F}^d$.

Example 2.2: Examples of isomorphisms

- Let $\mathcal{V}$ be a d-dimensional vector space over $\mathbb{F}$ with basis
$$\mathcal{B} = \{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_d\}$$
For each $\mathbf{v} \in \mathcal{V}$, we have uniquely $\mathbf{v} = \sum_{i=1}^{d} \alpha_i \mathbf{b}_i$. So the mapping $\mathbf{f} : \mathcal{V} \to \mathbb{F}^d$ defined as:
$$\mathbf{f}(\mathbf{v}) = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_d \end{pmatrix} \triangleq [\mathbf{v}]_{\mathcal{B}}$$
is an isomorphism between $\mathcal{V}$ and $\mathbb{F}^d$. Note that $[\mathbf{v}]_{\mathcal{B}}$ denotes the coordinate vector of $\mathbf{v}$ relative to $\mathcal{B}$.
- $\mathcal{P}_2 = \text{Span}\{1, t, t^2\}$ is the space of polynomials of at most 2. The mapping $\mathbf{f}(a + bt + c^2) = \begin{pmatrix} a \\ b \\ c \end{pmatrix}$ is an isomorphism between $\mathcal{P}_2$ and $\mathbb{R}^3$.

2.2 Introduction to matrices

The fundamental object of study in numerical linear algebra is the **matrix**. (You go to bed with them??) One can think of matrices in two ways:

- Rectangular array of scalars (the traditional way)
- Linear maps between two vector spaces (far more consequences).

The second viewpoint gives rise to an enormous amount of theoretical properties.

Note 2.3

Matrices are typically denoted by boldface uppercase letters in this course, eg. **A**.

2.3 Matrices as rectangular array of scalars

$$\mathbf{A} = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix}$$

Number of rows: m, number of columns: n. Sometimes in textbooks $n \times m$ is used, this is a pain point for a number of colleagues.

Note 2.4: Elements

Sometimes, a_{ij} is denoted by $a_{ij} = [\mathbf{A}]_{ij}$.

Note 2.5: Characteristics

We sometimes call matrices:

- Square: $m = n$;
- Tall and skinny: $m > n$;
- Short and fat: $m < n$

Note 2.6

$\mathbb{F}^{m \times n}$ is isomorphic to $\mathbb{F}^{mn}$. We can put numbers in a vector $m \times n$ long and it is isomorphic to the matrix.

Definition 2.4: Transpose

The transpose of a matrix $\mathbf{A}$, denoted $\mathbf{A}^\top$, is defined as for any $\mathbf{A} \in \mathbb{F}^{m \times n}$:

$$[\mathbf{A}^\top]_{ij} = [\mathbf{A}]_{ji}$$

Definition 2.5: Hermitian transpose

The conjugate transpose, adjoint or Hermitian transpose of a matrix $\mathbf{A}$, denoted $\mathbf{A}^*$ (or $\mathbf{A}^H$) is defined as the following: for any $\mathbf{A} \in \mathbb{C}^{m \times n}$:

$$[\mathbf{A}^*]_{ij} = [\bar{\mathbf{A}}]_{ji} \text{ or } \mathbf{A}^* = (\bar{\mathbf{A}})^\top$$

Note 2.7

The complex conjugate is defined as following:

$$c = a + bi \implies \bar{c} = a - bi$$

Fact 2.2: Characteristics of matrices

Note the following characteristics of matrices (sum and multiplication):

- $(\mathbf{AB})^\top = \mathbf{B}^\top\mathbf{A}^\top$;
- $(\mathbf{AB})^* = \mathbf{B}^*\mathbf{A}^*$;
- $(\mathbf{A}+\mathbf{B})^\top = \mathbf{A}^\top + \mathbf{B}^\top$;
- $(\mathbf{A}+\mathbf{B})^* = \mathbf{A}^* + \mathbf{B}^*$;

Definition 2.6: Symmetric

A square matrix $\mathbf{A} \in \mathbb{F}^{n\times n}$ is symmetric if

$$\mathbf{A}^\top = \mathbf{A}$$

Definition 2.7: Skew-symmetric

A square matrix $\mathbf{A} \in \mathbb{F}^{n\times n}$ is skew-symmetric if

$$\mathbf{A}^\top = -\mathbf{A}$$

Definition 2.8: Orthogonal

A square matrix $\mathbf{A} \in \mathbb{F}^{n\times n}$ is orthogonal if

$$\mathbf{A}^\top\mathbf{A} = \mathbf{I}$$

where $\mathbf{I}$ is the $n \times n$ identity matrix.

Definition 2.9: Hermitian

A square matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$ is Hermitian if

$$\mathbf{A}^* = \mathbf{A}$$

Definition 2.10: Skew-Hermitian

A square matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$ is skew-Hermitian if

$$\mathbf{A}^* = -\mathbf{A}$$

Definition 2.11: Unitary

A square matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$ is unitary if

$$\mathbf{A}^*\mathbf{A} = \mathbf{I}$$

where $\mathbf{I}$ is the $n \times n$ identity matrix.

Definition 2.12: Normal

A square matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$ is normal if

$$\mathbf{A}^*\mathbf{A} = \mathbf{A}\mathbf{A}^*$$

This table should help with remembering which is which:

$\mathbb{F}^{n\times n}$		$\mathbb{C}^{n\times n}$
symmetric	→	Hermitian
skew-symmetric	→	skew-Hermitian
orthogonal	→	unitary

Exercise 2.1

- Show that a Hermitian matrix has real main diagonal entries;
- Show that a skew-Hermitian matrix has pure imaginary main diagonal entries;
- What are the main diagonal entries of a real skew-symmetric matrix?

We define some operations on matrices:

Definition 2.13: Sum of two matrices

For any $\mathbf{A}, \mathbf{B} \in \mathbb{F}^{m\times n}$, the sum of $\mathbf{A}$ and $\mathbf{B}$ is:

$$[\mathbf{A}+\mathbf{B}]_{ij} = [\mathbf{A}]_{ij} + [\mathbf{B}]_{ij}$$

Definition 2.14: Scalar multiplication of matrices

For any $\mathbf{A} \in \mathbb{F}^{m\times n}$, the scalar multiplication of that matrix by λ is defined as:

$$[\lambda\mathbf{A}]_{ij} = \lambda[\mathbf{A}]_{ij}$$

Definition 2.15: Matrix inner product

For any $\mathbf{A} \in \mathbb{F}^{m\times n}$ and $\mathbf{B} \in \mathbb{F}^{n\times p}$, we have $\mathbf{AB} \in \mathbb{F}^{m\times p}$, where:

$$[\mathbf{AB}]_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

where $[\mathbf{AB}]_{ij}$ is the inner-product of the ith row of $\mathbf{A}$ and the jth column of $\mathbf{B}$.

Definition 2.16: Matrix outer product

Let

$$\mathbf{A} = [\mathbf{a}_1 \mid \mathbf{a}_2 \mid \ldots \mid \mathbf{a}_n]$$
$$\mathbf{B} = [\mathbf{b}_1 \mid \mathbf{b}_2 \mid \ldots \mid \mathbf{b}_n]^\top$$

We can combine this to obtain **AB**:

$$\mathbf{AB} = \sum_{i=1}^{n} \mathbf{a}_i \mathbf{b}_i^\top$$

where **AB** is the sum of outer-products of columns of **A** and the corresponding rows of **B**.

Note 2.8: Outer product

The outer product of two vectors **v**, **w** is:

$$v \times w = vw^*$$

whereas typically the inner product is:

$$\langle v, w \rangle = v^* w$$

The outer product is not commutative, whereas the inner product is commutative.

Sometimes multiplication in block forms is easier to think about:

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

Definition 2.17: Determinant

The determinant of a matrix **A** is a function $\det : \mathbb{F}^{n \times n} \to \mathbb{F}$ defined as (the Leibniz formula):

$$\det \mathbf{A} = \sum_{\pi \in \mathcal{P}} \operatorname{sgn}(\pi) \prod_{i=1}^{n} a_{i\pi_i}$$

Fact 2.3: Facts about determinants:

- $\det(\mathbf{A}) = \det(\mathbf{A}^\top)$;
- $\det(\mathbf{A}^*) = \det(\bar{\mathbf{A}}) = \overline{\det(\mathbf{A})}$;
- $\det(\mathbf{AB}) = \det(\mathbf{A})\det(\mathbf{B})$;
- $\det(\mathbf{A}^{-1}) = \frac{1}{\det(\mathbf{A})}$;
- $\det(\alpha \mathbf{A}) = \alpha^n \det(\mathbf{A}), \forall \alpha \in \mathbb{F}$.

Proposition 2.1

For any unitary matrix $\mathbf{U} \in \mathbb{C}^{n \times n}$, $|\det(\mathbf{U})| = 1$.

Proof. Since **U** is unitary, we have $\mathbf{U}^*\mathbf{U} = \mathbf{I}$. Therefore:

$$\begin{aligned} 1 &= \det(\mathbf{I}) \\ &= \det(\mathbf{U}^*\mathbf{U}) \end{aligned}$$

$$
\begin{aligned}
&= \det(\mathbf{U}^*)\det(\mathbf{U}) \\
&= \overline{\det(\mathbf{U})}\det(\mathbf{U}) \\
&= |\det(\mathbf{U})|^2 \\
\implies |\det(\mathbf{U})| &= \sqrt{1} = 1
\end{aligned}
$$

□

Definition 2.18: Trace

The trace of a matrix **A** is a function $\text{Trace} : \mathbb{F}^{n\times n} \to \mathbb{F}$ that is defined by:

$$\text{Trace}(\mathbf{A}) = \sum_i a_{ii}$$

Fact 2.4: Properties of trace

- $\text{Trace}(\mathbf{A}) = \text{Trace}(\mathbf{A}^\top)$;
- $\text{Trace}(\mathbf{A}^*) = \text{Trace}(\bar{\mathbf{A}})$;
- $\text{Trace}(\mathbf{AB}) \neq \text{Trace}(\mathbf{A})\,\text{Trace}(\mathbf{B})$;
- $\text{Trace}(\mathbf{ABC}) = \text{Trace}(\mathbf{CAB}) = \text{Trace}(\mathbf{BCA})$ (this is known as the **cyclic property**).

2.4 Matrices as linear maps between two vector spaces

Let us work towards a theorem about matrices and linear maps. Let $\mathcal{U}$ be a vector space of dimension n over $\mathbb{F}$ with a basis $\mathcal{B}_\mathcal{U}$:

$$\mathcal{B}_\mathcal{U} = \{\mathbf{u}_1, \dots, \mathbf{u}_n\} \implies \mathbf{U} \triangleq [\mathbf{u}_1 \mid \dots \mid \mathbf{u}_n]$$

Similarly, let $\mathcal{V}$ be a vector space of dimension m over $\mathbb{F}$ with a basis $\mathcal{B}_\mathcal{V}$:

$$\mathcal{B}_\mathcal{V} = \{\mathbf{v}_1, \dots, \mathbf{v}_n\} \implies \mathbf{V} \triangleq [\mathbf{v}_1 \mid \dots \mid \mathbf{v}_n]$$

Let us consider the linear transformation $\mathbf{f} : \mathcal{U} \to \mathcal{V}$ defined as:

$$\mathbf{f}(\alpha\mathbf{u}_1 + \beta\mathbf{u}_2) = \alpha\mathbf{f}(\mathbf{u}_1) + \beta\mathbf{f}(\mathbf{u}_2), \quad \forall \mathbf{u}_1, \mathbf{u}_2 \in \mathcal{U}, \alpha, \beta \in \mathbb{F}$$

Let us take any vector in $\mathcal{B}_\mathcal{U}$. For any $\mathbf{u}_i \in \mathcal{B}_\mathcal{U}$, $i = 1, \dots, n$, we have $\mathbf{f}(\mathbf{u}_i \in \mathcal{V})$, and therefore:

$$
\begin{aligned}
\mathbf{f}(\mathbf{u}_i) &= \sum_{j=1}^{m} a_{ji}\mathbf{v}_j \\
&= (\mathbf{v}_1 \mid \dots \mid \mathbf{v}_m) \cdot \begin{pmatrix} a_{1i} \\ \vdots \\ a_{mi} \end{pmatrix} \\
&= \mathbf{V} \cdot [\mathbf{f}(\mathbf{u}_i)]_{\mathcal{B}_\mathcal{V}}
\end{aligned}
$$

Note 2.9

An important skill is to write matrices in ‘matricised’ form. Much easier to read.

Similarly, for any $\mathbf{x} \in \mathcal{U}$, we have:

$$\mathbf{x} = \sum_{i=1}^{n} x_i \mathbf{u}_i = \mathbf{U} \cdot [\mathbf{x}]_{\mathcal{B}_{\mathcal{U}}}$$

This allows us to get a new expression for $\mathbf{f}(\mathbf{x})$:

$$\begin{aligned}
\mathbf{f}(\mathbf{x}) &= \mathbf{f}\left(\sum_{i=1}^{n} x_i \mathbf{u}_i\right) \\
&= \sum_{i=1}^{n} x_i \mathbf{f}(\mathbf{u}_i) \\
&= \sum_{i=1}^{n} x_i \sum_{j=1}^{m} a_{ji} \mathbf{v}_j \\
&= \sum_{i=1}^{n} \sum_{j=1}^{m} a_{ji} x_i \mathbf{v}_j \\
&= \sum_{j=1}^{m} \mathbf{v}_j \left(\sum_{i=1}^{n} a_{ji} x_i\right) \\
&= \mathbf{V}\mathbf{A}[\mathbf{x}]_{\mathcal{B}_{\mathcal{U}}} \\
&= (\mathbf{v}_1 \mid \dots \mid \mathbf{v}_n) \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}
\end{aligned}$$

Since $\mathbf{f}(\mathbf{x}) = \mathbf{V} \cdot [\mathbf{f}(\mathbf{x})]_{\mathcal{B}_{\mathcal{V}}}$, and $\mathbf{V}$ is a basis (implying full rank), we have:

$$\mathbf{V}[\mathbf{f}(\mathbf{x})]_{\mathcal{B}_{\mathcal{V}}} = \mathbf{V}\mathbf{A}[\mathbf{x}]_{\mathcal{B}_{\mathcal{U}}} \implies [\mathbf{f}(\mathbf{x})]_{\mathcal{B}_{\mathcal{V}}} = \mathbf{A}[\mathbf{x}]_{\mathcal{B}_{\mathcal{U}}}$$

Definition 2.19: Matrix representation

The $m \times n$ matrix $\mathbf{A}$ defined by the scalars a_{ij} is called the **matrix representation** of $\mathbf{f}$ in the ordered bases $\mathcal{B}_{\mathcal{U}}$ and $\mathcal{B}_{\mathcal{V}}$.

Theorem 2.3: Matrices and linear maps

Let $\mathbf{f} : \mathbb{F}^n \to \mathbb{F}^m$ be a linear map. Then $\exists! \mathbf{A} \in \mathbb{F}^{m \times n}$ such that:

$$\mathbf{f}(\mathbf{x}) = \mathbf{A}\mathbf{x} \quad \forall \mathbf{x} \in \mathbb{F}^n$$

Conversely, if $\mathbf{A} \in \mathbb{F}^{m \times n}$ then the function defined above is a **linear map** from $\mathbb{F}^n$ to $\mathbb{F}^m$.

Example 2.3: Givens rotation matrix

The counterclockwise rotation of a vector $\mathbf{v} \in \mathbb{R}^2$ by an angle θ is given using the linear map/-matrix:

$$\mathbf{A} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

This is known as the Givens rotation matrix and it is orthogonal.

Example 2.4: Simple example

Let $\mathbf{f} : \mathbb{R}^2 \to \mathbb{R}^3$ be defined as $\mathbf{f}(x, y) = (x + y, 2x - y, x - y)$. We use the canonical bases for 2 and 3-space:

$$\mathcal{B}_{\mathbb{R}^2} = \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}, \quad \mathcal{B}_{\mathbb{R}^3} = \left\{ \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\}$$

The corresponding matrix is:

$$\mathbf{f}(x, y) = \begin{pmatrix} x + y \\ 2x - y \\ x - y \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 2 & -1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

What if we change the basis?

$$\tilde{\mathcal{B}}_{\mathbb{R}^2} = \left\{ \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right\}, \quad \tilde{\mathcal{B}}_{\mathbb{R}^3} = \left\{ \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \\ -2 \end{pmatrix} \right\}$$

We need to find some $\mathbf{A}$ such that $\mathbf{f}(\mathbf{x}) = \mathbf{A}\mathbf{x}$, or:

$$\begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix} = \mathbf{A} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}$$

where:

$$\left[\begin{pmatrix} x \\ y \end{pmatrix} \right]_{\tilde{\mathcal{B}}_{\mathbb{R}^2}} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}, \quad \left[\begin{pmatrix} x + y \\ 2x - y \\ x - y \end{pmatrix} \right]_{\tilde{\mathcal{B}}_{\mathbb{R}^3}} = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix}$$

We know these are the bases, so we can uniquely write them uniquely as a linear combination in terms of the new basis as:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}$$

We can do the same for $\mathbf{f}(x, y)$:

$$\mathbf{f}\left(\begin{pmatrix} x \\ y \end{pmatrix} \right) = \begin{pmatrix} x + y \\ 2x - y \\ x - y \end{pmatrix}$$

We need to find the representation in the new basis, not the canonical basis, so we apply the inverse of the matrix to it.

$$\begin{pmatrix}\beta_1\\\beta_2\\\beta_3\end{pmatrix} = \begin{pmatrix}1&1&1\\1&2&0\\1&1&-2\end{pmatrix}^{-1}\begin{pmatrix}x+y\\2x-y\\x-y\end{pmatrix}$$

Overall:

$$\underbrace{\begin{pmatrix}\beta_1\\\beta_2\\\beta_3\end{pmatrix}}_{[\mathbf{f}(\mathbf{x})]_{\tilde{\mathcal{B}}_{\mathbb{R}^3}}} = \underbrace{\begin{pmatrix}1&1&1\\1&2&0\\1&1&-2\end{pmatrix}^{-1}\begin{pmatrix}1&1\\2&-1\\1&-1\end{pmatrix}\begin{pmatrix}2&1\\1&1\end{pmatrix}}_{\mathbf{A}}\underbrace{\begin{pmatrix}\alpha_1\\\alpha_2\end{pmatrix}}_{[\mathbf{x}]_{\tilde{\mathcal{B}}_{\mathbb{R}^2}}} \implies \mathbf{A} = \begin{pmatrix}7&7\\-2&-3\\2&2\end{pmatrix}$$

2.5 Matrix subspaces

Through isomorphism, any n-dimensional vector space over $\mathbb{F}$ may be identified with $\mathbb{F}^n$. To any linear mapping from $\mathbb{F}^n$ to $\mathbb{F}^m$, we may always assign a matrix $\mathbf{A} \in \mathbb{F}^{m\times n}$:

$$\mathbf{A} : \mathbb{F}^n \to \mathbb{F}^m$$

From now on, we will only look at this case. We define some functions:

Definition 2.20: Domain of a matrix

The domain of $\mathbf{A}$ is $\mathsf{Domain}(\mathbf{A}) = \mathbb{F}^n$.

Definition 2.21: Range of a matrix

The range of $\mathbf{A}$ is

$$\mathsf{Range}(\mathbf{A}) = \{\mathbf{y} \in \mathbb{F}^m \mid \mathbf{y} = \mathbf{A}\mathbf{x} \text{ for some } \mathbf{x} \in \mathbb{F}^n\}$$

Note that $\mathsf{Range}(\mathbf{A})$ is a subspace of $\mathbb{F}^m$ (doesn't have to be m-dimensional, just has to be $\leq m$).

Definition 2.22: Rank of a matrix

The rank of a matrix $\mathbf{A}$ is the dimension of the range of that matrix: $\dim(\mathsf{Range}(\mathbf{A})) = \mathsf{Rank}(A)$.

Definition 2.23: Full-rank

A full-rank matrix $\mathbf{A} \in \mathbb{F}^{m\times n}$ is a matrix with rank $= \min\{m, n\}$.

Definition 2.24: Rank-deficient

A rank-deficient matrix $\mathbf{A} \in \mathbb{F}^{m\times n}$ is a matrix with rank $< \min\{m, n\}$.

Definition 2.25: Nullspace

The nullspace of a matrix, denoted $\text{Null}(\mathbf{A})$ or $\text{Kernel}(\mathbf{A})$ is the set of all $\mathbf{x} \in \mathbb{F}^n$ such that $\mathbf{A}\mathbf{x} = \mathbf{0}$:

$$\text{Null}(\mathbf{A}) = \text{Kernel}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{F}^n \mid \mathbf{A}\mathbf{x} = \mathbf{0}\}$$

Note that $\text{Null}(\mathbf{A})$ is a subspace of $\mathbb{F}^n$ (once again, doesn't have to be n-dim.)

Definition 2.26: Nullity of a matrix

The nullity of a matrix $\mathbf{A}$ is the dimension of the nullspace of that matrix: $\dim(\text{Null}(\mathbf{A})) = \text{Nullity}(A)$.

Theorem 2.4: Rank-Nullity Theorem

Important! If $\mathbf{A} \in \mathbb{F}^{m \times n}$:

$$\dim(\text{Range}(\mathbf{A})) + \dim(\text{Null}(\mathbf{A})) = n$$

Definition 2.27: Orthogonal complement

The orthogonal complement of a subspace $\mathcal{S}$, denoted $\mathcal{S}^\perp$, is:

$$\mathcal{S}^\perp = \{\mathbf{v} \mid \langle \mathbf{v}, \mathbf{w} \rangle = 0, \forall \mathbf{w} \in \mathcal{S}\}$$

Essentially all the vectors that are orthogonal to the whole subspace.

Theorem 2.5: Four Fundamental Subspaces

If $\mathbf{A} \in \mathbb{C}^{m \times n}$ (an m-by-n matrix in complex space), then:

$$\text{Null}(\mathbf{A}) = \text{Range}(\mathbf{A}^*)^\perp \text{ and } \text{Null}(\mathbf{A}^*) = \text{Range}(\mathbf{A})^\perp$$

Proof. Let $\mathbf{x} \in \text{Null}(\mathbf{A})$. Take any $\mathbf{y} \in \text{Range}(\mathbf{A}^*)$, we have that $\mathbf{y} = \mathbf{A}^*\mathbf{z}$ for some $\mathbf{z}$. So we have:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{A}^*\mathbf{z} \rangle = \langle \mathbf{A}\mathbf{x}, \mathbf{z} \rangle = 0$$

which implies that $\mathbf{x} \in \text{Range}(\mathbf{A}^*)^\perp$, and hence $\text{Null}(\mathbf{A}) \subseteq \text{Range}(\mathbf{A}^*)^\perp$.

Conversely, let $\mathbf{x} \in \text{Range}(\mathbf{A}^*)^\perp$, which means that for any $\mathbf{y} \in \text{Range}(\mathbf{A}^*)$, we have that their inner product $\langle \mathbf{x}, \mathbf{y} \rangle = 0$. In particular, choosing $\mathbf{y} = \mathbf{A}^*\mathbf{A}\mathbf{x}$ implies:

$$0 = \langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{A}^*\mathbf{A}\mathbf{x} \rangle = \langle \mathbf{A}\mathbf{x}, \mathbf{A}\mathbf{x} \rangle = ||\mathbf{A}\mathbf{x}||^2$$

which gives $\mathbf{A}\mathbf{x} = 0$, which tells us that $\text{Range}(\mathbf{A}^*)^\perp \subseteq \text{Null}(\mathbf{A})$. Since we have shown that they are subsets of each other (a common proof technique), they are equal. The other statement is proved similarly. □

Definition 2.28: Column space

The column space of a matrix $\mathbf{A}$, denoted $\text{colsp}(\mathbf{A})$, is simply the range of $\mathbf{A}$:

$$\text{colsp}(\mathbf{A}) = \text{Range}(\mathbf{A}) = \{\mathbf{Ax} \mid \mathbf{x} \in \mathbb{C}^m\}$$

Definition 2.29: Row space

The row space of a matrix $\mathbf{A}$, denoted $\text{rowsp}(\mathbf{A})$, is simply the range of $\mathbf{A}^\top$:

$$\text{rowsp}(\mathbf{A}) = \text{Range}(\mathbf{A}^\top) = \{\mathbf{A}^\top\mathbf{x} \mid \mathbf{x} \in \mathbb{C}^n\}$$

Fact 2.5: Rank and (col/row)sp

Similar to earlier, we have that:

$$\dim(\text{colsp}(\mathbf{A})) = \dim(\text{rowsp}(\mathbf{A})) = \text{Rank}(\mathbf{A}) \le \min\{m, n\}$$

You can't have a higher rank than $\min\{m, n\}$!!!!! Remember this!!!

Fact 2.6: Characteristics of rank

- $\text{Rank}(\mathbf{A}) = \text{Rank}(\mathbf{A}^\top) = \text{Rank}(\mathbf{A}^*)$;
- $\text{Rank}(\mathbf{A}^*\mathbf{A}) = \text{Rank}(\mathbf{A})$.

Theorem 2.6: Full-rank factorisation

$\mathbf{A}$ has rank r if and only if:

$$\mathbf{A} = \mathbf{XY}^\top \text{ for some } \mathbf{X} \in \mathbb{C}^{m\times r}, Y \in \mathbb{C}^{n\times r}$$

(matrix outer product) each having full rank (independent columns.)

Theorem 2.7: Bounds on rank

If $\mathbf{A} \in \mathbb{C}^{m\times p}$ and $\mathbf{B} \in \mathbb{C}^{p\times n}$:

$$\text{Rank}(\mathbf{A}) + \text{Rank}(\mathbf{B}) - p \le \text{Rank}(\mathbf{AB}) \le \min\{\text{Rank}(\mathbf{A}), \text{Rank}(\mathbf{B})\}$$

3 Norms of vectors and matrices

3.1 Singularity and pseudoinverses

We don't normally care about actually computing inverses in the real world...

Definition 3.1: Non-singular

$\mathbf{A} \in \mathbb{F}^{n \times n}$ is said to be non-singular if $\mathbf{Ax} = 0 \iff \mathbf{x} = 0$.

Fact 3.1: Short and fat matrices are necessarily singular

$\mathbf{A} \in \mathbb{F}^{m \times n}$ with $m < n$ (short and fat) is necessarily singular

Fact 3.2: Equivalent to non-singular

- $\text{Rank}(\mathbf{A}) = n$;
- $\exists! \mathbf{A}^{-1} \in \mathbb{F}^{n \times n}$ such that $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$;
- $\det(\mathbf{A}) \neq 0$
- $\dim(\text{Range}(\mathbf{A})) = n$ and $\dim(\text{Null}(\mathbf{A})) = 0$;
- $\text{Null}(\mathbf{A}) = \{\mathbf{0}\}$;
- $\mathbf{A}$ has linearly independent rows and columns;
- The linear system $\mathbf{Ax} = \mathbf{b}$ has a unique solution for each $\mathbf{b} \in \mathbb{F}^n$.

Fact 3.3: Swapping inverse with transpose

You can swap the inverse with the transpose or Hermitian conjugate:

If $\mathbf{A} \in \mathbb{F}^{n \times n}$ is non-singular, then $(\mathbf{A}^{-1})^\top = (\mathbf{A}^\top)^{-1} \triangleq \mathbf{A}^{-\top}$.

If $\mathbf{A} \in \mathbb{C}^{n \times n}$ is non-singular, then $(\mathbf{A}^{-1})^* = (\mathbf{A}^*)^{-1} \triangleq \mathbf{A}^{-*}$.

Definition 3.2: Pseudo-inverse

For any $\mathbf{A} \in \mathbb{F}^{m \times n}$ matrix, $\exists! \mathbf{A}^\dagger \in \mathbb{F}^{n \times m}$ called the pseudo-inverse that satisfies the following four properties;

- $\mathbf{A}\mathbf{A}^\dagger\mathbf{A} = \mathbf{A}$;
- $\mathbf{A}^\dagger\mathbf{A}\mathbf{A}^\dagger = \mathbf{A}^\dagger$;
- $(\mathbf{A}\mathbf{A}^\dagger)^* = \mathbf{A}\mathbf{A}^\dagger$;
- $(\mathbf{A}^\dagger\mathbf{A})^* = \mathbf{A}^\dagger\mathbf{A}$.

Note 3.1

Full-column rank means more rows than columns.

Fact 3.4

When $\mathbf{A}$ is full-column rank, we have a left inverse:

$$\mathbf{A}^\dagger = (\mathbf{A}^*\mathbf{A})^{-1}\mathbf{A}^*$$

and so $\mathbf{A}^\dagger\mathbf{A} = \mathbf{I}$.

Fact 3.5

When $\mathbf{A}$ is full-row rank, we have a right inverse:

$$\mathbf{A}^{\dagger} = \mathbf{A}^{*}(\mathbf{A}\mathbf{A}^{*})^{-1}$$

and so $\mathbf{A}\mathbf{A}^{\dagger} = \mathbf{I}$.

Fact 3.6: Pseudoinverse equals inverse

If $\mathbf{A}$ is invertible, its pseudoinverse is its inverse.

Fact 3.7: More properties of pseudoinverse

- $(\mathbf{A}^{\dagger})^{\dagger} = \mathbf{A}$;
- $(\mathbf{A}^{\dagger})^{\top} = (\mathbf{A}^{\top})^{\dagger}$;
- $(\mathbf{A}^{*})^{\top} = (\mathbf{A}^{*})^{\dagger}$.

Fact 3.8

Unlike the inverse, where this is valid:

$$(\mathbf{A}\mathbf{B})^{\dagger} \neq \mathbf{B}^{\dagger}\mathbf{A}^{\dagger}$$

3.2 Vector norms

Norms encapsulate the notions of size and distance in a vector space. A vector is thought of as "small" if its norm is small, and also vectors $\mathbf{x}, \mathbf{y}$ are close if the norm of the difference $\mathbf{x} - \mathbf{y}$ is small. Approximations and convergence throughout numerical linear algebra and optimisation is measured using norms.

Definition 3.3: Vector norm

Given a vector space $\mathcal{V}$ over $\mathbb{F}$, a norm is a non-negative real-valued function $\|\cdot\| : \mathcal{V} \to [0, \infty)$ with the following properties, namely:

- **Sub-additivity/triangle inequality**: $\|\mathbf{u} + \mathbf{v}\| \leq \|\mathbf{u}\| + \|\mathbf{v}\|$;
- **Absolute homogeneity**: $\|\alpha\mathbf{u}\| = |\alpha|\|\mathbf{u}\|$;
- **Positive definiteness**: $\|\alpha\mathbf{u}\| = 0 \iff \mathbf{u} = \mathbf{0}$.

Lemma 3.1: Reverse triangle inequality

Any norm satisfies:

$$|\|\mathbf{x}\| - \|\mathbf{y}\|| \leq \|\mathbf{x} - \mathbf{y}\|$$

Proof. We write:

$$\begin{aligned}
\mathbf{x} = \mathbf{y} + (\mathbf{x}-\mathbf{y}) &\implies \|\mathbf{x}\| = \|\mathbf{y} + (\mathbf{x}-\mathbf{y})\| \leq \|\mathbf{y}\| + \|(\mathbf{x}-\mathbf{y})\| \\
\mathbf{y} = \mathbf{x} + (\mathbf{y}-\mathbf{x}) &\implies \|\mathbf{y}\| = \|\mathbf{x} + (\mathbf{y}-\mathbf{x})\| \leq \|\mathbf{x}\| + \|(\mathbf{y}-\mathbf{x})\|
\end{aligned}$$

Therefore $\|\mathbf{x}\| - \|\mathbf{y}\| \leq \|\mathbf{x}-\mathbf{y}\|$. □

Definition 3.4: Vector p-norms

The p-norms are the following:

ℓ_1	Manhattan norm	$\|\mathbf{x}\|_1 = \sum_{i=1}^{d} \|x_i\|$
ℓ_2	Euclidean norm	$\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{d} \|x_i\|^2} = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}$
ℓ_∞	max norm	$\|\mathbf{x}\|_\infty = \max_{i=1,\ldots,d} \|x_i\|$
ℓ_p		$\left(\sum_{i=1}^{d} \|x_i\|^p \right)^{1/p}$

Lemma 3.2: Hölder's Inequality for $\mathbb{C}^d$

For any $1 \leq p, q \leq \infty$ such that $\frac{1}{p} + \frac{1}{q} = 1$, we have:

$$|\langle \mathbf{x}, \mathbf{y} \rangle| \leq \|\mathbf{x}\|_p \|\mathbf{y}\|_q$$

with equality if and only if $\mathbf{x}$ and $\mathbf{y}$ are linearly dependent. For $p = 2$, this is the **Cauchy-Schwarz** inequality.

Proposition 3.1: Equivalence of norms in $\mathbb{C}^d$

For all $\mathbf{x} \in \mathbb{C}^d$, we have:

$$\|\mathbf{x}\|_\infty \leq \|\mathbf{x}\|_2 \leq \|\mathbf{x}\|_1 \leq \sqrt{d}\|\mathbf{x}\|_2 \leq d\|\mathbf{x}\|_\infty$$

In other words, for $\mathbb{R}^n/\mathbb{C}^n$, all norms are equivalent up to a constant.

Definition 3.5: Weighted Euclidean norm

Let $\mathbf{W}$ be a diagonal matrix with positive diagonal elements. The **weighted Euclidean norm** is defined as:

$$\|\mathbf{x}\|_\mathbf{W} \triangleq \sqrt{\langle \mathbf{x}, \mathbf{W}\mathbf{x} \rangle}$$

Theorem 3.1: Unitary invariance of Euclidean norm in $\mathbb{C}^d$

Given any matrix $\mathbf{U} \in \mathbb{C}^{m \times d}$ with $m \geq d$ and orthonormal columns, we have:

$$\|\mathbf{Ux}\|_2 = \|\mathbf{x}\|_2$$

Proof. One-liner:

$$\|\mathbf{Ux}\|_2^2 = \langle \mathbf{Ux}, \mathbf{Ux} \rangle = \langle \mathbf{x}, \mathbf{U}^*\mathbf{Ux} \rangle = \langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|^2$$

□

3.3 Entry-wise matrix norms

Remark 3.1

An $m \times n$ matrix can be viewed as a vector in an mn-dimensional space.

