PVK: Numerical Methods for CSE HS23 ETHZ
D-MATH: Seminar for Applied Mathematics

Mael Macuglia maelm@ethz.ch

DISCLAIMER:

The main script for Numerical Methods for CSE includes **all** the necessary content for the exam and is **given** during the exam. Therefore, it should be studied carefully! Knowing where to look during the exam is a significant advantage.

The PVK script should be studied **in advance** of the PVK. The aim is to provide a condensed version of chapters 2-3 from the main script, as well as some important tools from linear algebra, which are the building blocks for this course. The remaining content will be covered during the first three days of the PVK. On the last day, we will solve some old exam exercises.

The condensed content in this script will also be very useful for future bachelor courses like Introduction to Machine Learning, NPDE, but also for advanced courses during the master's, such as Optimization for Data Science or Computational Statistics.

Contents

1 Linear Algebra Review **2**
- 1.1 Vector Spaces . . . 2
- 1.2 Some Matrix Properties/Definitions . . . 2
- 1.3 Eigenvalues and Eigenvectors . . . 4
- 1.4 Similarity Transformations . . . 4
- 1.5 Singular Value Decomposition . . . 5
- 1.6 Innerproduct, Norms and Metric Spaces . . . 6
- 1.7 Matrix Norm . . . 7

2 Direct Methods for Linear Systems of Equations **9**
- 2.1 Gaussian Elimination Methods and LU Factorization . . . 9
- 2.2 Direct Solvers in Eigen . . . 10
- 2.3 Exploiting Structure when Solving LSE . . . 10
- 2.4 Not Discussed . . . 10

3 Direct Methods for Linear Least Squares **11**
- 3.1 Least Squares Definition and Setup . . . 11
- 3.2 Geometric Interpretation of Least Squares . . . 11
- 3.3 Normal Equation . . . 12
  - 3.3.1 Solving the N.EQ . . . 13
- 3.4 Orthogonal Transformation for Solving Least Squares Problems . . . 13
  - 3.4.1 QR decomposition . . . 13
  - 3.4.2 Computation of QR decomposition . . . 13
  - 3.4.3 Eigen: QR Decomposition for Solving LS Systems . . . 14
- 3.5 Moore-Penrose Pseudoinverse . . . 14
- 3.6 Not Discussed in This Script . . . 15

4 SVD Based Methods for Least Square Problems **15**
- 4.1 SVD for Solving General Least Squares Problems . . . 15
- 4.2 SVD in Eigen . . . 16

1 Linear Algebra Review

Useful lecture notes (also for NumPDE)

- lecture note

1.1 Vector Spaces

Definition 1.1 Vector Space

A Vector Space over the numeric field $K(K = \mathbb{R}, K = \mathbb{C})$ is a non empty set V, whose elements are called *vectors* and in which two operations are defined, called *addition* and *scalar multiplication*, that enjoys the following properties:

1. addition is commutative and associative
2. there exists an element $0 \in V$ called the *zero vectors* (or *null vector*) such that $v + 0 = v \quad \forall v \in V$
3. $0 \cdot v = 0$ and $1 \cdot v = v \quad \forall v \in V$
4. $\forall v \in V$ there exists an opposite element $-v$ such that $v + (-v) = 0$
5. the following distributive properties hold

$$\forall \alpha \in K, \forall v, w \in V, \alpha(v + w) = \alpha v + \alpha w$$
$$\forall \alpha, \beta \in K, \forall v \in V, (\alpha + \beta)v = \alpha v + \beta v$$

6. the folloing associative property hold

$$\forall \alpha, \beta \in K, \forall v \in V, (\alpha\beta)v = \alpha(\beta v)$$

Examples of vector spaces:

- $V = \mathbb{R}^n$ (respectively $V = \mathbb{C}^n$): the set of the n-tuples of real (reps. complex) numbers, $n \geq 1$
- $V = \mathbb{P}_n$ the set of polynomials $p_n(x) = \sum_{k=0}^{n} \alpha_k x^k$ with real or complex coefficients α_k
- $V = C^p([a, b])$ the set of real (or complex) valued functions which are continous on $[a, b]$ up to their $p - th$ derivative, $0 \leq p < \infty$

Definition 1.2 Vector Subspace

We say that a nonempty part W of V is a *vector subspace* of $V \leftrightarrow W$ is a vector space over K

Examples

- $\mathbb{P}_n$ is a vector subspace of $C^\infty(\mathbb{R})$ (space of infinite continously differentiablke functions on real line)
- trivial subspace of any vector space: $V_{sub} = \{0\}$

Definition 1.3 Linearly Independent Systems

A system of vectors $\{v_1, \ldots, v_m\}$ of a vector space V is called *linearly independent* if

$$\alpha_1 v_1 + \cdots + \alpha_m v_m = 0 \implies \alpha_1 = \cdots = \alpha_m = 0$$

If the last relation does not hold, we call the system *linearly dependent*

Definition 1.4 Basis of Vector Space

We call *basis* of V any system of linearly independent generators of V. If $\{u_1, \ldots, u_n\}$ is a basis of V, the expression $v = \nu_1 u_1 + \cdots + \nu_n u_n$ is called the *decomposition* of v with respect to the basis and the scalars $\nu_1, \ldots \nu_n \in K$ are called the *components* of v **with respect to the given basis**

Note 1.1 Finite vs Infinite Dimensional Vector Space

Let V be a vector space which admits a basis of n-vectors. Then every system of linearly independent vectors of V has at most n elements and **any other basis** of V has n elements. The number n is called the dimension of V and we write $\mathbf{dim(V\ ) = n}$.
If, instead for any n there **always exist** n linearly independent vectors of V, then the vector space is called **infinite dimensional**

1.2 Some Matrix Properties/Definitions

Definition 1.5 Invertible/Regular Matrix

$A \in K^{n \times n}$ is **invertible/regular/nonsingular** $\Leftrightarrow \exists B \in K^{n \times n}$ s.t $AB = BA = \mathbb{I}$.
B is called **inverse** of A
A matrix which is **not** invertible is called **singular**

Note 1.2

A square matrix is invertible iff its columns are linearly independent

Definition 1.6 Transpose Matrix

We call the *transpose* of a matrix $A \in \mathbb{R}^{m\times n}$ denoted by A^T, that is obtained by exchanging the rows of A with the columns of A. We list some of the properties below:

$$(A^T)^T = A \quad | \quad (A+B)^T = A^T + B^T \quad | \quad (AB)^T = B^T A^T \quad | \quad (A^T)^{-1} = (A^{-1})^T =: A^{-T}$$

Definition 1.7 Adjoint/ Conjugate transpose

Let $A \in \mathbb{C}^{m\times n}$; the matrix $B = A^H \in \mathbb{C}n \times m$ is called **the conjugate transpose** of A if $b_{ij} = \overline{a_{ji}}$ (complex conjugate of a_{ij}). Some properties

$$(A^H)^H = A \quad | \quad (A+B)^H = A^H + B^H \quad | \quad (AB)^H = B^H A^H \quad | \quad (A^H)^{-1} = (A^{-1})^H =: A^{-H}$$

