Chapter 1

Linear Systems of Equations

Introduction

Linear Systems of Equations: Example

- Consider an open economy with two very basic industries: **goods** and **services**.

- To produce €1 of their products ($\rightsquigarrow$ **internal demand**),
  - the **goods** industry must spend €0.40 on goods and €0.20 on services
  - the **services** industry must spend €0.30 on goods and €0.30 on services

- Assume also that during a period of one week, the economy has an **external demand** of **€75,000 in goods** and **€50,000 in services.**

- **Question:** How much should each sector produce to meet both **internal and external demand?**

Formulating the equations

- Let x_1 **be the Euro value of goods** produced and x_2 **the Euro value of services** produced.

- The total Euro value of **goods consumed** is $\underbrace{0.4x_1 + 0.3x_2}_{\text{internal}} + \underbrace{75000}_{\text{external}}$.

- The total Euro value of **services consumed** is $0.2x_1 + 0.3x_2 + 50000$.

- If we assume that **production equals consumption**, then we get

$$\begin{aligned} x_1 &= 0.4x_1 + 0.3x_2 + 75000 \\ x_2 &= 0.2x_1 + 0.3x_2 + 50000 \end{aligned} \Leftrightarrow \begin{bmatrix} 0.6 & -0.3 \\ -0.2 & 0.7 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 75000 \\ 50000 \end{bmatrix}$$

- The **solution** is $x_1 = 187500, \quad x_2 = 125000$. Can be checked easily...

Formal solution

- Main inside: triangular systems can be easily solved by substitution
  $\rightsquigarrow$ transform system to (upper) triangular.

- Do all operations on **augmented matrix** $\begin{bmatrix} A & \boldsymbol{b} \end{bmatrix}$.

$$\begin{bmatrix} \color{red}{0.6} & -0.3 & 75000 \\ -0.2 & 0.7 & 50000 \end{bmatrix} \Rightarrow \begin{bmatrix} 0.6 & -0.3 & 75000 \\ -0.6 & 2.1 & 150000 \end{bmatrix} \Rightarrow \begin{bmatrix} 0.6 & -0.3 & 75000 \\ 0 & 1.8 & 225000 \end{bmatrix}$$

$$\Rightarrow 1.8 x_2 = 225000 \Rightarrow x_2 = 125000 \Rightarrow x_1 = 187500.$$

- **Elimination step:** subtract a multiple of eq. 2 from eq. 1.
  $\rightsquigarrow$ **Gaussian elimination**

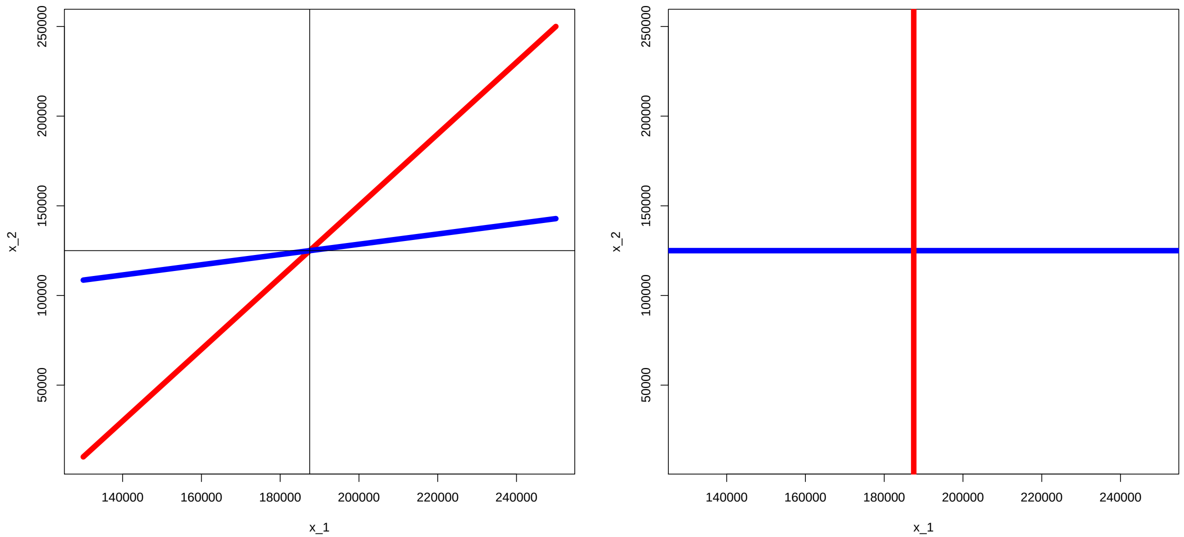
Formal solution

Instead of substituting, we could have continued with the elimination:

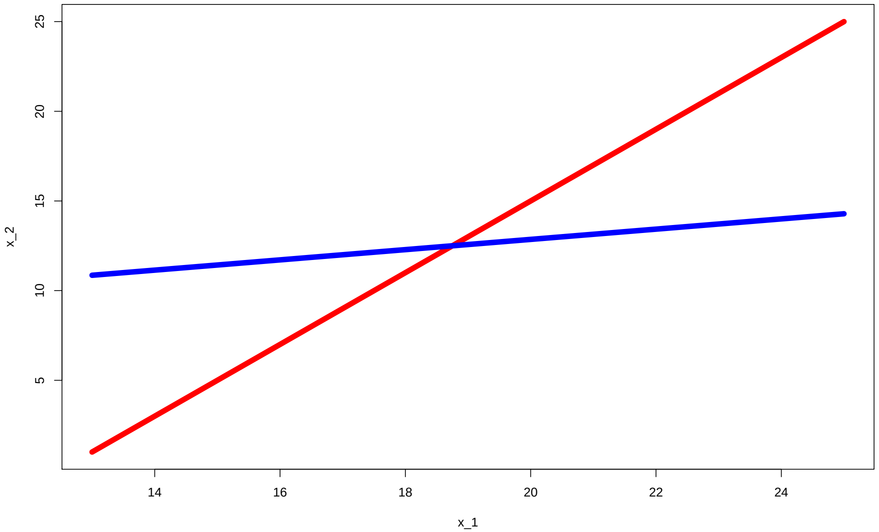
$$\begin{bmatrix} 0.6 & -0.3 & 75000 \\ 0 & {\color{red}1.8} & 225000 \end{bmatrix} \Rightarrow \begin{bmatrix} 3.6 & -1.8 & 450000 \\ 0 & 1.8 & 225000 \end{bmatrix} \Rightarrow \begin{bmatrix} 3.6 & 0 & 675000 \\ 0 & 1.8 & 225000 \end{bmatrix}$$

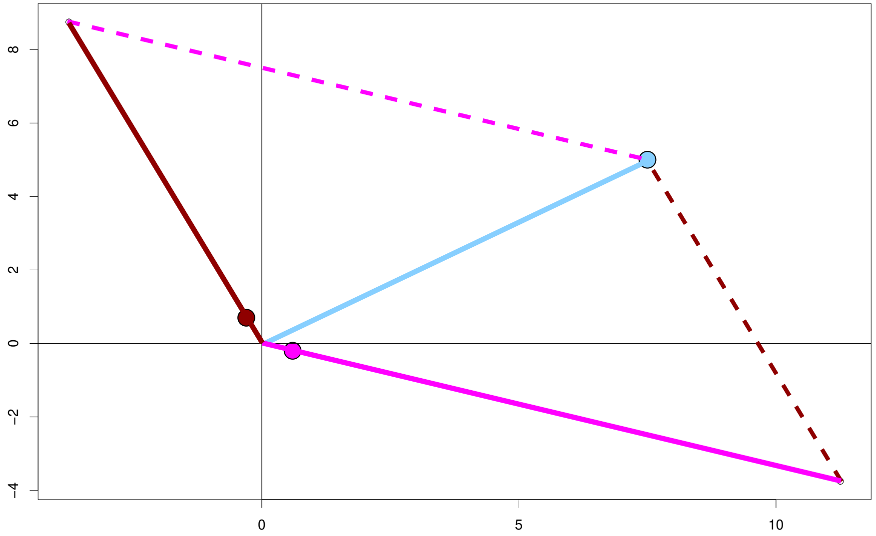
$\rightsquigarrow$ **Gauss-Jordan elimination**