Therefore, any mn-dimensional p-norm can be used for measuring the "size" of a matrix:

$$\|\mathbf{A}\|_p = \|\operatorname{vec}(\mathbf{A})\|_p = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^p \right)^{\frac{1}{p}}$$

This is sometimes called an entry-wise matrix norm. The most famous of these is the Frobenius norm:

Definition 3.6: Frobenius norm

Given any $\mathbf{A} \in \mathbb{C}^{m \times n}$, the ℓ_2 norm of the associated mn-dimensional vector is the Frobenius norm of the matrix:

$$\|\mathbf{A}\|_F \triangleq \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2 \right)^{\frac{1}{2}}$$

It is easy to see that $\|\mathbf{A}\|_F = \sqrt{\operatorname{Trace}(\mathbf{A}^*\mathbf{A})}$. We'll see later why we necessarily have that the trace of $\mathbf{A}^*\mathbf{A}$ is non-negative.

Theorem 3.2: Unitary invariance of Frobenius norm in $\mathbb{C}^{m \times n}$

Given any matrix $\mathbf{U} \in \mathbb{C}^{p \times m}$ with $p \geq m$ and orthonormal columns, we have:

$$\|\mathbf{UA}\|_F = \|\mathbf{A}\|_F$$

Proof. Another one-liner:

$$\|\mathbf{UA}\|_F^2 = \operatorname{Trace}(\mathbf{A}^*\mathbf{U}^*\mathbf{UA}) = \operatorname{Trace}(\mathbf{A}^*\mathbf{A}) = \|\mathbf{A}\|_F^2$$

□

Theorem 3.3: Sub-multiplicativity of entry-wise matrix norms

For any two matrices $\mathbf{A} \in \mathbb{C}^{m \times n}$, $\mathbf{B} \in \mathbb{C}^{n \times p}$:

$$\begin{aligned} \|\mathbf{AB}\|_F &\leq \|\mathbf{A}\|_F \|\mathbf{B}\|_F \\ \|\mathbf{AB}\|_1 &\leq \|\mathbf{A}\|_1 \|\mathbf{B}\|_1 \end{aligned}$$

(note Frobenius norm is the entry-wise ℓ_2 norm)

In general, however, the max-norm is not submultiplicative!

Example 3.1: Counterexample

Let $\mathbf{A} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$. We have:

$$\begin{aligned} \|\mathbf{A}^2\|_\infty = 2\|\mathbf{A}\|_\infty &= 2 \\ \|\mathbf{A}\|_\infty^2 &= 1 \\ 2 &\neq 1 \end{aligned}$$

Definition 3.7: Induced matrix norm

Consider an arbitrary matrix $\mathbf{A} \in \mathbb{F}^{m \times s}$. Given any two norms $\| \cdot \|_p$ and $\| \cdot \|_q$ respectively on $\mathrm{Domain}(\mathbf{A})$, $\mathrm{Range}(\mathbf{A})$, the corresponding induced matrix norm is defined as:

$$\|\mathbf{A}\|_{p,q} \triangleq \max_{\substack{\mathbf{x} \in \mathbb{F}^m \\ \mathbf{x} \neq \mathbf{0}}} \frac{\|\mathbf{Ax}\|_q}{\|\mathbf{x}\|_p} = \max_{\substack{\mathbf{x} \in \mathbb{F}^m \\ \|\mathbf{x}\|_p = 1}} \|\mathbf{Ax}\|_q$$

A common abbreviation if $p = q$ is to shorten $\|\mathbf{A}\|_{p,p}$ to $\|\mathbf{A}\|_p$.

Note 3.2

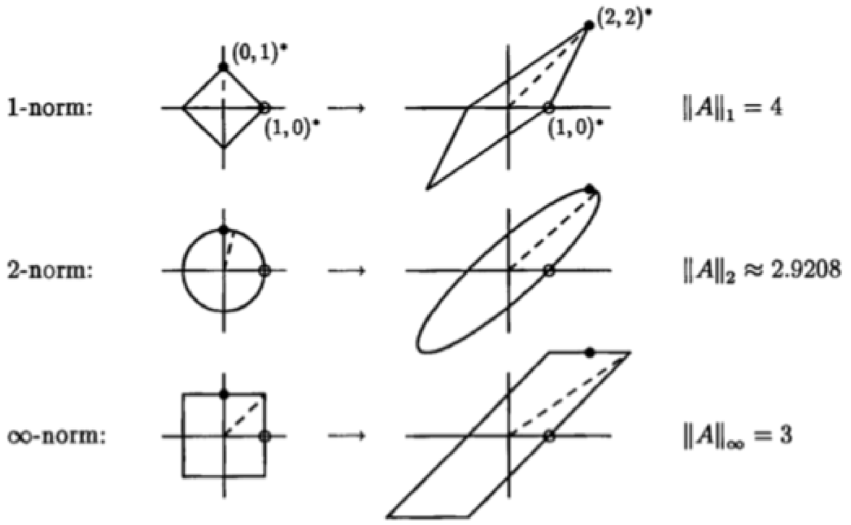
Induced norms give an indication of the maximum factor by which **A** can “stretch” a vector.

Exercise 3.1

Show that the above definition is indeed a norm, i.e., satisfies the norm properties mentioned earlier on.

Example 3.2

Consider the matrix $\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 0 & 2 \end{pmatrix}$ as a mapping from $\mathbb{R}^2$ to $\mathbb{R}^2$.



3.4 Matrix norms induced by vector norms

Theorem 3.4: Unitary invariance of induced 2-norm in $\mathbb{F}^{m\times n}$

Given any matrix $\mathbf{U} \in \mathbb{F}^{p\times m}$ orthonormal columns, we have:

$$\|\mathbf{UA}\|_2 = \|\mathbf{A}\|_2$$

where the norm here is the induced 2-norm (Euclidean norm)

Proof. Immediate, by noticing that for any $\mathbf{x} \in \mathbb{F}^m$, we have:

$$\|\mathbf{UAx}\|_2 = \|\mathbf{Ax}\|_2$$

□

Theorem 3.5: All induced matrix norms are submultiplicative

Let $\|\cdot\|_p, \|\cdot\|_q, \|\cdot\|_r$ be vector norms on, respectively, $\text{Domain}(\mathbf{B}), \text{Range}(\mathbf{B}), \text{Range}(\mathbf{A})$. We have:

$$\|\mathbf{AB}\|_{p,r} \leq \|\mathbf{A}\|_{q,r}\|\mathbf{B}\|_{p,q}$$

Proof. For any $\mathbf{x}$, we have that:

$$\|\mathbf{ABx}\|_r \leq \|\mathbf{A}\|_{q,r}\|\mathbf{Bx}\|_q \leq \|\mathbf{A}\|_{q,r}\|\mathbf{B}\|_{p,q}\|\mathbf{x}\|_p$$

□

Theorem 3.6: Equivalence of induced matrix norms

For any $\mathbf{A} \in \mathbb{C}^{m\times n}$ with $\text{Rank}(\mathbf{A}) = r$, we have:

$$\|\mathbf{A}\|_2 \leq \|\mathbf{A}\|_F \leq \sqrt{r}\|\mathbf{A}\|_2$$

$$\|\mathbf{A}\|_\infty \leq \sqrt{n}\|\mathbf{A}\|_2 \leq \sqrt{mn}\|\mathbf{A}\|_\infty$$
$$\|\mathbf{A}\|_1 \leq \sqrt{m}\|\mathbf{A}\|_2 \leq \sqrt{mn}\|\mathbf{A}\|_1$$

Remark 3.2

Consider $\mathbf{Ax} = \mathbf{b}$. An important question we will run into later is to ask, "how accurate can we expect the solution of a linear system to be if the right hand side $\mathbf{b}$ is perturbed by $\delta\mathbf{b}$?" More specifically, suppose $\mathbf{A} \in \mathbb{C}^{n\times n}$ is invertible, and we have $\mathbf{b} \in \mathbb{C}^n$. Let $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$. Now suppose $\hat{\mathbf{b}} = \mathbf{b} + \delta\mathbf{b}$, and $\hat{\mathbf{x}} = \mathbf{A}^{-1}\hat{\mathbf{b}}$. How close is $\hat{\mathbf{x}}$ to $\mathbf{x}$? This can be encapsulated in the notion of the **matrix condition number**.

Consider the matrix $\mathbf{A} \in \mathbb{C}^{m\times n}$ as a mapping from $\mathbb{C}^m$ to $\mathbb{C}^n$, and consider any vector norm $\|\cdot\|$. For a given input $\mathbf{x}$ and a corresponding change in the input $\delta\mathbf{x}$, we have:

$$\left(\frac{\|\mathbf{A}(\mathbf{x}+\delta\mathbf{x}) - \mathbf{Ax}\|}{\mathbf{Ax}} \bigg/ \frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|}\right) = \left(\frac{\|\mathbf{A}\delta\mathbf{x}\|}{\|\delta\mathbf{x}\|} \bigg/ \frac{\|\mathbf{Ax}\|}{\|\mathbf{x}\|}\right)$$

Definition 3.8: Condition of MVP

Let $\mathbf{A} \in \mathbb{C}^{m\times n}$, and consider any vector norm $\|\cdot\|$ with its induced matrix norm. For a given vector $\mathbf{x}$, the condition of MVP for $\mathbf{A}$ is defined as:

$$\kappa(\mathbf{A};\mathbf{x}) \triangleq \max_{\delta\mathbf{x}}\left(\|\frac{\mathbf{A}\delta\mathbf{x}\|}{\|\delta\mathbf{x}\|} \bigg/ \frac{\|\mathbf{Ax}\|}{\|\mathbf{x}\|}\right) = \frac{\|\mathbf{A}\|\|\mathbf{x}\|}{\|\mathbf{Ax}\|}$$

Note 3.3

The condition number of $\mathbf{A}$ is the worst case scenario over all of the changes of all inputs.

Definition 3.9: Condition Number

In the above, if $m \geq n$ and $\mathbf{A}$ has full column rank, then the condition number of $\mathbf{A}$, relative to $\|\cdot\|$, is defined as:

$$\kappa(\mathbf{A}) = \max_{\mathbf{x}} \kappa(\mathbf{A};\mathbf{x}) = \|\mathbf{A}\|\|\mathbf{A}^\dagger\|$$

Definition 3.10: Well and ill-conditioned

If $\kappa(\mathbf{A})$ is small, $\mathbf{A}$ is said to be well-conditioned. If $\kappa(\mathbf{A})$ is large, $\mathbf{A}$ is ill-conditioned.

Example 3.3: Back to our linear system example

A question arises as to whether the bound is pessimistic or whether there are examples of $\mathbf{b}$ and $\delta\mathbf{b}$ for which the relative error in $\mathbf{b}$ is magnified by exactly $\kappa(\mathbf{A})$. For a given $\mathbf{A}$, are there $\mathbf{b}$ and $\delta\mathbf{b}$ for which we have:

$$\frac{\|\hat{\mathbf{x}} - \mathbf{x}\|}{\|\mathbf{x}\|} \overset{?}{=} \kappa(\mathbf{A})\frac{\|\delta\mathbf{b}\|}{\|\mathbf{b}\|}$$

The answer is: **the bound is not pessimistic and is sharp for certain problems**. We know there is a $\mathbf{y}_0$ for which:

$$||\mathbf{A}|| = \max_{\mathbf{y}\neq\mathbf{0}} \frac{||\mathbf{A}\mathbf{y}||}{||\mathbf{y}||} = \frac{||\mathbf{A}\mathbf{y}_0||}{||\mathbf{y}_0||}$$

Let $\mathbf{x} = \mathbf{y}_0$. We also know there is a $\mathbf{z}_0$ for which:

$$||\mathbf{A}^{-1}|| = \max_{\mathbf{z}\neq\mathbf{0}} \frac{||\mathbf{A}^{-1}\mathbf{z}||}{||\mathbf{z}||} = \frac{||\mathbf{A}^{-1}\mathbf{z}_0||}{||\mathbf{z}_0||}$$

Let $\delta\mathbf{b} = \mathbf{z}_0$. Go through the above derivation and note that with these choices, we have:

$$||\mathbf{b}|| = ||\mathbf{A}\mathbf{x}|| = ||\mathbf{A}||||\mathbf{x}|| \quad \text{and} \quad ||\mathbf{A}^{-1}\delta\mathbf{b}|| = ||\mathbf{A}^{-1}||||\delta\mathbf{b}||$$

Lemma 3.3

We always have $\kappa(\mathbf{A}) \geq 1$.

Proof. One liner:

$$1 = ||\mathbf{A}\mathbf{A}^{\dagger}|| \leq ||\mathbf{A}||||\mathbf{A}^{\dagger}||$$

□

Exercise 3.2

What is the condition number of a unitary matrix?

Remark 3.3

Even though we motivated condition number by perturbing **b** or **x**, we could also have perturbed **A**. However, all these perturbations result in the same notion of $\kappa(\mathbf{A})$.

Note 3.4: Ill conditioned problems are hard to solve

Generally speaking, in solving $\mathbf{A}\mathbf{x} = \mathbf{b}$, one can expect to “lose $\log_{10}\kappa(\mathbf{A})$ digits” in computing the solution, except under very special lucky circumstances.

4 Matrix spectral properties

4.1 Eigenvalues and eigenvectors

Let’s consider a problem we’ve seen many times before. Given a matrix $\mathbf{a} \in \mathbb{C}^{n\times n}$, are there **nonzero** vectors that are mapped to a possibly scaled version of themselves (their direction is invariant under **A**):

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$$

Definition 4.1: Eigenvalue and eigenvector

Let $\mathbf{A} \in \mathbb{C}^{n\times n}$. If we have:

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v}, \quad \mathbf{v} \in \mathbb{C}^n, \mathbf{v} \neq \mathbf{0}, \lambda \in \mathbb{C}$$

then:

- λ is called an eigenvalue of $\mathbf{A}$;
- $\mathbf{v}$ is called an eigenvector of $\mathbf{A}$ associated with λ;
- the pair $(\lambda, \mathbf{v})$ is an eigenpair for $\mathbf{A}$.

Note 4.1: Facts about eigen-x

- Eigenpairs are only defined for square matrices $\mathbf{A}$.
- An eigenvector can never be the zero vector, otherwise the definition is vacuous

Fact 4.1

$$(\lambda, \mathbf{v}) \text{ is an eigenpair } \iff (\lambda\mathbf{I} - \mathbf{A})\mathbf{v} = \mathbf{0} \text{ and } \mathbf{v} \neq \mathbf{0}$$

Fact 4.2

Eigenvalues are the roots of the characteristic polynomial of $\mathbf{A}$, i.e. $\det(\lambda\mathbf{I}-\mathbf{A}) = 0$. $\det(\lambda\mathbf{I}-\mathbf{A})$ is a polynomial of degree exactly n in λ, i.e.:

$$\det(\lambda\mathbf{I} - \mathbf{A}) = p_n(\lambda) = \sum_{k=0}^{n} c_k \lambda^i, c_n \neq 0$$

Remark 4.1

Does any matrix have an eigenpair? Over $\mathbb{C}$, yes, but over $\mathbb{R}$, no. Counterexample: $\mathbf{A} = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$. $\det(A - \lambda\mathbf{I}) \implies \lambda = 1 \pm i$.

Note 4.2

Finding eigenvalues is equivalent to solving polynomial equations. By the Fundamental Theorem of Algebra, any matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ has exactly n eigenvalues, not necessarily all distinct, among the complex numbers.

Definition 4.2: Spectrum

The spectrum of $\mathbf{A} \in \mathbb{C}^{n \times n}$, denoted by $\text{spec}(\mathbf{A})$, is the set of all eigenvalues of $\mathbf{A}$:

$$\text{spec}(\mathbf{A}) = \{\lambda \in \mathbb{C} \mid \exists \mathbf{v} \neq \mathbf{0}, \mathbf{A}\mathbf{v} = \lambda\mathbf{v}\}$$

Definition 4.3: Spectral radius

The spectral radius of a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ is the maximum magnitude of an eigenvector in the spectrum of that matrix:

$$\rho(\mathbf{A}) \triangleq \max_{\lambda \in \mathrm{spec}(\mathbf{A})} |\lambda|$$

Fact 4.3: Facts about eigenpairs and conjugates

- $(\lambda, c\mathbf{v})$ is an eigenpair for $\mathbf{A}$ for any $c \in \mathbb{C}$ e.g. $c = \frac{1}{\|\mathbf{v}\|_2}$;
- If $\mathbf{A} \in \mathbb{C}^{n\times n}$ and $\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$, then $\bar{\mathbf{A}}\bar{\mathbf{v}} = \bar{\lambda}\bar{\mathbf{v}}$;
- If $\mathbf{A} \in \mathbb{R}^{n\times n}$ and $\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$, then $\mathbf{A}\bar{\mathbf{v}} = \bar{\lambda}\bar{\mathbf{v}}$, i.e. for real matrices, if $(\lambda, \mathbf{v})$ is an eigenpair, then so is $(\bar{\lambda}, \bar{\mathbf{v}})$;
- If $\mathbf{A} \in \mathbb{C}^{n\times n}$ and λ is an eigenvalue of $\mathbf{A}$, then $\bar{\lambda}$ is an eigenvalue of $\mathbf{A}^*$, but **eigenvectors might not be related**.

Fact 4.4: Determinants and trace

Amazingly:

$$\det(\mathbf{A}) = \prod_{i=1}^{n} \lambda_i, \quad \mathrm{Trace}(\mathbf{A}) = \sum_{i=1}^{n} \lambda_i$$

Fact 4.5: Facts about spectrum and its radius

- $\mathrm{spec}(\mathbf{A}) = \mathrm{spec}(\mathbf{A}^\top)$;
- $\mathrm{spec}(\bar{\mathbf{A}}) = \mathrm{spec}(\mathbf{A}^*)$;
- $\mathrm{spec}(\mathbf{A}) \neq \mathrm{spec}(\mathbf{A}^*)$ but **always** $\rho(\mathbf{A}) = \rho(\mathbf{A}^*)$;
- $\forall \alpha \in \mathbb{C}$:

$$\rho(\alpha\mathbf{A}) = |\alpha|\rho(\mathbf{A}), \quad \rho(\mathbf{A}^k) = [\rho(\mathbf{A})]^k$$

Let $p(t)$ be a given polynomial of degree k, ie.

$$p(t) = \sum_{i=0}^{k} a_i t^i$$

We can also define matrix polynomial in an analogous way:

Definition 4.4: Matrix polynomial

Let $\mathbf{A} \in \mathbb{C}^{n\times n}$. Then a matrix polynomial of degree k is defined as:

$$p(\mathbf{A}) = \sum_{i=0}^{k} a_i \mathbf{A}^i$$

for $a_i \in \mathbb{C}$, $i = 1, 2, \ldots, k$.

Fact 4.6

The polynomial is monic if $a_k = 1$. Polynomial factorisation also carries over to matrices:

$$p(\mathbf{A}) = \prod_{i=1}^{k} (\mathbf{A} - \beta_i \mathbf{I}), \beta_i \in \mathbb{C}, i = 1, \ldots, k$$

Theorem 4.1: Spectral mapping theorem for matrix polynomials

- If $(\lambda, \mathbf{v})$ is an eigenpair $\mathbf{A} \in \mathbb{C}^{n \times n}$, then $(p(\lambda), \mathbf{v})$ is an eigenpair of $p(\mathbf{A})$.
- Conversely, if $k \geq 1$ and μ is an eigenvalue of $p(\mathbf{A})$, then there is some eigenvalue of λ of $\mathbf{A}$ such that $\mu = p(\lambda)$.

Proof. Note that $\mathbf{A}^i \mathbf{v} = \mathbf{A}^{i-1} \mathbf{A} \mathbf{v} = \lambda \mathbf{A}^{i-1} \mathbf{v} = \ldots = \lambda^i \mathbf{v}$. So:

$$p(\mathbf{A})\mathbf{v} = \sum_{i=0}^{k} a_i \lambda^i \mathbf{v} = p(\lambda)\mathbf{v}$$

Let's define $q(t) = p(t) - \mu$. Since $k \geq 1$, $q(\mathbf{A}) = p(\mathbf{A}) - \mu \mathbf{I}$ has degree k, so it can be factorised as:

$$p(\mathbf{A}) - \mu \mathbf{I} = q(\mathbf{A}) = \prod_{i=1}^{k} (\mathbf{A} - \beta_i \mathbf{I})$$

$p(\mathbf{A}) - \mu \mathbf{I}$ is singular so some factor $(\mathbf{A} - \beta_j \mathbf{I})$ must be singular, which means that β_j is an eigenvalue of $\mathbf{A}$. But:

$$0 = q(\beta_j) = p(\beta_j) - \mu \implies \mu = p(\beta_j)$$

□

Exercise 4.1

Suppose that $\mathbf{A} \in \mathbb{C}^{n \times n}$. If $\mathrm{spec}(\mathbf{A}) = \{-1, 1\}$, what is $\mathrm{spec}(\mathbf{A}^2)$? Hint: Use the first part of the previous theorem to identify a point in $\mathrm{spec}(\mathbf{A}^2)$, and then use the second part to ascertain if it is the only point in $\mathrm{spec}(\mathbf{A}^2)$.

Theorem 4.2

$\mathbf{A}$ is singular if and only if $0 \in \mathrm{spec}(\mathbf{A})$.

Proof. A lot of if and only ifs:

$$\begin{aligned} \mathbf{A} \text{ is singular} &\iff \exists \mathbf{x} \neq \mathbf{0} \text{ s.t. } \mathbf{A}\mathbf{x} = \mathbf{0} \\ &\iff \exists \mathbf{x} \neq \mathbf{0} \text{ s.t. } \mathbf{A}\mathbf{x} = 0\mathbf{x} \\ &\iff 0 \in \mathrm{spec}(\mathbf{A}) \end{aligned}$$

□

Theorem 4.3

Let $\mathbf{A} \in \mathbb{C}^{n\times n}$ and $\lambda, \mu \in \mathbb{C}$. Then:

$$\lambda \in \text{spec}(\mathbf{A}) \iff \lambda + \mu \in \text{spec}(\mathbf{A} + \mu\mathbf{I})$$

Proof. More iffs:

$$\begin{aligned} \lambda \in \text{spec}(\mathbf{A}) &\iff \exists \mathbf{v} \neq \mathbf{0} \text{ s.t. } \mathbf{A}\mathbf{v} = \lambda\mathbf{v} \\ &\iff \mathbf{A}\mathbf{v} + \mu\mathbf{v} = \lambda\mathbf{v} + \mu\mathbf{v} \\ &\iff (\mathbf{A} + \mu\mathbf{I})\mathbf{v} = (\lambda + \mu)\mathbf{v} \\ &\iff \lambda + \mu \in \text{spec}(\mathbf{A} + \mu\mathbf{I}) \end{aligned}$$

□

Theorem 4.4

If $\mathbf{A} \in \mathbb{C}^{n\times n}$ is Hermitian, then all its eigenvalues are real.

Proof. Suppose $\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$, $v \neq \mathbf{0}$. We have:

$$\lambda\mathbf{v}^*\mathbf{v} = \mathbf{v}^*\mathbf{A}\mathbf{v} = \mathbf{v}^*\mathbf{A}^*\mathbf{v}$$

On the other hand:

$$\lambda\mathbf{v}^*\mathbf{v} = \mathbf{v}^*\mathbf{A}\mathbf{v} \iff (\lambda\mathbf{v}^*\mathbf{v})^* = (\mathbf{v}^*\mathbf{A}\mathbf{v})^* \iff \bar{\lambda}\mathbf{v}^*\mathbf{v} = \mathbf{v}^*\mathbf{A}^*\mathbf{v}$$

So $\lambda = \bar{\lambda}$, which means that $\lambda \in \mathbb{R}$. □

Exercise 4.2

Show that skew-Hermitian matrices have pure imaginary eigenvalues.

Theorem 4.5

If $\mathbf{A} \in \mathbb{C}^{n\times n}$ is Hermitian, then eigenvectors corresponding to distinct eigenvalues are mutually orthogonal.

Proof. Suppose we have two vectors $\mathbf{v}, \mathbf{w}$ such that:

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v}, \mathbf{v} \neq \mathbf{0}$$
$$\mathbf{A}\mathbf{w} = \mu\mathbf{w}, \mathbf{w} \neq \mathbf{0}$$

with $\lambda \neq \mu$ (unique eigenpairs). We have:

$$\begin{aligned} \lambda\langle\mathbf{v}, \mathbf{w}\rangle = \langle\lambda\mathbf{v}, \mathbf{w}\rangle = \langle\mathbf{A}\mathbf{v}, \mathbf{w}\rangle &= \langle\mathbf{v}, \mathbf{A}^*\mathbf{w}\rangle \\ &= \langle\mathbf{v}, \mathbf{A}\mathbf{w}\rangle = \langle\mathbf{v}, \mu\mathbf{w}\rangle = \mu\langle\mathbf{v}, \mathbf{w}\rangle \end{aligned}$$

So $(\lambda - \mu)\langle\mathbf{v}, \mathbf{w}\rangle = 0$, which since $\mu \neq \lambda$, we get $\langle\mathbf{v}, \mathbf{w}\rangle = 0$. Recall that an inner product of 0 is equivalent to orthogonality. □

For certain types of matrices, computing eigenvalues is immediate, for example a block triangular matrix:

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_{11} & \dots & \mathbf{A}_{1k} \\ 0 & \ddots & \vdots \\ 0 & 0 & \mathbf{A}_{kk} \end{pmatrix}$$

Its polynomial and spectrum are given by:

$$p_{\mathbf{A}}(\lambda) = \prod_{i=1}^{k} p_{\mathbf{A}_{ii}}(\lambda) \implies \operatorname{spec}(\mathbf{A}) = \bigcup_{i=1}^{k} \operatorname{spec}(\mathbf{A}_{ii})$$

Example 4.1

Compute the eigenpair of:

$$\mathbf{A} = \begin{pmatrix} 2 & 1 & 2 \\ 0 & 1 & -1 \\ 0 & 0 & 2 \end{pmatrix}$$

The eigenvalues are 2 and 1.

$$\mathbf{A} - 2\mathbf{I} = \begin{pmatrix} 0 & 1 & 2 \\ 0 & -1 & -2 \\ 0 & 0 & 0 \end{pmatrix} \to \begin{cases} (\mathbf{A} - 2\mathbf{I}) \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \mathbf{0} \\ (\mathbf{A} - 2\mathbf{I}) \begin{pmatrix} 0 \\ -2 \\ 1 \end{pmatrix} = \mathbf{0} \end{cases}$$

Note that $\begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^\top$ and $\begin{pmatrix} 0 & -2 & 1 \end{pmatrix}^\top$ are two eigenvectors of **A** associated with eigenvalue 2 which are also linearly independent.

Example 4.2

Consider the matrix:

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

Its eigenvalues are 2 and 1.

$$\mathbf{B} - 2\mathbf{I} = \begin{pmatrix} 0 & 1 & 1 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{pmatrix} \to (\mathbf{A} - 2\mathbf{I}) \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \mathbf{0}$$

Note that $\begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^\top$ is the only eigenvector of **A** for eigenvalue 2.

Remark 4.2

Not only can an eigenvalue occur as multiple roots of the characteristic polynomial, but also any given eigenvalue might correspond to multiple eigenvectors.

Definition 4.5: Eigenspace

The eigenspace associated with an eigenvalue λ is the subspace defined as:

$$\begin{aligned}\mathcal{E}_\lambda(\mathbf{A}) &= \mathrm{Null}(\mathbf{A} - \lambda\mathbf{I}) \\ &= \{\mathbf{v} \mid (\mathbf{A} - \lambda\mathbf{I})\mathbf{v} = 0\} \\ &= \{\text{all eigenvectors of } \mathbf{A} \text{ associated with } \lambda\} \cup \{\mathbf{0}\}\end{aligned}$$

Note 4.3

Because the eigenspace is a subspace, one can find an orthonormal basis for this space (the basis vectors are themselves eigenvectors of **A**).

Definition 4.6: Algebraic multiplicity

The algebraic multiplicity of λ is the multiplicity of λ as a root of the characteristic polynomial.

Definition 4.7: Geometric multiplicity

The geometric multiplicty of λ is the dimension of the associated eigenspace:

$$\dim(\mathcal{E}_\lambda(\mathbf{A})) = \dim(\mathrm{Null}(\mathbf{A} - \lambda\mathbf{I}))$$

Definition 4.8: Simple eigenvalue

The eigenvalue λ of **A** is said to be simple if its algebraic multiplicity is 1.

Theorem 4.6

Let $m(\lambda)$ be the algebraic multiplicity of λ. Then there are bounds on the geometric multiplicity:

$$1 \leq \dim(\mathcal{E}_\lambda(\mathbf{A})) \leq m(\lambda)$$

Definition 4.9: Defective matrix

A matrix is defective if it has an eigenvalue λ for which:

$$\dim(\mathcal{E}_\lambda(\mathbf{A})) < m(\lambda)$$

4.2 Similarity transformation

Since diagonal/triangular matrices are nice, is there a way to "transform" all matrices to look diagonal/triangular without affecting their spectrum?

Definition 4.10: Similarity transformation

Let $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n\times n}$. We say that $\mathbf{B}$ is similar to $\mathbf{A}$ if there exists a non-singular matrix $\mathbf{S} \in \mathbb{C}^{n\times n}$ such that:

$$\mathbf{B} = \mathbf{S}^{-1}\mathbf{A}\mathbf{S}$$

Fact 4.7

Two similar matrices share the same spectrum and the same characteristic polynomial.

Theorem 4.7

If $\mathbf{A}$ and $\mathbf{B}$ are similar, then they have the same characteristic polynomial.

Proof.

$$\begin{aligned} p_{\mathbf{B}}(\lambda) &= \det(\mathbf{B} - \lambda\mathbf{I}) \\ &= \det(\mathbf{S}^{-1}\mathbf{A}\mathbf{S} - \lambda\mathbf{S}^{-1}\mathbf{S}) \\ &= \det(\mathbf{S}^{-1}(\mathbf{A} - \lambda\mathbf{I})\mathbf{S}) \\ &= \det(\mathbf{S}^{-1}\det(\mathbf{A} - \lambda\mathbf{I})\det(\mathbf{S})) \\ &= \det(\mathbf{A} - \lambda\mathbf{I}) = p_{\mathbf{A}}(\lambda) \end{aligned}$$

□

Theorem 4.8

$(\lambda, \mathbf{v})$ is an eigenpair of $\mathbf{A}$ if and only if $(\lambda, \mathbf{S}^{-1}\mathbf{v})$ is an eigenpair for $\mathbf{B} = \mathbf{S}^{-1}\mathbf{A}\mathbf{S}$.

Proof.

$$\begin{aligned} \mathbf{A}\mathbf{v} = \lambda\mathbf{v} &\iff \mathbf{A}\mathbf{S}\mathbf{S}^{-1}\mathbf{v} = \lambda\mathbf{v} \\ &\iff \mathbf{S}^{-1}\mathbf{A}\mathbf{S}\mathbf{S}^{-1}\mathbf{v} = \lambda\mathbf{S}^{-1}\mathbf{v} \\ &\iff \mathbf{B}\mathbf{S}^{-1}\mathbf{v} = \lambda\mathbf{S}^{-1}\mathbf{v} \\ &\iff \mathbf{B}\mathbf{w} = \lambda\mathbf{w} \end{aligned}$$

where we define $\mathbf{w} = \mathbf{S}^{-1}\mathbf{v}$. □

Since diagonal matrices are the simplest, what matrices are similar to diagonal matrices?

Definition 4.11: Diagonalisable matrix

If $\mathbf{A} \in \mathbb{C}^{n\times n}$ is similar to a diagonal matrix (pre and post multiplied), then $\mathbf{A}$ is said to be diagonalisable.

5 Diagonalisation

5.1 Eigendecomposition

Theorem 5.1

The matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ is diagonalisable if and only if it has n linearly independent eigenvectors. In other words, $\mathbf{A}$ is diagonalisable if and only if it is not defective, i.e.:

$$\dim(\mathcal{E}_\lambda(\mathbf{A})) = m(\lambda), \quad \forall \lambda \in \operatorname{spec}(\mathbf{A})$$

A simple criterion: if all eigenvalues of $\mathbf{A}$ are simple, then $\mathbf{A}$ is diagonalisable.

Theorem 5.2: Eigendecomposition

Let $\mathbf{A}$ be diagonalisable and define $\mathbf{V} = \begin{pmatrix} \mathbf{v}_1 & \dots & \mathbf{v}_n \end{pmatrix} \in \mathbb{C}^{n \times n}$ to be the set of linearly independent eigenvectors of $\mathbf{A}$. Then:

$$\mathbf{V}^{-1}\mathbf{A}\mathbf{V} = \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix} \triangleq \mathbf{\Lambda}$$

Here, λ_i are eigenvalues of $\mathbf{A}$. The matrix $\mathbf{V}$ is not necessarily unique, e.g. $\alpha\mathbf{V}$ for any $\alpha \neq 0$ works too.

Note 5.1

As a rule of thumb, almost every matrix in $\mathbb{C}^{n \times n}$ is diagonalisable. The set of matrices $\mathbb{C}^{n \times n}$ that are not diagonalisable has Lebesgue measure zero.

Note 5.2

Even though not every matrix is diagonalisable, it turns out that:

- every matrix $\mathbf{A}$ is unitarily similar to an **upper triangular matrix** by Schur decomposition;
- every matrix $\mathbf{A}$ is similar to a **block diagonal matrix** by Jordan canonical form.

5.2 Schur decomposition/triangularisation

Theorem 5.3: Schur decomposition/triangularisation

For any $\mathbf{A} \in \mathbb{C}^{n\times n}$, there exists unitary $\mathbf{U} \in \mathbb{C}^{n\times n}$ such that:

$$\mathbf{U}^*\mathbf{A}\mathbf{U} = \begin{pmatrix} \lambda_1 & b_{12} & \dots & b_{1n} \\ & \lambda_2 & b_{23} & b_{2n} \\ & & \ddots & \vdots \\ & & & \lambda_n \end{pmatrix}$$

Here, λ_i are the eigenvalues of **A**. The matrices **U** and **T** are not necessarily unique, for example, if $\dim(\mathcal{E}_\lambda(\mathbf{A})) > 1$, then any orthonormal basis for this space will work. If **A** and its eigenvalues are real, then **U** can be chosen to be real orthogonal.