Definition 1.8 Orthogonal and Unitary Matrix

A matrix $A \in \mathbb{R}^{\times n}$ is called **Orthogonal** if $AA^T = A^T A = \mathbb{I}$, that is $A^{-1} = A^T$
A matrix $A \in \mathbb{C}^{m\times n}$ is called **Unitary** if $AA^H = A^H A = \mathbb{I}$. A unitary matrix is then $A^{-1} = A^H$

Definition 1.9 Symmetric and Hermitian/Self-Adjoint Matrix

A matrix $A \in \mathbb{R}^{m\times n}$ is called **Symmetric** if $A = A^T$, while it is called **Antisymetric** if $A = -A^T$
A matrix $A \in \mathbb{C}^{m\times n}$ is called **Hermitian** or self-adjoint if $A^T = \overline{A}$ that is if $A = A^H$

Definition 1.10 Range and Kernel of a Matrix

Given a matrix $A \in K^{m\times n}$ the **image space** or. **range** of A is the subspace of K^m spanned by the columns of A i.e

$$R(A) := \{x \in K^n : Ax\}$$

The **Kernel** or **Null Space** of A is defined as the subspace

$$Ker(A) = \{x \in K^n : Ax = 0\}$$

Definition 1.11 Rank of a Matrix

The **rank** of a matrix $A \in K^{m\times n}$, denoted by $rank(A)$, is the maximum number of linearly independent rows/columns of A. Equivalently we have

$$rank(A) = Dim(R(A))$$

Definition 1.12 Normal Matrix

Let $A \in K^{n\times n}$. We say A is **normal** iff $A^H A = AA^H$

Lemma 1.1 Range and Kernel Relations between Hermitian/Transpose Matrices (without proof)

For any matrix $A \in K^{m,n}$ holds

- $\mathcal{N}(A) = \mathcal{R}(A^H)^\perp$
- $\mathcal{N}(A)^\perp = \mathcal{R}(A^H)$

Note 1.3 Some Equivalent Properties of Regular Matrices

Let $A \in K^{n\times n}$ then the following holds

$$\begin{aligned}
&\text{A is nonsingular} \quad \Leftrightarrow \\
&det(A) \neq 0 \quad \Leftrightarrow \\
&Ker(A) = \{0\} \quad \Leftrightarrow \\
&rank(A) = n \quad \Leftrightarrow \\
&\text{A has linearly independent rows and columns.}
\end{aligned}$$

1.3 Eigenvalues and Eigenvectors

Definition 1.13 Eigenvalues/Eigenvectors and some Nomenclature

Let $A \in K^{n\times n}$:

- $\lambda \in \mathbb{C}$ is called **eigenvalue** of A $\Leftrightarrow \underbrace{det(A - \lambda\mathbb{I} = 0)}_{=:p(\lambda)} \Leftrightarrow \exists x \in K^n \quad s.t \quad Ax = \lambda x$
- $p(\lambda)$ is called **characteristic polynomial**
- Given an eigenvalue λ we call x **righteigenvector** and y **lefteigenvector** of A

$$Ax = \lambda x$$
$$y^H A = \lambda y^H$$

- The set of eigenvalues is called **Spectrum** of A

$$\sigma(A) := \{\lambda \in \mathbb{C} : \lambda \text{ Eigenvalue of} A\}$$

- The maximum module of the eigenvalues of A is called **spectral radius** of A

$$\rho(A) := \max_{\lambda \in \sigma(A)} |\lambda|$$

- The set of vectors associated with eigenvalue λ is called **Eigenspace** of λ denoted Eig_λ. and it corresponds by definition to

$$ker(A - \lambda\mathbb{I})$$

- The dimension of Eig_λ is called **geometric multiplicity**

$$dim[ker(A - \lambda I)] = n - rank(A - \lambda I)$$

- The **algebraic multiplicity** of λ is the multiplicity of λ as a root of the charachteristic polynomial.

Note 1.4 Some Properties/Results wihtout Proofs

- $det(A) = \prod_{i=1}^n \lambda_i$ and $Tr(A) = \sum_{i=1}^n \lambda_i$ (the proof is easy, try it home !) The first result has the following implication

$$\text{A is singular} \Leftrightarrow \exists\lambda = 0, \lambda \in \sigma(A)$$

- we notice the following

$$det(A^T - \lambda I) = det((A - \lambda I)^T) = det(A - \lambda I) \implies \sigma(A^T) = \sigma(A)$$

- Given an eignvector x we can determine the corresponding eigenvalue with the **Rayleigh quotient**

$$\lambda = \frac{x^H Ax}{x^H x}$$

1.4 Similarity Transformations

Definition 1.14 Similar Matrix

Let $A, B \in K^{n\times n}$. We say that A and B are **similar** if there exists an invertible matrix $P \in K^{n\times n}$ such that

$$B = P^{-1}AP$$

The linear map $A \to P^{-1}AP$ is called a **similarity transformation**
Remark: Similar matrices share have same Eigen spectrum (easy to show)

Note 1.5 Property: Schur Decomposition

Given $A \in K^{n\times n}$ there exists U unitary such that

$$U^{-1}AU = U^H AU = \begin{bmatrix} \lambda_1 & b_{12} & b_{13} & \cdots & b_{1n} \\ 0 & \lambda_2 & b_{23} & \cdots & b_{2n} \\ 0 & 0 & \lambda_3 & \cdots & b_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_n \end{bmatrix} = T$$

Remark: the matrices U and T are not unique.

Lemma 1.2 Results from Schur Decomposition (without proofs)

- Every **Hermitian** (symetric) matrix is **unitarly similar** to a diagonal real matrix (every Schur decompostion of A is diagonal).

$$U^{-1}AU = U^HAU = D = diag(\lambda_1, \dots, \lambda_n) \Leftrightarrow AU = UD \Leftrightarrow Au_i = \lambda_i u_i \quad i \in [n]$$

  Hence the columns vector of U are the Eigenvectors of A. Moreover as the matrix U is unitary, the span of its columns generate the whole space K^n

- A matrix $A \in K^{n \times n}$ is **normal** iff it is **unitarly similar** to a diagonal matrix D. as a consequence, a normal matrix A admits the following **spectral decomposition**

$$A = UDU^H = \sum_{i \in [n]} \lambda_i u_i u_i^H$$

1.5 Singular Value Decomposition

Theorem 1.1 Singular Value Decomposition (without proof)

For any $A \in K^{m \times n}$ there are **unitary / Orhtogonal** matrices $U \in K^{m \times m}$, $V \in K^{n \times n}$ and a diagonal matrix $\Sigma = diag(\sigma_1, \dots, \sigma_p) \in \mathbb{R}^{m \times n}$ with $p := min\{m, n\}$ and $\sigma_1 \geq \dots \geq \sigma_p$ such that

$$A = U\Sigma V^H$$

The last decompostion is called **Singular Value Decomposition** of the matrix A, with **singular values** $= \{\sigma_1, \dots, \sigma_p\}$