Geometric interpretation: have transformed original equations into a new space in which they are aligned with the coordinate axis:



Row and column view

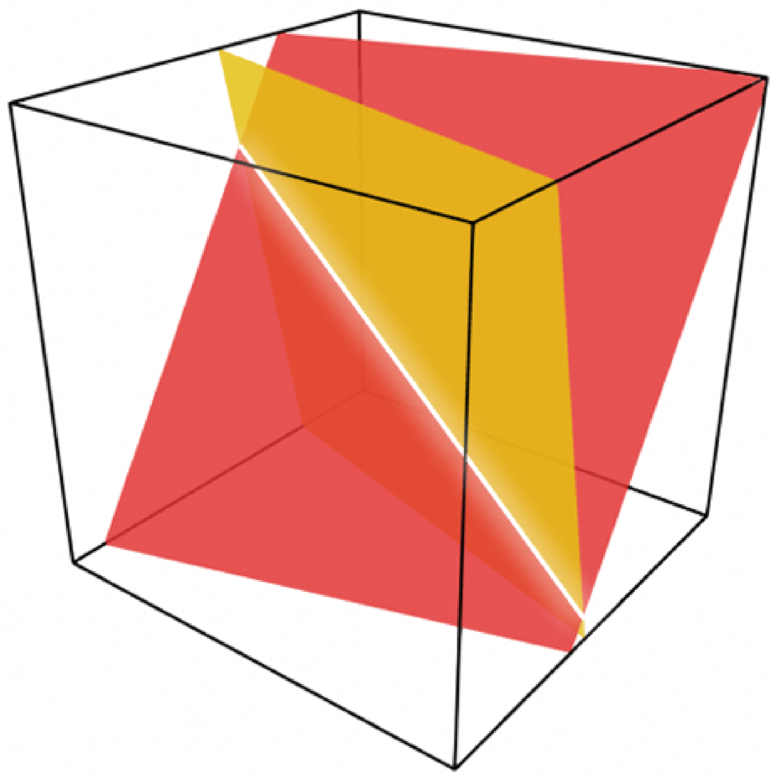




$$\begin{aligned} 0.6x_1 - 0.3x_2 &= 7.5 \\ -0.2x_1 + 0.7x_2 &= 5 \end{aligned} \Leftrightarrow \begin{bmatrix} 0.6 & -0.3 \\ -0.2 & 0.7 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 7.5 \\ 5 \end{bmatrix}$$

Column view (right): $\begin{bmatrix} 0.6 \\ -0.2 \end{bmatrix} x_1 + \begin{bmatrix} -0.3 \\ 0.7 \end{bmatrix} x_2 = \begin{bmatrix} 7.5 \\ 5 \end{bmatrix}$

Some examples and concepts



The solution set for two equations in three variables is usually a line.

This is an example of an **underdetermined** system.

Chapter 1

Linear Systems of Equations

Linear Algebra I

Vector spaces and subspaces

A **subspace** of a vector space is a **nonempty subset** that satisfies the

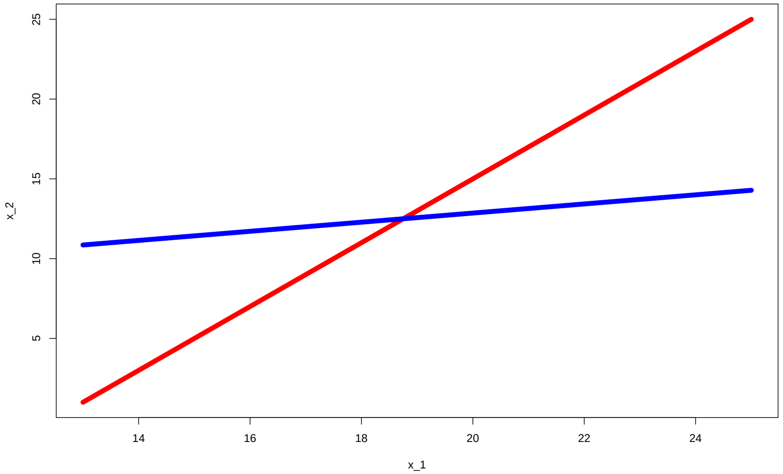
Requirements for a **vector space:**
"Linear combinations stay in the subspace"
(i) If we add any vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ in the subspace,
$\boldsymbol{x} + \boldsymbol{y}$ is in the subspace.
(ii) If we multiply any vector $\boldsymbol{x}$ in the subspace by **any scalar** c,
$c\boldsymbol{x}$ is in the subspace.