Example 5.1

Show that $\mathrm{Trace}(\mathbf{A}) = \sum_{i=1}^n \lambda_i$. Let **U** be as in Schur Triangularisation Theorem. Then:

$$\begin{aligned} \mathrm{Trace}(\mathbf{A}) &= \mathrm{Trace}(\mathbf{UTU}^*) \\ &= \mathrm{Trace}(\mathbf{U}^*\mathbf{UT}) \\ &= \mathrm{Trace}(\mathbf{T}) = \sum_{i=1}^n \lambda_i \end{aligned}$$

Example 5.2

Prove that $\det(\mathbf{A}) = \prod_{i=1}^n \lambda_i$. Exercise. This in particular implies that **A** is non-singular $\iff$ $0 \notin \mathrm{spec}(\mathbf{A})$, i.e. all its eigenvalues are non-zero.

5.3 Jordan blocks and canonical form

Definition 5.1: Jordan block

A **Jordan block** $\mathbf{J}_k(\lambda)$ is a $k \times k$ upper triangular matrix of the form:

$$\mathbf{J}_k(\lambda) = \begin{pmatrix} \lambda & 1 & & & \\ & \lambda & 1 & & \\ & & \ddots & \ddots & \\ & & & \lambda & 1 \\ & & & & \lambda \end{pmatrix}$$

In particular, $\mathbf{J}_1(\lambda) = (\lambda)$ and $\mathbf{J}_2(\lambda) = \begin{pmatrix} \lambda & 1 \\ 0 & \lambda \end{pmatrix}$.

Theorem 5.4: Jordan canonical form

For any $\mathbf{A} \in \mathbb{C}^{n\times n}$, there is a non-singular $\mathbf{S} \in \mathbb{C}^{n\times n}$, positive integers $k, n_1, n_2, \dots, n_k$ with

$n_1 + n_2 + \ldots + n_k = n$ and scalars $\lambda_1, \ldots, \lambda_k \in \mathbb{C}$ such that:

$$\mathbf{A} = \mathbf{S} \overbrace{\begin{pmatrix} \mathbf{J}_{n_1}(\lambda_1) & & \\ & \ddots & \\ & & \mathbf{J}_{n_k}(\lambda_k) \end{pmatrix}}^{\mathbf{J_A}} \mathbf{S}^{-1}$$

Fact 5.1: Facts about Jordan

- The Jordan matrix $\mathbf{J_A}$ is uniquely determined up to permutation of Jordan blocks;
- If $\mathbf{A}$ is real and has only real eigenvalues, then $\mathbf{S}$ can be chosen to be real;
- The number of Jordan blocks, k, is the maximum number of linearly independent eigenvectors of $\mathbf{A}$;
- Given an eigenvalue λ, its geometric multiplicity is the number of its corresponding Jordan blocks;
- The sum of the sizes of all Jordan blocks corresponding to an eigenvalue λ is its algebraic multiplicity;
- If an eigenvalue is defective, the size of at least one of its corresponding Joradn blocks is greater than one, so a matrix is diagonalisable if and only if all its Jordan blocks are 1×1.

Example 5.3

$$\mathbf{J_A} = \begin{pmatrix} 4 & 1 & & & & & & & \\ & 4 & 1 & & & & & & \\ & & 4 & & & & & & \\ & & & 4 & 1 & & & & \\ & & & & 4 & & & & \\ & & & & & 3 & 1 & & \\ & & & & & & 3 & & \\ & & & & & & & 2 & \\ & & & & & & & & 2 \end{pmatrix}$$

$\mathbf{A}$ has three distinct eigenvalues, namely:

$$\text{spec}(\mathbf{A}) = \{4, 3, 2\}$$

- Its algebraic multiplicities are $(4) = 5, (3) = 2, (2) = 2$;
- Its geometric multiplicites are $(4) = 2, (3) = 1, (2) = 2$.

Example 5.4

Find all possible Jordan canonical forms for a 3×3 matrix whose eigenvalues are $\{-2, 3, 3\}$. Each Jordan block $\mathbf{J}_d$ contributes d eigenvalues to the matrix. So for -2, we can only have one 1×1 Jordann block. For 3, however, we can have two 1×1 or one 2×2. Hence, possible choices are:

$$\begin{pmatrix} -2 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 3 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} -2 & 0 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 3 \end{pmatrix}$$

Note 5.3

We saw that a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ is diagonalisable if and only if it posses a complete linearly independent set of eigenvectors, i.e. the matrix $\mathbf{V} = \begin{pmatrix} \mathbf{v}_1 & \dots & \mathbf{v}_n \end{pmatrix}$ is invertible, in which case we have $\mathbf{V}^{-1}\mathbf{A}\mathbf{V} = \mathbf{\Lambda}$ is a diagonal matrix with eigenvalues of $\mathbf{A}$ on the diagonal.

Fact 5.2

For diagonalisable matrices, we have:

$$\text{Jordan canonical form} \quad \equiv \quad \text{eigendecomposition}$$

Now recall Schur decomposition as $\mathbf{U}^*\mathbf{A}\mathbf{U} = \mathbf{T}$ with eigenvalues of $\mathbf{A}$ on the diagonal of $\mathbf{T}$. When do we have eigendecomposition $\equiv$ Schur decomposition?

Definition 5.2: Unitarily diagonalisable matrix

We say that $\mathbf{A}$ is unitarily diagonalisable if it is unitarily similar to a diagonal matrix.

Remark 5.1

$\mathbf{A} \in \mathbb{C}^{n \times n}$ is normal if and only if $\mathbf{A}\mathbf{A}^* = \mathbf{A}^*\mathbf{A}$.

Theorem 5.5

A matrix is unitarily diagonalisable if and only if it is normal.

Proof. ($\Longrightarrow$) Suppose $\mathbf{A}$ is unitarily diagonalisable, that is $\mathbf{U}^*\mathbf{A}\mathbf{U} = \mathbf{\Lambda}$. So we have $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^*$, hence:

$$\begin{aligned} \mathbf{A}\mathbf{A}^* &= \mathbf{U}\mathbf{\Lambda}\mathbf{U}^*\mathbf{U}\mathbf{\Lambda}^*\mathbf{U}^* \\ &= \mathbf{U}\mathbf{\Lambda}\mathbf{\Lambda}^*\mathbf{U}^* \\ &= \mathbf{U}\mathbf{\Lambda}^*\mathbf{\Lambda}\mathbf{U}^* \\ &= \mathbf{U}\mathbf{\Lambda}^*\mathbf{U}^*\mathbf{U}\mathbf{\Lambda}\mathbf{U}^* \\ &= \mathbf{A}^*\mathbf{A} \end{aligned}$$

($\Longleftarrow$) Conversely, suppose $\mathbf{A}$ is normal and consider its Schur decomposition, ie $\mathbf{U}^*\mathbf{A}\mathbf{U} = \mathbf{T}$. We have:

$$\begin{aligned} \mathbf{T}^*\mathbf{T} &= \mathbf{U}^*\mathbf{A}^*\mathbf{U}\mathbf{U}^*\mathbf{A}\mathbf{U} \\ &= \mathbf{U}^*\mathbf{A}^*\mathbf{A}\mathbf{U} \end{aligned}$$

$$\begin{aligned} &= \mathbf{U}^*\mathbf{A}\mathbf{A}^*\mathbf{U} \\ &= \mathbf{U}^*\mathbf{A}\mathbf{U}\mathbf{U}^*\mathbf{A}^*\mathbf{U} \\ &= \mathbf{T}\mathbf{T}^* \end{aligned}$$

but since **T** is upper-triangular, it has to be diagonal. □

Exercise 5.1

A normal triangular matrix is diagonal.

Theorem 5.6

For normal matrices:

$$\text{Schur decomposition} \quad \equiv \quad \text{eigendecomposition}$$

or:

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^* = \sum_{i=1}^{n} \lambda_i \mathbf{u}_i \mathbf{u}_i^*$$

Fact 5.3: Some final facts on matrices

- Among complex matrices, all unitary, Hermitian and skew-Hermitian matrices are normal;
- Among real matrices, all orthogonal, symmetric and skew-symmetric matrices are normal;
- It is **not** the case that all normal matrices are either unitary or (skew-)Hermitian, e.g. $\forall a, b \in \mathbb{C}$, $\begin{bmatrix} a & b \\ -b & a \end{bmatrix}$ is normal and has $\lambda_i = a \pm ib$;
- A normal matrix is Hermitian $\iff$ all its eigenvalues are real

6 Singular value decomposition

6.1 Singular value decomposition

We saw when a square matrix is unitarily diagonalisable, i.e. it is similar to a diagonal matrix by unitary similarity transformation:

$$\mathbf{U}^*\mathbf{A}\mathbf{U} = \mathbf{\Lambda}$$

Question: if we relax “similarity” and allow for unitary matrices to be selected independently, can we still “diagonalise” a matrix? i.e. instead of unitary similarity, we seek unitary equivalence. The solution:

Any matrix of any size can be “diagonalised” by a suitable pre- and post-multiplication by unitary matrices.

Theorem 6.1: Singular value decomposition

Let $\mathbf{A} \in \mathbb{C}^{m\times n}, q = \min\{m, n\}$ and $\text{Rank}(\mathbf{A}) \triangleq r \leq q$. There exists two unitary matrices $\mathbf{U} \in \mathbb{C}^{m\times m}$ and $\mathbf{V} \in \mathbb{C}^{n\times n}$, and a square diagonal matrix:

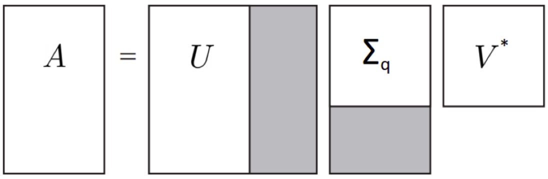
$$\mathbf{\Sigma}_q = \begin{pmatrix} \sigma_1 & & & & & & \\ & \sigma_2 & & & & & \\ & & \ddots & & & & \\ & & & \sigma_r & & & \\ & & & & 0 & & \\ & & & & & \ddots & \\ & & & & & & 0 \end{pmatrix} \in \mathbb{R}^{q\times q}$$

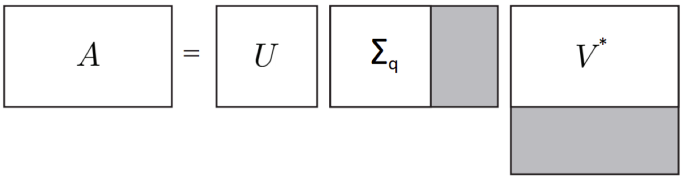
such that $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0$, and $\mathbf{U}^*\mathbf{A}\mathbf{V} = \mathbf{\Sigma} \in \mathbb{R}^{m\times n}$, where:

$$\mathbf{\Sigma} = \begin{cases} \mathbf{\Sigma}_m & m = n \\ \begin{pmatrix} \mathbf{\Sigma}_m & \mathbf{0}_{m\times(n-m)} \end{pmatrix} & m \leq n \\ \begin{pmatrix} \mathbf{\Sigma}_n \\ \mathbf{0}_{(m-n)\times n} \end{pmatrix} & m \geq n \end{cases}$$

The diagonal entries of $\mathbf{\Sigma}_q$ are called the **singular values** of $\mathbf{A}$. The columsn of $\mathbf{U}$ and $\mathbf{V}$ are called, respectively, the **left and right** singular vectors of $\mathbf{A}$:

$$\mathbf{A}\mathbf{v}_i = \sigma_i \mathbf{u}_i, \quad i = 1, \ldots, \min\{m, n\}$$





The unitary factors in SVD are never unique e.g. we may replace $\mathbf{U}$ with $-\mathbf{U}$ and $\mathbf{V}$ with $-\mathbf{V}$. The singular values of A are uniquely detemrined by the eigenvalues of $\mathbf{A}^*\mathbf{A}$, or equivalently by the eigenvalues of $\mathbf{A}\mathbf{A}^*$) The matrix $\mathbf{\Sigma}$ is determined up to permutation of its diagonal entries.

Note 6.1: Convention

To make $\mathbf{\Sigma}$ unique, one often requires that the signular values be arranged in non-increasing order (as in our definition here), but other choices are possible.

Note 6.2

Any real matrix has a singular value decomposition in which all three factors are real.

Now we have a nice theorem relating singular values and eigenvalues:

Theorem 6.2

For any $\mathbf{A} \in \mathbb{C}^{m\times n}$, we have:

$$\sigma_i = \sqrt{\lambda_i(\mathbf{A}^*\mathbf{A})} = \sqrt{\lambda_i(\mathbf{A}\mathbf{A}^*)}, \quad i = 1, 2, \dots, \mathrm{Rank}(\mathbf{A})$$

Proof. Assume without loss of generality that $m \geq n$. let $\mathbf{A} = \mathbf{U\Sigma V}^*$. We have:

$$\begin{aligned} \mathbf{A}^*\mathbf{A} &= \mathbf{V\Sigma}^\top\mathbf{U}^*\mathbf{U\Sigma V}^* \\ &= \mathbf{V\Sigma}^\top\mathbf{\Sigma V}^* \\ &= \mathbf{V\Sigma}_n^2\mathbf{V}^* \\ &\implies \dots \end{aligned}$$

Similarly:

$$\begin{aligned} \mathbf{AA}^* &= \mathbf{U\Sigma V}^*\mathbf{V\Sigma}^\top\mathbf{U}^* \\ &= \mathbf{U\Sigma\Sigma}^*\mathbf{U}^* \\ &= \mathbf{U}\begin{pmatrix} \mathbf{\Sigma}_n^2 & \mathbf{0}_{n\times(m-n)} \\ \mathbf{0}_{(m-n)\times n} & \mathbf{0}_{(m-n)\times(m-n)} \end{pmatrix}\mathbf{U}^* \\ &\implies \dots \end{aligned}$$

□

Definition 6.1: Compact SVD

Discard zero entries on $\mathbf{\Sigma}$ to get $\mathbf{A} = \mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^*$, where $\mathbf{U}_r \in \mathbb{C}^{m\times r}, \mathbf{\Sigma}_r \in \mathbb{C}^{r\times r}, \mathbf{V}_r \in \mathbb{C}^{m\times r}$ as:

$$\mathbf{A} = \underbrace{\begin{pmatrix} \mathbf{u}_1 & \dots & \mathbf{u}_r \end{pmatrix}}_{\mathbf{U}_r} \underbrace{\mathrm{diag}(\sigma_1, \dots, \sigma_r)}_{\mathbf{\Sigma}_r} \underbrace{\begin{pmatrix} \mathbf{v}_1 & \dots & \mathbf{v}_r \end{pmatrix}}_{\mathbf{V}_r^*} = \sum_{i=1}^r \sigma_i\mathbf{u}_i\mathbf{v}_i^*$$

Theorem 6.3

Let $\mathbf{A} = \mathbf{U\Sigma V}^*$ be an SVD of $\mathbf{A} \in \mathbb{C}^{m\times n}$ and assume that for some r, we have $\sigma_r \neq 0$ and $\sigma_{r+1} = 0$. (Since singular values are conventionally ordered, this implies all singular values past this point are zero).

Then we have the following:

- $\mathrm{Rank}(\mathbf{A}) = r$

- $\text{Null}(\mathbf{A}) = \text{Span}\{\mathbf{v}_{r+1}, \mathbf{v}_{r+2}, \dots, \mathbf{v}_n\}$
- $\text{Range}(\mathbf{A}) = \text{Span}\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_r\}$.
- $\mathbf{A}^\dagger = \mathbf{V}\mathbf{\Sigma}^\dagger\mathbf{U}^* = \mathbf{V}_r\mathbf{\Sigma}_r^{-1}\mathbf{U}_r^*$

$$= \underbrace{[\mathbf{u}_1 \quad \mathbf{u}_2 \quad \dots \quad \mathbf{u}_r}_{\text{Range}(\mathbf{A})} \quad \underbrace{\mathbf{u}_{r+1} \quad \dots \quad \mathbf{u}_m]}_{\text{Null}(\mathbf{A}^*)} \begin{bmatrix} \sigma_1 & & & & & & \\ & \sigma_2 & & & & & \\ & & \ddots & & & & \\ & & & \sigma_r & & & \\ & & & & 0 & & \\ & & & & & \ddots & \\ & & & & & & 0 \\ & & & \mathbf{0} & & & \\ & & & & & & \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^* \\ \mathbf{v}_2^* \\ \vdots \\ \mathbf{v}_r^* \\ \mathbf{v}_{r+1}^* \\ \vdots \\ \mathbf{v}_n^* \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ a \\ a \\ a \end{matrix}}\right\} \text{Range}(\mathbf{A}^*) \\ \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} \text{Null}(\mathbf{A}) \end{matrix}$$

Theorem 6.4

Let $\mathbf{A} \in \mathbb{C}^{n \times n}$ be a normal matrix whose (not necessarily distinct) eigenvalues are $\lambda_1, \dots, \lambda_n$. Show that the singular values of $\mathbf{A}$ are $|\lambda_1|, |\lambda_2|, \dots, |\lambda_n|$.

Proof. Since $\mathbf{A}$ is a normal matrix, $\implies$ unitarily diagonalisable as $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^*$, where $\mathbf{\Lambda}$ is the diagonal matrix of eigenvalues of $\mathbf{A}$. Now, we have:

$$\begin{aligned} \sigma_i &= \sqrt{\lambda_i(\mathbf{A}^*\mathbf{A})} \\ &= \sqrt{\bar{\lambda}_i(\mathbf{A})\lambda_i(\mathbf{A})} \\ &= \sqrt{|\lambda_i(\mathbf{A})|^2} = |\lambda_i(\mathbf{A})| \end{aligned}$$

□

6.2 Schatten norm and matrix low-rank approximations

Note 6.3

There is also a third notion based on singular values, and is referred to as the **Schatten** norm.

Definition 6.2: Schatten norm

Schatten p-norm of a matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$ is defined by applying vector p-norm to the vector of singular values, i.e.:

$$\|\mathbf{A}\|_p = \left(\sum_{i=1}^{\min m,n} \sigma_i^p\right)^{1/p}$$

Note 6.4: Properties of the Schatten norm

- All Schatten norms are sub-multiplicative
- Unitarily invariant
- Famous norms:
  - $p = 1$: nuclear or trace norm;
  - $p = 2$: Frobenius norm;
  - $p = \infty$: spectral or ℓ_2 induced norm.

Recall that a matrix $\mathbf{A} \in \mathbb{C}^{m\times n}$ with $\mathrm{Rank}(\mathbf{A})$ can be written as a sum of rank-one matrices as:

$$\mathbf{A} = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^*$$

So for $1 \leq k \leq r$, the partial sum:

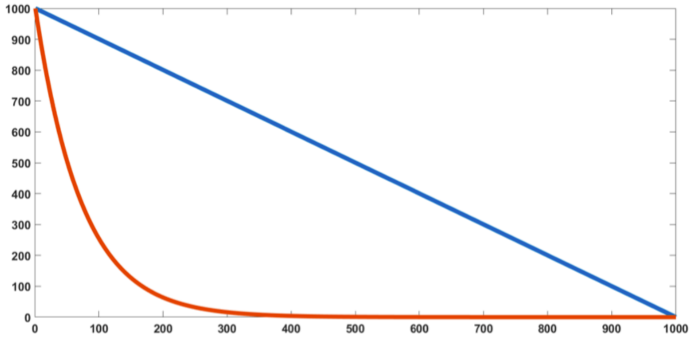
$$\mathbf{A}_k = \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^*$$

captures some "energy" of $\mathbf{A}$. We can make this notion precise by introducing the concept of *matrix low-rank approximations*.

Theorem 6.5: Matrix low-rank approximation: spectral norm

$$\|\mathbf{A} - \mathbf{A}_k\|_2 = \min_{\substack{A\in\mathbb{C}^{m\times n} \\ \mathrm{Rank}(\mathbf{B})\leq k}} \|\mathbf{A} - \mathbf{B}\|_2 = \sigma_{k+1}$$

where $\|\cdot\|_2$ is the matrix spectral norm and $\sigma_{k+1} = 0$ for $k = \min\{m, n\}$.



Here is some intuition behind low rank approximation.

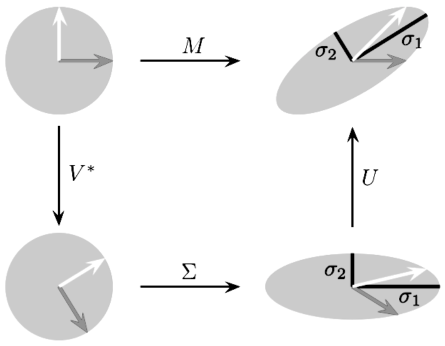
- What is the best approximation of an n-dimensional hyper-ellipsoid by a line segment? **Take the line segment to be the longest axis.**
- What is the best approximation by a two-dimensional ellipsoid? **Take the ellipsoid spanned byhte longest and second-longest axis.**
- At each step, include the largest axis of the hyper-ellipsoid not yet included. After r steps, we have captured all of $\mathbf{A}$.

Theorem 6.6: Matrix low-rank approximation: Frobenius norm

$$\|\mathbf{A}-\mathbf{A}_k\|_{\mathsf{F}} = \min_{\substack{\mathbf{A}\in\mathbb{C}^{m\times n}\\ \mathrm{Rank}(\mathbf{B})\le k}} \|\mathbf{A}-\mathbf{B}\|_{\mathsf{F}} = \sqrt{\sum_{i=k+1}^{r}\sigma_i}$$

where $\|\cdot\|_{\mathsf{F}}$ is the matrix Frobenius norm.

6.3 A more geometric view



$$M = U\cdot\Sigma\cdot V^*$$

Consider a linear mapping $\mathbf{f}:\mathbb{R}^n\to\mathbb{R}^m$ given by $\mathbf{f}(\mathbf{x})=\mathbf{A}\mathbf{x}$ where $\mathbf{A}\in\mathbb{R}^{m\times n}$. Assume $m\ge n$, $\mathrm{Rank}(\mathbf{A})=n$. Let $\mathbf{A}=\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$ be the economy SVD of $\mathbf{A}$, i.e. $\mathbf{U}\in\mathbb{R}^{m\times n}$, $\boldsymbol{\Sigma},\mathbf{V}\in\mathbb{R}^{n\times n}$. Consider the unit sphere $\mathcal{S}=\{\mathbf{z}\in\mathbb{R}^n \mid \|\mathbf{z}\|=1\}$. We can show that the image of $\mathcal{S}$ under $\mathbf{f}$, i.e. $\mathcal{E}=\{\mathbf{A}\mathbf{z}\in\mathbb{R}^m \mid \mathbf{z}\in\mathcal{S}\}$ is an ellipsoid.

Theorem 6.7

$\mathcal{E}=\mathbf{U}\mathcal{E}_0$, where $\mathcal{E}_0=\{\mathbf{y}\in\mathbb{R}^n \mid \sum_{i=1}^n \frac{y_i^2}{\sigma_i^2}=1\}$.

Proof. Suppose $\mathbf{z}\in\mathcal{S}$. We have $\mathbf{A}\mathbf{z}=\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top\mathbf{z}=\mathbf{U}\mathbf{y}$ where $\mathbf{y}=\boldsymbol{\Sigma}\mathbf{V}^\top\mathbf{z}$. We just need to show that $\mathbf{y}\in\mathcal{E}_0$. We have $\mathbf{z}=\mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{y}$, which implies that:

$$\begin{aligned}1 &= \|\mathbf{z}\|^2\\ &= \|\mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{y}\|^2\\ &= \|\boldsymbol{\Sigma}^{-1}\mathbf{y}\|^2\end{aligned}$$

$$= \sum_{i=1}^{n} \frac{y_i^2}{\sigma_i^2}$$

This implies that $\mathbf{y} \in \mathcal{E}_0$. □

Recall that for a given matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$, we saw that the condition number allows us to measure the sensitivity of problems with respect to perturbations. When the Euclidean norm is used, condition number can also be equivalently written (assuming $\mathbf{A}$ has full column rank) as:

$$\kappa(\mathbf{A}) = \|\mathbf{A}\|\|\mathbf{A}^\dagger\| = \frac{\sigma_1}{\sigma_n}$$

This way, we also obtain a nice geometric interpretation of $\kappa(\mathbf{A})$ as the amount of distortion, caused by $\mathbf{A}$, to a unit sphere i.e. it measures how badly $\mathbf{A}$ distorts the geometry of the space.

7 Special matrix types

7.1 Diagonal matrices

Definition 7.1: Diagonal matrix

A diagonal matrix $\mathbf{D}$ is of the form:

$$\mathbf{D} = \begin{pmatrix} d_{11} & & & \\ & d_{22} & & \\ & & \ddots & \\ & & & d_{nn} \end{pmatrix}$$

Fact 7.1: Properties of a diagonal matrix

- $\text{spec}(\mathbf{D}) = \{d_{11}, d_{22}, \dots, d_{nn}\}$;
- $\det(\mathbf{D}) = \prod_{i=1}^{n} d_{ii}$
- $\mathbf{D}$ is non-singular $\iff d_{ii} \neq 0, \forall i$

Note 7.1: Multiplication with diagonal matrices

Diagonal matrices multiplied with regular matrices give nice properties:

$$\begin{pmatrix} d_{11} & & & \\ & d_{22} & & \\ & & \ddots & \\ & & & d_{nn} \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} = \begin{pmatrix} d_{11}a_{11} & d_{11}a_{12} & \cdots & d_{11}a_{1n} \\ d_{22}a_{21} & d_{22}a_{22} & \cdots & d_{22}a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ d_{nn}a_{n1} & d_{nn}a_{n2} & \cdots & d_{nn}a_{nn} \end{pmatrix}$$

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} d_{11} & & & \\ & d_{22} & & \\ & & \ddots & \\ & & & d_{nn} \end{pmatrix} = \begin{pmatrix} d_{11}a_{11} & d_{22}a_{12} & \cdots & d_{nn}a_{1n} \\ d_{22}a_{21} & d_{22}a_{22} & \cdots & d_{nn}a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ d_{11}a_{n1} & d_{22}a_{n2} & \cdots & d_{nn}a_{nn} \end{pmatrix}$$

7.2 Block diagonal matrices

Definition 7.2: Block diagonal matrices

A block diagonal matrix **D** consists of submatrices like the following:

$$\mathbf{D} = \begin{pmatrix} \mathbf{D}_{11} & & & \\ & \mathbf{D}_{22} & & \\ & & \ddots & \\ & & & \mathbf{D}_{kk} \end{pmatrix}$$

Fact 7.2: Properties of block-diagonal matrices

- $\text{spec}(\mathbf{D}) = \bigcup_{i=1}^{k} \text{spec}(\mathbf{D}_{ii})$
- $\det(\mathbf{D}) = \prod_{i=1}^{k} \det(\mathbf{D}_{ii})$
- **D** is non-singular $\iff$ $\mathbf{D}_{ii}$ is nonsingular

Note 7.2: Multiplication with block diagonal matrices

Block diagonal matrices multiplied with regular block matrices give nice properties:

$$\begin{pmatrix} \mathbf{D}_{11} & & & \\ & \mathbf{D}_{22} & & \\ & & \ddots & \\ & & & \mathbf{D}_{kk} \end{pmatrix} \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \cdots & \mathbf{A}_{1k} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \cdots & \mathbf{A}_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{A}_{k1} & \mathbf{A}_{k2} & \cdots & \mathbf{A}_{kk} \end{pmatrix} = \begin{pmatrix} \mathbf{D}_{11}\mathbf{A}_{11} & \mathbf{D}_{11}\mathbf{A}_{12} & \cdots & \mathbf{D}_{11}\mathbf{A}_{1k} \\ \mathbf{D}_{22}\mathbf{A}_{21} & \mathbf{D}_{22}\mathbf{A}_{22} & \cdots & \mathbf{D}_{22}\mathbf{A}_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{D}_{kk}\mathbf{A}_{k1} & \mathbf{D}_{kk}\mathbf{A}_{k2} & \cdots & \mathbf{D}_{kk}\mathbf{A}_{kk} \end{pmatrix}$$

$$\begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \cdots & \mathbf{A}_{1k} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \cdots & \mathbf{A}_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{A}_{k1} & \mathbf{A}_{k2} & \cdots & \mathbf{A}_{kk} \end{pmatrix} \begin{pmatrix} \mathbf{D}_{11} & & & \\ & \mathbf{D}_{22} & & \\ & & \ddots & \\ & & & \mathbf{D}_{kk} \end{pmatrix} = \begin{pmatrix} \mathbf{D}_{11}\mathbf{A}_{11} & \mathbf{D}_{22}\mathbf{A}_{12} & \cdots & \mathbf{D}_{kk}\mathbf{A}_{1k} \\ \mathbf{D}_{22}\mathbf{A}_{21} & \mathbf{D}_{22}\mathbf{A}_{22} & \cdots & \mathbf{D}_{kk}\mathbf{A}_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{D}_{11}\mathbf{A}_{k1} & \mathbf{D}_{22}\mathbf{A}_{k2} & \cdots & \mathbf{D}_{kk}\mathbf{A}_{kk} \end{pmatrix}$$

$$\begin{pmatrix} \mathbf{D}_{11} & & & \\ & \mathbf{D}_{22} & & \\ & & \ddots & \\ & & & \mathbf{D}_{kk} \end{pmatrix}^{-1} = \begin{pmatrix} \mathbf{D}_{11}^{-1} & & & \\ & \mathbf{D}_{22}^{-1} & & \\ & & \ddots & \\ & & & \mathbf{D}_{kk}^{-1} \end{pmatrix}$$

7.3 Triangular matrices

Definition 7.3: Triangular matrix

A triangular matrix **T** is of the following form:

$$\mathbf{T} = \begin{pmatrix} t_{11} & t_{12} & \cdots & t_{1n} \\ & t_{22} & \cdots & t_{2n} \\ & & \ddots & \vdots \\ & & & t_{nn} \end{pmatrix}$$

Fact 7.3: Properties of a triangular matrix

- $\operatorname{spec}(\mathbf{T}) = \{t_{11}, t_{22}, \ldots, t_{nn}\}$;
- $\det(\mathbf{T}) = \prod_{i=1}^{k} t_{ii}$
- $\mathbf{T}$ is non-singular $\iff$ all $t_{ii} \neq 0$
- $\operatorname{Rank}(\mathbf{T}) \geq$ the number of nonzero t_{ii}. For example, the singular values of the strictly upper triangular matrix:
$$\begin{pmatrix} 0 & t_{12} & & & \\ & 0 & t_{23} & & \\ & & \ddots & \ddots & \\ & & & & t_{n-1,n} \\ & & & & 0 \end{pmatrix}$$
are $0, |t_{12}|, \ldots, |t_{n-1,n}|$.
- Sparsity patterns: the inverse of a triangular matrix is triangular.
- The product of two triangular matrices is triangular.

7.4 Block-triangular matrices

Definition 7.4: Block-triangular matrix

A block triangular matrix $\mathbf{T}$ is a matrix of the form:

$$\mathbf{T} = \begin{pmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} & \cdots & \mathbf{T}_{1k} \\ & \mathbf{T}_{22} & \cdots & \mathbf{T}_{2k} \\ & & \ddots & \vdots \\ & & & \mathbf{T}_{kk} \end{pmatrix}$$

Fact 7.4: Properties of block-triangular matrices

- $\operatorname{spec}(\mathbf{T}) = \bigcap_{i=1}^{k} \operatorname{spec}(\mathbf{T}_{ii})$
- $\det(\mathbf{T}) = \prod_{i=1}^{k} \det(\mathbf{T}_{ii})$
- $\mathbf{T}$ is non-singular $\iff$ all $\mathbf{T}_{ii}$ are non-singular
- $\operatorname{Rank}(\mathbf{T}) \geq \sum_{i=1}^{k} \operatorname{Rank}(\mathbf{T}_{ii})$
- The sparsity pattern is similar to the triangular case, but with respect to blocks.

7.5 Permutation matrices

Definition 7.5: Permutation matrix

A permutation matrix $\mathbf{P}$ is a matrix where exactly one entry in each row and column is equal to 1, and all other entries are 0. For example:

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} = \begin{pmatrix} 2 \\ 4 \\ 3 \\ 1 \end{pmatrix}$$

Fact 7.5: Facts about permutation matrices

- $\mathbf{P}^\top \mathbf{P} = \mathbf{P}\mathbf{P}^\top = \mathbf{I}$, i.e. $\mathbf{P}$ is orthogonal;
- $\det(\mathbf{P}) = \pm 1$, that is, permutation matrices are non-singular
- Left-multiplication of a matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$ and $n \times n$ permutation matrix $\mathbf{P}$, i.e. $\mathbf{PA}$, permutes the rows of $\mathbf{A}$;
- Right-multiplication of a matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$, and $n \times n$ permutation matrix $\mathbf{P}$, i.e. $\mathbf{AP}$, permutes the columns of $\mathbf{A}$.
- If $\mathbf{P}$ and $\mathbf{Q}$ are permutation matrices, then so is $\mathbf{PQ}$ and $\mathbf{QP}$ (generally $\mathbf{PQ} \neq \mathbf{QP}$)

7.6 Upper and lower Hessenberg matrices

Definition 7.6: Hessenberg matrix

A Hessenberg matrix (upper shown here, but lower is easily seen) $\mathbf{A}$ or $\mathbf{H}$ is a matrix of the form:

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ & a_{32} & a_{33} & \cdots & a_{3n} \\ & & \ddots & \ddots & \vdots \\ & & & a_{n,n-1} & a_{nn} \end{pmatrix}$$

Definition 7.7: Unreduced matrix

A Hessenberg matrix $\mathbf{A}$ is said to be unreduced if all of its super(sub)-diagonal entries are non-zero.

Fact 7.6

The rank of an unreduced matrix is at least $n-1$ since its first $n-1$ columns are independent.

7.7 Projection matrices

Definition 7.8: Projection matrix

A matrix $\mathbf{P} \in \mathbb{C}^{n \times n}$ is a projection, or idempotent, if $\mathbf{P}^2 = \mathbf{P}$.