Note 1.6 Economical SVD

drop the **bottom zero rows** of Σ yield the so called **"economical" SVD**

$$\begin{array}{llllll} m \geq n: & A = U\Sigma V^H & , \; U \in K^{m,n} \quad V \in K^{n,n} & , \; U^HU = \mathbb{I}_n & , \; V \text{ Unitary} \\ m < n: & A = U\Sigma V^H & , \; U \in K^{m,m} \quad V \in K^{n,m} & , \; U \text{ Unitary} & , \; VV^H = \mathbb{I}_m \end{array}$$

Lemma 1.3 Uniqueness of SVD

The matrix U and V from SVD are **not unique**.
The singular values are **unique**.
<u>Proof:</u>
Assume two decomposition for a matrix $A \in \mathbb{R}^{m,n}$ i.e

$$A = U_1\Sigma_1V_1^T = U_2\Sigma_2V_2^T \implies U_1\Sigma_1\Sigma_1^TU_1^T = AA^T = U_2\Sigma_2\Sigma_2^TU_2^T$$

hence by noticing that $\Sigma_1\Sigma_1^T$ and $\Sigma_2\Sigma_2^T$ are similar, both matrices shares the same eigenvalues which agrees with there. $\square$

Lemma 1.4 Relation Eigenvalue SVD

Note

$$\sigma_i(A) = \sqrt{\lambda_i(A^HA)} \quad , \quad i \in [p]$$

Proof: recall full SVD U and V **Unitary** hence

$$A^HA = V\Sigma^H\Sigma V^H \implies \lambda_i(A^HA) = \lambda_i(\Sigma^H\Sigma) = (\sigma_i(A))^2$$

Lemma 1.5 SVD and rank of matrix (no proof)

Let $r : 1 \leq r \leq p := min\{m, n\}$. If the singular values of A satisfy

$$\sigma_1 \geq \dots \geq \sigma_r \geq \sigma_{r+1} = \dots \sigma_p = 0$$

then the following holds

- $Rank(A) = r$
- $\mathcal{N}(A) = Span\{(V)_{:,r+1}, \dots, (V)_{:,p}\}$
- $\mathcal{R}(A) = Span\{(U)_{:,1}, \dots, (U)\}$

Lemma 1.6 Rank of Matrix

If $A \in \mathbb{R}^{m,n}$ has rank $r \leq min\{m, n\}$ then there exist $X \in \mathbb{R}^{m,r}$ and $Y \in \mathbb{R}^{n,r}$ such that $A = XY^T$

Proof: By SVD theorem $A = U\Sigma V^T$, choose

$$X := (U)_{:,1:r}(\Sigma)_{1:r,1:r}$$
$$Y := (V)_{:,1:r}$$

Theorem 1.2 Best Low Rank Approximation (without proof)

Let $A = U\Sigma V^H$ be the SVD of $A \in K^{m,n}$ then for $1 \leq k \leq Rank(A)$ set

$$A_k := U_k \Sigma_k V_k^H = \sum_{l \in [k]} \sigma_l (U)_{:,l} (V)_{:,l}^H \quad \text{with} \quad \begin{cases} U_k := [(U)_{:,1}, \dots, (U)_{:,k}] \in K^{m,k} \\ V_k := [(V)_{:,1}, \dots, (V)_{:,k}] \in K^{n,k} \\ \Sigma_k := diag(\sigma_1, \dots, \sigma_k) \in K^{k,k} \end{cases}$$

Then

$$\|A - A_k\|_F \leq \|A - F\|_F \quad \forall F \in \mathcal{R}_k(m,n) := \{F \in K^{m,n} : rank(A) \leq k\}$$

that is the matrix A_k is the **best rank-k approximation** of the matrix A

1.6 Innerproduct, Norms and Metric Spaces

Think: aim at quantifying differences between objects in vector spaces .

Definition 1.15 Inner Product Space

An **inner product space** is a vector space V over the Field F toegether with an **inner product**, that is a map

$$\langle \cdot, \cdot \rangle : V \times V \to F$$

having the following properties

- Hermitian: $\langle y, x \rangle = \overline{\langle x, y \rangle}$
- Linearity : $\langle \gamma x + \lambda z, y \rangle = \gamma \langle x, y \rangle + \lambda \langle z, y \rangle$
- Positive Definite: $\langle x, x \rangle > 0, \forall x \neq 0$ and $\langle x, x \rangle = 0 \Leftrightarrow x = 0$

Examples of valid inner product spaces:

- $(\mathbb{R}^n, \langle x, y \rangle) := x^T y$ Euclidean inner product
- $(\mathbb{R}^n, \langle x, y \rangle) := x^T A y$ for A spd matrix
- $(\mathbb{R}^{m,n}, \langle A, B \rangle) := Tr\{A^T B\}$
- Let X and Y be random variables: then the expected value is a inner product $\langle X, Y \rangle = \mathbb{E}XY$
- $(L_2(\mathbb{R}), \langle f, g \rangle := \int_{\mathbb{R}} f(t) g(t) dt)$

Definition 1.16 Normed Space

A **normed space** is a vector space V over a field F together with a **norm**, that is a map

$$\| \cdot \| : F \to \mathbb{R}_+$$

satifying

- $\|v\| \geq 0 \quad \forall v \in V$ and $\|v\| = 0 \Leftrightarrow v = 0$
- Homogeneity: $\|\alpha v\| = |\alpha| \|v\| \quad \forall \alpha \in F \quad \forall v \in V$ where $|\alpha|$ is the absolute value if $F = \mathbb{R}$ or the module if $F = \mathbb{C}$
- Triangular Inequality: $\|v + w\| \leq \|v\| + \|w\| \quad \forall v, w \in V$

Examples of valid normed vector spaces:

- $(\mathbb{R}^n, \| \cdot \|_1)$
- $(\mathbb{R}^n, \| \cdot \|_2)$
- $(L_2(\mathbb{R}), \| \cdot \|_1)$

Definition 1.17 Metric Space

A metric is a pair $(\mathcal{X}, \rho)$ where $\mathcal{X}$ is a set and ρ is a so called **metric**, that is a mapping

$$\rho : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$$

satisfying $\forall x, y, z \in \mathcal{X}$

- $\rho(x, x) = 0$ (distance to itself)
- Positivity: $x \neq y \implies \rho(x, y) > 0$
- Symmetry: $\rho(x, y) = \rho(y, x)$
- Triangle Inequality: $\rho(x, z) \leq \rho(x, y) + \rho(y, z)$

Examples:

- $(\mathbb{R}, \rho(x, y) := |y - x|)$

- $(\mathbb{R}^+, \rho(x,y) := ln(y/x))$
- $(\mathbb{R}^{m,n}, \rho(X,Y) := rank(X-Y))$

Note 1.7

- From a **normed vector** space $(V, \|\cdot\|)$ we can define a valid distance metric as $\rho(x,y) := \|x-y\|$
- From an **inner product** space $(V, \langle\cdot,\cdot\rangle)$ we can define a valid "associated" norm $\|\cdot\|$ as

$$\|x\| = \sqrt{\langle x,x\rangle}$$

  which in turn gives rise to a distance metric $\rho(x,y) := \|x-y\|$
- **inner product space** $\implies$ **norm space** $\implies$ **metric space**

Note 1.8 p-norm

A **p -norm** or Holder norm is defined as

$$\|x\|_p = (\sum_{i\in[n]} |x_i|^p)^{\frac{1}{p}}$$

Lemma 1.7 Cauchy-Schwartz

Given x, y in an **inner product** space $(V, \langle\cdot,\cdot\rangle)$ the **Cauchy - Bunyakovsk -Schwarz -inequality** says

$$|\langle x,y\rangle|^2 \leq \langle x,x\rangle \cdot \langle y,y\rangle \Leftrightarrow$$
$$|\langle x,y\rangle| \leq \sqrt{\langle x,x\rangle} \cdot \sqrt{\langle y,y\rangle}$$

Examples:

- From 2-d Geometry assuming $x, y \in \mathbb{R}^2$: $\langle x,y\rangle = \|u\|_2\|v\|_2 cos(\theta) \leq \|u\|_2\|v\|_2$
- Assume $f, g \in (L_2(\mathbb{R}^n), \langle f,g\rangle := \int_R^n f(x)g(x)dx)$ then cauchy-schwartz tells us

$$\int_{R^n} f(x)g(x)dx \leq \int_{\mathbb{R}^n} |f(x)|^2 dx \int_{\mathbb{R}^n} |g(x)|^2 dx$$

Lemma 1.8 Holder's Inequality (not most general result: for probability measure take the result over a measurable space)

Assume $x, y \in \mathbb{R}^n$ then $|\langle x,y\rangle| \leq \|x\|_p\|y\|_q$ with $\frac{1}{p} + \frac{1}{q} = 1$

Definition 1.18 Equivalent norm

Two norms $\|\cdot\|_p$ and $\|\cdot\|_q$ on V are **equivalent** if there exist two positive constants c_{pq} and C_{pq} such that

$$c_{pq}\|x\|_q \leq \|x\|_p \leq C_{pq}\|x\|_q \quad \forall x \in V$$

Hence **p-norm** with $p = 1, 2, \infty$ are equivalent (easy to show)

Note 1.9 Convergence Finite Dimensional Vector Space

Let $\|\cdot\|$ be a norm in a finite dimensional space V. Then

$$\lim_{k\to\infty} x^k = x \Leftrightarrow \lim_{k\to\infty} \|x - x^k\| = 0$$

where $x \in V$ and $\{x^k\}$ is a sequence in V

1.7 Matrix Norm

Definition 1.19 Matrix Norm

A **matrix norm** is a mapping $\|\cdot\| : \mathbb{R}^{m\times n} \to \mathbb{R}$ such that

- $\|A\| \geq 0 \quad \forall A \in \mathbb{R}^{m\times n}$ and $\|A\| = 0 \Leftrightarrow A = 0$
- $\|\| = \|alpha\|\|A\| \quad \forall \alpha \in \mathbb{R} \quad \forall A \in \mathbb{R}^{m\times n}$
- $\|A + B\| \leq \|A\| + \|B\| \quad \forall A, B \in \mathbb{R}^{m\times n}$

Definition 1.20 Consistentency with vector norm

We say that a matrix norm $\|\cdot\|$ is **consistent** with a vector norm $\|\cdot\|$ if

$$\|Ax\| \leq \|A\|\|x\| \quad \forall x \in \mathbb{R}^n$$

Definition 1.21 Sub-multiplicative

We say that a matrix norm $\|\cdot\|$ is **sub- multiplicative** if $\forall A \in \mathbb{R}^{m\times n} \quad \forall B \in \mathbb{R}^{m\times q}$

$$\|AB\| \leq \|A\|\|B\|$$

CAREFUL: this property is not fullfied by every matrix norm

Note 1.10 Forbenius Norm

The **Forbenius** norm defined as follow

$$\|A\|_F = \sqrt{\langle A, A\rangle} \underbrace{=}_{InnerProd.} \sqrt{Tr(AA^H)} \underbrace{=}_{SVD} \sqrt{TrV\Sigma^2V^H} = \sqrt{Tr\Sigma^2} = \sqrt{\sum_{i\in[r]} \sigma_i^2}$$

where $r = Rank(A)$

Definition 1.22 Matrix Norm Associated with Two Normed Spaces

Given vector norms $\|\cdot\|_x$ on K^n and $\|\cdot\|_y$ on K^m the **associated** mapping

$$\|A\| = \sup_{x\neq 0} \frac{\|Ax\|_y}{\|x\|_x}$$

is a valid matrix norm, where $A \in K^{m,n}$

Definition 1.23 Induced p-Matrix-Norm

Let $\|\cdot\|_p$ be a p-norm on K^n. The function

$$\|A\|_p = \sup_{x\neq 0} \frac{\|Ax\|_p}{\|x\|_p}$$

is a valid matrix norm, where $A \in K^{n,n}$

Note 1.11 Spectral Norm and Nuclear Norm

Two import matrix norms are:

- p=2 matrix norm is called **spectral** or **operator** norm

$$\|A\|_2 = \sup_{x\neq 0} \frac{\|Ax\|_2}{\|x\|_2} \underbrace{=}_{\text{proof omitted}} \sigma(A)_1$$

- The **nuclear** norm defined as

$$\|A\|_* = \sum_{i\in[r]} \sigma(A)_i$$

2 Direct Methods for Linear Systems of Equations

Definition 2.1 Square LSE

Given a square matrix $A \in K^{n,n}$ and a vector $b \in K^n$ find $x \in K^n$ such that

$$Ax = b$$

Existence of a solution is guaranteed iff $A \in K_*^{n,n} := \{A \in K^{n,n} : A \text{ Regular see 1.3}\}$, where the solution takes the form

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

Let denote the **data** space $\mathcal{X} := K_*^{n,n} \times K^n$ and the **result** space $\mathcal{Y} := K^n$ we define the mapping F from data to output space as

$$F : \begin{cases} \mathcal{X} \to \mathcal{Y} \\ (A, b) \to A^{-1}b \end{cases}$$

2.1 Gaussian Elimination Methods and LU Factorization

The Gaussian elimination method aims at reducing the system Ax=b to an equivalent system (that is, having the same solution) of the form Ux=b, where U is an upper triangular matrix and b is an updated right side vector. We recall a useful lemma

Lemma 2.1 Group of regular diagonal/triangular matrices

$$A, B = \begin{cases} \text{diagonal} \\ \text{upper triangular} \\ \text{lower triangular} \end{cases} \implies AB \text{ and } A^{-1} = \begin{cases} \text{diagonal} \\ \text{upper triangular} \\ \text{lower triangular} \end{cases}$$

Definition 2.2 Permutation Matrix

An **n-permutation** $n \in \mathbb{N}$ is a bijective mapping $\pi : \{1, \dots, n\} \to \{1, \dots, n\}$. The corresponding **permutation** matrix $P_\pi \in K^{n,n}$ is defined by

$$(P_\pi)_{ij} = \begin{cases} 1 \text{ if } j = \pi(j) \\ 0 \text{ else} \end{cases}$$