Theorem 7.1: Various facts about projections which should be proven

1. $\mathbf{Pv} = \mathbf{v} \iff \mathbf{v} \in \text{Range}(\mathbf{P})$
2. If $\mathbf{P}$ is a projection, then so is $\mathbf{I} - \mathbf{P}$
3. $\text{Range}(\mathbf{I} - \mathbf{P}) = \text{Null}(\mathbf{P})$
4. $\text{Range}(\mathbf{P}) \cap \text{Range}(\mathbf{I} - \mathbf{P}) = \{\mathbf{0}\}$
5. $\text{Range}(\mathbf{P}) \oplus \text{Range}(\mathbf{I} - \mathbf{P}) = \mathbb{C}^n$
6. $\lambda \in \{0, 1\}$

Proof.

1. $(\implies)$ $\mathbf{v} = \mathbf{Pv} \implies \exists \mathbf{w} \in \mathbb{C}^n$s.t.$\mathbf{v} = \mathbf{Pw}$, namely $\mathbf{w} = \mathbf{v}$.
   $(\impliedby)$ $\mathbf{v} \in \text{Range}(\mathbf{P}) \implies \exists \mathbf{w} \in \mathbb{C}^n$ such that $\mathbf{v} = \mathbf{Pw} \implies \mathbf{Pv} = \mathbf{P}^2\mathbf{w} = \mathbf{Pw} = \mathbf{v}$.
2. $(\mathbf{I} - \mathbf{P})^2 = \mathbf{I} - 2\mathbf{P} + \mathbf{P}^2 = \mathbf{I} - \mathbf{P}$
3. $(\implies)$ $\mathbf{v} \in \text{Range}(\mathbf{I} - \mathbf{P}) \implies \exists \mathbf{w} \in \mathbb{C}^n$ such that $\mathbf{v} = (\mathbf{I} - \mathbf{P})\mathbf{w} \implies \mathbf{Pv} = \mathbf{P}(\mathbf{I} - \mathbf{P})\mathbf{w} = (\mathbf{P} - \mathbf{P}^2)\mathbf{w} = \mathbf{0} \implies \mathbf{v} \in \text{Null}(\mathbf{P})$
   $(\impliedby)$ $\mathbf{v} \in \text{Null}(\mathbf{P}) \implies \mathbf{Pv} = \mathbf{0} \implies \mathbf{v} = (\mathbf{I} - \mathbf{P})\mathbf{v} \implies \mathbf{v} \in \text{Range}(\mathbf{I} - \mathbf{P})$
4. $\mathbf{v} \in \text{Range}(\mathbf{P}) \cap \text{Range}(\mathbf{I} - \mathbf{P}) \implies \mathbf{v} = \mathbf{Pv} = \mathbf{P}(\mathbf{I} - \mathbf{P})\mathbf{v} = \mathbf{0}$
5. $\text{Range}(\mathbf{P}) \oplus \text{Range}(\mathbf{I} - \mathbf{P}) \subseteq \mathbb{C}^n$, but also $\mathbf{x} = \mathbf{Px} + (\mathbf{I} - \mathbf{P})\mathbf{x} \implies \mathbb{C}^n \subseteq \text{Range}(\mathbf{P}) \oplus \text{Range}(\mathbf{I} - \mathbf{P})$
6. $\mathbf{Pv} = \lambda\mathbf{v} \implies \mathbf{P}^2\mathbf{v} = \lambda\mathbf{Pv} \implies \mathbf{Pv} = \lambda^2\mathbf{v} \implies \lambda = \lambda^2 \implies \lambda \in \{0, 1\}$.

□

Definition 7.9: Orthogonal projection

A matrix $\mathbf{P} \in \mathbb{C}^{n \times n}$ is an orthogonal projection if $\mathbf{P}^2 = \mathbf{P}$ and $\mathbf{P}^* = \mathbf{P}$.

Note 7.3

Orthogonal projectors are not orthogonal matrices!!!!

Fact 7.7: Facts about orthogonal projections

- $\text{Range}(\mathbf{P}) \perp \text{Range}(\mathbf{I} - \mathbf{P})$
- $\|\mathbf{v}\|^2 = \|\mathbf{Pv}\|^2 + \|(\mathbf{I} - \mathbf{P})\mathbf{v}\|^2$
- Given any matrix $\mathbf{Q} \in \mathbb{C}^{m \times n}$ with orthonormal columns, $\mathbf{P} = \mathbf{QQ}^*$ is an orthogonal projection onto the $\text{Range}(\mathbf{Q})$
- Given any matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$, $\mathbf{P} = \mathbf{AA}^\dagger$ is an orthogonal projection onto the $\text{Range}(\mathbf{A})$.
- **Rank-one orthogonal projector:** $\mathbf{P} = \mathbf{vv}^*/\|\mathbf{v}\|^2$ is an orthogonal projection along the direction given by $\mathbf{v} \in \mathbb{C}^n$.

7.8 Positive (semi-)definite matrices

Definition 7.10: Positive (semi-)definite

If $\mathbf{A} \in \mathbb{R}^{n \times n}$ is symmetric, it is positive definite if:

$$\mathbf{A} \succ \mathbf{0} \iff \langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle > 0, \quad \forall \mathbf{x} \neq \mathbf{0} \in \mathbb{R}^n$$

It is positive semi-definite if:

$$\mathbf{A} \succeq \mathbf{0} \iff \langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle \geq 0, \quad \forall \mathbf{x} \in \mathbb{R}^n$$

If $\mathbf{A} \in \mathbb{C}^{n \times n}$ is Hermitian (implied), it is positive definite if:

$$\mathbf{A} \succ \mathbf{0} \iff \langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle > 0, \quad \forall \mathbf{x} \neq \mathbf{0} \in \mathbb{C}^n$$

It is positive semi-definite if:

$$\mathbf{A} \succeq \mathbf{0} \iff \langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle \geq 0, \quad \forall \mathbf{x} \in \mathbb{C}^n$$

Note 7.4

The latter definition implies that **A** is Hermitian:

$$\begin{aligned} \mathbf{B} &\triangleq \frac{1}{2}(\mathbf{A} + \mathbf{A}^*) \\ \mathbf{C} &\triangleq \frac{1}{2i}(\mathbf{A} - \mathbf{A}^*) \\ \implies \mathbf{A} &= \mathbf{B} + i\mathbf{C} \\ \implies \mathbf{x}^*\mathbf{A}\mathbf{x} &= \mathbf{x}^*\mathbf{B}\mathbf{x} + i\mathbf{x}^*\mathbf{C}\mathbf{x}, \quad \forall \mathbf{x} \in \mathbb{C}^n \text{ (all components are real)} \\ \implies \mathbf{C} = 0 &\implies \mathbf{A} = \mathbf{B} \end{aligned}$$

Note 7.5

The former definition without symmetry imposed is not enough to obtain positive (semi-)definiteness, eg.

$$\mathbf{A} = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \implies \begin{pmatrix} 1 & -i \end{pmatrix} \mathbf{A} \begin{pmatrix} 1 \\ i \end{pmatrix} = 2 + 2i \notin \mathbb{R}$$

Note 7.6

For real symmetric matrices, these two defintiions are equivalent - $\mathbf{A} \in \mathbb{R}^{ntimesn}$ being PD in real sense implies **A** is PD in complex sense.

Fact 7.8: Facts about positive (semi-)definite matrices

- $\mathbf{A} \in \mathbb{C}^{m \times n} \implies \mathbf{A}^*\mathbf{A} \succeq \mathbf{0}$
- $\mathbf{A} \in \mathbb{C}^{m \times n} \implies \mathbf{A}\mathbf{A}^* \succeq \mathbf{0}$

- $\mathbf{A} \succ \mathbf{0} \iff \lambda_i(\mathbf{A}) > 0, i = 1, \dots, n$
- $\mathbf{A} \succeq \mathbf{0} \iff \lambda_i(\mathbf{A}) \geq 0, i = 1, \dots, n$
- $\mathbf{A} \prec \mathbf{0} \iff \lambda_i(\mathbf{A}) < 0, i = 1, \dots, n$
- $\mathbf{A} \preceq \mathbf{0} \iff \lambda_i(\mathbf{A}) \leq 0, i = 1, \dots, n$
- Every PD matrix is invertible, and its inverse is also PD
- For $\mathbf{A}, \mathbf{B} \succ \mathbf{0}$ and $\alpha > 0$, $\alpha\mathbf{A} \succ \mathbf{0}$ and $\mathbf{A} + \mathbf{B} \succ \mathbf{0}$
- $\mathbf{A} \succeq \mathbf{0} \iff \exists! \mathbf{B} \succeq \mathbf{0}$ such that $\mathbf{B}^2 = \mathbf{A}$ (not to be confused with Cholesky factor)
- For $\mathbf{A} \succ \mathbf{0}$, the Schur decomposition, spectral decomposition and SVD all coincide
- If $\mathbf{A} \succeq \mathbf{0}$, then $\mathbf{B}^*\mathbf{A}\mathbf{B} \succeq \mathbf{0}, \forall \mathbf{B} \in \mathbb{C}^{n \times m}$
- If $\mathbf{A} \succ \mathbf{0}$ and $\mathbf{B}$ has full column rank, then $\mathbf{B}^*\mathbf{A}\mathbf{B} \succ \mathbf{0}$.

Note 7.7

$\mathbf{A} \nsucceq \mathbf{0}$ **does not** imply $\mathbf{A} \prec \mathbf{0}$. This is due to being a partial-order - we define the Loewner partial order as follows:

Definition 7.11: Loewner Partial-Order

$$\mathbf{A} \succ \mathbf{B} \iff \mathbf{A} - \mathbf{B} \succ \mathbf{0}$$
$$\mathbf{A} \succeq \mathbf{B} \iff \mathbf{A} - \mathbf{B} \succeq \mathbf{0}$$

Fact 7.9: Properties of the Loewner partial-order

- $\mathbf{A} \succeq \mathbf{B}$, then $\lambda_i(\mathbf{A}) \geq \lambda_i(\mathbf{B}), i = 1, 2, \dots, n$
- $\mathbf{A} \succeq \mathbf{B}$ and $\mathbf{A} \neq \mathbf{B}$, then $\exists i \in \{1, 2, \dots, n\}, \lambda_i(\mathbf{A}) > \lambda_i(\mathbf{B})$
- $\mathbf{A} \succ \mathbf{B}$, then $\lambda_i(\mathbf{A}) > \lambda_i(\mathbf{B}), i = 1, 2, \dots, n$ (after ordering eigenvalues).

Definition 7.12: Schur complement

Let $\mathbf{M} = \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^* & \mathbf{C} \end{pmatrix}$. The Schur complement of $\mathbf{A}$ in $\mathbf{B}$ is $\mathbf{C} - \mathbf{B}^*\mathbf{A}^{-1}\mathbf{B}$.

Fact 7.10: Properties of the Schur complement

- $\mathbf{M} \succ \mathbf{0} \iff \mathbf{A} \succ \mathbf{0}$ and $\mathbf{C} - \mathbf{B}^*\mathbf{A}^{-1}\mathbf{B} \succ \mathbf{0}$;
- $\mathbf{M} \succeq \mathbf{0} \iff \mathbf{A} \succ \mathbf{0}$ and $\mathbf{C} - \mathbf{B}^*\mathbf{A}^{-1}\mathbf{B} \succeq \mathbf{0}$.

7.9 Diagonally dominant matrices

Definition 7.13: Diagonally dominant matrix

A matrix **A** is diagonally dominant if it is of the form

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$$

and the magnitude of the diagonal entry in a row is larger than or equal to the sum of the magnitudes of all the other (non-diagonal) entries in that row:

$$|a_{ii}| \geq \sum_{i \neq j} |a_{ij}|, \quad \forall i.$$

Theorem 7.2: Levy-Desplanques Theorem

A strictly diagonally dominant matrix is non-singular.

Fact 7.11: Implications on positive (semi-)definiteness

- If $\mathbf{A} \in \mathbb{C}^{n \times n}$ is Hermitian, diagonally dominant with non-negative diagonals $a_{ii} \geq 0 \; \forall i$, then **A** is positive semi-definite.
- If $\mathbf{A} \in \mathbb{C}^{n \times n}$ is Hermitian, strictly diagonally dominant with positive diagonals $a_{ii} > 0 \; \forall i$, then **A** is positive definite.

8 Matrix factorisation

We have already seen a number of matrix factorisations, namely eigendecomposition, Schur decomposition, Jordan normal form and singular value decomposition. Now, we see a few more important matrix factorisations that are used in various applications.

8.1 LU factorisation

The LU factorisation involves splitting **A** into two matrices, **L** a lower triangular and **U** an upper triangular matrix:

$$\mathbf{A} = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} \end{pmatrix} = \underbrace{\begin{pmatrix} \ell_{11} & & \\ \vdots & \ddots & \\ \ell_{1n} & \dots & \ell_{nn} \end{pmatrix}}_{\mathbf{L}} \underbrace{\begin{pmatrix} u_{11} & \dots & u_{1n} \\ & \ddots & \vdots \\ & & u_{nn} \end{pmatrix}}_{\mathbf{U}}$$

LU factorisation is an example of triangular factorisations - these factors result from Gaussian elimination! LU factorisation of a given matrix may or may not exist, and if it exists, it need not be unique.

Fact 8.1

If **A** is invertible, then it admits an LU factorisation if and only if all its leading principal minors are nonzero.

Fact 8.2

We can uniquely write $\mathbf{A} = \mathbf{LDU}$, where **D** is a diagonal matrix and $\mathbf{L}, \mathbf{U}$ are *unit* triangular matrices:

$$\begin{pmatrix} 1 & & & \\ \frac{\ell_{21}}{\ell_{11}} & 1 & & \\ \vdots & \vdots & \ddots & \\ \frac{\ell_{n1}}{\ell_{11}} & \frac{\ell_{n2}}{\ell_{22}} & \cdots & 1 \end{pmatrix} \begin{pmatrix} \ell_{11}u_{11} & & & \\ & \ell_{22}u_{22} & & \\ & & \ddots & \\ & & & \ell_{nn}u_{nn} \end{pmatrix} \begin{pmatrix} 1 & \frac{u_{12}}{u_{11}} & \cdots & \frac{u_{1n}}{u_{11}} \\ & 1 & \cdots & \frac{u_{2n}}{u_{22}} \\ & & \ddots & \vdots \\ & & & 1 \end{pmatrix}$$

Note 8.1

Without proper permutations, LU factorisations may not exist, or can be numerically (backward) unstable. For example:

Example 8.1

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} \ell_{11} & 0 \\ \ell_{21} & \ell_{22} \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} \\ 0 & u_{22} \end{pmatrix} \implies \ell_{11}u_{11} = 0$$

So either **L** or **U** must be singular, but **A** is not singular.

Example 8.2

$$\begin{pmatrix} 10^{-20} & 1 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 10^{20} & 1 \end{pmatrix} \begin{pmatrix} 10^{-20} & 1 \\ 0 & 1 - 10^{20} \end{pmatrix}$$

$$\overset{\text{machine repn.}}{\implies} \begin{pmatrix} 1 & 0 \\ 10^{20} & 1 \end{pmatrix} \begin{pmatrix} 10^{-20} & 1 \\ 0 & -10^{20} \end{pmatrix} = \begin{pmatrix} 10^{-20} & 1 \\ 1 & 0 \end{pmatrix}$$

which is different from the initial.

Note 8.2

Singular matrices can have LU factorisation.

Example 8.3

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

Note 8.3

For singular matrices, LU need not be unique even if we require **L** or **U** to be unit lower (or upper) triangular.

Example 8.4

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ 0 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ a & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 0 & 2-a \end{pmatrix}, \quad \forall a$$

8.2 PLU factorisation

It turns out that a proper permutation in rows (or columns) is all that we need for LU factorisation to exist and to be numerically stable in practice.

Theorem 8.1: PLU Factorisation

For each $\mathbf{A} \in \mathbb{C}^{n \times n}$, there exists a permutation matrix $\mathbf{P} \in \mathbb{R}^{n \times n}$, a unit lower triangular $\mathbf{L} \in \mathbb{C}^{n \times n}$ and an upper triangular $\mathbf{U} \in \mathbb{C}^{n \times n}$ such that $\mathbf{A} = \mathbf{PLU}$.

The factors need not be unique. There are two main types of LU factorisation:

- LU factorisation with partial pivoting: $\mathbf{P}^\top \mathbf{A} = \mathbf{LU}$ - partial pivoting is explosively unstable for certain matrices, yet stable in practice for almost all real applications.
- LU factorisation with full pivoting: $\mathbf{P}^\top \mathbf{AQ} = \mathbf{LU}$ - improvement in stability over partial pivoting is marginal.

Computing LU factorisation requires $2n^3/3 + \mathcal{O}(n^2)$ flops (e.g. Doolittle algorithm, Crout algorithm etc). Partial pivoting adds only a quadratic term - full pivoting is significantly more expensive. If $\mathbf{A} \succ \mathbf{0}$ ($\mathbf{A}$ is positive definite), then $\mathbf{A} = \mathbf{LDL}^*$ where $\mathbf{D}$ is real positive diagonal. So, for positive definite matrices, this factorisation can be done at half the memory/flops. This brings us to Cholesky decomposition.

8.3 Cholesky factorisation

We previously saw that every matrix of the form $\mathbf{AA}^*$ is positive semi-definite. Cholesky gives us the converse of this in a certain sense. Cholesky factorisation is similar to LU factorisation in that it gives triangular factors, but only for positive semidefinite matrices. For a PSD matrix, Cholesky factorisation **always exists**; for a PD matrix, Cholesky factorisation **is unique**. By exploiting symmetry, Cholesky factorisation requires $\frac{n^3}{3} + \mathcal{O}(n^2)$ flops, i.e. half as much as LU decomposition for general matrices. It can be done by imposing the equality $\mathbf{A} = \mathbf{LL}^*$ element wise! All of the issues with the stability of Gaussian elimination vanish for Cholesky factorisation, i.e. this algorithm is always stable.

Theorem 8.2: Cholesky factorisation

Let $\mathbf{A} \in \mathbb{C}^{n\times n}$ be Hermitian. Then the following are true:

- $\mathbf{A}$ is positive semidefinite (respectively, positive definite) if and only if there is a lower triangular matrix $\mathbf{L} \in \mathbb{C}^{n\times n}$ with nonnegative (respectively, positive) diagonal entries such that $\mathbf{A} = \mathbf{LL}^*$;
- Furthermore, if $\mathbf{A}$ is positive definite, $\mathbf{L}$ is unique, i.e. there is only one lower triangular matrix $\mathbf{L}$ with strictly positive diagonal entries such that $\mathbf{A} = \mathbf{LL}^*$;
- $\mathbf{A}$ is real $\implies$ $\mathbf{L}$ is real.

Proof. Is this even a theorem? □

8.4 QR factorisation

So what if $\mathbf{A}$ isn't positive (semi-)definite? We can use QR factorisation. It is applicable to any $\mathbf{A} \in \mathbb{C}^{m\times n}$. The decomposition is into a form $\mathbf{A} = \mathbf{QR}$ where $\mathbf{Q} \in \mathbb{C}^{m\times n}$ is a unitary matrix and $\mathbf{R} \in \mathbb{C}^{m\times n}$ is an upper triangular matrix. In general, it is **not unique**, but if $\mathbf{A}$ is of full rank, then there exists a single $\mathbf{R}$ that has all positive diagonal elements. If $\mathbf{A}$ is square, then $\mathbf{Q}$ is unique. Just like LU is a reduction to upper-triangular form by Gaussian elimination, QR is a reduction to upper-triangular form by Gram-Schmidt. We'll see how to compute QR later in the course. Many algorithms exist.

Theorem 8.3: QR factorisation

Let $\mathbf{A} \in \mathbb{C}^{m\times n}$. Then:

- There exists a unitary $\mathbf{Q} \in \mathbb{C}^{m\times m}$ and an upper triangular $\mathbf{R} \in \mathbb{C}^{m\times n}$ with nonnegative diagonal entries such that $\mathbf{A} = \mathbf{QR}$.
- If $m \geq n$, there exists a $\mathbf{Q} \in \mathbb{C}^{m\times n}$ with orthonormal columns and an upper triangular $\mathbf{R} \in \mathbb{C}^{n\times n}$ with nonnegative main diagonal entries such that $\mathbf{A} = \mathbf{QR}$. This is called **"Thin QR" or "Reduced QR"**.
- If $\text{Rank}(\mathbf{A}) = n$, then the factors $\mathbf{Q}$ and $\mathbf{R}$ are uniquely determined and the diagonal entries of $\mathbf{R}$ are all positive.
- If $m = n$, then the factor $\mathbf{Q}$ is unitary.
- If $\mathbf{A}$ is real, then the factors $\mathbf{Q}$ and $\mathbf{R}$ may be taken to be real.

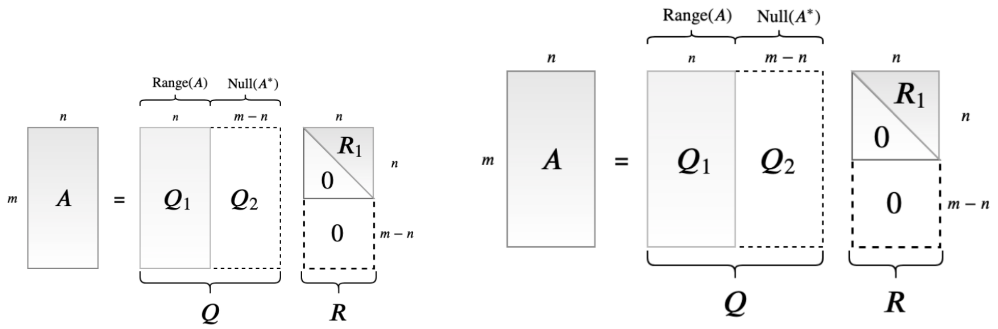
Fact 8.3: Different forms of QR

- Let $\mathbf{A} \in \mathbb{C}^{m\times n}$ with $\text{Rank}(\mathbf{A}) = n \leq m$. Then:

$$\mathbf{A} = \mathbf{QR} = \begin{pmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{pmatrix} \begin{pmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{pmatrix} = \mathbf{Q}_1\mathbf{R}_1$$

- Let $\mathbf{A} \in \mathbb{C}^{m\times n}$ with $\text{Rank}(\mathbf{A} = m \leq n)$. Then:

$$\mathbf{A} = \mathbf{QR} = \mathbf{Q} \begin{pmatrix} \mathbf{R}_1 & \mathbf{R}_2 \end{pmatrix}$$

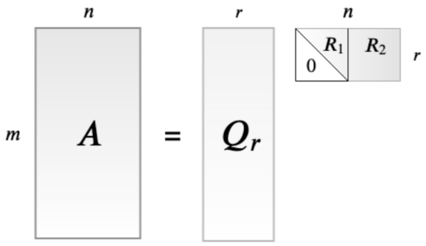


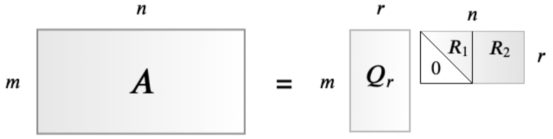
Fact 8.4: Permuting QR

Let $\mathbf{A} \in \mathbb{C}^{m\times n}$ with $\operatorname{Rank}(\mathbf{A}) = r < n \leq m$. Then we have $r_{ii} = 0$ for some i. One can permute the columns of $\mathbf{A}$ to obtain

$$\mathbf{AP} = \mathbf{P}\begin{pmatrix}\mathbf{R}_1 & \mathbf{R}_2\\ \mathbf{0} & \mathbf{0}\end{pmatrix} = \mathbf{Q}_r\begin{pmatrix}\mathbf{R}_1 & \mathbf{R}_2\end{pmatrix},$$

where $\mathbf{P}$ is a permutation matrix, $\mathbf{R}_1 \in \mathbb{C}^{r\times r}$ is non-singular and upper triangular and $\mathbf{Q}_r \in \mathbb{C}^{m\times r}$ has orthonormal columns. Amazingly, this holds in the case where $\operatorname{Rank}(\mathbf{A}) = r < m \leq n$.





Remark 8.1: Notes on FLOPs

A flop is a floating-point operation - its plural is flops. “flops” can also mean flops per second; ignore this, this isn’t a CS course. The number of flops required to execute an algorithm is called the flop count of the algorithm. Given an expression for the number of flops, we say the algorithm is $\mathcal{O}(n)$ if the dominant term in the flop count is Cn where C is a constant.

Remark 8.2

Evaluating complexity of algorithms using flops considers only part of the picture. It does *not* take into account data structures, memory footprint, memory hierarchy, parallelism, bandwidth, latency etcetera. Data locality can be crucial to the execution of an algorithm. *Smaller flop count is not always better. But noting that the run time of an algorithm is $\mathcal{O}(n^2)$ rather than $\mathcal{O}(n^3)$ can be meaningful in practice.*

9 Direct methods

Here begins the section on **numerical linear algebra**. Generally speaking, numerical linear algebra algorithms can be categorised into two main groups:

- **Direct methods:** the procedure takes finite number of steps. Upon completion, one has the exact solution (up to round-off errors). *No useful approximation if one stops early.*
- **Iterative methods:** the approximate solution, called the iterate, is updated iteratively. Some iterative methods take infinite number of steps. Some others give the exact solution in a finite number of steps. Most often, the solution is approximated well if one stops early.

Remark 9.1

Linear algebra is everwhere! Linear systems arise very frequently in virtually any branch of science, engineering, economics, finance and data analysis.

9.1 Linear systems

We now consider numerical methods for solving a linear system of equations of the form:

$$\mathbf{Ax} = \mathbf{b}, \quad \mathbf{A} \in \mathbb{C}^{n\times n}, \quad \mathbf{b} \in \mathbb{C}^n$$

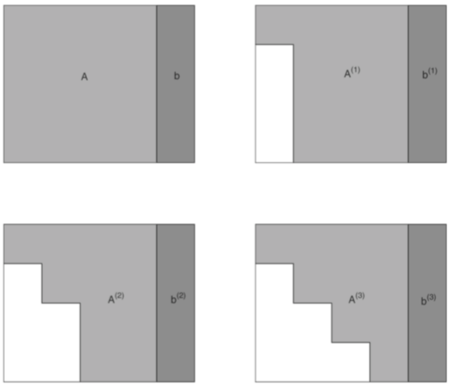
Note that the linear system can have no solution, one solution of inifnitely many solutions. We only consider the **consistent case**, in which the linear system has a solution of some sort.

$$\text{consistent} \iff \mathbf{b} \in \mathrm{Range}(\mathbf{A}) \iff \mathrm{Rank}(\mathbf{A}) = \mathrm{Rank}(\begin{pmatrix}\mathbf{A} & \mathbf{b}\end{pmatrix})$$

Two cases:

- If consistent and **A** is non-singular, then there is a unique solution
- If consistent and **A** is singular, then there are infinitely many solutions.

Recall Gaussian elimination. . .



$$\mathbf{Ax} = \mathbf{b} \quad \Longrightarrow \quad \mathbf{LUx} = b \quad \Longrightarrow \quad \underbrace{\mathbf{U}}_{\mathbf{A}^{(3)}}\mathbf{x} = \underbrace{\mathbf{L}^{-1}\mathbf{b}}_{\mathbf{b}^{(3)}}$$

So Gaussian elimination is essentially LU decomposition. Note the notation $\mathbf{A}^{(i)}$ means the ith iteration of Gaussian elimination.

$$\begin{aligned}\tilde{\mathbf{L}}^{(1)}\mathbf{A} &= \mathbf{A}^{(1)}\\ \tilde{\mathbf{L}}^{(2)}\mathbf{A}^{(1)} &= \mathbf{A}^{(2)}\\ &\vdots\end{aligned}$$

$$\begin{pmatrix} 1 & & & & \\ \frac{-a_{21}}{a_{11}} & 1 & & & \\ \frac{-a_{31}}{a_{11}} & & 1 & & \\ \vdots & & & \ddots & \\ \frac{-a_{n1}}{a_{11}} & & & & 1 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \dots & a_{2n}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & \dots & a_{2n}^{(1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(1)} & a_{n3}^{(1)} & \dots & a_{nn}^{(1)} \end{pmatrix}$$

$$\begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & \frac{-a_{32}^{(1)}}{a_{22}^{(1)}} & 1 & & \\ & \vdots & & \ddots & \\ & \frac{-a_{n2}^{(1)}}{a_{22}^{(1)}} & & & 1 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \dots & a_{2n}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & \dots & a_{2n}^{(1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(1)} & a_{n3}^{(1)} & \dots & a_{nn}^{(1)} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \dots & a_{2n}^{(1)} \\ 0 & 0 & a_{33}^{(2)} & \dots & a_{2n}^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & a_{n3}^{(2)} & \dots & a_{nn}^{(2)} \end{pmatrix}$$

With this, we can develop a form for the inverse of $\mathbf{L}$:

$$\underbrace{\tilde{\mathbf{L}}^{(n-1)}\dots\tilde{\mathbf{L}}^{(2)}\tilde{\mathbf{L}}^{(1)}}_{\tilde{\mathbf{L}}}\mathbf{A} = \mathbf{U} \implies \mathbf{A} = \underbrace{\tilde{\mathbf{L}}^{-1}}_{\mathbf{L}}\mathbf{U}$$

$$\Downarrow$$

$$\begin{aligned}\tilde{\mathbf{L}}^{-1} &= \left(\tilde{\mathbf{L}}^{(n-1)}\dots\tilde{\mathbf{L}}^{(2)}\tilde{\mathbf{L}}^{(1)}\right)^{-1}\\ &= \left[\tilde{\mathbf{L}}^{(1)}\right]^{-1}\left[\tilde{\mathbf{L}}^{(2)}\right]^{-1}\dots\left[\tilde{\mathbf{L}}^{(n-1)}\right]^{-1}\end{aligned}$$

Fact 9.1: Two strokes of luck

We obtain "two strokes of luck" from this result:

1. We can obtain the inverse of $\tilde{\mathbf{L}}^{(i)}$ by simply taking the negative:

$$\tilde{\mathbf{L}}^{(2)} = \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & \frac{-a_{32}^{(1)}}{a_{22}^{(1)}} & 1 & & \\ & \vdots & & \ddots & \\ & \frac{-a_{n2}^{(1)}}{a_{22}^{(1)}} & & & 1 \end{pmatrix} \implies [\tilde{\mathbf{L}}^{(2)}]^{-1} = \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & \frac{a_{32}^{(1)}}{a_{22}^{(1)}} & 1 & & \\ & \vdots & & \ddots & \\ & \frac{a_{n2}^{(1)}}{a_{22}^{(1)}} & & & 1 \end{pmatrix}$$

2. Let ℓ_k denote a vector with 0s above and at the diagonal, and $\ell_{k+1,k}$ below. It can be

seen that a matrix formed with these vectors plus identity gives us the **L** matrix:

$$\boldsymbol{\ell}_k = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \ell_{k+1,k} \\ \vdots \\ \ell_{n,k} \end{bmatrix} \implies \left[\tilde{\mathbf{L}}^{(k)}\right]^{-1}\left[\tilde{\mathbf{L}}^{(k+1)}\right]^{-1} = \left(\mathbf{I} + \ell_k \mathbf{e}_k^\top\right)\left(\mathbf{I} + \boldsymbol{\ell}_{k+1}\mathbf{e}_{k+1}^\top\right)$$

$$= \mathbf{I} + \boldsymbol{\ell}_k \mathbf{e}_k^\top + \boldsymbol{\ell}_{k+1}\mathbf{e}_{k+1}^\top$$

From this we can gather:

$$[\tilde{\mathbf{L}}^{(1)}]^{-1}[\tilde{\mathbf{L}}^{(2)}]^{-1} = \begin{pmatrix} 1 & & & & \\ \frac{a_{21}}{a_{11}} & 1 & & & \\ \frac{a_{31}}{a_{11}} & \frac{a_{32}^{(1)}}{a_{22}^{(1)}} & 1 & & \\ \vdots & \vdots & & \ddots & \\ \frac{a_{n1}}{a_{11}} & \frac{a_{n2}^{(1)}}{a_{22}^{(1)}} & & & 1 \end{pmatrix}$$

We now have an algorithm for solving a linear system using PLU factorisation:

1. $\tilde{\mathbf{b}} = \mathbf{P}^\top \mathbf{b}$
2. Forward Solve: $\mathbf{L}\mathbf{y} = \tilde{\mathbf{b}}$
3. Backward Solve: $\mathbf{U}\mathbf{x} = \mathbf{y}$

Algorithm 1: Solving $\mathbf{A}\mathbf{x} = \mathbf{b}$ using PLU factorisation

Note 9.1: Remarks on computation

Recall that PLU factorisation takes $2n^3/3 + \mathcal{O}(n^2)$ flops. The forward/backward substitutions, i.e. triangular system solves, cost only $\mathcal{O}(n^2)$ flops. In practice, **P** is not stored as a matrix; instead, the permutations are represented in a vector. For diagonally dominant and positive definite matrices, partial pivoting is **not** required. We may perform LU/PLU decomposition once and then solve different linear systems for different **b** at an $\mathcal{O}(n^2)$ cost for each, however, **we have to store the factors**.

Note 9.2: Never form $\mathbf{A}^{-1}$ explicitly!