- $P^T = P^{-1}$ i.e permutation matrices are **Orthogonal/ Unitary**
- **left-multiplication** $P_\pi A$ effects π permutation of **rows**
- **right-multiplication** AP_π effects π permutation of **columns**

Note 2.1 Gaussian Elimination as Row Manipulation

Using permutation and scaling matrices to manipulate the rows of A we can derive (not rigorous) the following gaussian elimination

$$\begin{aligned} Ax = b &\Leftrightarrow \\ L_1 Ax = L_1 b &\Leftrightarrow \\ \vdots &\Leftrightarrow \\ \underbrace{L_{n-1} \dots L_1 A}_{:=U} x = \underbrace{L_{n-1} \dots L_1 b}_{\tilde{b}} &\Leftrightarrow \\ Ux = \tilde{b} & \end{aligned}$$

where $L_1, \dots, L_{n-1}$ are lower triangluar matrices. Notice that

$$L_1 \cdots L_{n-1} A =: L^{-1} A = U \Leftrightarrow A = LU$$

where L is lower triangular and U is upper triangular. It can be shown that the so called LU factorisation is algebraically equivalent to gaussian elimination.

Definition 2.3 LU-decomposition/factorization

Given a square matrix $A \in K^{n,n}$, an **upper triangular matrix** $U \in K^{n,n}$ and a **normalized lower triangular matrix** $L \in K^{n,n}$ form an LU factorisation of A if

$$A = LU$$

Theorem 2.1 Existence and Uniqueness (without proofs)

The LU-decompostion of $A \in K^{n,n}$ **exists** if all submatrices $(A)_{1:k,1:k}$ $\quad 1 \le k \le n$ are **regular** Moreover the LU factorisation is **unique**

Note 2.2 Solving LSE with LU Factorisation

- Step 1: Get Decompositon: $A = LU$ cost: $\mathcal{O}(n^3)$
- Step 2: **forward** elimination : Solve $L\tilde{b} = b$ for $\tilde{b}$ Cost: $\mathcal{O}(n^2)$
- Step 3: **backward** elimination: Solve for x $Ux = \tilde{b}$ for x Cost: $\mathcal{O}(n^2)$

Total Cost: Settup phase $\mathcal{O}(n^3)$ + Elimination Phase $\mathcal{O}(n^2)$

2.2 Direct Solvers in Eigen

Given $A \in K_*^{n,n}$ and a l right handside vectors as columns of a matrix $B = [b_1, \dots, b_l] \in K^{n,l}$ find

$$X = A^{-1}B = [A^{-1}b_1, \dots, A^{-1}b_l] \quad \text{Costs: } \mathcal{O}(n^3 + n^2 l)$$

```
Eigen::Solvertype<Eigen::MatrixXd> solver(A);
Eigen::VectorXd x = solver.solve(b);
```

or in one line

```
Eigen::VectorXd A.Solvertype().solve(b);
```

The **SolverType** are listed in Eigen Documentation.
Example of a solver type is gaussian elimination with partial pivoting yielding

```
Eigen::VectorXd A.lu().solve(b);
```

We can also extract the matrix U and L as follow

```
    auto ludec = A.lu() ;
    Eigen::MatrixXd U{ludec.matrixU().triangularView<Eigen::upper>() };
        MatrixXd L {lude.matrixLU(). triangularView <Eigen::UnitLower>()};
```

2.3 Exploiting Structure when Solving LSE

- A has Triangular Structure (See Script)
- Decompostion already available and matrix A affected by rank-k modifications i.e

$$\tilde{A} = A + UV^H \quad \text{with} \quad rank(UV^H) = k$$

with available solution

$$X = A^{-1}B$$

Lemma 2.2 Sherman-Morrison-Woodbury formula

For regular $A \in K^{n,n}$, and $U, V \in K^{n,k}$, $\quad k \leq n \in \mathbb{N}$, holds

$$(A + UV^H)^{-1} = A^{-1} - A^{-1}U(I + V^H A^{-1}U)^{-1}V^H A^{-1}$$

if $I + V^H A^{-1}U$ is regular.

Using last lemma we conclude that we only require to solve

$$A\hat{X} = U$$

in order to compute the modified solution $\tilde{X}$

$$\tilde{X} = A^{-1}B - A^{-1}U(I + V^H A^{-1}U)^{-1}V^H A^{-1}B = X - A^{-1}U(I + V^H A^{-1}U)^{-1}V^H X$$

which yield an asymptotic complexity $\mathcal{O}(kn^2)$

2.4 Not Discussed

- Stabily of gaussian elimination without pivoting for diagonally dominant matrices
- LSE for **Sparse** matrices. Here check Eigen/Sparse documentation. The LSE deinition is the same, but Eigen will use the fact that the data matrix A is sparse.

3 Direct Methods for Linear Least Squares

3.1 Least Squares Definition and Setup

Note 3.1 Setup for this Chapter

Consider the linear system of equation

$$x \in \mathbb{R}^n \quad Ax = b \quad b \in \mathbb{R}^m \quad A \in \mathbb{R}^{m,n} \qquad \underbrace{m \geq n}_{=:\text{possibly overdetermined}}$$

Last chapter tells us

$$\exists x \in \mathbb{R}^n \quad s.t. \quad Ax = b \quad \Leftrightarrow \quad b \in \mathcal{R}(A) = \{Ax, x \in \mathbb{R}^n\} \subset \mathbb{R}^m$$

What if $b \notin \mathcal{R}(A)$ **?**

When $\mathcal{R}(A) \neq \mathbb{R}^m$ we say that the image space $\mathcal{R}(A)$ is "a set of measure zero " in $\mathbb{R}^m$, which (not formally) means that the subspace formed by the columns of A is relatively small compared to the entire space $\mathbb{R}^m$. Hence it is unlikely that the vector b belongs to $\mathcal{R}(A)$. Even in the case where modelling constraints the vector b to be in $\mathcal{R}(A)$, almost all perturbation of b , e.g measurements errors , will destroy $b \in \mathcal{R}(A)$

Definition 3.1 Least Squares Solution

For given $A \in \mathbb{R}^{m,n}, b \in \mathbb{R}^m$ the vector $x \in \mathbb{R}^n$ is a **least square solution (LS)** of the linear system of equations $Ax = b$ if

$$x \in \arg\min_{y \in \mathbb{R}^n} \|Ay - b\|_2^2$$

We write the set of least squares solutions as follow

$$Lsq(A, b) := \{x \in \mathbb{R}^n : x \text{is least square solution of } Ax = b\} \subset \mathbb{R}^n$$

Lemma 3.1 Existence of LS solution

For any $A \in \mathbb{R}^{m,n}, b \in \mathbb{R}^m$ a least square solution of $Ax = b$ exists.
Proof:

$$\left[\underbrace{F(x) := \|Ax - b\|_2^2 \geq 0}_{\text{bounded from below}} \quad \text{continous}\right] \text{ and } \left[x \to \infty \implies F(x) \to \infty\right] \implies \exists x^* \in \mathbb{R}^n F(x^*) \text{ is minimum}$$

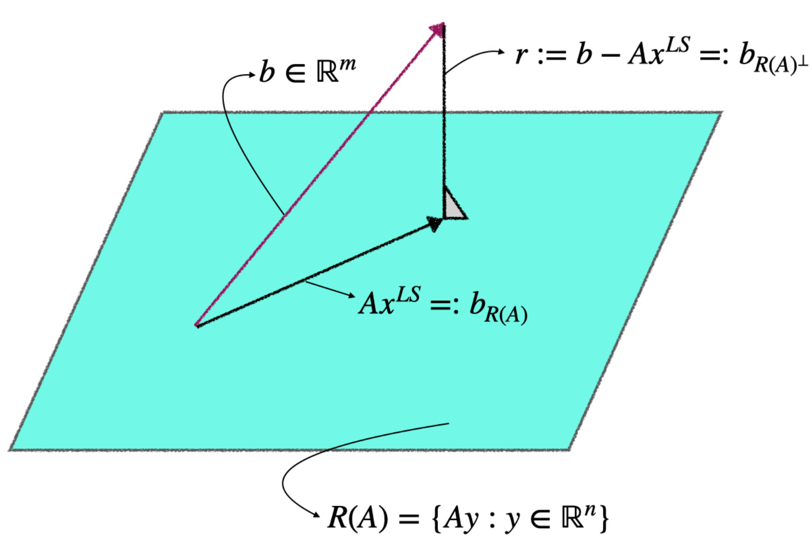
3.2 Geometric Interpretation of Least Squares

The closest (wrt to euclidean norm) vector in a subspace ($\mathcal{R}(A) \subset \mathbb{R}^n$) to a vector ($b \in \mathbb{R}^m$) is the orthogonal projection of the vector b onto the subspace $\mathcal{R}(A)$. We can then decompose the vector b with its projection onto $\mathcal{R}(A)$ and the orthogonal complement i.e

$$b = b_{\mathcal{R}(A)} + b_{\mathcal{R}(A)^\perp}$$

the space of least squares solutions is then

$$lsq(A, b) = \{Ax = b_{\mathcal{R}(A)} : x \in \mathbb{R}^n\}$$



3.3 Normal Equation

From the geometric intuition in last section we conclude that

$$b - Ax^{LS} \perp Span\{(A)_{:,1}, \dots, (A)_{:,n}\} \Leftrightarrow A^T(Ax^{LS} - b) = 0$$

which results in the following theorem

Theorem 3.1 Solving Normal Equation Yield LS Solutions

The vector $x \in \mathbb{R}^n$ is a **least squares solution** of the linear system $Ax = b, A \in \mathbb{R}^{m,n}, b \in \mathbb{R}^m$, **if and only if** it solves the following square system called the **Normal Equation**

$$A^T Ax = A^T b$$

<u>Proof:</u>
Hint: Employ inner product notation for enhanced clarity! The objective of presenting this proof is to illustrate that leveraging clever notation can be transformative.
We want to show $\|Ay - b\|_2^2 > \|Ax - b\|_2^2$ for x, y sucht that $\forall y \neq x \in \mathbb{R}^n \quad , \quad x \in \{x :\in \mathbb{R}^n : A^T(Ax - b) = 0\}$
the following holds:

$$\begin{aligned}
\|Ay - b\|_2^2 - \|Ax - b\|_2^2 &= \langle Ay - b, Ay - b\rangle - \langle Ax - b, Ax - b\rangle \\
&= \langle Ay, Ay\rangle - 2\langle Ay, b\rangle + \langle b, b\rangle - \underbrace{\langle Ax - b, b\rangle}_{=0 \text{ Normal Eq}} + \langle Ax, b\rangle - \langle b, b\rangle \\
&= \langle Ay, Ay\rangle - 2\langle Ay, b\rangle + \langle Ax, b\rangle
\end{aligned}$$

plugging the geometric view point of normal equation we have $b = r + Ax = b - Ax + Ax$ we have

$$\begin{aligned}
\langle Ay, Ay\rangle - 2\langle Ay, b\rangle + \langle Ax, b\rangle &= \langle Ay, Ay\rangle - 2\langle Ay, (b - Ax) + Ax\rangle + \langle Ax, (b - Ax) + Ax\rangle \\
&= \langle Ay, Ay\rangle - 2\underbrace{\langle Ay, (b - Ax)\rangle}_{=0} - 2\langle Ay, Ax\rangle + \underbrace{\langle b - Ax, Ax\rangle}_{=0} + \langle Ax, Ax\rangle \\
&= \langle Ay - Ax, Ay - Ax\rangle \\
&= \|Ay - Ax\|_2^2 > 0 \quad \forall y \neq x
\end{aligned}$$

□

Note 3.2 Normal Equation from Convexity Argument

Notice that the function $F(x) = \|Ax - b\|_2^2$ is convex (easy to show) and differentiable over a convex set ($\mathbb{R}^n$). If $\nabla F(x) = 0$, then x is a minimizer. Using differention rule for inner product we have

$$\begin{aligned}
\nabla F(x)^T h = DF(x)h &= \langle D(Ax - b)h, Ax - b\rangle + \langle Ax - b, D(Ax - b)h\rangle \\
&= \langle Ah, Ax - b\rangle + \langle Ax - b, Ah\rangle \\
&= 2\langle Ax - b, Ah\rangle \\
&\Leftrightarrow \nabla F(x) = 2A^T(Ax - b) \\
&A^T(Ax - b) = 0 \Leftrightarrow x \text{minimizer}
\end{aligned}$$

Lemma 3.2 Uniqueness of Least Squares

for $m \geq n$ the linear system of equation from the normal equation has a unique solution iff $\mathcal{N}(A^T A) = \{0\}$ (from section on linear systems). Notice that

$$\mathcal{N}(A^T A) = \mathcal{N}(A)$$

Hence the system $Ax = b, A \in \mathbb{R}^{m,n}, b \in \mathbb{R}^n$ has a unique Least Square solution solving

$$A^T Ax^{LS} = A^T b$$

iff

$$\mathcal{N}(A^T A) = \mathcal{N}(A) = \{0\} \Leftrightarrow Rank(A) = n \quad \text{(Full Rank)}$$

Note 3.3 Rank Defect

In case the rank of the design matrix $A \in \mathbb{R}^{m,n}, m \geq n$ fails to have full rank, it hints at **inadequate modelling**. In this case parameters are redundant, because different sets of parameters (i.e LS solution not unique) yield the same output quantities: the parameters are not "observable".

3.3.1 Solving the N.EQ

Note 3.4 Recipe for Soling Normal Equation

Assuming $A \in \mathbb{R}^{m,n}$ has full rank, solving the normal equation is consists of the following three steps

1. Compute the regular s.p.d matrix $C = A^T A \in \mathbb{R}^{n,n}$
2. Compute the right handside vector $c := A^T b$
3. Solve the s.p.d linear system of equations: $Cx = c$

Total Asymptotic Costs:

$$\mathcal{O}(mn^2 + n^3) \quad n \to \infty, m \to \infty$$

We point out that for symmetric positive definite matrices, there exists a theorem saying that gaussian elimination is **stable** without pivoting. This is taken into account by Eigen when using the **Cholesky** decomposition by calling the llt() method.