If we had $\mathbf{A}^{-1}$, then we could have found $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ in $\mathcal{O}(n^2)$ flops. So suppose we would like to explicitly find $\mathbf{A}^{-1}$. We could form the LU decomposition of **A** and apply it to columns of identity:

$$\mathbf{LU}\mathbf{x}_i = \mathbf{e}_i, i = 1, \ldots, n \quad \implies \quad \mathbf{A}^{-1} = \begin{pmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \ldots & \mathbf{x}_n \end{pmatrix}$$

The initial LU takes $2n^3/3$ flops, each forward or backward will take $2n^2$ so then you end up with $2n^3/3 + 2n^2 \cdot n = 8n^3/2$ flops. Explicitly forming $\mathbf{A}^{-1}$ and multiplying with **b** is not recommended because:

- More computationally expensive than by simply using LU decomposition
- **A** can be sparse, $\mathbf{A}^{-1}$ casn be dense
- Round off errors get progressively worse as matrix size increases
- *No advantages!!!*

Definition 9.1: Banded matrices

A matrix **A** is banded if other than inside a band of diagonals, all other elements are nonzero, eg.

$$\mathbf{A} = \begin{pmatrix} a_{11} & \cdots & a_{1q} & & & & \\ \vdots & \ddots & \ddots & \ddots & & & \\ a_{p1} & \ddots & \ddots & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \ddots & \ddots & a_{n-q+1,n} \\ & & & \ddots & \ddots & \ddots & \vdots \\ & & & & a_{n,n-p+1} & \cdots & a_{nn} \end{pmatrix}$$

Banded matrices are fantastic for LU decomposition. If $p, q \ll n$ then LU decomposition can be done in merely $\mathcal{O}(n)$ time and storage! But with pivoting, not as fantastic.

$$\mathbf{L} = \begin{pmatrix} 1 & & & & & & \\ \vdots & 1 & & & & & \\ \ell_{p1} & \ddots & 1 & & & & \\ & \ddots & \ddots & \ddots & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & \ell_{n,n-p+1} & \cdots & 1 \end{pmatrix}, \mathbf{U} = \begin{pmatrix} u_{11} & \cdots & u_{1q} & & & & \\ & \ddots & \ddots & \ddots & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & \ddots & \ddots & u_{n-q+1,n} \\ & & & & & \ddots & \vdots \\ & & & & & & u_{nn} \end{pmatrix}$$

9.2 Ordinary least squares (OLS)

Consider the optimisation problem:

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2, \quad \mathbf{A} \in \mathbb{R}^{m \times n}, \mathbf{b} \in \mathbb{R}^m$$

The ordinary least squares, or OLS, problem often arises in many data fitting applications such as machine learning, computer graphics, statistics etc.

Example 9.1: Polynomial data fitting

Suppose we have m data points $\{(t_i, b_i)\}_{i=1}^m$. Suppose we'd like to fit a polynomial of degree at most $n-1$, of the form:

$$p_{n-1}(t) = \sum_{j=0}^{n-1} x_j t^j$$

If $m > n$, then we cannot hope to fit the data exactly (e.g. interpolating 3 points with a line is not possible). But we *can* approximate b_i's somewhat. Our polynomial approximation for each b_i can be written as:

$$p_{n-1}(t_i) = \sum_{j=0}^{n-1} x_j t_i^j = \langle \mathbf{x}, \mathbf{t}_i \rangle = \hat{b}_i, \quad i = 1, \ldots, m$$

where we define the following vectors:

$$\mathbf{x} \triangleq \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \quad \mathbf{t}_i \triangleq \begin{pmatrix} 1 \\ t_i \\ \vdots \\ t_i^{n-1} \end{pmatrix}$$

Here, we denote $\hat{b}_i$ as an approximation of b_i. We can collect all this information together and obtain the following linear system:

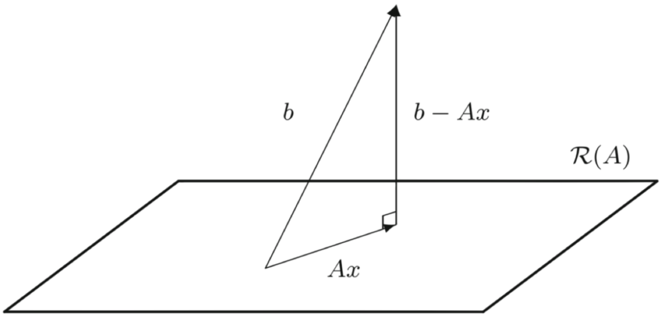
$$\begin{pmatrix} 1 & t_1 & t_1^2 & \cdots & t_1^{n-1} \\ 1 & t_2 & t_2^2 & \cdots & t_2^{n-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & t_m & t_m^2 & \cdots & t_m^{n-1} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} \hat{b}_1 \\ \hat{b}_2 \\ \vdots \\ \hat{b}_m \end{pmatrix}$$

We would like $\hat{\mathbf{b}}$ to be close to $\mathbf{b}$, hence we choose:

$$\min_{\mathbf{x}} \frac{1}{2} \|\hat{\mathbf{b}}(\mathbf{x}) - \mathbf{b}\|^2 = \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$$

Here $\mathbf{A}$ depends on the choice of basis for polynomials - we used monomial basis (which is only good for really low degree polynomials).

You can also fit data in other norms. For example minimising in the Manhattan norm is more robust to outliers, whereas minimising in the max-norm helps keep the worst-case error at check. These two formulations can be solved using linear programming, which we'll discuss later in the course.



So we need to minimise $f(\mathbf{x}) \triangleq \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$. Since $f(\mathbf{x})$ is convex (which will be explained later in the course), we just need to find a $\hat{\mathbf{x}}$ such that $\nabla f(\hat{\mathbf{x}}) = 0$.

$$
\begin{aligned}
f(\mathbf{x}) &= \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|^2 = \frac{1}{2}(\mathbf{A}\mathbf{x}-\mathbf{b})^\top(\mathbf{A}\mathbf{x}-\mathbf{b}) \\
&= \frac{1}{2}\left(\mathbf{x}^\top\mathbf{A}^\top - \mathbf{b}^\top\right)(\mathbf{A}\mathbf{x}-\mathbf{b}) \\
&= \frac{1}{2}\left(\mathbf{x}^\top\mathbf{A}^\top\mathbf{A}\mathbf{x} - \mathbf{x}^\top\mathbf{A}^\top\mathbf{b} - \mathbf{b}^\top\mathbf{A}\mathbf{x} + \mathbf{b}^\top\mathbf{b}\right) \\
&= \frac{1}{2}\left(\mathbf{x}^\top\mathbf{A}^\top\mathbf{A}\mathbf{x} - 2\mathbf{b}^\top\mathbf{A}\mathbf{x} + \mathbf{b}^\top\mathbf{b}\right) \\
&= \frac{1}{2}\mathbf{x}^\top\mathbf{A}^\top\mathbf{A}\mathbf{x} - \mathbf{b}^\top\mathbf{A}\mathbf{x} + \frac{1}{2}\mathbf{b}^\top\mathbf{b}
\end{aligned}
$$

Now note the following fact about inner products and gradients:

Fact 9.2

For all $\mathbf{b} \in \mathbb{R}^p$ and $\mathbf{B} \in \mathbb{R}^{p\times p}$, we have:

$$
\begin{aligned}
f(\mathbf{x}) = \langle \mathbf{b}, \mathbf{x}\rangle &\implies \nabla f(\mathbf{x}) = \mathbf{b} \\
f(\mathbf{x}) = \langle \mathbf{b}, \mathbf{B}\mathbf{x}\rangle &\implies \nabla f(\mathbf{x}) = (\mathbf{B}+\mathbf{B}^\top)\mathbf{x}
\end{aligned}
$$

We obtained:

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top\mathbf{A}^\top\mathbf{A}\mathbf{x} - b^\top\mathbf{A}\mathbf{x} + \frac{1}{2}\mathbf{b}^\top\mathbf{b}$$

which tells us:

$$\nabla f(\mathbf{x}) = \mathbf{A}^\top\mathbf{A}\mathbf{x} - \mathbf{A}^\top\mathbf{b}$$

Since we want when the gradient is zero, we need to find a $\hat{\mathbf{x}}$ such that:

$$\mathbf{A}^\top\mathbf{A}\hat{\mathbf{x}} = \mathbf{A}^\top\mathbf{b}$$

In general, solutions to $\mathbf{A}^\top\mathbf{A}\hat{\mathbf{x}} = \mathbf{A}^\top\mathbf{b}$ are of the form:

$$\hat{\mathbf{x}} = \mathbf{A}^\dagger\mathbf{b} + (\mathbf{I} - \mathbf{A}^\dagger\mathbf{A})\mathbf{y}, \quad \forall \mathbf{y} \in \mathbb{R}^m$$

where $\mathbf{A}^\dagger\mathbf{b}$ is the least-norm solution, and $(\mathbf{I} - \mathbf{A}^\dagger\mathbf{A})$ is the projection onto $\text{Null}(\mathbf{A})$. When $\mathbf{A}$ is full rank, we have that $\mathbf{A}^\dagger = (\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top$ and $\mathbf{A}^\dagger\mathbf{A} = \mathbf{I}$, and so we obtain the unique solution:

$$\hat{\mathbf{x}} = (\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top\mathbf{b}$$

So we can just use Cholesky to solve the normal equation $\mathbf{A}^\top\mathbf{A}\mathbf{x} = \mathbf{A}^\top\mathbf{b}$? No.

Example 9.2: Why you can't just do that

Consider the matrix $\mathbf{A} = \begin{pmatrix} 1 & 1 & 1 \\ \varepsilon & 0 & 0 \\ 0 & \varepsilon & 0 \\ 0 & 0 & \varepsilon \end{pmatrix}$. We have:

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

If $\eta \leq \varepsilon \leq \sqrt{\eta}$, where η is the rounding unit, then $\mathbf{A}^\top\mathbf{A}$ is **numerically singular**, even if $\mathbf{A}$ is full rank.

Proposition 9.1

Let $\mathbf{A} \in \mathbb{R}^{m\times n}$ be full column rank. We have $\kappa(\mathbf{A}^\top\mathbf{A}) = \kappa^2(\mathbf{A})$.

Proof. Let $\mathbf{A} = \mathbf{U\Sigma V}^\top$ be the economy SVD of $\mathbf{A}$. We have $\mathbf{A}^\top\mathbf{A} = \mathbf{V\Sigma}^2\mathbf{V}^\top$ and hence $\kappa(\mathbf{A}^\top\mathbf{A}) = \frac{\sigma_1^2}{\sigma_n^2} = \kappa^2(\mathbf{A})$. □

Example 9.3: Solving using QR factorisation

Suppose $\mathbf{A} \in \mathbb{R}^{m\times n}$ is full column rank. We compute its economy QR factorisation as $\mathbf{A} = \mathbf{QR}$. Let $\mathbf{Q}_\perp$ be the vectors **not** included in the economy QR. We verify that $\mathbf{QQ}^\top$ is an orthogonal projection onto $\mathtt{Range}(\mathbf{A})$, and that $\mathbf{Q}_\perp\mathbf{Q}_\perp^\top$ is an orthogonal projection onto $\mathsf{Null}(\mathbf{A}^\top)$, i.e. that $\mathbf{Q}_\perp\mathbf{Q}_\perp^\top = \mathbf{I} - \mathbf{QQ}^\top$. We have that:

$$\begin{aligned} \|\mathbf{Ax} - \mathbf{b}\|^2 = \|\mathbf{QRx} - \mathbf{b}\|^2 &= \left\|\mathbf{QRx} - \left(\mathbf{QQ}^\top + \mathbf{Q}_\perp\mathbf{Q}_\perp^\top\right)\mathbf{b}\right\|^2 \\ &= \left\|\mathbf{QRx} - \mathbf{QQ}^\top\mathbf{b}\right\|^2 + \left\|\mathbf{Q}_\perp\mathbf{Q}_\perp^\top\mathbf{b}\right\|^2 \\ &= \left\|\mathbf{Rx} - \mathbf{Q}^\top\mathbf{b}\right\|^2 + \left\|\mathbf{Q}_\perp\mathbf{Q}_\perp^\top\mathbf{b}\right\|^2 \end{aligned}$$

So therefore:

$$\arg\min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|^2 = \arg\min_{\mathbf{x}} \|\mathbf{Rx} - \mathbf{Q}^\top\mathbf{b}\|^2 = \mathbf{R}^{-1}\mathbf{Q}^\top\mathbf{b}$$

Example 9.4: Solving using SVD

Suppose $\mathbf{A} \in \mathbb{R}^{m\times n}$ with $r \triangleq \mathsf{Rank}(\mathbf{A}) < \min\{m, n\}$. There are infinitely many solutions! We typically choose the least norm solution $\mathbf{A}^\dagger\mathbf{b}$, which is the solution to the minimisation problem:

$$\begin{aligned} \min \quad & \|\mathbf{x}\|_2 \\ \text{s.t.} \quad & \mathbf{x} \in \arg\min_{\hat{\mathbf{x}}\in\mathbb{R}^n} \|\mathbf{A}\hat{\mathbf{x}} - \mathbf{b}\| \end{aligned}$$

We compute its economy SVD as $\mathbf{A} = \mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^\top$. Same as before, we obtain:

$$\|\mathbf{Ax} - \mathbf{b}\|^2 = \left\|\mathbf{\Sigma}_r\mathbf{V}_r^\top\mathbf{x} - \mathbf{U}_r^\top\mathbf{b}\right\|^2 + \left\|\left(\mathbf{I} - \mathbf{U}_r\mathbf{U}_r^\top\right)\mathbf{b}\right\|^2$$

But $\mathbf{\Sigma}_r$ is invertible, so $\mathbf{V}_r^\top\mathbf{x} = \mathbf{\Sigma}_2^{-1}\mathbf{U}_2^\top\mathbf{b}$. Any $\mathbf{x} = \mathbf{V}_r\mathbf{\Sigma}_r^{-1}\mathbf{U}_r^\top\mathbf{b} + (\mathbf{I} - \mathbf{V}_r\mathbf{V}_r^\top)\mathbf{z}, \forall\mathbf{z} \in \mathbb{R}^n$ is a solution. In particular:

$$\mathbf{x}^\star = \mathbf{V}_r\mathbf{\Sigma}_r^{-1}\mathbf{U}_r^\top\mathbf{b}$$

is the minimum norm solution.

There are three main methods to solving ordinary least squares:

- **Normal equation:** working with normal equations is fast, simple and intuitive. Solving the normal equation takes $mn^2 + n^3/3$ flops - solving it naively is generally strongly advised against. There are clever techniques to do this implicitly and more stably using iterative methods (more to come!)
- **QR factorisation:** the standard direct method - takes $2mn^3 - 2n^3/3$ flops. So QQR approach is roughly twice as expensive as the normal equation.
- **SVD factorisation:** takes $2mn^2 + 11n^3$. So SVD is roughly the same as QR for $m \gg n$, but more expensive when $m \approxeq n$. SVD is more robust for (nearly) rank deficient cases.

10 Gram-Schmidt and Householder reflections

There are generally two intuitive ways to compute QR factorisations. The first is **Gram-Schmidt procedure**, or triangular orthogonalisation (find $\mathbf{R}$):

$$\mathbf{A}\underbrace{\mathbf{R}_1\mathbf{R}_2\cdots\mathbf{R}_n}_{\mathbf{R}^{-1}} = \mathbf{Q}$$

The second method is **Householder reflections** or Givens rotation, which is orthogonal triangularisation (find $\mathbf{Q}$):

$$\underbrace{\mathbf{Q}_n\cdots\mathbf{Q}_2\mathbf{Q}_1}_{\mathbf{Q}^\top}\mathbf{A} = \mathbf{R}$$

10.1 Gram-Schmidt orthogonalisation

Let $\mathbf{A} = \mathbf{QR}$:

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

So, we obtain $\mathbf{a}_j = \sum_{i=1}^{j} r_{ij}\mathbf{q}_i$ by inspection. Start by setting:

$$r_{11} = \|\mathbf{a}_1\|, \quad \mathbf{q}_1 = \frac{\mathbf{a}_1}{\|\mathbf{a}_1\|}$$

We then require that $\mathbf{a}_2 = r_{12}\mathbf{q}_1 + r_{22}\mathbf{q}_2$, which gives us:

$$\begin{aligned} r_{12} &= \langle \mathbf{q}_1, \mathbf{a}_2 \rangle \\ r_{22} &= \|\mathbf{a}_2 - r_{12}\mathbf{q}_1\| \\ \mathbf{q}_2 &= \frac{\mathbf{a}_2 - r_{12}\mathbf{q}_1}{r_{22}} \end{aligned}$$

We can generalise the above procedure to the jth column, obtaining:

$$\mathbf{a}_j = r_{1j}\mathbf{q}_1 + r_{2j}\mathbf{q}_2 + \ldots + r_{jj}\mathbf{q}_j$$

giving us a general form:

$$\begin{aligned} r_{ij} &= \langle \mathbf{q}_i, \mathbf{a}_j \rangle, i = 1, \ldots, j-1 \\ r_{ij} &= \left\| \mathbf{a}_j - \sum_{i=1}^{j-1} r_{ij}\mathbf{q}_i \right\| \\ \mathbf{q}_j &= \frac{\mathbf{a}_j - \sum_{i=1}^{j-1} r_{ij}\mathbf{q}_i}{r_{ij}} \end{aligned}$$

Overall, this algorithm takes $2mn^2$ flops. The so-called classical Gram-Schmidt algorithm is as follows:

```
Input: $\mathbf{A}$
for $j = 1, 2, \ldots, n$ do
    $\mathbf{q}_j = \mathbf{a}_j$
    for $i = 1, \ldots, j-1$ do
        $r_{ij} = \langle \mathbf{q}_i, \mathbf{a}_j \rangle$
        $\mathbf{q}_j = \mathbf{q}_j - r_{ij}\mathbf{q}_i$
    end for
    $r_{jj} = \|\mathbf{q}_j\|$
    $\mathbf{q}_j = \mathbf{q}_j / r_{jj}$
end for
```

Algorithm 2: Classical Gram-Schmidt

Note 10.1: Issues with classical Gram-Schmidt

Simple, but highly numerically unstable if the columns of **A** are nearly linearly dependent, i.e. the vectors $\mathbf{q}_i$ are not quite orthogonal. Denote $\mathbf{P} = \begin{pmatrix} \mathbf{q}_1 & \mathbf{q}_2 & \ldots & \mathbf{q}_{j-1} \end{pmatrix}$. What classical Gram-Schmidt is doing is:

$$\begin{aligned} \tilde{\mathbf{q}}_j &= (\mathbf{I} - \mathbf{P}\mathbf{P}^*)\,\mathbf{a}_j \\ &= \left(\mathbf{I} - \sum_{i=1}^{j-1} \mathbf{q}_i\mathbf{q}_i^* \right) \mathbf{a}_j \end{aligned}$$

$$= \mathbf{a}_j - \sum_{i=1}^{j-1} \langle \mathbf{q}_i, \mathbf{a}_j \rangle \mathbf{q}_i$$

We can remedy this by multiplying instead:

$$\begin{aligned}\tilde{\mathbf{q}}_j &= (\mathbf{I} - \mathbf{PP}^*)\,\mathbf{a}_j \\ &= \left(\prod_{i=1}^{j-1} (\mathbf{I} - \mathbf{q}_i\mathbf{q}_i^*)\right)\mathbf{a}_j \\ &= \left(\mathbf{I} - \mathbf{q}_{j-1}\mathbf{q}_{j-1}^*\right) \dots (\mathbf{I} - \mathbf{q}_2\mathbf{q}_2^*)\,(\mathbf{I} - \mathbf{q}_1\mathbf{q}_1^*)\,\mathbf{a}_j\end{aligned}$$

```
Input: $\mathbf{A}$
for $j = 1, 2, \dots, n$ do
    $\mathbf{q}_j = \mathbf{a}_j$
    for $i = 1, \dots, j-1$ do
        $r_{ij} = \langle \mathbf{q}_i, \boxed{\mathbf{q}_j} \rangle$
        $\mathbf{q}_j = \mathbf{q}_j - r_{ij}\mathbf{q}_i$
    end for
    $r_{jj} = \|\mathbf{q}_j\|$
    $\mathbf{q}_j = \mathbf{q}_j / r_{jj}$
end for
```

Algorithm 3: Modified Gram-Schmidt

10.2 Householder reflections

Given a vector $\mathbf{v}$, we wish to find an orthogonal transformation $\mathbf{P}$ such that:

$$\mathbf{Pv} = \begin{pmatrix} \star \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Given a **unit vector** $\mathbf{u}$, let $\mathbf{P} = \mathbf{I} - 2\mathbf{uu}^\top$. $\mathbf{P}^\top\mathbf{P} = \mathbf{I}$, so $\mathbf{P}$ is orthogonal. Also note that $\mathbf{Pu} = -\mathbf{u}$, so $\mathbf{P}$ is also a reflector. Finding $\mathbf{P}$ will correspond to finding $\mathbf{u}$:

$$\alpha\mathbf{e}_1 = \mathbf{Pv} = (\mathbf{I} - 2\mathbf{uu}^\top)\mathbf{v} = \mathbf{v} - \beta\mathbf{u}$$

By orthogonality of $\mathbf{P}$, we have $\alpha = \pm\|\mathbf{v}\|$. So $\mathbf{u}$ must be parallel to $\mathbf{v} \pm \|\mathbf{v}\|\mathbf{e}_1$. We pick the sign, positive or negative, according to $\operatorname{sgn}(v_1)$, i.e. $\operatorname{sgn}(\star)$, to reduce cancellation error. So:

$$\mathbf{u} = \frac{\mathbf{v} + \operatorname{sgn}(v_1)\|\mathbf{v}\|\mathbf{e}_1}{\|\mathbf{v} + \operatorname{sgn}(v_1)\|\mathbf{v}\|\mathbf{e}_1\|}$$

So let $\mathbf{Q}_1$ be the Householder reflector for $\mathbf{a}_1$, and we have:

$$\mathbf{Q}_1\mathbf{A} = \begin{pmatrix} \star & \star & \star & \cdots & \star \\ 0 & \star & \star & \cdots & \star \\ 0 & \star & \star & \cdots & \star \\ 0 & \star & \star & \cdots & \star \\ \vdots & \vdots & \vdots & \cdots & \star \\ 0 & \star & \star & \star & \star \end{pmatrix}$$

Now let $\mathbf{P}_2$ be the Householder reflector corresponding to the boxed part, and set $\mathbf{Q}_2 = \begin{pmatrix} 1 & \\ & \mathbf{P}_2 \end{pmatrix}$. This gives us:

$$\mathbf{Q}_2\mathbf{Q}_1\mathbf{A} = \begin{pmatrix} \star & \star & \star & \cdots & \star \\ 0 & \star & \star & \cdots & \star \\ 0 & 0 & \star & \cdots & \star \\ 0 & 0 & \star & \cdots & \star \\ \vdots & \vdots & \vdots & \cdots & \star \\ 0 & 0 & \star & \star & \star \end{pmatrix}.$$

This pattern continues to give us:

$$\mathbf{Q}_n \cdots \mathbf{Q}_2\mathbf{Q}_1\mathbf{A} = \begin{pmatrix} \star & \star & \cdots & \star \\ & \star & \cdots & \star \\ & & \ddots & \vdots \\ & & & \star \\ & & & 0 \\ & & & \vdots \\ & & & 0 \end{pmatrix}$$

Householder transformations are the most suitable for general-purpose QR, as it is very robust. it has better numerical stability properties than modified Gram-Schmidt and its storage requirements are modest. Overall, it takes $2mn^2 - 2n^3/3$ flops.

11 Public holiday - no lecture

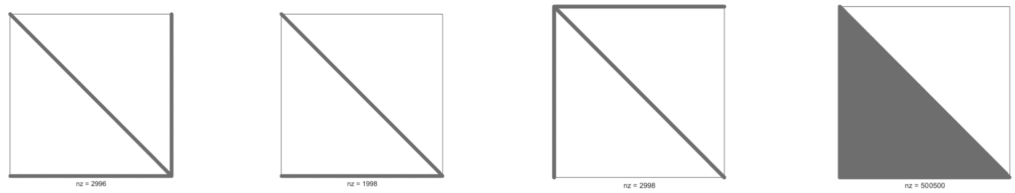
12 Iterative methods: stationary methods

12.1 Motivation and introduction

There are some serious drawbacks to direct methods:

- **They cannot solve inexactly:** Sometimes we don't raelly need to solve the system exactly (linear systems as sub-problems of nonlinear solvers, for example). So we need n steps each costing $\mathcal{O}(n^2)$, totalling $\mathcal{O}(n^3)$. For iterative methods, we can stop after $k \ll n$ iterations, hence needing only $\mathcal{O}(n^2)$.
- **They require A to be available explicitly**: Sometimes $\mathbf{A}$ is only known as a "black-box" operator (for example, we only have access to MVP $\mathbf{A} \cdot \mathbf{v}$).

- **They cannot use any prior info on the solution**: Sometimes, we have a good guess for an approximate solution, and we'd like to use that to "warm start" the method, for example, the solution to the previous step in a time-dependent problem.
- **They can result in fill-ins**: When $\mathbf{A}$ is large and sparse, but not tightly banded, LU decomposition can result in dense factors. Even if it is banded, it will result in fill-in within the band.



(a) $\mathbf{A} \in \mathbb{R}^{1000\times 1000}$: nnz = 2998 (b) $\mathbf{L}$: nnz = 1998 (c) $\mathbf{A} \in \mathbb{R}^{1000\times 1000}$: nnz = 2998 (d) $\mathbf{L}$: nnz = 500500

Remark 12.1: Before diving into iterative methods

Direct methods such as Gaussian elimination and its variations are the method of choice for many problems. Despite the disadvantages, one should not discount them altogether. For example, if one can "afford" direct methods, they might be preferable since they are less affected by the problem conditioning. Obviously, the correct choice of the algorithm is highly problem dependent.

Note 12.1: Motivation

Stationary methods are the result of work especially by Jacobi, Gauss and von Seidel. Stationary methods can be motivated as fixed point iterations. Suppose we need to solve $\mathbf{f}(\mathbf{x}) = \mathbf{0}$. We might be able to "split" $\mathbf{f}$ such that:

$$\mathbf{f}(\mathbf{x}) = \mathbf{0} \iff \mathbf{g}(\mathbf{x}) = \mathbf{x}$$

Fixed point iterations for splitting are of the form:

$$\mathbf{x}_{k+1} = \mathbf{g}(\mathbf{x}_k)$$

Here, $\mathbf{f}(\mathbf{x}) = \mathbf{A}\mathbf{x} - \mathbf{b}$. So if $\mathbf{g}(\mathbf{x}) = (\mathbf{I} - \mathbf{A})\mathbf{x} + \mathbf{b}$, then $\mathbf{x} = (\mathbf{I} - \mathbf{A})\mathbf{x} + \mathbf{b}$. Then we have fixed point iterations of the form:

$$\mathbf{x}_{k+1} = (\mathbf{I} - \mathbf{A})\mathbf{x}_k + \mathbf{b} = \mathbf{x}_k + (\mathbf{b} - \mathbf{A}\mathbf{x}_k)$$

More generally:

$$\mathbf{A} = \mathbf{M} - \mathbf{N} \implies \mathbf{M}\mathbf{x} = \mathbf{N}\mathbf{x} + \mathbf{b} \implies \mathbf{M}\mathbf{x}_{k+1} = \mathbf{N}\mathbf{x}_k + \mathbf{b}$$

Setting $\mathbf{M} = \mathbf{I}$ gives rise to our basic iterations on the previous slide, which is known as **Richardson iteration**:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + (\mathbf{b} - \mathbf{A}\mathbf{x}_k)$$

The convergent Richardson's method uses a "damping" factor:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha(\mathbf{b} - \mathbf{A}\mathbf{x}_k)$$

Why do we split? $\mathbf{M}\mathbf{x}_{k+1} = \mathbf{N}\mathbf{x}_k + \mathbf{b}$? If inverting $\mathbf{A}$ is very difficult, it might be worth inverting a much "easier" matrix several time, as opposed to directly inverting $\mathbf{A}$ only once. So we aim to split in such a way that ivnerting $\mathbf{M}$ can be done easily.

Fact 12.1: General stationary iterations

$$\mathbf{x}_{k+1} = \mathbf{M}^{-1}(\mathbf{N}\mathbf{x}_k + \mathbf{b}) = \mathbf{x}_k + \mathbf{M}^{-1}(\mathbf{b} - \mathbf{A}\mathbf{x}_k)$$

They are called stationary iterations because we are looking for a stationary point (a fixed point), and $\mathbf{M}$, $\mathbf{N}$ are constant and do not depend on iterations.

Note 12.2: Preconditioning

Relationship to preconditioning: $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{M}^{-1}(\mathbf{b} - \mathbf{A}\mathbf{x}_k)$. The matrix $\mathbf{M}$ can also be thought of as a preconditioning matrix, i.e.:

$$\begin{aligned}\mathbf{A}\mathbf{x} = \mathbf{b} &\Leftrightarrow \mathbf{M}^{-1}\mathbf{A}\mathbf{x} = \mathbf{M}^{-1}\mathbf{b} \\ &\Leftrightarrow \mathbf{x} = \mathbf{x} + \mathbf{M}^{-1}\mathbf{b} - \mathbf{M}^{-1}\mathbf{A}\mathbf{x} \\ &\Leftrightarrow \mathbf{x} = \mathbf{x} + \mathbf{M}^{-1}(\mathbf{b} - \mathbf{A}\mathbf{x})\end{aligned}$$

In other words, this is just applying Richardson iterations to the system $\mathbf{M}^{-1}\mathbf{A}\mathbf{x} = \mathbf{M}^{-1}\mathbf{b}$. If $\mathbf{M}^{-1}\mathbf{A}$ is better conditioned than $\mathbf{A}$, then this iteration produces better results faster than applying Richardson iterations to the original system $\mathbf{A}\mathbf{x} = \mathbf{b}$.

Note 12.3: How to choose M?

How do we choose $\mathbf{M}$ in $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{M}^{-1}(\mathbf{b} - \mathbf{A}\mathbf{x}_k)$?

1. Choose $\mathbf{M}$ such that inverting it is easy.
2. Choose $\mathbf{M}$ such that $\mathbf{M}^{-1}$ is close to $\mathbf{A}^{-1}$.

Different choices of $\mathbf{M}$ lead to a variety of methosd, from the simple iterative methods to very complicated multi-resolution ones.

Note 12.4: Important note

We do not have to form $\mathbf{M}^{-1}$ explicitly. We only need the solution to $\mathbf{M}\mathbf{p}_k = b - \mathbf{A}\mathbf{x}_k$, followed by $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{p}_k$.

12.2 Relaxation methods

Consider the typical Richardson iteration $\mathbf{A} = \mathbf{N}\mathbf{x} + \mathbf{b} \implies \mathbf{M}\mathbf{x}_{k+1} = \mathbf{N}\mathbf{x}_k + \mathbf{b}$. Let $\mathbf{A} = \mathbf{D} + \mathbf{E} + \mathbf{F}$, where:

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} a_{11} & & & \\ & a_{22} & & \\ & & \ddots & \\ & & & a_{nn} \end{pmatrix}$$

$$\mathbf{E} = \begin{pmatrix} 0 & & & \\ a_{21} & \ddots & & \\ \vdots & \ddots & \ddots & \\ a_{n1} & \cdots & a_{n,n-1} & 0 \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} 0 & a_{12} & \cdots & a_{1n} \\ & \ddots & \ddots & \vdots \\ & & \ddots & a_{n-1,n} \\ & & & 0 \end{pmatrix}$$

Fact 12.2: Types of iteration methods

- $\mathbf{M} = \mathbf{D}$: Jacobi method (simultaneous relaxation)
- $\mathbf{M} = \mathbf{D} + \mathbf{E}$: Gauss-Seidel method (GS)
- $\mathbf{M} = \omega^{-1}\mathbf{D} + \mathbf{E}$: Successive over-relaxation (SOR). $0 < \omega < 2$ is necessary for convergence (for $\mathbf{A}$ that has nonzero diagonal elements), and sufficient for PD systems. $\omega = 1$ then just Gauss-Seidel - *the best results are usually obtained for* $1 \leq \omega < 2$. There is also symmetric SOR (SSOR), and other variants.
- Block version of these splittings.

Note 12.5

When $\mathbf{A}$ is implicit, these methods may not be possible.

12.3 Convergence results

Note 12.6: Convergence of basic relaxation methods

We define $\mathbf{e}_k \triangleq \mathbf{x}_k - \mathbf{x}^\star$ as the error vector, and $\mathbf{T} \triangleq \mathbf{I} - \mathbf{M}^{-1}\mathbf{A}$ as athe iteration matrix. Let us denote the true solution by $\mathbf{x}^\star$, i.e. $\mathbf{A}\mathbf{x}^\star = \mathbf{b}$. We have:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{x}_k + \mathbf{M}^{-1}(\mathbf{b} - \mathbf{A}\mathbf{x}_k) \\ \mathbf{x}_{k+1} - \mathbf{x}^\star &= \mathbf{x}_k - \mathbf{x}^\star + \mathbf{M}^{-1}(\mathbf{A}\mathbf{x}^\star - \mathbf{A}\mathbf{x}_k) \\ \mathbf{x}_{k+1} - \mathbf{x}^\star &= \mathbf{x}_k - \mathbf{x}^\star - \mathbf{M}^{-1}\mathbf{A}(\mathbf{x}_k - \mathbf{x}^\star) \\ \mathbf{x}_{k+1} - \mathbf{x}^\star &= \left(\mathbf{I} - \mathbf{M}^{-1}\mathbf{A}\right)(\mathbf{x}_k - \mathbf{x}^\star) \\ \mathbf{x}_{k+1} - \mathbf{x}^\star &= \left(\mathbf{I} - \mathbf{M}^{-1}\mathbf{A}\right)^{k+1}(\mathbf{x}_0 - \mathbf{x}^\star) \\ \mathbf{e}_{k+1} &= \mathbf{T}^{k+1}\mathbf{e}_0 \end{aligned}$$

Note that $\lim_{k\to\infty} \mathbf{e}_k = 0 \iff \lim_{k\to\infty} \mathbf{T}^k = 0$.