CODE:: Solve Normal Equation with llt() Method:

```
auto x = (A.transpose()*A).llt().solve(A.transpose()*b);
```

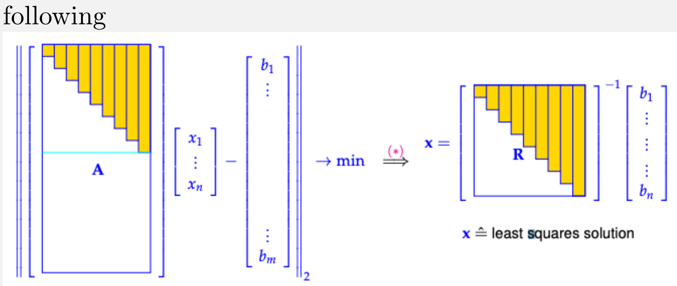
3.4 Orthogonal Transformation for Solving Least Squares Problems

Note 3.5 What LS squares Systems are "Easy" to solve ?

Consider the full rank linear least squares setup, i.e

$$A \in \mathbb{R}^{m,n} \quad Rank(A) = n \quad b \in \mathbb{R}^m \quad \text{find} \quad x = \arg\min_{y \in \mathbb{R}^n} \|Ay - b\|_2$$

Notice that if A is upper triangular, then the LS squares solution would be "easy" to solve as it can be seen by the following



If A is not upper triangular, we can obtain at an equivalent least squares system with an upper triangular matrix by means of **orthogonal transformation**, i.e assuming a transformation matrix $T \in \{T \in \mathbb{R}^{m,m} : T \text{ Unitary}\}$ such that $TA =$ Upper triangular then

$$\arg\min_{x \in \mathbb{R}^n} \|Ax - b\|_2^2 = \arg\min_{x \in \mathbb{R}^n} \|T(Ax - b)\|_2^2 \Leftrightarrow$$
$$\|T(Ax - b)\|_2^2 = (Ax - b)^T \underbrace{T^T T}_{=I}(Ax - b) = \|Ax - b\|_2^2$$

3.4.1 QR decomposition

Theorem 3.2 QR-decomposition

For any matrix $A \in R^{m,n}$ with $rank(A) = n$ there exists

- a unique **unitary** matrix $Q_0 \in \mathbb{R}^{m,n}$ and a unique **upper triangular** matrix $R_0 \in \mathbb{R}^{n,n}$ with $(R_0)_{i,i} > 0 \quad i \in [n]$ such that

$$A = Q_0 R_0 \quad \textbf{"economical" QR Decomposition}$$

there also exists

- a unique **unitary** matrix $Q \in \mathbb{R}^{m,m}$ and a unique **upper triangular** matrix $R \in \mathbb{R}^{m,n}$ with $(R)_{i,i} > 0 \quad i \in [n]$ such that

$$A = QR \quad \textbf{"full" QR Decomposition}$$

3.4.2 Computation of QR decomposition

To compute the QR decomposition, two methods are proposed:

- Householders Reflections
- Given Rotations

both methods rely on **successive orthogonal transformations** annihilating columns entries in order to form an upper triangular matrix, i.e

$$\underbrace{Q_{n-1}Q_{n-2},\dots,Q_1}_{:=Q^T} A = R$$

Recalling that composition of orthogonal matrices of the same size is again orthogonal.

3.4.3 Eigen: QR Decomposition for Solving LS Systems

In Eigen the so called **Householder** reflexion methods is implemented (together ith other methods see Eigen Documentation). We point out the asymptotic complexity for QR decomposition is

$$\mathcal{O}(mn^2)$$

Hence for solving a least square solution we require

$$\underbrace{\mathcal{O}(mn^2)}_{\text{QR decomp.}} + \underbrace{\mathcal{O}(n^2)}_{\text{back substitution}}$$

CODE:: Solve Least Squares with QR Decomposition:

```
auto x = A.householderQr().solve(b);
auto residual = (A*x -b).norm();
```

3.5 Moore-Penrose Pseudoinverse

What if non-trivial Kernel of $A \in \mathbb{R}^{m,n}$?
Idea: Single out one solution from the solution space.

Definition 3.2 Generalized Solution of Linear System of Equations

The generalized solution $x^+ \in \mathbb{R}^n$ of a linear system of equations $Ax = b, A \in \mathbb{R}^{m,n}, b \in \mathbb{R}^m$, is defined as

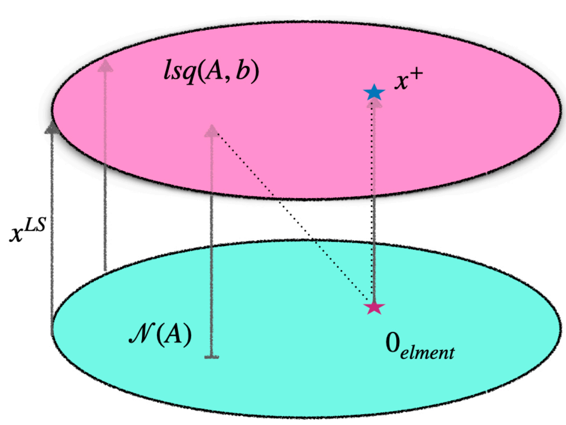
$$x^+ := \arg\min\{x \in lsq(A,b), \|x\|_2\}$$

The general solution uniquely identifies the solution with the minimum 2-norm from the solution space.

Geometric intuition:
Assume a solution to the normal equation x^{LS}, then for any $y \in \mathcal{N}(A^TA) = \mathcal{N}(A) \quad x^{LS} + y$ is also a solution

$$A^T(A(x^{LS} + y) - b) = 0 \Leftrightarrow A^TAx + \underbrace{A^TAy}_{=0} - A^Tb = 0$$

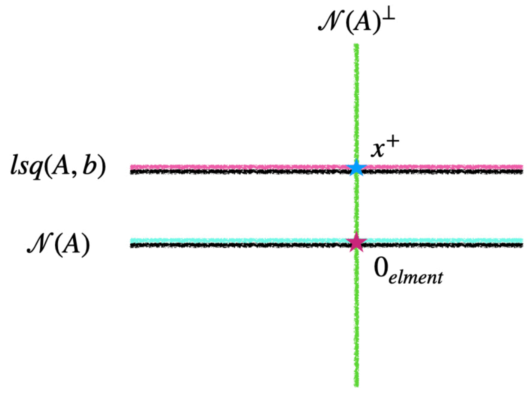
hence the space of solutions $lsq(A,b)$ is an affine subspace parallel to $\mathcal{N}(A)$. Notice that the zero element belongs to $\mathcal{N}(A)$, hence the LS solution with minimal 2-norm, i.e the **generalized solution** is the closest to the zero vector (see picture below)



Also notice that the generalised solution belong to the orthogonal complement of the null space $\mathcal{N}(A)$ i.e

$$x^{LS} \in \mathcal{N}(A)^\perp = \{x \in \mathbb{R}^n : \langle x, y\rangle = 0 \quad \forall y \in \mathcal{N}(A)\}$$

The last equation is supported by the picture below:



Hence given a basis $\{v_1,\dots,v_k\} \subset \mathbb{R}^n$ of $\mathcal{N}(A)^\perp, k := dim\mathcal{N}(A)^\perp = n - dim\mathcal{N}(A)$ we can find $y \in \mathbb{R}^k$ such that

$$x^+ = Vy \quad , V := [v_1,\dots,v_k] \in \mathbb{R}^{n,k}$$

Hence we can rewrite our linear system for a generalised solution as follow

$$\begin{aligned}
&Ax^+ = b \Leftrightarrow AVy = b \\
&\text{N.EQ} \quad \underbrace{V^T A^T AV}_{k\times k} y = V^T A^T b && \Leftrightarrow \\
&y = (V^T A^T AV)^{-1} V^T A^T b && \Leftrightarrow \\
&x^+ = Vy = V(V^T A^T AV)^{-1} V^T A^T b =: A^+ b
\end{aligned}$$

Note that by **construction** the k by k matrix $V^T A^T AV$ has a trivial nullspace meaning it is regular. Recalling

$$\mathcal{N}(V^T A^T AV) = \mathcal{N}(AV) = \{x \in \mathbb{R}^k : AVx = 0\}$$

we see with the help of picture above (3.5) that $Vx \in \mathcal{N}(A)^\perp$ hence the only vector $z \in \mathcal{N}(A)^perp$ yielding $Az = 0$ is the zero vector $z = 0$.

Theorem 3.3 Formula for Generalised Solution

Given $A \in \mathbb{R}^{m,n}, b \in \mathbb{R}^m$, the generalised solution x^+ of the linear system of equation $Ax = b$ is given by

$$x^+ = V(V^T A^T AV)^{-1} V^T A^T b =: A^+ b$$

where A^+ is called the **pseudo inverse** of the matrix A

3.6 Not Discussed in This Script

Due to the restricted amount of time, some more advanced subchapters that are also important for succeeding in the exam could not be covered. Please refer to the script of the main lecture, particularly for the following

- Modifications Techniques for QR-Decomposition
- Householder Reflexion
- Given Rotations

4 SVD Based Methods for Least Square Problems

Review:

- Section 1.5 on SVD
- Section 3.5 on the moore pseudo inverse for **generalised** least squares solution

4.1 SVD for Solving General Least Squares Problems

Assume the most general setting

$$Ax = b \in \mathbb{R}^m \quad \text{with} \quad A \in \mathbb{R}^{m,n} \quad x \in \mathbb{R}^n \quad rank(A) = r \leq \min\{m,n\}$$

i.e **no full rank assumption**.

We have the following full SVD decomposition:

$$A = U\Sigma V =: \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}$$

where

$$U_1 \in \mathbb{R}^{m,r} \quad , \quad U_2 \in \mathbb{R}^{m,m-r}$$
$$V_1 \in \mathbb{R}^{n,r} \quad , \quad V_2 \in \mathbb{R}^{n,n-r}$$
$$\Sigma_r = diag\{\sigma_1, \dots, \sigma_r\} \in \mathbb{R}^{r,r}$$

Hence the Least Squares objective read

$$\|Ax - b\|_2 = \| \begin{bmatrix} \Sigma_r V_1^T x - U_1^T b \\ U_2^T b \end{bmatrix} \|_2$$

hence to a minimizer of the above would be a solution to

$$\arg\min_{x \in \mathbb{R}^n} \|\Sigma_r V_1^T x - U_1^T b\|_2$$

Now if $r < n$ we have an underdetermined system i.e no unique solution of the system, i.e we need the **generalised** least square solution x^+.
We now from 3.5 and from lemma 1.1 that the following holds

$$V_1^T x^+ = \Sigma_r^{-1} U_1^T b \Leftrightarrow \quad x^+ \in \mathcal{N}(V_1^T)^\perp \underbrace{=}_{1.1} \mathcal{R}(V_1)$$

Hence we can write the following

$$\exists y \in \mathcal{R}(V_1) \quad \text{s.t} \quad x^+ = V_1 y \implies$$
$$V_1^T x^+ = V_1^T V_1 y = y = \Sigma_r^{-1} U_1^T b$$

Last equation implies the following result which yield a more general version of the pseudo-inverse matrix

$$x^+ = V_1 \Sigma_r^{-1} U_1^T b = A^+ b$$

Theorem 4.1 Pseudo Inverse and SVD

If matrix $A \in \mathbb{R}^{m,n}$ with rank r admits the following SVD

$$A = U\Sigma V =: \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}$$

then its **Moore Pseudo Inverse** is given by

$$A^+ = V_1 \Sigma_r^{-1} U_1^T b$$

Note: The same yield for $A \in \mathbb{C}^{m,n}$
Note: The proof of the theorem is given in the explanations above.

4.2 SVD in Eigen

Note: try to code it yourself !

C++-code 3.4.2.1: Computing SVDs in EIGEN ⇢ GITLAB

```cpp
#include <Eigen/SVD>
using MatrixXd = Eigen::MatrixXd;
// Computation of (full) SVD A = UΣV^H → Thm. 3.4.1.1
// SVD factors are returned as dense matrices in natural order
std::tuple<MatrixXd,MatrixXd,MatrixXd> svd_full(const MatrixXd& A) {
  Eigen::JacobiSVD<MatrixXd> svd(A, Eigen::ComputeFullU | Eigen::ComputeFullV);
  MatrixXd U = svd.matrixU(); // get unitary (square) matrix U
  MatrixXd V = svd.matrixV(); // get unitary (square) matrix V
  VectorXd sv = svd.singularValues(); // get singular values as vector
  MatrixXd Sigma = MatrixXd::Zero(A.rows(), A.cols());
  const unsigned p = sv.size(); // no. of singular values
  Sigma.block(0,0,p,p) = sv.asDiagonal(); // set diagonal block of Σ
  return std::tuple<MatrixXd,MatrixXd,MatrixXd>(U,Sigma,V);
}

// Computation of economical (thin) SVD A = UΣV^H, see (3.4.1.5)
// SVD factors are returned as dense matrices in natural order
std::tuple<MatrixXd,MatrixXd,MatrixXd> svd_eco(const MatrixXd& A) {
  Eigen::JacobiSVD<MatrixXd> svd(A, Eigen::ComputeThinU | Eigen::ComputeThinV);
  MatrixXd U = svd.matrixU(); // get matrix U with orthonormal columns
  MatrixXd V = svd.matrixV(); // get matrix V with orthonormal columns
  VectorXd sv = svd.singularValues(); // get singular values as vector
  MatrixXd Sigma = sv.asDiagonal(); // build diagonal matrix Σ
  return std::tuple<MatrixXd,MatrixXd,MatrixXd>(U,Sigma,V);
}
```