Proposition 12.1: Sufficient condition on convergence

$$\|\mathbf{T}\| < 1 \implies \lim_{k\to\infty} \mathbf{e}_k = 0$$

Proof.

$$\begin{aligned}\|\mathbf{e}_{k+1}\| &= \left\|\mathbf{T}^{k+1}\mathbf{e}_k\right\| \\ &\leq \left\|\mathbf{T}^{k+1}\right\| \|\mathbf{e}_0\| \\ &\leq \|\mathbf{T}\|^{k+1} \|\mathbf{e}_0\|\end{aligned}$$

□

It turns out we have necessary and sufficient conditions on convergence:

Theorem 12.1: Necessary and sufficient condition on convergence

$$\rho(\mathbf{T}) < 1 \iff \lim_{k\to\infty} \mathbf{e}_k = 0$$

Proof. It follows immediately from the fact that:

$$\lim_{k\to\infty} \mathbf{A}^k = \mathbf{0} \iff \rho(\mathbf{A}) < 1$$

□

Definition 12.1: Non-asymptotic rate of convergence

If $\|\mathbf{T}\| < 1$, from $\|\mathbf{e}_k\| \leq \|\mathbf{T}\|^k \|\mathbf{e}_0\|$, it follows that after $k \geq \log(\varepsilon)/\log(\|\mathbf{T}\|)$, we have $\|\mathbf{e}_k\| \leq \varepsilon \|\mathbf{e}_0\|$ If $\|\mathbf{T}\| < 1$, then the factor $\|\mathbf{T}\|$ is called the non-asymptotic rate of convergence.

Theorem 12.2: Asymptotic rate of convergence

$$\limsup_{k\to\infty} \left(\frac{\|\mathbf{e}_k\|}{\|\mathbf{e}_0\|}\right)^{1/k} \leq \rho(\mathbf{T}), \quad \forall \mathbf{x}_0$$

Note 12.7: Notes on convergence

Jacobi, Gauss-Seidel and SOR do not necessarily converge for just any non-signular matrix $\mathbf{A}$, e.g. they are not even well-defined if $a_{ii} = 0$ for some i. They converge if $\mathbf{A}$ is strictly diagonally dominant. SOR converges for $\omega \in (0, 2)$ if $\mathbf{A}$ is positive definite. This is necessary for any matrix - if $\omega \notin (0, 2)$, the method will diverge for any matrix.

Note 12.8: Termination criteria

How should we terminate the iterations? Some possible options are:

1. $\|\mathbf{r}_k\| \leq \tau \|\mathbf{b}\|$
2. $\|\mathbf{x}_k - \mathbf{x}_{k-1}\| \leq \tau$

Elaborating:

1. $$\frac{\|\mathbf{x}_k - \mathbf{x}^\star\|}{\|\mathbf{x}^\star\|} \leq \kappa(\mathbf{A})\frac{\|\mathbf{r}_k\|}{\|\mathbf{b}\|} \leq \kappa(\mathbf{A})\tau,$$

   so when $\kappa(\mathbf{A}) \gg 1$, then τ has to be chosen small.
2. $$\begin{aligned} \|\mathbf{x}_k - \mathbf{x}^\star\| &\leq \|\mathbf{T}\|\,\|\mathbf{x}_{k-1} - \mathbf{x}^\star\| \\ &= \|\mathbf{T}\|\,\|\mathbf{x}_{k-1} - \mathbf{x}_k + \mathbf{x}_k - \mathbf{x}^\star\| \\ &\leq \|\mathbf{T}\|\left(\|\mathbf{x}_{k-1} - \mathbf{x}_k\| + \|\mathbf{x}_k - \mathbf{x}^\star\|\right) \\ \implies \|\mathbf{x}_k - \mathbf{x}^\star\| &\leq \left(\frac{\|\mathbf{T}\|}{1 - \|\mathbf{T}\|}\right)\|\mathbf{x}_k - \mathbf{x}_{k-1}\| \leq \left(\frac{\|\mathbf{T}\|}{1 - \|\mathbf{T}\|}\right)\tau \end{aligned}$$

   So when $\|\mathbf{T}\| \approx 1$, then τ has to be chosen small.

13 Iterative methods: polynomial acceleration

13.1 Introduction

Polynomial acceleration is a technique pioneered by Pafnuty Chebyshev. It is a technique to accelerate the convergence of stationary methods. Throughout this lecture, we assume **A** is positive definite, and define new variables M and m in terms of the minimum and maximum eigenvalues:

$$M \triangleq \lambda_{\max}(\mathbf{A}), \quad m \triangleq \lambda_{\min}(\mathbf{A})$$

For the stationary Richardson iteration $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha(\mathbf{b} - \mathbf{A}\mathbf{x}_k)$, the optimal step-size is given by $\alpha = \frac{2}{m+M}$. The convergence is given by:

$$\|\mathbf{e}_k\| \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)\|\mathbf{e}_{k-1}\|$$

where we define $\kappa \triangleq M/m$ as the condition number of **A**. Non-stationary Richardson aims to accelerate convergence by assigning a vector of α rather than a single α over the iterations:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \boxed{\alpha_k}(\mathbf{b} - \mathbf{A}\mathbf{x}_k)$$

Let $\mathbf{A}\mathbf{x}^\star = \mathbf{b}$ and $\mathbf{e}_k \triangleq \mathbf{x}_k - \mathbf{x}^\star$. We can easily see that:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_k\left(\mathbf{b} - \mathbf{A}\mathbf{x}_k\right) \\ \mathbf{x}_{k+1} - \mathbf{x}^\star &= \mathbf{x}_k - \mathbf{x}^\star + \alpha_k\left(\mathbf{A}\mathbf{x}^\star - \mathbf{A}\mathbf{x}_k\right) \end{aligned}$$

$$
\begin{aligned}
\mathbf{e}_{k+1} &= \mathbf{e}_k - \alpha_k \mathbf{A}\mathbf{e}_k = \left(\mathbf{I} - \alpha_k \mathbf{A}\right)\mathbf{e}_k \\
&= \left(\mathbf{I} - \alpha_k \mathbf{A}\right)\left(\mathbf{I} - \alpha_{k-1} \mathbf{A}\right)\mathbf{e}_{k-1} \\
&\vdots \\
\mathbf{e}_{k+1} &= \prod_{i=0}^{k} \left(\mathbf{I} - \alpha_i \mathbf{A}\right)\mathbf{e}_0
\end{aligned}
$$

where $\prod_{i=0}^{k}(\mathbf{I} - \alpha_i \mathbf{A})\mathbf{e}_0$ is a polynomial of degree $k+1$ in $\mathbf{A}$. So we have that $\mathbf{e}_k = p_k(\mathbf{A})\mathbf{e}_0$, where we define $p_k(\mathbf{A}) = \prod_{i=0}^{k-1}(\mathbf{I} - \alpha_i \mathbf{A})\mathbf{e}_0$ (noting that $p_k(\mathbf{0}) = \mathbf{I}$).

A similar method can be used for the residual - let $\mathbf{r}_k \triangleq \mathbf{b} - \mathbf{A}\mathbf{x}_k$. We easily see that:

$$
\begin{aligned}
\mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_k \left(\mathbf{b} - \mathbf{A}\mathbf{x}_k\right) \\
\mathbf{A}\mathbf{x}_{k+1} - \mathbf{b} &= \mathbf{A}\mathbf{x}_k - \mathbf{b} + \alpha_k \mathbf{A}\left(\mathbf{b} - \mathbf{A}\mathbf{x}_k\right) \\
\mathbf{r}_{k+1} &= \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{r}_k = \left(\mathbf{I} - \alpha_k \mathbf{A}\right)\mathbf{r}_k \\
&= \left(\mathbf{I} - \alpha_k \mathbf{A}\right)\left(\mathbf{I} - \alpha_{k-1} \mathbf{A}\right)\mathbf{r}_{k-1} \\
&\vdots \\
\mathbf{r}_{k+1} &= \prod_{i=0}^{k} \left(\mathbf{I} - \alpha_i \mathbf{A}\right)\mathbf{r}_0
\end{aligned}
$$

where $\prod_{i=0}^{k}\left(1 - \alpha_i \mathbf{A}\right)\mathbf{r}_0$ is a polynomial of degree $k+1$. We have $\mathbf{r}_k = p_k(\mathbf{A})\mathbf{r}_0$, where $p_k(\mathbf{A}) \triangleq \prod_{i=0}^{k-1}\left(1 - \alpha_i \mathbf{A}\right)\mathbf{r}_0$. Note that $p_k(\mathbf{0}) = \mathbf{I}$. So in general, we obtain the following equivalence:

$$
\mathbf{r}_k = p_k(\mathbf{A})\mathbf{r}_0, \quad \text{and} \quad \mathbf{e}_k = p_k(\mathbf{A})\mathbf{e}_0
$$

where $p_k(\mathbf{0}) = \mathbf{I}$.

Recall the equivalence between polynomials of matrices and numbers:

$$
p_k(t) = \prod_{i=0}^{k-1}\left(1 - \alpha_i t\right) \iff p_k(\mathbf{A}) = \prod_{i=0}^{k-1}\left(1 - \alpha_i \mathbf{A}\right)
$$

where $1/\alpha_i$ are the roots of the polynomial $p_k(t)$.

Definition 13.1: Residual polynomial

A residual polynomial is a polynomial of degree k where $p_k(0) = 1$.

We now have (where $||\cdot||_2$ Euclidean vector norm with ℓ_2 induced matrix norm) that:

$$
\|\mathbf{r}_k\|_2 \leq \|p_k(\mathbf{A})\|_2 \|\mathbf{r}_0\|_2 \quad \text{and} \quad \|\mathbf{e}_k\|_2 \leq \|p_k(\mathbf{A})\|_2 \|\mathbf{e}_0\|_2
$$

Therefore:

$$
\begin{aligned}
\|\mathbf{r}_k\|_2 \leq \|p_k(\mathbf{A})\|_2 \|\mathbf{r}_0\|_2 \\
\|\mathbf{e}_k\|_2 \leq \|p_k(\mathbf{A})\|_2 \|\mathbf{e}_0\|_2
\end{aligned}
$$

where

$$p_k(\mathbf{A}) = \prod_{i=0}^{k-1} (\mathbf{I} - \alpha_i \mathbf{A})$$

The idea of polynomial acceleration is to find a residual polynomial which minimises the 2-norm:

$$\min_{\{\alpha_0,\ldots,\alpha_{k-1}\}} \|p_k(\mathbf{A})\|$$

Recalling the first part of the spectral mapping theorem, if $(\lambda, \mathbf{v})$ is an eigenpair $\mathbf{A} \in \mathbb{C}^{n\times n}$, then $(p(\lambda), \mathbf{v})$ is an eigenpair of $p(\mathbf{A})$. Let λ_i be eigenvalues of $\mathbf{A}$. Suppose $\mathbf{A}$ is normal. Then:

$$\|\mathbf{A}\| = \rho(\mathbf{A}) = \max_{i\in\{1,\ldots,n\}} |\lambda_i|$$

By chasing definition, $p_k(\mathbf{A})$ is normal so:

$$\|p_k(\mathbf{A})\| = \rho\left(p_k(\mathbf{A})\right) = \max_{i\in\{1,\ldots,n\}} |p_k\left(\lambda_i\right)|$$

As a result:

$$\begin{aligned}
\|\mathbf{r}_k\|_2 &\leq \max_{i\in\{1,\ldots,n\}} |p_k(\lambda_i)|\|\mathbf{r}_0\|_2 \\
\|\mathbf{e}_k\|_2 &\leq \max_{i\in\{1,\ldots,n\}} |p_k(\lambda_i)|\|\mathbf{e}_0\|_2
\end{aligned}$$

Minimising $\|p_k(\mathbf{A})\|_2$ gives the following min-max problem:

$$\min_{\{\alpha_0,\ldots,\alpha_{k-1}\}} \max_{i\in\{1,\ldots,n\}} |p_k\left(\lambda_i\right)|$$

In general, we might not know the eigenvalues of $\mathbf{A}$, but suppose we are able to find some set that-contains all the eigenvalues $\{\lambda_1, \lambda_2, \ldots, \lambda_n\} \subseteq \mathcal{S}$. Since $\max_{\lambda\in\{\lambda_1,\lambda_2,\ldots,\lambda_n\}} |p_k(\lambda)| \leq \max_{\lambda\in\mathcal{S}} |p_k(\lambda)|$, we can loosen the bound to

$$\begin{aligned}
\|\mathbf{e}_k\|_2 &\leq \min_{\{\alpha_0,\ldots,\alpha_{k-1}\}} \max_{\lambda\in\mathcal{S}} |p_k(\lambda)| \, \|\mathbf{e}_0\|_2 \\
&= \min_{\{\alpha_0,\ldots,\alpha_{k-1}\}} \max_{\lambda\in\mathcal{S}} \left| \prod_{i=0}^{k-1} (1 - \alpha_i \lambda) \right| \|\mathbf{e}_0\|_2
\end{aligned}$$

Define the following set

$$\Pi_k \triangleq \{p(.) \text{ is a polynomial} \mid p(0) = 1, \text{ degree } (p) \leq k\}.$$

The problem of finding optimal α_i is equivalent to

$$\|\mathbf{e}_k\|_2 \leq \min_{p_k\in\Pi_k} \max_{\lambda\in\mathcal{S}} |p_k(\lambda)| \, \|\mathbf{e}_0\|_2 \, .$$

13.2 Chebyshev polynomials

Chebyshev polynomials are the key to optimal α_i, but we won't go into depth about them in this lecture. Refer to slides for more detail.

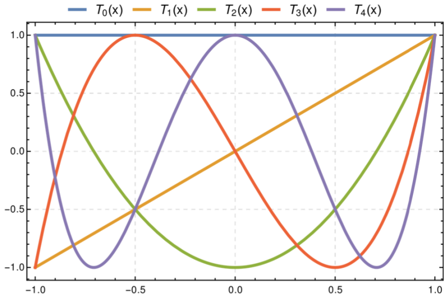
Definition 13.2: Chebyshev polynomials of the first kind

Chebyshev polynomials of the first kind are defined recursively as:

$$\begin{aligned} T_0(x) &= 1 \\ T_1(x) &= x \\ T_{k+1}(x) &= 2xT_k(x) - T_{k-1}(x), \quad k \geq 1 \end{aligned}$$

Alternatively, we have explicit expressions:

$$T_k(x) = \begin{cases} \cos(k \arccos(x)) & |x| \leq 1 \\ \frac{1}{2}\left[(x + \sqrt{x^2-1})^k + (x + \sqrt{x^2-1})^{-k}\right] & |x| \geq 1 \end{cases}$$



One can show that the min-max of $\min_{p_k \in \Pi_k} \max_{t \in [m,M]} |p_k(t)|$ is attained uniquely by the polynomial

$$p_k^\star(t) = \frac{T_k\left(1 + \frac{2(t-M)}{M-m}\right)}{T_k\left(-\frac{M+m}{M-m}\right)}.$$

Also,

$$\max_{t \in [m,M]} |p_k^\star(t)| \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k$$

The optimal step-sizes are given by the reciprocals of the roots of $p_k^\star(t)$ as:

$$\alpha_i^{-1} = \frac{1}{2}\left(m + M + (M-m)\cos\left(\frac{\pi(2(i+1)-1)}{2k}\right)\right), \quad i = 0, \ldots, k-1$$

We compare this with the stationary Richardson's step size of $\alpha = \frac{2}{m+M}$. So, for the non-stationary Richardson iteration, we get:

$$\|\mathbf{e}_k\| \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k \|\mathbf{e}_0\|$$

Compare this with the convergence of the stationary Richardson iteration:

$$\|\mathbf{e}_k\| \leq \left(\frac{\kappa-1}{\kappa+1}\right)^k \|\mathbf{e}_0\|$$

Note 13.1: Drawbacks of naive Chebyshev acceleration

1. In exact arithmetic, the order in which the parameters α_i are used does not matter. In practice, however, this can be very important; the "natural" ordering may lead to large round-off errors. Hence, without appropriate re-ordering methods, the calculations can be very unstable.
2. One also has to choose the number of iterations k in advance!
3. For Chebyshev acceleration to be effective, a fairly accurate knowledge of the interval $[a, b]$ is required. If this estimate is too crude, the method can be very inefficient.

To alleviate the first two problems, Chebyshev acceleration is stably implemented using the three-term recurrence relation

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

We will see later an alternative method, called CG or conjugate gradient, which also alleviates the last draw-back, i.e. it is parameter free.

$$\begin{aligned}
\mathbf{x}_1 &= \mathbf{x}_0 + \alpha\mathbf{r}_0, \quad k = 0 \\
\mathbf{x}_{k+1} &= \mathbf{x}_{k-1} + \alpha\omega_k\mathbf{r}_k + \omega_k\left(\mathbf{x}_k - \mathbf{x}_{k-1}\right), \quad k \geq 1
\end{aligned}$$

where

$$\begin{aligned}
\alpha &= \frac{2}{m+M} \\
\omega_0 &= 2 \\
\omega_k &= \left(1 - \frac{\omega_{k-1}}{4\beta^2}\right)^{-1}, \quad k \geq 1 \\
\beta &\triangleq 1 - \frac{2M}{M-m}
\end{aligned}$$

More generally, such polynomial acceleration techniques can be applied to many stationary methods under certain conditions. Chebyshev acceleration in general requires real eigenvalues for **A** (real positive eigenvalues for Richardson iteration acceleration). However, polynomial acceleration can be used in more general settings, even when **A** has complex eigenvalues, if the polynomials are chosen correctly (e.g. if the spectrum is bounded by an ellipse in the complex plane). It is even possible to derive acceleration schemes adapted to the case when the spectrum is contained in two intervals $[a, b] \cup [c, d]$, where $a < b < 0 < c < d$.

14 Iterative methods: Krylov subspaces

14.1 Preliminaries

Sometimes it's hard to find a lower or upper bound with polynomial acceleration, $[a, b]$. You can get b if you're lucky, but a is pretty much impossible. This can be remedied with another class of algorithms which will be covered over the next 5 lectures: Krylov subspace methods.

Krylov subspace methods are among the top ten "algorithms with the greatest influence on the development and practice of science and engineering in the 20th century"! Perhaps the most important class of iterative methods for solving linear systems, they do not require any explicit form of **A** and are particularly well suited for large sparse linear systems. Many different Krylov subspace methods exist: some are suitable for nonsymmetric matrices, and others require symmetry or even PD. When only approximate solutions are needed, these methods possess some attractive approximation/convergence properties. In fact, most Krylov methods (all?) give an exact solution after a finite number of iterations.

Note 14.1: Building up

Suppose $\mathbf{A} \in \mathbb{C}^{n\times n}$ with $0 < \lambda_i(\mathbf{A}) < 2$, $i = 1, \ldots, n$. It follows that $\rho(\mathbf{I} - \mathbf{A}) < 1$ (since $1 - 2 = -1$ so $|-1| = 1$), and so:

$$\mathbf{A}^{-1} = (\mathbf{I} - (\mathbf{I} - \mathbf{A}))^{-1} = \sum_{k=0}^{\infty} (\mathbf{I} - \mathbf{A})^k$$

This geometric series is known as a Neumann series. So for any $\mathbf{x}_0 \in \mathbb{C}^n$, we have:

$$\begin{aligned} \mathbf{x}^\star &= \mathbf{A}^{-1}\mathbf{b} \\ &\overset{\mathbf{b}=\mathbf{A}\mathbf{x}^\star}{=} \mathbf{A}^{-1}(\mathbf{A}\mathbf{x}^\star - \mathbf{A}\mathbf{x}_0 + \mathbf{A}\mathbf{x}_0) \\ &= \mathbf{x}_0 + \mathbf{A}^{-1}\mathbf{r}_0 \\ &= \mathbf{x}_0 + \sum_{k=0}^{\infty} (\mathbf{I} - \mathbf{A})^k \mathbf{r}_0 \end{aligned}$$

Do we actually need to compute infinitely many vectors to be able to find $\mathbf{x}^\star$? Of course not.

Theorem 14.1: Cayley-Hamilton

Let $p_n(\lambda) = \sum_{i=0}^{n} c_i \lambda^i$ be the characteristic polynomial of the matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$. Then we have $p_n(\mathbf{A}) = 0$.

This implies for us that (since **A** is invertible):

$$\begin{aligned} c_n\mathbf{A}^n + c_{n-1}\mathbf{A}^{n-1} + \ldots + c_1\mathbf{A} + c_0 &= \mathbf{0} \\ c_n\mathbf{A}^{n-1} + c_{n-1}\mathbf{A}^{n-2} + \ldots + c_1 + c_0\mathbf{A}^{-1} &= \mathbf{0} \\ \alpha_n\mathbf{A}^{n-1} + \alpha_{n-1}\mathbf{A}^{n-2} + \ldots + \alpha_2\mathbf{A} + \alpha_1 &= \mathbf{A}^{-1} \end{aligned}$$

where $\alpha_i = -c_i/c_0$. So we can represent $\mathbf{A}^{-1}$ as a matrix polynomial of degree $n-1$ (coefficient will depend on $\mathtt{spec}(\mathbf{A})$), i.e. we should be able to find $\mathbf{x}^\star$ after having only finitely many vectors. $\mathbf{x}^\star$ should be in some subspace of linear combinations of powers of $\mathbf{A}\mathbf{r}_0 + \mathbf{x}_0$. This is where Krylov subspaceds arise.

Note 14.2: The central question of Krylov subspace theory

Suppose we only consider a subspace of this, i.e. for some $k < n$:

$$\mathbf{x}_k \in \mathbf{x}_0 + \mathrm{Span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \mathbf{A}^2\mathbf{r}_0, \ldots, \mathbf{A}^{k-1}\mathbf{r}_0\}$$

How good is this approximation $\mathbf{x}_k$? And what is "good"? We've already seen an example of this subspace approximation in Richardson iterations:

$$\begin{aligned}
\mathbf{x}_k &= \mathbf{x}_{k-1} + \alpha_{k-1}(\mathbf{b} - \mathbf{A}\mathbf{x}_{k-1}) \\
&= \mathbf{x}_{k-1} + \alpha_{k-1}\mathbf{r}_{k-1} \\
&= \mathbf{x}_{k-2} + \alpha_{k-1}\mathbf{r}_{k-1} + \alpha_{k-2}\mathbf{r}_{k-2} = \ldots = \mathbf{x}_0 + \sum_{i=0}^{k-1} \alpha_i \mathbf{r}_i \\
&= \mathbf{x}_0 + \sum_{i=0}^{k-1} \alpha_i \prod_{j=0}^{i-1} (\mathbf{I} - \alpha_j \mathbf{A})\mathbf{r}_0
\end{aligned}$$

So $\mathbf{x}_k \in \mathbf{x}_0 + \mathrm{Span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \mathbf{A}^2\mathbf{r}_0, \ldots, \mathbf{A}^{k-1}\mathbf{r}_0\}$. But depending on our α_k dramatically different convergence.

Note 14.3: The central quest

The main quest is to find the "best" $\mathbf{x}_k$ for some $k \ll n$ such that $\mathbf{x}_k \approx \mathbf{x}^\star$ in some sense.

14.2 Defining Krylov subspaces

Definition 14.1: Krylov subspace

The Krylov subspace of order k generated by the matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ and the vector $\mathbf{v} \in \mathbb{C}^n$ is defined as:

$$\mathcal{K}_k(\mathbf{A}, \mathbf{v}) = \mathrm{Span}\{\mathbf{v}, \mathbf{A}\mathbf{v}, \ldots, \mathbf{A}^{k-1}\mathbf{v}\}, \quad k \geq 1$$

and where $\mathcal{K}_0(\mathbf{A}, \mathbf{v}) = \{\mathbf{0}\}$ (since all subspaces have to contain zero).

Now we note a few elementary properties of this subspace:

- Scaling: $\mathcal{K}_k(\mathbf{A}, \mathbf{v}) = \mathcal{K}_k(\alpha\mathbf{A}, \beta\mathbf{v}), \alpha \neq 0, \beta \neq 0$;
- Triangularisation: $\mathcal{K}_k(\mathbf{A}, \mathbf{v}) = \mathcal{K}_k(\mathbf{A} - \mu\mathbf{I}, \mathbf{v}), \forall \mu \in \mathbb{F}$
- Similarity: $\mathcal{K}_k(\mathbf{B}^{-1}\mathbf{A}\mathbf{B}, \mathbf{B}^{-1}\mathbf{v}) = \mathbf{B}^{-1}\mathcal{K}_k(\mathbf{A}, \mathbf{v})$
- Nested property: $\mathcal{K}_k(\mathbf{A}, \mathbf{v}) \subseteq \mathcal{K}_{k+1}(\mathbf{A}, \mathbf{v}), \forall k \geq 1$ (Fred comment: duh!)
- $\mathcal{K}_{k+1}(\mathbf{A}, \mathbf{v}) = \mathrm{Span}(\mathbf{v}) + \mathbf{A}\mathcal{K}_k(\mathbf{A} < \mathbf{v}), \forall k \geq 1$. So in particular, we have $\mathbf{A}\mathcal{K}_k(\mathbf{A}, \mathbf{v}) \subseteq \mathcal{K}_{k+1}(\mathbf{A}, \mathbf{v}), \forall k \geq 1$. Also if $\notin \mathrm{Range}(\mathbf{A})$, then $\exists k$ such that $\mathbf{A}\mathcal{K}_k(\mathbf{A}, \mathbf{v}) \subsetneq \mathcal{K}_{k+1}(\mathbf{A}, \mathbf{v})$.

So $\mathcal{K}_k(\mathbf{A}, \mathbf{v})$ is a subspace. A natural question is what is $\dim(\mathcal{K}_k(\mathbf{A}, \mathbf{v}))$? Since these spaces are nested, it might be reasonable to expect that their dimension cannot grow indefinitely. At some spoint, the Krylov subspace is large enough that in some sense contains all the information we can extract from $\mathbf{A}$ through its multiplication by $\mathbf{v}$. For example, if $\mathbf{v}$ is an eigenvector of $\mathbf{A}$, then $\dim(\mathcal{K}_k(\mathbf{A}, \mathbf{v})) = 1, \forall k$ it turns out that this is indeed the case.

Theorem 14.2: Grade of v with respect to A

There exists a positive integer $t \triangleq t(\mathbf{v}, \mathbf{A})$ called the grade of $\mathbf{v}$ with respect to $\mathbf{A}$ such that:

$$\dim(\mathcal{K}_k(\mathbf{A}, \mathbf{v})) = \begin{cases} k & k \leq t \\ t & k \geq t \end{cases}$$

In words, for all $k \leq t$, the vectors forming a Krylov subspace, i.e. $\mathbf{A}^i\mathbf{v}$, $i = 0, \ldots, k-1$ remian linearly independent, i.e. they form a basis, and hence $\mathcal{K}_{k-1}(\mathbf{A}, \mathbf{v}) \subsetneq \mathcal{K}_k(\mathbf{A}, \mathbf{v})$. After the cutoff, new vectors will be linearly dependent on previous and hence for $k > t$:

$$\mathcal{K}_{k-1}(\mathbf{A}, \mathbf{v}) = \mathcal{K}_k(\mathbf{A}, \mathbf{v})$$

Proof. Suppose t is the smallest integer such that $\mathbf{A}^t\mathbf{v} + \sum_{i=0}^{t-1} \alpha_i \mathbf{A}^i\mathbf{v} = \mathbf{0}$ for some α_i. In other words, the vectors $\mathbf{v}, \mathbf{A}\mathbf{v}, \mathbf{A}^2\mathbf{v}, \ldots, \mathbf{A}^t\mathbf{v}$ are linearly dependent. So we must have that $\dim(\mathcal{K}_{t+1}(\mathbf{A}, \mathbf{v})) \leq t$. It easily follows that:

$$\mathbf{A}^{t+1}\mathbf{v} + \sum_{i=0}^{t-1} \alpha_i \mathbf{A}^{i+1}\mathbf{v} = \mathbf{A}\left(\mathbf{A}^t\mathbf{v} + \sum_{i=0}^{t-1} \alpha_i \mathbf{A}^i\mathbf{v}\right) = \mathbf{0}$$

In other words, the vectors $\mathbf{A}\mathbf{v}, \ldots, \mathbf{A}^{t+1}\mathbf{v}$ will also be linearly dependent, which in turn implies that $\mathbf{v}, \mathbf{A}\mathbf{v}, \ldots, \mathbf{A}^{t+1}\mathbf{v}$ are linearly dependent. So we must have that $\dim(\mathcal{K}_{t+1}(\mathbf{A}, \mathbf{v})) \leq t$. We can continue this way, hence we have $\dim(\mathcal{K}_k(\mathbf{A}, \mathbf{v})) \leq t, \forall k \geq t$.

Since t is the smallest integer with such property, for any $k < t$, we have $\mathbf{A}^k\mathbf{v} + \sum_{i=0}^{k-1} \alpha_i \mathbf{A}^i\mathbf{v} \neq \mathbf{0}$ for **all** α_i, $i = 0, \ldots, k-1$. This implies that all the vectors $\mathbf{v}, \mathbf{A}\mathbf{v}, \mathbf{A}^2\mathbf{v}, \ldots, \mathbf{A}^k\mathbf{v}$ are linearly independent. Indeed, consider any α_i, $i = 0, \ldots, k$ with $\alpha_k \neq 0$. From the above assumption, we have:

$$\alpha_k \mathbf{A}^k\mathbf{v} + \sum_{i=0}^{k-1} \alpha_i \mathbf{A}^i\mathbf{v} = \mathbf{A}^k\mathbf{v} + \sum_{i=0}^{k-1} \frac{\alpha_i}{\alpha_k} \mathbf{A}^i\mathbf{v} \neq \mathbf{0}$$

Now consider the case where $\alpha_k = 0$ and suppose $\sum_{i=0}^{k-1} \alpha_i \mathbf{A}^i\mathbf{v} = \mathbf{0}$ for some α_i, $i = 0, \ldots, k-1$ that are not all zero. Let i be the largest index with non-zero α_i. We have $\mathbf{A}^i\mathbf{v} = \sum_{\ell=0}^{i} \left(\frac{\alpha_\ell}{\alpha_i}\right) \mathbf{A}^\ell\mathbf{v}$ which contradicts the assumption on t. So $\dim(\mathcal{K}_k(\mathbf{A}, \mathbf{v})) = k \; \forall k \leq t$. □

Remark 14.1

Why is the grade of $\mathbf{v}$ with respect to $\mathbf{A}$ important? Because $\mathbf{A}^{-1}\mathbf{v} \notin \mathcal{K}_k(\mathbf{A}, \mathbf{v})$ for $k < t$. This gives us the following corollary:

Corollary 14.1

$$t = \min\{k \mid \mathbf{A}^{-1}\mathbf{v} \in \mathcal{K}_k(\mathbf{A}, \mathbf{v})\}$$

Proof. Recall that an application of the Cayley-Hamilton theorem implied that:

$$\mathbf{A}^{-1}\mathbf{v} = \sum_{i=0}^{n-1} \alpha_i \mathbf{A}^i\mathbf{v}$$

But since $\mathcal{K}_k(\mathbf{A},\mathbf{v}) = \mathcal{K}_{K+1}(\mathbf{A},\mathbf{v})$, $k \geq t$, we can write:

$$\mathbf{A}^{-1}\mathbf{v} = \sum_{i=0}^{t-1} \beta_i \mathbf{A}^i \mathbf{v}$$

So $\mathbf{A}^{-1}\mathbf{v} \in \mathcal{K}_k(\mathbf{A},\mathbf{v})$, $k \geq t$. Now suppose this also holds for $k = t-1$, i.e. $\mathbf{A}^{-1}\mathbf{v} = \sum_{i=0}^{t-2} \gamma_i \mathbf{A}^i \mathbf{v}$. But then this gives $\mathbf{v} = \sum_{i=0}^{t-2} \gamma_i \mathbf{A}^{i+1}\mathbf{v}$. In other words, $\{\mathbf{v}, \mathbf{Av}, \ldots, \mathbf{A}^{t-1}\mathbf{v}\}$ are linearly dependent, which implies $\dim(\mathcal{K}_t(\mathbf{A},\mathbf{v})) < t$ which is a contradiction. □

So we have seen that $\mathbf{x}^\star \in \mathbf{x}_0 + \mathcal{K}_n(\mathbf{A},\mathbf{r}_0)$. But now after all of this we have a more general result.

Note 14.4

In other literature $\mathbf{r}_0 = \mathbf{b}$, $\mathbf{x}_0 = \mathbf{0}$.

Corollary 14.2

For any $\mathbf{x}_0$, we have

$$\mathbf{x}^\star \in \mathbf{x}_0 + \mathcal{K}_t(\mathbf{A},\mathbf{r}_0)$$

where $\mathbf{r}_0 = b - \mathbf{A}\mathbf{x}_0$ and t is the grade of $\mathbf{r}_0$ with respect to $\mathbf{A}$.

Even though it is not possible to give a precise formal definition of Krylov subspace solvers, we can describe standard versions as follows:

Algorithm 14.1: A standard Krylov subspace method for solving $\mathbf{Ax} = \mathbf{b}$

A standard Krylov subspace method is an iterative method which starting from some $\mathbf{x}_0$ generates an appropriate sequence of iterates:

$$\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A},\mathbf{r}_0)$$

until it finds $\mathbf{x}^\star$ in exactly t steps. (Take this statement with a grain of salt as it comes with many caveats! There are situations where such $\mathbf{x}_k$ for some k is not even defined) The iterates are chosen appropriately such that if we terminate early, we have still $\mathbf{x}_k \approx \mathbf{x}^\star$ in some sense.

Note 14.5

Not all Krylov subspaces are of this standard form. Some non-standard Krylov subspace methods build upon different Krylov subspaces, such as $\mathcal{K}_k(\mathbf{A},\mathbf{Av})$ and some others work with more than one Krylov subspace such as $\mathcal{K}_k(\mathbf{A}^*,\mathbf{w})$.

Remark 14.2

A bunch of remarks:

- In the late 1970s Nemirovskii and Judin showed that Krylov contain "optimal information" about a system of linear equations;
- Krylov subspace solvers differ among themselves in many aspects, including the subspace, the sense in which a chosen iterate is considered appropriate and the sense in which $\mathbf{x}_k \approx \mathbf{x}^\star$ is measured.
- Although Krylov subspace solvers are iterative, in exact arithmetic, they have finite termi-

nation property. In fact the original methods were designed to replace Gaussian elimination as alternative exact methods!

- In finite precision, finite termination property no longer holds.
- In practice, most often, these methods require some form of preconditioning to perform well (more on this later.)

15 Iterative methods: computing basis

15.1 Fundamentals of computing bases

How can we construct vectors that lie in any subspace? Using a basis for that subspace! $\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$.

What kind of basis of $\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$ should we pick? Is the most obvious choice of vectors, which is $\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \dots, \mathbf{A}^{k-1}\mathbf{r}_0\}$ any good?

Suppose that the grade of $\mathbf{r}_0$ with respect to $\mathbf{A}$ is n, in other words, $\dim(\mathcal{K}_n(\mathbf{A}, \mathbf{r}_0)) = n$. We kow that in this case the basis matrix

$$\mathbf{K} = \begin{pmatrix} \mathbf{r}_0 & \cdots & \mathbf{A}^{n-1}\mathbf{r}_0 \end{pmatrix} \in \mathbb{R}^{n \times n}$$

is invertible. So we have:

$$\begin{aligned} \mathbf{A}\mathbf{K} &= \begin{pmatrix} \mathbf{A}\mathbf{r}_0 & \cdots & \mathbf{A}^n\mathbf{r}_0 \end{pmatrix} \\ &= \mathbf{K} \begin{pmatrix} \mathbf{e}_2 & \mathbf{e}_3 & \cdots & \mathbf{K}^{-1}\mathbf{A}^n\mathbf{r}_0 \end{pmatrix} \end{aligned}$$

which implies

$$\mathbf{K}^{-1}\mathbf{A}\mathbf{K} = \underbrace{\overbrace{\begin{bmatrix} 0 & 0 & & 0 & 0 & c_1 \\ 1 & 0 & & 0 & 0 & c_2 \\ 0 & 1 & & 0 & 0 & c_3 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ 0 & 0 & & 1 & 0 & c_{n-1} \\ 0 & 0 & & 0 & 1 & c_n \end{bmatrix}}^{\text{Upper Hessenberg}}}_{c}$$

Even though $\mathbf{C}$ is very sparse and easy to work with, such a basis is practically very useless, for a multitude of reasons:

- We need n matrix-vector products, but we were hoping to get a good approximate in $k \ll n$
- We need to solve a linear system with $\mathbf{K}$, which can be very dense even if $\mathbf{A}$ is sparse. So this linear system might actually be harder than the original $\mathbf{A}\mathbf{x} = \mathbf{b}$:

$$\begin{aligned} \mathbf{K}^{-1}\mathbf{A}\mathbf{K} &= \mathbf{C} \\ \mathbf{A}\mathbf{x} &= \mathbf{b} \\ \mathbf{K}^{-1}\mathbf{A}\mathbf{K}\mathbf{K}^{-1}\mathbf{x} &= \mathbf{K}^{-1}\mathbf{b} \\ \mathbf{C}\mathbf{y} &= \mathbf{K}^{-1}\mathbf{b} \end{aligned}$$

- The matrix $\mathbf{K}$ is very ill-conditioned...why? (think power method). Recall that very parallel $\Longrightarrow$ very ill-conditioned. How close to singular says how ill-conditioned.

Suppose $\mathbf{K} = \mathbf{QR}$ has a $\mathbf{QR}$ factorisation. Note that $\mathrm{Range}(\mathbf{K}) = \mathrm{Range}(\mathbf{Q})$. Then

$$\mathbf{Q}^\top \mathbf{AQ} = \mathbf{RK}^{-1}\mathbf{AKR}^{-1} = \mathbf{RCR}^{-1} \triangleq \mathbf{H}.$$

Since $\mathbf{R}$ and $\mathbf{R}^{-1}$ are both upper triangular, and $\mathbf{C}$ is upper Hessenberg, it is easy to see that $\mathbf{H}$, though dense, is upper Hessenberg. For $\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$ with $k \ll n$, we search for a basis matrix $\mathbf{Q}_k \in \mathbb{R}^{n\times k}$, for which $\mathbf{Q}_k^\top \mathbf{AQ}_k \triangleq \mathbf{H}_k \in \mathbb{R}^{k\times k}$ is also upper Hessenberg. But not only is it upper Hessenberg, it is also only $k \times k$, which allows us to perform all computations using direct methods on this smaller method.

Note 15.1: Desired ingredients

We wish to find $\mathbf{Q}_k = \begin{pmatrix}\mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_k\end{pmatrix} \in \mathbb{R}^{n\times k}$ such that:

- $\{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_k\}$ is a basis for $\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$;
- $\{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_k\}$ is orthonormal;
- $\mathbf{Q}_k^\top \mathbf{AQ}_k \triangleq \mathbf{H}_k \in \mathbb{R}^{k\times k}$ is upper Hessenberg.

We want $\mathbf{Q}_k^\top \mathbf{AQ}_k = \mathbf{H}_k$ with $\mathbf{H}_k$ to be upper-Hessenberg. In general, $\mathbf{AQ}_k \neq \mathbf{Q}_k\mathbf{H}_k$ for any $k < n$. To obtain equality, we need to adjust with an error, i.e.

$$\mathbf{AQ}_k = \mathbf{Q}_k\mathbf{H}_k + \mathbf{E}_k, \quad \mathbf{E}_k \in \mathbb{R}^{n\times k}.$$

But necessarily, we must have $\mathbf{Q}_k^\top \mathbf{E}_k = \mathbf{0}$:

$$\mathbf{Q}^\top \mathbf{AQ}_k = \mathbf{Q}_k^\top \mathbf{Q}_k\mathbf{H}_k + \mathbf{Q}_k^\top \mathbf{E}_k$$

so let's aim to find the simplest $\mathbf{E}_k$ we can.

Suppose we have a vector $\mathbf{q}_{k+1}$ such that $\mathbf{q}_{k+1} \perp \mathbf{q}_i$, $i = 1, \dots, k$. Let

$$\mathbf{E} = \mathbf{q}_{k+1}\mathbf{h}_k^\top$$

for any $\mathbf{h}_k \in \mathbb{R}^k$. We obtain

$$\mathbf{Q}_k^\top \mathbf{E} = \mathbf{Q}_k^\top \left(\mathbf{q}_{k+1}\mathbf{h}_k^\top\right) = \left(\mathbf{Q}_k^\top \mathbf{q}_{k+1}\right)\mathbf{h}_k^\top = \mathbf{0}$$

This is true for any $\mathbf{h}_k \in \mathbb{R}^k$, so to make this even simpler and preserve as much sparsity as possible, we can set

$$\mathbf{h}_k = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ h_{k+1,k} \end{bmatrix}$$

So $\mathbf{AQ}_k = \mathbf{Q}_k\mathbf{H}_k + \mathbf{q}_{k+1}\mathbf{h}_k^\top$, and we can write

$$\mathbf{AQ}_k = \begin{pmatrix}\mathbf{Q}_k & \mathbf{q}_{k+1}\end{pmatrix}\begin{pmatrix}\mathbf{H}_k \\ \mathbf{h}_k^\top\end{pmatrix}, \quad \mathbf{h}_k = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ h_{k+1,k} \end{pmatrix}$$

Recall that $\mathbf{H}_k$ is upper Hessenberg, so

$$\mathbf{A}\begin{pmatrix}\mathbf{q}_1 & \cdots & \mathbf{q}_k\end{pmatrix} = \begin{pmatrix}\mathbf{q}_1 & \cdots & \mathbf{q}_{k+1}\end{pmatrix}\begin{pmatrix} h_{11} & \cdots & \cdots & \cdots & h_{1k} \\ h_{21} & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_{k,k-1} & h_{kk} \\ 0 & \cdots & 0 & 0 & h_{k+1,k} \end{pmatrix}$$

15.2 Arnoldi algorithm

The Arnoldi algorithm, at times referred to as the Arnoldi process, is based on a modified version of Gram-Schmidt orthogonalisation to find the desired $\mathbf{Q}_k$.

1. When $k = 1$, the only vector in $\mathcal{K}_1(\mathbf{A}, \mathbf{r}_0)$ is $\mathbf{r}_0$. So all we need to compute is $\mathbf{q}_1 = \mathbf{r}_0/\|\mathbf{r}_0\|$.
2. For $k = 2$, we require $\mathbf{A}\mathbf{q}_1 = h_{11}\mathbf{q}_1 + h_{12}\mathbf{q}_2$. This gives

$$\begin{aligned} h_{11} &= \langle \mathbf{q}_1, \mathbf{A}\mathbf{q}_1 \rangle \\ h_{21} &= \|\mathbf{A}\mathbf{q}_1 - h_{11}\mathbf{q}_1\| \\ \mathbf{q}_2 &= \frac{\mathbf{A}\mathbf{q}_1 - h_{11}\mathbf{q}_1}{h_{21}} \end{aligned}$$

3. For $k = j$, we get $\mathbf{A}\mathbf{q}_j = h_{1j}\mathbf{q}_1 + h_{2j}\mathbf{q}_2 + \ldots + h_{j+1,j}\mathbf{q}_{j+1}$. This gives

$$\begin{aligned} h_{ij} &= \langle \mathbf{q}_i, \mathbf{A}\mathbf{q}_j \rangle \\ h_{j+1,j} &= \left\| \mathbf{A}\mathbf{q}_j - \sum_{i=1}^{j} h_{ij}\mathbf{q}_i \right\| \\ \mathbf{q}_{j+1} &= \frac{\mathbf{A}\mathbf{q}_j - \sum_{i=1}^{j} h_{ij}\mathbf{q}_i}{h_{j+1,j}} \end{aligned}$$

This gives us what we wanted:

$$\mathbf{A}\begin{pmatrix}\mathbf{q}_1 & \cdots & \mathbf{q}_k\end{pmatrix} = \begin{pmatrix}\mathbf{q}_1 & \cdots & \mathbf{q}_{k+1}\end{pmatrix}\begin{pmatrix} h_{11} & \cdots & \cdots & \cdots & h_{1k} \\ h_{21} & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_{k,k-1} & h_{kk} \\ 0 & \cdots & 0 & 0 & h_{k+1,k} \end{pmatrix} \equiv \mathbf{A}\mathbf{Q}_k = \mathbf{Q}_{k+1}\mathbf{H}_{k+1,k}$$

15.3 Related results

Theorem 15.1

Assume the Arnoldi process does not terminate before k steps. Then the vectors $\{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k\}$ form an orthonormal basis for $\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$.

```
Input: $\mathbf{A}, \mathbf{r}_0, k$
$\mathbf{q}_1 = \mathbf{r}_0 / \|\mathbf{r}_0\|$
for $j = 1, \ldots, k$ do
    $\mathbf{z} = \mathbf{A}\mathbf{q}_j$
    for $i = 1, \ldots, j$ do
        $h_{ij} = \langle \mathbf{q}_i, \mathbf{z} \rangle$
        $\mathbf{z} = \mathbf{z} - h_{ij}\mathbf{q}_i$
    end for
    $h_{j+1,j} = \|\mathbf{z}\|$
    if $h_{j+1,j} = 0$ then
        STOP
    end if
    $\mathbf{q}_{j+1} = \mathbf{z} / h_{j+1,j}$
end for
```

Algorithm 4: Arnoldi process

Proof. First note that $\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0) = \mathcal{K}_k(\mathbf{A}, \mathbf{q}_1)$. Orthonormality is clear from the construction. For $j = 1$, we trivially have $\mathbf{q}_1 = p_0(\mathbf{A})\mathbf{q}_1$, where $p_{i-1}(t)$ $p_0(\mathbf{A}) = \mathbf{1}$. Suppose for all $i \leq j$ we have $\mathbf{q}_i = p_{i-1}(\mathbf{A})\mathbf{q}_1$, where $p_{i-1}(t)$ is a polynomial of degree $i-1$. For $j+1$, it follows that

$$h_{j+1,j}\mathbf{q}_{i+1} = \mathbf{A}\mathbf{q}_j - \sum_{i=1}^{j} h_{ij}\mathbf{q}_i = \mathbf{A}p_{j-1}(\mathbf{A})\mathbf{q}_1 - \sum_{i=1}^{j} h_{ij}p_{i-1}(\mathbf{A})\mathbf{q}_i$$

so we have $\mathbf{q}_{j+1} = p_j(\mathbf{A})\mathbf{q}_1$. In other words, each column of $\mathbf{Q}_k$ can be written as linear combination of vectors $\left\{\mathbf{q}_1, \mathbf{A}\mathbf{q}_1, \ldots, \mathbf{A}^{k-1}\mathbf{q}_1\right\}$, and since $\mathbf{q}_j$'s are independent, they must span the same space, i.e., $\mathcal{K}_k(\mathbf{A}, \mathbf{q}_1)$. □

Theorem 15.2

The Arnoldi process breaks down at step j, i.e. $h_{j+1,j} = 0$ if and only if the grade of $\mathbf{r}_0$ with respect to $\mathbf{A}$ is j, i.e. $t(\mathbf{r}_0, \mathbf{A}) = j$.

Proof. ($\Longleftarrow$) First, note that $t(\mathbf{r}_0, \mathbf{A}) = t(\mathbf{q}_1, \mathbf{A})$. Suppose $t(\mathbf{q}_1, \mathbf{A}) = j$ which implies that $\dim(\mathcal{K}_{j+1}(\mathbf{A}, \mathbf{q}_1)) = j$. Hence, we must have $\mathbf{A}\mathbf{q}_j - \sum_{i=1}^{j} h_{ij}\mathbf{q}_i = \mathbf{0}$. Otherwise $\mathbf{q}_{i+1}$ could be defined, which in turn implies that $\dim(\mathcal{K}_{j+1}(\mathbf{A}, \mathbf{q}_1)) = \dim(\{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_{j+1}\}) = j+1$, which is a contradiction. Hence we get $h_{j+1,j} = \left\|\mathbf{A}\mathbf{q}_j - \sum_{i=1}^{j} h_{ij}\mathbf{q}_i\right\| = 0$.

($\Longrightarrow$) To prove the converse, suppose $h_{j+1,j} = 0$, which means $\mathrm{A}q_j - \sum_{i=1}^{j} h_{ij}\mathbf{q}_i = 0$. Now since by previous theorem, Span $\{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_j\} = \mathcal{K}_j(\mathbf{A}, \mathbf{q}_1)$, we have $\mathbf{A}\mathbf{q}_j \in \mathcal{K}_j(\mathbf{A}, \mathbf{q}_1)$. But similar to the proof of the previous theorem, we can get that $\mathbf{A}\mathbf{q}_j = p_j(\mathbf{A})\mathbf{q}_1$, where $p_j(\mathbf{A})$ is a matrix polynomial of degree exactly j. This in particular implies $\mathbf{A}^j\mathbf{q}_1 \in \mathcal{K}_j(\mathbf{A}, \mathbf{q}_1)$. Hence, we must have $t(\mathbf{q}_1, \mathbf{A}) \leq j$. However, we cannot have $t(\mathbf{q}_1, \mathbf{A}) < j$, as otherwise by the first part of the proof, the algorithms would have already stopped. □

16 Iterative methods: optimality conditions

16.1 Projection methods

What have we done so far? We say that $\mathbf{x}^\star \in \mathbf{x}_0 + \mathcal{K}_t(\mathbf{A}, \mathbf{r}_0)$, where $t = t(\mathbf{r}_0, \mathbf{A})$ is the **grade** of $\mathbf{r}_0$ with respect to $\mathbf{A}$. To find $\mathbf{x}^\star$ or its approximation, Krylov subspace methods generate iterates in progressively nested affine spaces

$$\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0), \quad k = 1, \ldots.$$

We found a way to construct a basis for $\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$. So, now we can generate vectors inside these affine spaces. Now, among the infinitely many possibilities, what is the "best" $\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$? Once we've found what point is the "best", how can we efficiently generate it? Today's lecture will focus on the first point.

Note 16.1

For the remainder of this lecture, for simplicity, we assume $\mathbf{A} \in \mathbb{R}^{n\times n}$ and $\mathbf{b} \in \mathbb{R}^n$, i.e. **everything is real valued**. But the same methods apply to complex matrices – in fact, the only technicality is just computing complex derivatives here and there.

Some natural optimality conditions include (note all $\arg\min$ are subscript $\mathbf{x} \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$):

- $\mathbf{x}_k = \arg\min \|\mathbf{x} - \mathbf{x}^\star\|_2$: uh, we can't really do this;
- If $\mathbf{A} \succ \mathbf{0}$, $\mathbf{x}_k = \arg\min \|\mathbf{x}-\mathbf{x}^\star\|_\mathbf{A} = \arg\min \frac{1}{2}\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle - \langle \mathbf{b}, \mathbf{x}\rangle$: conjugate gradient or CG method (the second equality is proven in assignment 3);
- $\mathbf{x}_k = \arg\min \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2$: MINRES for symmetric $\mathbf{A}$, GMRES for nonsymmetric $\mathbf{A}$.
- Find $\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$ such that $\mathbf{r}_k \perp \mathcal{L}_k$, where $\mathcal{K}_k$ is some k-dimensional subspace: $\mathbf{A}$ is PD, means CG. $\mathbf{A}$ is symmetric, means SYMMLQ. $\mathbf{A}$ is nonsymmetric, a variant of GMRES.

In fact, the very last optimality condition can be used to unify many aspects of the analysis and to present a general framework to represent all Krylov methods (and many iterative methods in scientific computing) as **projection methods**.

Note 16.2: Intuition behind projection methods

Once a basis has been constructed, there are k degrees of freedom for picking a point in k-dimensional affine subspace. So, to uniquely determine a point, we need in general k constraints. A typical way is to impose that the residual is orthogonal to k linearly independent vectors, e.g. basis of another k-dimensional subspace.

Definition 16.1: Projection method

A projection method consists of a search subspace $\mathcal{K}_k$ with $\dim(\mathcal{K}_k) = k$, a constraint subspace $\mathcal{L}_k$ with $\dim(\mathcal{L}_k) = k$ and the Petrov-Galerkin conditions, which are to find some $\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}_k$ such that $\mathbf{r}_k \perp \mathcal{L}_k$. A projection method is orthogonal if we wish to find $\mathcal{L}_k = \mathcal{K}_k$, and oblique if we wish to find $\mathcal{L}_k = \mathbf{A}\mathcal{K}_k$. More formally, let $\mathbf{x}_k = \mathbf{x}_0 + \mathbf{z}_k, \mathbf{z}_k \in \mathcal{K}_k$. Then the

Petrov-Galerkin conditions imply $\mathbf{r}_0 - \mathbf{A}\mathbf{z}_k \perp \mathcal{L}_k$. So the projection method is defined as:

$$\text{find } \mathbf{x}_k = \mathbf{x}_0 + \mathbf{z}_k \text{ such that } \begin{cases} \mathbf{z}_k \in \mathcal{K}_k \\ \langle \mathbf{r}_0 - \mathbf{A}\mathbf{z}_k, \mathbf{w} \rangle = 0, \quad \forall \mathbf{w} \in \mathcal{L}_k \end{cases}$$

How can we impose the conditions

$$\mathbf{z}_k \in \mathcal{K}_k, \quad \langle \mathbf{r}_0 - \mathbf{A}\mathbf{z}_k, \mathbf{w} \rangle = 0, \quad \forall \mathbf{w} \in \mathcal{L}_k?$$

We form an appropriate basis for $\mathcal{K}_k$ and $\mathcal{L}_k$. Suppose we have found the basis for $\mathcal{K}_k$ and $\mathcal{L}_k$, i.e.

$$\begin{aligned} \mathcal{K}_k &= \mathrm{Span}\,\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\} \\ \mathcal{L}_k &= \mathrm{Span}\,\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_k\} \end{aligned}$$

Define the following matrices:

$$\begin{aligned} \mathbf{K}_k &\triangleq \begin{pmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_k \end{pmatrix} \in \mathbb{R}^{n \times k} \\ \mathbf{L}_k &\triangleq \begin{pmatrix} \mathbf{w}_1 & \mathbf{w}_2 & \cdots & \mathbf{w}_k \end{pmatrix} \in \mathbb{R}^{n \times k} \end{aligned}$$

Now we can easily enforce the constraints in a different manner:

$$\begin{aligned} \mathbf{z}_k \in \mathcal{K}_k &\equiv \mathbf{z}_k = \mathbf{K}_k \mathbf{y}_k, \ \mathbf{y}_k \in \mathbb{R}^k \\ \mathbf{r}_0 - \mathbf{A}\mathbf{z}_k \perp \mathcal{L}_k &\equiv \mathbf{L}_k^\top (\mathbf{r}_0 - \mathbf{A}\mathbf{K}_k \mathbf{y}_k) = \mathbf{0} \end{aligned}$$

Furthermore, if $\mathbf{L}_k^\top \mathbf{A}\mathbf{K}_k$ is non-singular, we can readily express $\mathbf{x}_k$ as

$$\mathbf{x}_k = \mathbf{x}_0 + \mathbf{K}_k \left(\mathbf{L}_k^\top \mathbf{A}\mathbf{K}_k \right)^{-1} \mathbf{L}_k^\top \mathbf{r}_0,$$

which gives rise to a prototype of basic projection methods.

```
for $k = 1, \dots$ until convergence do
    Select $\mathcal{K}_k$ and $\mathcal{L}_k$
    For the basis matrices $\mathbf{K}_k$ and $\mathbf{L}_k$
    Solve the linear system $(\mathbf{L}_k^\top \mathbf{A}\mathbf{K}_k)\mathbf{y}_k = \mathbf{L}_k^\top \mathbf{r}_0$
    $\mathbf{x}_k = \mathbf{x}_0 + \mathbf{K}_k \mathbf{y}_k$
end for
```

Algorithm 5: Prototype projection method

In most algorithms, the matrix $\mathbf{L}_k^\top \mathbf{A}\mathbf{K}_k$ need not be formed explicitly, as it becomes available as a by-product of some other part of the algorithm, for example, when we find $\mathbf{K}_k$ and $\mathbf{L}_k$. The above method is only well-defined when the matrix $\mathbf{L}_k^\top \mathbf{A}\mathbf{K}_k$ is non-singular, which can be violated even though $\mathbf{A}$ is non-singular, for example

$$\mathbf{A} = \begin{pmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{I} & \mathbf{I} \end{pmatrix}, \quad \mathbf{K} = \mathbf{L} = \begin{pmatrix} \mathbf{I} \\ \mathbf{0} \end{pmatrix} \implies \mathbf{L}^\top \mathbf{A}\mathbf{K} = \mathbf{0}$$

16.2 Projection method theorems

Proposition 16.1

The matrix $\mathbf{L}^\top\mathbf{A}\mathbf{K}$ is non-singular if either:

1. $\mathbf{A} \succ \mathbf{0}$ and $\mathcal{L} = \mathcal{K}$ or;
2. $\det(\mathbf{A}) \neq 0$ and $\mathcal{L} = \mathbf{A}\mathcal{K}$.

Proof.

1. Since $\mathcal{L} = \mathcal{K}$, any basis of $\mathcal{L}$ is also a basis for $\mathcal{K}$. In fact, we can write $\mathbf{L} = \mathbf{K}\mathbf{B}$ where $\mathbf{B} \in \mathbb{R}^{k\times k}$ is non-singular. Now, we have
$$\mathbf{L}^\top\mathbf{A}\mathbf{K} = \mathbf{B}^\top\mathbf{K}^\top\mathbf{A}\mathbf{K}$$
and since $\mathbf{A} \succ 0$, we have $\mathbf{K}^\top\mathbf{A}\mathbf{K} \succ \mathbf{0}$ and hence the entire product is non-singular.
2. Since $\mathcal{L} = \mathbf{A}\mathcal{K}$, we can write $\mathbf{L} = \mathbf{A}\mathbf{K}\mathbf{B}$ where $\mathbf{B} \in \mathbb{R}^{k\times k}$ is non-singular. Now we have $\mathbf{L}^\top\mathbf{A}\mathbf{K} = \mathbf{B}^\top\mathbf{K}^\top\mathbf{A}^\top\mathbf{A}\mathbf{K}$ and since $\mathbf{A}$ is non-singular, we have $\mathbf{A}^\top\mathbf{A} \succ \mathbf{0}$, which as above, implies that the entire product is non-singular.

□

How do all of the previously mentioned optimality conditions tie in with the projection framework?

Theorem 16.1

The case where $\mathbf{A} \succ \mathbf{0}$ and $\mathcal{L}_k = \mathcal{K}_k$ is equivalent to

$$\mathbf{x}_k = \underset{\mathbf{x}\in\mathbf{x}_0+\mathcal{K}_k}{\arg\min} \|\mathbf{x}-\mathbf{x}^\star\|_{\mathbf{A}} = \underset{\mathbf{x}\in\mathbf{x}_0+\mathcal{K}_k}{\arg\min} \frac{1}{2}\langle\mathbf{x},\mathbf{A}\mathbf{x}\rangle - \langle\mathbf{b},\mathbf{x}\rangle$$

Proof. Let $\mathbf{x} = \mathbf{x}_0 + \mathbf{K}_k\mathbf{y}$ for $\mathbf{y} \in \mathbb{R}^k$, where $\mathbf{K}_k$ is a basis matrix for $\mathcal{K}_k$. So:

$$\begin{aligned}\mathbf{x}_k &= \underset{\mathbf{x}\in\mathbf{x}_0+\mathcal{K}_k}{\arg\min} \frac{1}{2}\langle\mathbf{x},\mathbf{A}\mathbf{x}\rangle - \langle\mathbf{b},\mathbf{x}\rangle \\ &= \underset{\mathbf{y}\in\mathbb{R}^k}{\arg\min} \frac{1}{2}\left\langle\mathbf{x}_0+\mathbf{K}_k\mathbf{y},\mathbf{A}\left(\mathbf{x}_0+\mathbf{K}_k\mathbf{y}\right)\right\rangle - \left\langle\mathbf{b},\mathbf{x}_0+\mathbf{K}_k\mathbf{y}\right\rangle \\ &= \underset{\mathbf{y}\in\mathbb{R}^k}{\arg\min} \frac{1}{2}\left\langle\mathbf{K}_k\mathbf{y},\mathbf{A}\mathbf{K}_k\mathbf{y}\right\rangle - \left\langle\mathbf{b}-\mathbf{A}\mathbf{x}_0,\mathbf{K}\mathbf{y}\right\rangle\end{aligned}$$

Since $\mathbf{A} \succ \mathbf{0}$, it is necessary and sufficient for the optimal $\mathbf{y}_k$ to satisfy the equality $\mathbf{K}_k^\top\mathbf{A}\mathbf{K}_k\mathbf{y}_k - \mathbf{K}_k^\top\left(\mathbf{b}-\mathbf{A}\mathbf{x}_0\right) = \mathbf{0}$, which is the same as $\mathbf{K}_k^\top\left(\mathbf{A}\mathbf{x}_k-\mathbf{b}\right) = \mathbf{0}$, i.e., $\mathbf{r}_k \perp \mathcal{K}_k$ □

Theorem 16.2

The case where $\det\left(\mathbf{A}\right) \neq 0$ and $\mathcal{L}_k = \mathbf{A}\mathcal{K}_k$ is equivalent to

$$\mathbf{x}_k = \underset{\mathbf{x}\in\mathbf{x}_0+\mathcal{K}_k}{\arg\min} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2$$

Proof. Similarly as above, let $\mathbf{x} = \mathbf{x}_0 + \mathbf{K}_k\mathbf{y}$ for $\mathbf{y} \in \mathbb{R}^k$, where $\mathbf{K}_k$ is a basis matrix for $\mathcal{K}_k$:

$$\begin{aligned} \mathbf{x}_k &= \underset{\mathbf{x}\in\mathbf{x}_0+\mathcal{K}_k}{\arg\min} \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 \\ &= \underset{\mathbf{y}\in\mathbb{R}^k}{\arg\min} \frac{1}{2}\left\|\mathbf{A}\left(\mathbf{x}_0 + \mathbf{K}_k\mathbf{y}\right) - \mathbf{b}\right\|^2 \end{aligned}$$

Now, since A is non-singular, it is necessary and sufficient for the optimal y_k to satisfy

$$\mathbf{K}_k^\top\mathbf{A}^\top\mathbf{A}\mathbf{K}\mathbf{y}_k = \mathbf{K}_k^\top\mathbf{A}^\top\left(\mathbf{b} - \mathbf{A}\mathbf{x}_0\right)$$

which is the same as $\mathbf{K}_k^\top\mathbf{A}^\top\left(\mathbf{A}\mathbf{x}_k - \mathbf{b}\right) = \mathbf{0}$, i.e., $\mathbf{r}_k \perp \mathbf{A}\mathcal{K}_k$ □

The sense in which Krylov space methods decrease the "distance" to the real solution $\mathbf{x}^\star$ is difference.

Example 16.1

Since $\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$ are strictly nested, then

1. Conjugate gradient is monotonic in the "energy" norm:

$$\mathbf{x}_k = \underset{\mathbf{x}\in\mathbf{x}_0+\mathcal{K}_k(\mathbf{A},\mathbf{r}_0)}{\arg\min} \|\mathbf{x} - \mathbf{x}^\star\|_{\mathbf{A}} \implies \|\mathbf{x}_k - \mathbf{x}^\star\|_{\mathbf{A}} < \|\mathbf{x}_{k-1} - \mathbf{x}^\star\|_{\mathbf{A}}.$$

2. MINRES and GMRES are monotonic in norm of the residuals:

$$\mathbf{x}_k = \underset{\mathbf{x}\in\mathbf{x}_0=\mathcal{K}_k(\mathbf{A},\mathbf{r}_0)}{\arg\min} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2 \implies \|\mathbf{r}_k\| < \|\mathbf{r}_{k-1}\|.$$

17 Iterative methods: examples (MINRES & GMRES)

Recall our projection framework is

$$\mathbf{x}_k = \mathbf{x}_0 + \mathbf{z}_k, \quad \underbrace{\mathbf{z}_k \in \mathcal{K}_k,}_{1} \quad \underbrace{\mathbf{r}_0 - \mathbf{A}\mathbf{z}_k \perp \mathcal{L}_k}_{2}$$

If $\mathcal{L}_k = \mathcal{K}_k = \text{Range}(\mathbf{Q}_k)$, then

$$\mathbf{x}_k = x_0 + \mathbf{Q}_k\mathbf{y}$$

and

$$\mathbf{Q}_k^\top\left(\mathbf{r}_0 - \mathbf{A}\mathbf{Q}_k\mathbf{y}\right) = \mathbf{0} \iff \mathbf{Q}_k^\top\mathbf{A}\mathbf{Q}_k\mathbf{y} = \mathbf{Q}_k^\top\mathbf{r}_0 \iff \underbrace{\mathbf{H}_k\mathbf{y} = \|\mathbf{r}_0\|\,\mathbf{e}_1}_{2}.$$

If $\mathcal{L}_k = \mathbf{A}\cdot\mathcal{K}_k = \mathbf{A}\cdot\text{Range}(\mathbf{Q}_k)$, then $\mathbf{x}_k = \mathbf{x}_0 + \mathbf{Q}_k\mathbf{y}$ (1) and

$$\begin{aligned} \mathbf{Q}_k^\top\mathbf{A}^\top\left(\mathbf{r}_0 - \mathbf{A}\mathbf{Q}_k\mathbf{y}\right) = 0 &\iff \mathbf{Q}_k^\top\mathbf{A}^\top\mathbf{A}\mathbf{Q}_k\mathbf{y} = \mathbf{Q}_k^\top\mathbf{A}^\top\mathbf{r}_0 \\ &\iff \mathbf{y} = \underset{\mathbf{y}\in\mathbb{R}^k}{\arg\min}\,\frac{1}{2}\|\mathbf{A}\mathbf{Q}_k\mathbf{y} - \mathbf{r}_0\|^2 = \underset{\mathbf{y}\in\mathbb{R}^k}{\arg\min}\,\frac{1}{2}\|\mathbf{Q}_{k+1}\mathbf{H}_{k+1,k}\mathbf{y} - \mathbf{r}_0\|^2 \\ &\iff \mathbf{y} = \underset{\mathbf{y}\in\mathbb{R}^k}{\arg\min}\,\frac{1}{2}\|\mathbf{H}_{k+1,k}\mathbf{y} - \mathbf{Q}_{k+1}^\top\mathbf{r}_0\|^2 \end{aligned}$$

$$\iff \underbrace{\mathbf{y} = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \frac{1}{2}\|\mathbf{H}_{k+1,k}\mathbf{y} - \|\mathbf{r}_0\|\mathbf{e}_1\|^2}_{2}$$

When **A** is symmetric, things get significantly more easier on so many levels! This allows us to introduce the Lanczos process.

17.1 Lanczos process

When **A** is symmetric, then so is $\mathbf{Q}_k^\top \mathbf{A}\mathbf{Q}_k$, which means it is actually tridiagonal (it is often denoted as $\mathbf{T}_k$ instead of $\mathbf{H}_k$). In this case, the Arnoldi process becomes the **Lanczos process**. Specifically, we obtain

$$A\begin{pmatrix}\mathbf{q}_1 & \cdots & \mathbf{q}_k\end{pmatrix} = \begin{pmatrix}\mathbf{q}_1 & \cdots & \mathbf{q}_k & \mathbf{q}_{k+1}\end{pmatrix}\begin{pmatrix}\gamma_1 & \beta_1 & 0 & \cdots & 0\\ \beta_1 & \gamma_2 & \beta_2 & \ddots & \vdots\\ 0 & \ddots & \ddots & \ddots & 0\\ \vdots & \ddots & \ddots & \ddots & \beta_{k-1}\\ 0 & \cdots & 0 & \beta_{k-1} & \gamma_k\\ 0 & \cdots & 0 & 0 & \beta_k\end{pmatrix}$$

This can be compactly written as $\mathbf{A}\mathbf{Q}_k = \mathbf{Q}_{k+1}\mathbf{T}_{k+1,k}$. It can be verified that $\mathbf{Q}_k^\top \mathbf{A}\mathbf{Q}_k = \mathbf{T}_k$ is tridiagonal.

Furthermore, if $\mathcal{L}_k = \mathcal{K}_k = \text{Range}(\mathbf{Q}_k)$, then $\mathbf{x}_k = \mathbf{x}_0 = \mathbf{Q}_k\mathbf{y}$ (1) and

$$\underbrace{\mathbf{T}_k\mathbf{y} = \|\mathbf{r}_0\|\mathbf{e}_1}_{2}.$$

If $\mathcal{L}_k = \mathbf{A}\cdot\mathcal{K}_k = \mathbf{A}\cdot\text{Range}(\mathbf{Q}_k)$, then $\mathbf{x}_k = \mathbf{x}_0 + \mathbf{Q}_k\mathbf{y}$ (1) and

$$\mathbf{y} = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \frac{1}{2}\|\mathbf{T}_{k+1,k}\mathbf{y} - \|\mathbf{r}_0\|\mathbf{e}_1\|^2$$

Recall the breakdown of Arnoldi (or equivalently Lanczos for symmetric matrices). We now show that break-down is actually a **good thing**, and implies that we have found the exact solution! (This is why it is often called "lucky break-down").

Proposition 17.1

If Arnoldi (or Lanczos) Process breaks down at step $t = t(\mathbf{A}, \mathbf{r}_0)$, then $\mathbf{x}_t$ from any projection method onto $\mathcal{K}_t(\mathbf{A}, \mathbf{r}_0)$ or $\mathbf{A}\cdot\mathcal{K}_t(\mathbf{A}, \mathbf{r}_0)$ would be exact.

Proof. We show the proof for Arnoldi, as that for Lanczos is identical. First consider the projection method onto $\mathcal{K}_t(\mathbf{A}, \mathbf{r}_0)$, i.e., $\mathcal{L}_t = \mathcal{K}_t$. Recall again that $\mathbf{x}_t = \mathbf{x}_0 + \mathbf{Q}_t\mathbf{y}$, Range $(\mathbf{Q}_t) = \mathcal{K}_t(\mathbf{A}, \mathbf{r}_0)$, $\mathbf{y} \in \mathbb{R}^t$. Since $\mathbf{r}_0 \in \text{Range}(\mathbf{Q}_t)$, we have $\mathbf{Q}_t\mathbf{Q}_t^\top\mathbf{r}_0 = \mathbf{r}_0$. Also, since $h_{t+1,t} = 0$, we have $\mathbf{A}\mathbf{Q}_t = \mathbf{Q}_{t+1}\mathbf{H}_{t+1,t} = \mathbf{Q}_t\mathbf{H}_t$. It follows that $\mathbf{Q}_t\mathbf{Q}_t^\top(\mathbf{A}\mathbf{Q}_t\mathbf{y}) = \mathbf{Q}_t\mathbf{H}_t\mathbf{y} = \mathbf{A}\mathbf{Q}_t\mathbf{y}$. Hence, we have:

$$\begin{aligned}\mathbf{0} = \mathbf{Q}_t^\top(\mathbf{r}_0 - \mathbf{A}\mathbf{Q}_t\mathbf{y}) \iff \mathbf{0} &= \mathbf{Q}_t\mathbf{Q}_t^\top(\mathbf{r}_0 - \mathbf{A}\mathbf{Q}_t\mathbf{y})\\ &= \mathbf{r}_0 - \mathbf{A}\mathbf{Q}_t\mathbf{y}\end{aligned}$$

$$\begin{aligned} &= \mathbf{b} - \mathbf{A}\mathbf{x}_0 - \mathbf{A}\mathbf{Q}_t\mathbf{y} \\ &= \mathbf{b} - \mathbf{A}\mathbf{x}_t \end{aligned}$$

In other words, $\mathbf{x}_t$ is the exact solution. Now consider the projection method onto $\mathbf{A} \cdot \mathcal{K}_t(\mathbf{A}, \mathbf{r}_0)$. Just as before, we get:

$$\begin{aligned} \mathbf{y} &= \arg\min_{\mathbf{y}\in\mathbb{R}^t} \frac{1}{2}\left\|\mathbf{H}_{t+1,t}\mathbf{y} - \mathbf{Q}_{t+1}^\top \mathbf{r}_0\right\|^2 \\ &= \arg\min_{\mathbf{y}\in\mathbb{R}^t} \frac{1}{2}\left\|\mathbf{H}_t\mathbf{y} - \mathbf{Q}_t^\top \mathbf{r}_0\right\|^2 \\ &= \mathbf{H}_t^{-1}\mathbf{Q}_t^\top \mathbf{r}_0 \end{aligned}$$

where the last equality follows since H_t is an invertible square matrix. Again, noting that $\mathbf{r}_0 \in$ Range $(\mathbf{Q}_t)$, we have

$$\begin{aligned} \mathbf{H}_t\mathbf{y} = \mathbf{Q}_t^\top\mathbf{r}_0 \iff \mathbf{Q}_t\mathbf{H}_t\mathbf{y} = \mathbf{Q}_t\mathbf{Q}_t^\top\mathbf{r}_0 = \mathbf{r}_0 \iff \mathbf{A}\mathbf{Q}_t\mathbf{y} = \mathbf{r}_0 \\ \iff \mathbf{A}(\mathbf{x}_0 + \mathbf{Q}_t\mathbf{y}) = \mathbf{b} \iff \mathbf{A}\mathbf{x}_t = \mathbf{b} \end{aligned}$$

In other words, $\mathbf{x}_t$ is the exact solution. □

17.2 GMRES and MINRES

Consider the constraint subspace $\mathcal{L}_k = \mathbf{A} \cdot \mathcal{K}_k$. There are two cases:

- $\boxed{\mathbf{A} \neq \mathbf{A}^\top}$: this induces the generalised minimum residual (GMRES) method, and the subproblems

  $$\|\mathbf{H}_{k+1,k}\mathbf{y} - \| \, \mathbf{r}_0 \, \|\mathbf{e}_1\|$$

  are solved using the reduced QR factorisation of $\mathbf{H}_{k+1,k} = \mathbf{U}_{k+1,k}\mathbf{R}_k$ as

  $$\mathbf{y} = \|\mathbf{r}_0\| \, \mathbf{R}_k^{-1}\mathbf{U}_{k+1,k}^\top\mathbf{e}_1.$$

  Additionally, the residual can be computed as

  $$\|\mathbf{b} - \mathbf{A}\mathbf{x}_k\| = \|\mathbf{r}_0\| \sqrt{\left(1 - \left\|\mathbf{U}_{k+1,k}^\top\mathbf{e}_1\right\|^2\right)},$$

  without the need to explicitly compute $\mathbf{A}\mathbf{x}_k$.
- $\boxed{\mathbf{A} = \mathbf{A}^\top}$ (symmetric): induces minimum residual (MINRES) method, and

  $$\mathbf{T}_{k+1,}\,\mathbf{y} - \|\mathbf{r}_0\|\mathbf{e}_1\|$$

  is solved using reduced QR factorisation, similar to above.

In this course, we will only really examine GMRES in detail.

18 Iterative methods: examples (conjugate gradient)

Consider $\mathcal{L}_k = \mathcal{K}_k$. There are two cases:

- $\mathbf{A} \neq \mathbf{A}^\top$: requires using full orthogonalisation method (FOM), not covered in this course.
- $\mathbf{A} \succ \mathbf{0}$: conjugate gradient (CG) for positive definite matrices, and $\mathbf{T}_k\mathbf{y}_k = \|\mathbf{r}_0\|\mathbf{e}_1$ is solved using Cholesky factorisation.

```
$\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0, \mathbf{q}_1 = \mathbf{r}_0/\|\mathbf{r}_0\|, \mathbf{H}_{1,0} = \emptyset, \mathbf{Q}_1 = \begin{pmatrix} q_1 \end{pmatrix}$
for $k = 1, 2, \ldots$ until $\|\mathbf{r}_k\| \leq \tau$ do
    Run one step of Arnoldi process, with $\{\mathbf{q}_i\}_{i=1}^k$ and $\mathbf{A}\mathbf{q}_k$ to get $\mathbf{q}_{k+1}$
    Update $\mathbf{H}_{k,k-1}$ using $\{h_{ik}\}_{i=1}^{k+1}$ from the Arnoldi process to obtain $\mathbf{H}_{k+1,k}$
    $\mathbf{Q}_{k+1} = \begin{pmatrix} \mathbf{Q}_k & \mathbf{q}_{k+1} \end{pmatrix}$
    Form QR factorisation of $\mathbf{H}_{k+1,k} = \mathbf{U}_{k+1,k}\mathbf{R}_k$
    $\mathbf{y}_k = \|\mathbf{r}_0\|\mathbf{R}_k^{-1}\mathbf{U}_{k+1}^\top\mathbf{e}_1$
    $\mathbf{x}_k = \mathbf{x}_0 + \mathbf{Q}_k\mathbf{y}_k$
    $\|\mathbf{r}_k\| = \|\mathbf{r}_0\|\sqrt{\left(1 - \|\mathbf{U}_{k+1,k}^\top\mathbf{e}_1\|^2\right)}$
end for
```

Algorithm 6: Overview of GMRES

18.1 Derivation

Let's go into the method. Consider the Cholesky factorisation of $\mathbf{T}_k = \mathbf{L}_k\mathbf{D}_k\mathbf{L}_k^\top$, where $\mathbf{L}_k$ is a unit lower bi-diagonal and $\mathbf{D}_k$ is diagonal:

$$\mathbf{L}_k = \begin{pmatrix} 1 & & & \\ \ell_1 & \ddots & & \\ & \ddots & \ddots & \\ & & \ell_{k-1} & 1 \end{pmatrix}, \quad \mathbf{D}_k = \begin{pmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_k \end{pmatrix}.$$

We have

$$\begin{aligned}
\mathbf{x}_k &= \mathbf{x}_0 + \mathbf{Q}_k\mathbf{y}_k = \mathbf{x}_0 + \|\mathbf{r}_0\|\,\mathbf{Q}_k\mathbf{T}_k^{-1}\mathbf{e}_1 \\
\mathbf{x}_k &= \mathbf{x}_0 + \|\mathbf{r}_0\|\,\mathbf{Q}_k\left(\mathbf{L}_k\mathbf{D}_k\mathbf{L}_k^\top\right)^{-1}\mathbf{e}_1 \\
\mathbf{x}_k &= \mathbf{x}_0 + \underbrace{\left(\mathbf{Q}_k\mathbf{L}_k^{-\top}\right)}_{\tilde{\mathbf{P}}_k}\underbrace{\left(\|\mathbf{r}_0\|\,\mathbf{D}_k^{-1}\mathbf{L}_k^{-1}\mathbf{e}_1\right)}_{\tilde{\mathbf{y}}_k} \\
\mathbf{x}_k &= \mathbf{x}_0 + \tilde{\mathbf{P}}_k\tilde{\mathbf{y}}_k
\end{aligned}$$

Since $\mathbf{L}_k$ is unit lower bi-diagonal, denoting $\mathbf{a} \triangleq \ell_{k-1}\hat{\mathbf{e}}_{k-1}$, we can write it as

$$\mathbf{L}_k = \begin{pmatrix} L_{k-1} & \mathbf{0} \\ \mathbf{a}^\top & 1 \end{pmatrix} \Longrightarrow \mathbf{L}_k^{-1} = \begin{pmatrix} L_{k-1}^{-1} & \mathbf{0} \\ \star & 1 \end{pmatrix},$$

where "$\star$" is some appropriate row vector, and we don't care about its exact value. We obtain a simple recursion for $\tilde{\mathbf{y}}_k$ as

$$\begin{aligned}
\tilde{\mathbf{y}}_k &= \|\mathbf{r}_0\|\,\mathbf{D}_k^{-1}\mathbf{L}_k^{-1}\mathbf{e}_1^k \\
&= \|\mathbf{r}_0\|\begin{pmatrix} \mathbf{D}_k^{-1} & \mathbf{0} \\ \mathbf{0}^\top & \mathbf{d}_k^{-1} \end{pmatrix}\begin{pmatrix} \mathbf{L}_{k-1}^{-1} & \mathbf{0} \\ \star & 1 \end{pmatrix}\mathbf{e}_1^k \\
&= \|\mathbf{r}_0\|\begin{pmatrix} \mathbf{D}_{k-1}^{-1}\mathbf{L}_{k-1}^{-1} & \mathbf{0} \\ \star & \mathbf{d}_k^{-1} \end{pmatrix}\begin{pmatrix} \mathbf{e}_1^{k-1} \\ 0 \end{pmatrix} \\
&= \begin{pmatrix} \tilde{\mathbf{y}}_{k-1} \\ \eta_k \end{pmatrix}
\end{aligned}$$

We also get a simple recursion for $\hat{\mathbf{P}}_k$ as

$$\begin{aligned}\tilde{\mathbf{P}}_k &= \mathbf{Q}_k\mathbf{L}_k^{-\top}\\ &= (\mathbf{Q}_{k-1} \quad \mathbf{q}_k)\begin{pmatrix} \mathbf{L}_k^{-\top} & \star \\ \mathbf{0}^\top & 1\end{pmatrix}\\ &= \begin{pmatrix} \mathbf{Q}_{k-1}\mathbf{L}_{k-1}^{-\top} & \tilde{\mathbf{p}}_k\end{pmatrix}\\ &= \begin{pmatrix} \tilde{\mathbf{P}}_{k-1} & \tilde{\mathbf{p}}_k\end{pmatrix}\end{aligned}$$

Once again, we don't care about $\star$. So, we have:

$$\begin{aligned}\mathbf{x}_k &= \mathbf{x}_0 + \tilde{\mathbf{P}}_k\tilde{\mathbf{y}}_k\\ &= \mathbf{x}_0 = \begin{pmatrix}\tilde{\mathbf{P}}_{k-1} & \tilde{p}_k\end{pmatrix}\begin{pmatrix}\tilde{\mathbf{y}}_{k-1}\\ \eta_k\end{pmatrix}\\ &= \mathbf{x}_0 + \tilde{\mathbf{P}}_{k-1}\tilde{\mathbf{y}}_{k-1} + \eta_k\tilde{\mathbf{p}}_k\\ &= \mathbf{x}_{k-1} + \eta_k\tilde{\mathbf{p}}_k\end{aligned}$$

Also, from $\tilde{\mathbf{P}}_k\mathbf{L}_k^\top = \mathbf{Q}_k$, we get $\tilde{\mathbf{p}}_k = \mathbf{q}_k - \ell_{k-1}\tilde{\mathbf{p}}_{k-1}$:

$$\begin{pmatrix}\tilde{\mathbf{p}}_1 & \tilde{\mathbf{p}}_2 & \cdots & \tilde{\mathbf{p}}_k\end{pmatrix}\begin{pmatrix}1 & \ell_1 & & \\ & \ddots & \ddots & \\ & & \ddots & \ell_{k-1}\\ & & & 1\end{pmatrix}$$

Overall, we have short recursions as

$$\begin{aligned}\mathbf{x}_k &= \mathbf{x}_{k-1} + \eta_k\tilde{\mathbf{p}}_k\\ \mathbf{r}_k &= \mathbf{r}_{k-1} - \eta_k\mathbf{A}\tilde{\mathbf{p}}_k\\ \tilde{\mathbf{p}}_k &= \mathbf{q}_k - \ell_{k-1}\tilde{\mathbf{p}}_{k-1}.\end{aligned}$$

It seems like we need to store 4 vectors $(\mathbf{q}_k, \tilde{\mathbf{p}}_{k-1}, \mathbf{x}_{k_1}, \mathbf{r}_{k-1})$, but scientific computing researchers are incredibly stingy and made it better somehow.

Recall $\mathbf{A}\mathbf{Q}_k = \mathbf{Q}_k\mathbf{T}_k + \mathbf{q}_{k+1}\mathbf{t}_k^\top$, where

$$\mathbf{t}_k \triangleq \begin{pmatrix}0 & 0 & \cdots & 0 & \beta_k\end{pmatrix}^\top \in \mathbb{R}^k.$$

This gives us

$$\mathbf{r}_k = \mathbf{r}_0 - \mathbf{A}\mathbf{Q}_k\mathbf{y}_k = \underbrace{\mathbf{r}_0 - \mathbf{Q}_k\mathbf{T}_k\mathbf{y}}_{=0} - \underbrace{\langle\mathbf{t}_k, \mathbf{y}\rangle}_{\beta_k Y_k}\mathbf{q}_{k+1} = -\beta_k y_k\mathbf{q}_{k+1}$$

This implies that $\mathbf{r}_k$ is parallel to $\mathbf{q}_{k+1}$. It also implies that $\mathbf{r}_k$ is orthogonal to all $\mathbf{q}_i,\ i = 1, \ldots, k$. Hence, it also implies that $\mathbf{r}_k$ is orthogonal to all $\mathbf{r}_i,\ i = 0, \ldots, k-1$. So, we can replace $\mathbf{r}_{k-1} = \mathbf{q}_k/\gamma_k$, define $\mathbf{p}_k \triangleq \tilde{\mathbf{p}}_k/\gamma_k$ and consolidate all scalars to obtain this 3-term recursion:

$$\begin{aligned}\mathbf{x}_k &= \mathbf{x}_{k-1} + \alpha_k\mathbf{p}_k\\ \mathbf{r}_k &= \mathbf{r}_{k-1} - \alpha_k\mathbf{A}\mathbf{p}_k\\ \mathbf{p}_k &= \mathbf{r}_{k-1} + \beta_k\mathbf{p}_{k-1}\end{aligned}$$

We now only need to store 3 vectors $(\mathbf{r}_{k-1}, \mathbf{p}_{k-1}, \mathbf{x}_{k-1})$. So starting from $\mathbf{x}_0, \mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$ and $\mathbf{p}_0 = \mathbf{0}$, we can iterate as:

```
$\mathbf{p}_1 = \mathbf{r}_0 + \beta_1 \mathbf{p}_0 = \mathbf{r}_0$
for $k = 1, \dots$ do
    $\mathbf{x}_k = \mathbf{x}_{k-1} + \alpha_k \mathbf{p}_k$
    $\mathbf{r}_k = \mathbf{r}_{k-1} - \alpha_k \mathbf{A}\mathbf{p}_k$
    $\mathbf{p}_{k+1} = \mathbf{r}_k + \beta_{k+1} \mathbf{p}_k$
end for
```

So we need to find $\{\alpha_k\}_{k \geq 1}$ and $\{\beta_k\}_{k \geq 2}$ (note that β_1 does not matter). We can easily see that the columns of $\tilde{\mathbf{P}}_k$ are $\mathbf{A}$-conjugate, i.e. $\langle \tilde{\mathbf{p}}_i, \mathbf{A}\tilde{\mathbf{p}}_j \rangle = 0, i \neq j$. In fact, it is easy to show that

$$\tilde{\mathbf{P}}_k^\top \mathbf{A} \tilde{\mathbf{P}}_k = \mathbf{D}_k$$

So $\mathbf{p}_i$ are $\mathbf{A}$-conjugate, i.e. $\langle \mathbf{p}_i, \mathbf{A}\mathbf{p}_j \rangle = 0, i \neq j$. We have

$$\begin{aligned} \langle \mathbf{p}_i, \mathbf{A}\mathbf{p}_j \rangle &= 0, i \neq j \\ \langle \mathbf{r}_i, \mathbf{r}_j \rangle &= 0, i \neq j \end{aligned}$$

We first note that from $\mathbf{p}_k = \mathbf{r}_{k-1} + \beta_k \mathbf{p}_{k-1}$, we have

$$\langle \mathbf{A}\mathbf{p}_k, \mathbf{p}_k \rangle = \langle \mathbf{A}\mathbf{p}_k, \mathbf{r}_{k-1} + \beta_k \mathbf{p}_{k-1} \rangle = \langle \mathbf{A}\mathbf{p}_k, \mathbf{r}_{k-1} \rangle$$

We can now get α_k from $\mathbf{r}_k = \mathbf{r}_{k-1} - \alpha_k \mathbf{A}\mathbf{p}_k$ as

$$\begin{aligned} \langle \mathbf{r}_{k-1}, \mathbf{r}_k \rangle &= \langle \mathbf{r}_{k-1}, \mathbf{r}_{k-1} - \alpha_k \mathbf{A}\mathbf{p}_k \rangle \\ \implies \alpha_k &= \frac{\langle \mathbf{r}_{k-1}, \mathbf{r}_{k-1} \rangle}{\langle \mathbf{r}_{k-1}, \mathbf{A}\mathbf{p}_k \rangle} = \frac{\langle \mathbf{r}_{k-1}, \mathbf{r}_{k-1} \rangle}{\langle \mathbf{p}_k, \mathbf{A}\mathbf{p}_k \rangle} \end{aligned}$$

Again, from $\mathbf{p}_k = \mathbf{r}_{k-1} + \beta_k \mathbf{p}_{k-1}$, we have

$$\langle \mathbf{A}p_{k-1}, \mathbf{p}_k \rangle = \langle \mathbf{A}p_{k-1}, \mathbf{r}_{k-1} + \beta_k \mathbf{p}_{k-1} \rangle \implies \beta_k = -\frac{\langle \mathbf{A}\mathbf{p}_{k-1}, \mathbf{r}_{k-1} \rangle}{\langle \mathbf{A}\mathbf{p}_{k-1}, \mathbf{p}_{k-1} \rangle}$$

but this is one more dot product than we want because efficiency! From $\mathbf{r}_k = \mathbf{r}_{k-1} - \alpha_k \mathbf{A}\mathbf{p}_k$, we have

$$\langle \mathbf{r}_k, \mathbf{r}_k \rangle = \langle \mathbf{r}_k, \mathbf{r}_{k-1} - \alpha_k \mathbf{A}\mathbf{p}_k \rangle \implies \alpha_k = -\frac{\langle \mathbf{r}_k, \mathbf{r}_k \rangle}{\langle \mathbf{r}_k, \mathbf{A}\mathbf{p}_k \rangle}$$

By equating the two expressions for α_k, we obtain

$$\begin{aligned} -\frac{\langle \mathbf{r}_k, \mathbf{r}_k \rangle}{\langle \mathbf{r}_k, \mathbf{A}\mathbf{p}_k \rangle} = \frac{\langle \mathbf{r}_{k-1}, \mathbf{r}_{k-1} \rangle}{\langle \mathbf{p}_k, \mathbf{A}\mathbf{p}_k \rangle} \iff -\frac{\langle \mathbf{r}_k, \mathbf{r}_k \rangle}{\langle \mathbf{r}_{k-1}, \mathbf{r}_{k-1} \rangle} &= \frac{\langle \mathbf{r}_k, \mathbf{A}\mathbf{p}_k \rangle}{\langle \mathbf{p}_k, \mathbf{A}\mathbf{p}_k \rangle} \\ \implies \beta_k &= \frac{\langle \mathbf{r}_{k-1}, \mathbf{r}_{k-1} \rangle}{\langle \mathbf{r}_{k-2}, \mathbf{r}_{k-2} \rangle} \end{aligned}$$

After all this, we finally arrive at the celebrated conjugate gradient algorithm:

Note 18.1

In practice, $\mathbf{A}\mathbf{p}_k$ is computed once and reused in various lines. Also, $\|\mathbf{r}_k\|^2 = \langle \mathbf{r}_k, \mathbf{r}_k \rangle$ from each iteration is reused in the next iteration to check the termination criterion and also to compute α_k and β_k in the next iteration.

```
$\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0, \mathbf{p}_1 = \mathbf{r}_0$
for $k = 1, 2, \ldots$ until $\|\mathbf{r}_{k-1}\| \le \tau$ do
    $\alpha_k = \langle \mathbf{r}_{k-1}, \mathbf{r}_{k-1} \rangle / \langle \mathbf{p}_k, \mathbf{A}\mathbf{p}_k \rangle$
    $\mathbf{x}_k = \mathbf{x}_{k-1} + \alpha_k \mathbf{p}_k$
    $\mathbf{r}_k = \mathbf{r}_{k-1} - \alpha_k \mathbf{A}\mathbf{p}_k$
    $\beta_{k+1} = \langle \mathbf{r}_k, \mathbf{r}_k \rangle / \langle \mathbf{r}_{k-1}, \mathbf{r}_{k-1} \rangle$
    $\mathbf{p}_{k+1} = \mathbf{r}_k + \beta_{k+1} \mathbf{p}_k$
end for
```

Algorithm 7: Conjugate Gradient

18.2 Convergence properties

From $\mathbf{x} \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$ we have

$$\begin{aligned} \mathbf{x} = \mathbf{x}_0 + p_{k-1}(\mathbf{A})\mathbf{r}_0 &\Longrightarrow \mathbf{x} - \mathbf{x}^\star = \mathbf{x}_0 - \mathbf{x}^\star + p_{k-1}(\mathbf{A})\left(\mathbf{b} - \mathbf{A}\mathbf{x}_0\right) \\ &\Longrightarrow \mathbf{x} - \mathbf{x}^\star = \mathbf{x}_0 - \mathbf{x}^\star + p_{k-1}(\mathbf{A})\mathbf{A}\left(\mathbf{x}^\star - \mathbf{x}_0\right) \\ &\Longrightarrow \mathbf{x} - \mathbf{x}^\star = r_k(\mathbf{A})\left(\mathbf{x}_0 - \mathbf{x}^\star\right) \end{aligned}$$

Note that $r_k(\mathbf{A}) \triangleq \mathbf{I} - \mathbf{A}p_{k-1}(\mathbf{A})$ is a residual polynomial of degree k. For conjugate gradient, we had $\mathbf{x}_k = \arg\min_{\mathbf{x} \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)} \|\mathbf{x} - \mathbf{x}^\star\|_{\mathbf{A}}$, which implies

$$\begin{aligned} \|\mathbf{x}_k - \mathbf{x}^\star\|_{\mathbf{A}} = \min_{\mathbf{x} \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)} \|\mathbf{x} - \mathbf{x}^\star\|_{\mathbf{A}} &= \min_{r_k \in \Pi_k} \|r_k(\mathbf{A})\left(\mathbf{x}_0 - \mathbf{x}^\star\right)\|_{\mathbf{A}} \\ &\le \|\mathbf{x}_0 - \mathbf{x}^\star\|_{\mathbf{A}} \min_{r_k \in \Pi_k} \|r_k(\mathbf{A})\| \\ &\le \|\mathbf{x}_0 - \mathbf{x}^\star\|_{\mathbf{A}} \min_{r_k \in \Pi_k} \max_{\lambda \in \mathrm{spec}(\mathbf{A})} |r_k(\lambda)| \end{aligned}$$

so just as in the case of Chebyshev acceleration, we obtain

$$\|\mathbf{x}_k - \mathbf{x}^\star\|_{\mathbf{A}} \le 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k \|\mathbf{x}_0 - \mathbf{x}^\star\|_{\mathbf{A}} .$$

From $\mathbf{x} \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$, we have

$$\begin{aligned} \mathbf{x}_k = \mathbf{x}_0 + p_{k-1}(\mathbf{A})\mathbf{r}_0 &\Longrightarrow \mathbf{A}\mathbf{x}_k = \mathbf{A}\mathbf{x}_0 + \mathbf{A}p_{k-1}(\mathbf{A})\mathbf{r}_0 \\ &\Longrightarrow \mathbf{r}_k = \mathbf{r}_0 - \mathbf{A}p_{k-1}(\mathbf{A})\mathbf{r}_0 = q_k(\mathbf{A})\mathbf{r}_0 \end{aligned}$$

Note that $r_k(\mathbf{A}) \triangleq \mathbf{I} - \mathbf{A}p_{k-1}(\mathbf{A})$ is a residual polynomial of degree k. For GMRES, we had $\mathbf{x}_k = \arg\min_{\mathbf{x} \in \mathbf{x}_0 + \mathcal{K}_k} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2$, which implies

$$\|\mathbf{b} - \mathbf{A}\mathbf{x}_k\| = \min_{\mathbf{x} \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)} \|\mathbf{b} - \mathbf{A}\mathbf{x}\| = \min_{r_k \in \Pi_k} \|r_k(\mathbf{A})\mathbf{r}_0\|$$

so if $\mathbf{A}$ is diagonalisable as $\mathbf{A} = \mathbf{T}\mathbf{\Lambda}\mathbf{T}^{-1}$, we have

$$\begin{aligned} \|\mathbf{r}_k\| = \min_{r_k \in \Pi_k} \|r_k(\mathbf{A})\mathbf{r}_0\| &\le \min_{r_k \in \Pi_k} \|r_k(\mathbf{A})\| \, \|\mathbf{r}_0\| \\ &\le \mathrm{Cond}(\mathbf{T}) \, \|\mathbf{r}_0\| \min_{r_k \in \Pi_k, \lambda \in \mathrm{spec}(\mathbf{A})} |r_k(\lambda)| \end{aligned}$$

For example, when $\mathbf{A} \succ \mathbf{0}$, just as in the case of Chebyshev acceleration, we obtain for MINRES

$$\|\mathbf{r}_k\| \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k \|\mathbf{r}_0\|.$$

In both bounds, we obtained an expression of the form

$$\min_{r_k\in\Pi_k} \max_{\lambda\in\mathrm{spec}(\mathbf{A})} |r_k(\lambda)|,$$

where Π_k is the space of residual polynomials of degree k. Recall that any such polynomial can be written in a factorised form as

$$r_k \in \Pi_k \iff r_k(\lambda) = \prod_{i=1}^{k}\left(1-\frac{\lambda}{\gamma_i}\right) \quad \gamma_1,\ldots,\gamma_k \in \mathbb{C}.$$

So the above bound amounts to

$$\min_{\{\gamma_1,\ldots,\gamma_k\in\mathbb{C}\}} \max_{\lambda\in\mathrm{spec}(\mathbf{A})} \left|\prod_{i=1}^{k}\left(1-\frac{\lambda}{\gamma_i}\right)\right|$$

So if $|\,\mathrm{spec}(\mathbf{A})| = m$, there exists a residual polynomial of degree m whose roots are exactly these eigenvalues, i.e. after m iterations CG/GMRES/MINRES will find the exact solution! Even if all eigenvalues of the matrix are simple but they are "clumped" in m clusters, these algorithms find a very good approximate solution in around m iterations.

18.3 Preconditioning

When $\kappa(\mathbf{A}) \gg 1$, most iterative methods perform very poorly! In such cases, it might be worthwhile to transform the problem to the one that is either better conditioned, or whose eigenvalues are more tightly clustered.

$$\begin{aligned}
\mathbf{P}^{-1}\mathbf{A}\mathbf{x} &= \mathbf{P}^{-1}\mathbf{b} && \text{(left preconditioner)}\\
\mathbf{A}\mathbf{P}^{-1}\mathbf{y} &= \mathbf{b}, \quad \mathbf{P}\mathbf{x} = \mathbf{y} && \text{(right preconditioner)}\\
\mathbf{P}_L^{-1}\mathbf{A}\mathbf{P}_R^{-1}\mathbf{y} &= \mathbf{P}_L^{-1}\mathbf{b}, \quad \mathbf{P}_R\mathbf{x} = \mathbf{y} && \text{(split preconditioner)}
\end{aligned}$$

A good preconditioner matrix $\mathbf{P}$ is easily invertible and $\kappa(\mathbf{P}^{-1}\mathbf{A}) \ll \kappa(\mathbf{A})$, or its eigenvalues are more tightly clustered. Examples of preconditioners include:

- Matrices from the stationary methods (Jacobi, block variant);
- Incomplete Cholesky (IC) factorisation for when $\mathbf{A} \succ \mathbf{0}$
- Incomplete LU (ILU) factorisation for general matrices;
- Domain decomposition, etc...

Preconditioning is often an art, and application specific.

18.4 Preconditioning conjugate gradient

One might think that since we need to maintain symmetry and positive definiteness, we can only do split preconditioning for CG. Well, it turns out that if **P** is positive definite, we can easily do left preconditioning as

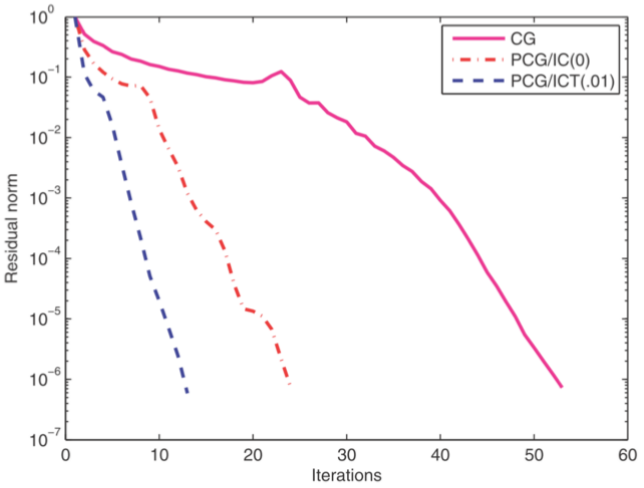
$$\mathbf{P}^{-1}\mathbf{A}\mathbf{x} = \mathbf{P}^{-1}\mathbf{b}$$

Obviously, $\mathbf{P}^{-1}\mathbf{A}$ need not be symmetric in general. But $\mathbf{P}^{-1}\mathbf{A}$ is self-adjoint for the **P**-inner product

$$\begin{aligned}\left\langle \mathbf{P}^{-1}\mathbf{A}\mathbf{x}, \mathbf{y} \right\rangle_{\mathbf{p}} &= \langle \mathbf{A}\mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{A}\mathbf{y} \rangle \\ &= \left\langle \mathbf{x}, \mathbf{P}\mathbf{P}^{-1}\mathbf{A}\mathbf{y} \right\rangle = \left\langle \mathbf{x}, \mathbf{P}^{-1}\mathbf{A}\mathbf{y} \right\rangle_{\mathbf{p}}\end{aligned}$$

So in the original CG, all we need to do is replace

- $\langle \cdot, \cdot \rangle \leftarrow \langle \cdot, \cdot \rangle_{\mathbf{P}}$;
- $\mathbf{A} \leftarrow \mathbf{P}^{-1}\mathbf{A}$;
- $\mathbf{r}_k \leftarrow \mathbf{P}^{-1}\mathbf{r}_k$.



Definition 18.1: Ritz values

Ritz values are the eigenvalues of $\mathbf{H}_k$ or $\mathbf{T}_k$.

Eigenvalues of $\mathbf{H}_k$ or $\mathbf{T}_k$ from the Arnoldi or Lanczos methods typically are very good approximations to the eigenvalues of **A**. When $k \ll n$, extremal eigenvalues of **A**, typically the largest ones, are well approximated. The eigenvectors can also be approximated using $\mathbf{Q}_k$. There also exists a restarted GMRES method which is more efficient.

Note 18.2

CG can be derived from a multitude of angles, but most often it is obtained from the geometric perspective of optimization of quadratic objective and obtaining downhill search directions. This is what you find in most textbooks. Here, we took a different view point and derived CG algebraically based on the decomposition of a symmetric positive definite tridiagonal matrix.

Both view points gives the same algorithm.

Part II

Optimisation

19 Introduction

20 Topology and geometry