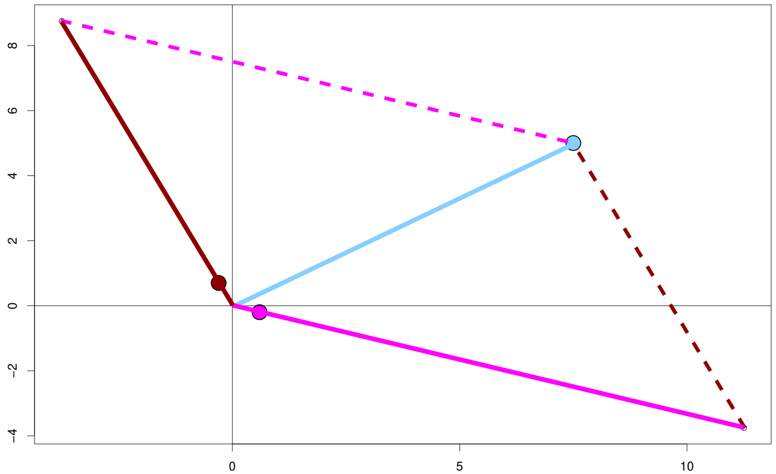
Rule (ii) with $c = 0 \rightsquigarrow$ **Every subspace contains the zero vector.**

The smallest subspace Z contains only the zero vector.
Why? Rules (i) and (ii) are satisfied:
$\mathbf{0} + \mathbf{0}$ is in this one-point space, and so are all multiples $c\mathbf{0}$.

The **largest subspace** is the whole of the original space.

The column space of a matrix





Column view (right): $\begin{bmatrix} 0.6 \\ -0.2 \end{bmatrix} x_1 + \begin{bmatrix} -0.3 \\ 0.7 \end{bmatrix} x_2 = \begin{bmatrix} 7.5 \\ 5 \end{bmatrix}$

The **column space $C(A)$** contains all linear combinations of the columns of $A_{m\times n} \rightsquigarrow$ subspace of $\mathbb{R}^m$.

The system $Ax = b$ is solvable iff b is in the column space of A.

Nullspace

A system with right-hand side $\boldsymbol{b} = \mathbf{0}$ always allows the solution $\boldsymbol{x} = \mathbf{0}$, but there may be **infinitely many other solutions.**

The solutions to $A\boldsymbol{x} = \mathbf{0}$ form the **nullspace of A.**

The **nullspace** $N(A)$ of a matrix A consists of all vectors $\boldsymbol{x}$ such that $A\boldsymbol{x} = \mathbf{0}$. It is a subspace of $\mathbb{R}^n$:
(i) If $A\boldsymbol{x} = \mathbf{0}$ and $A\boldsymbol{x}' = \mathbf{0}$, then $A(\boldsymbol{x} + \boldsymbol{x}') = \mathbf{0}$.
(ii) If $A\boldsymbol{x} = \mathbf{0}$ then $A(c\boldsymbol{x}) = cA\boldsymbol{x} = \mathbf{0}$.

For an invertible matrix A:

- $N(A)$ **contains only** $\boldsymbol{x} = \mathbf{0}$ (multiply $A\boldsymbol{x} = \mathbf{0}$ by A^{-1}).
- **The column space is the whole space.**
  ($A\boldsymbol{x} = \boldsymbol{b}$ has a solution for every $\boldsymbol{b}$)
- **The columns of A are independent.**

Nullspace

Singular matrix example:

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 6 \end{bmatrix}.$$

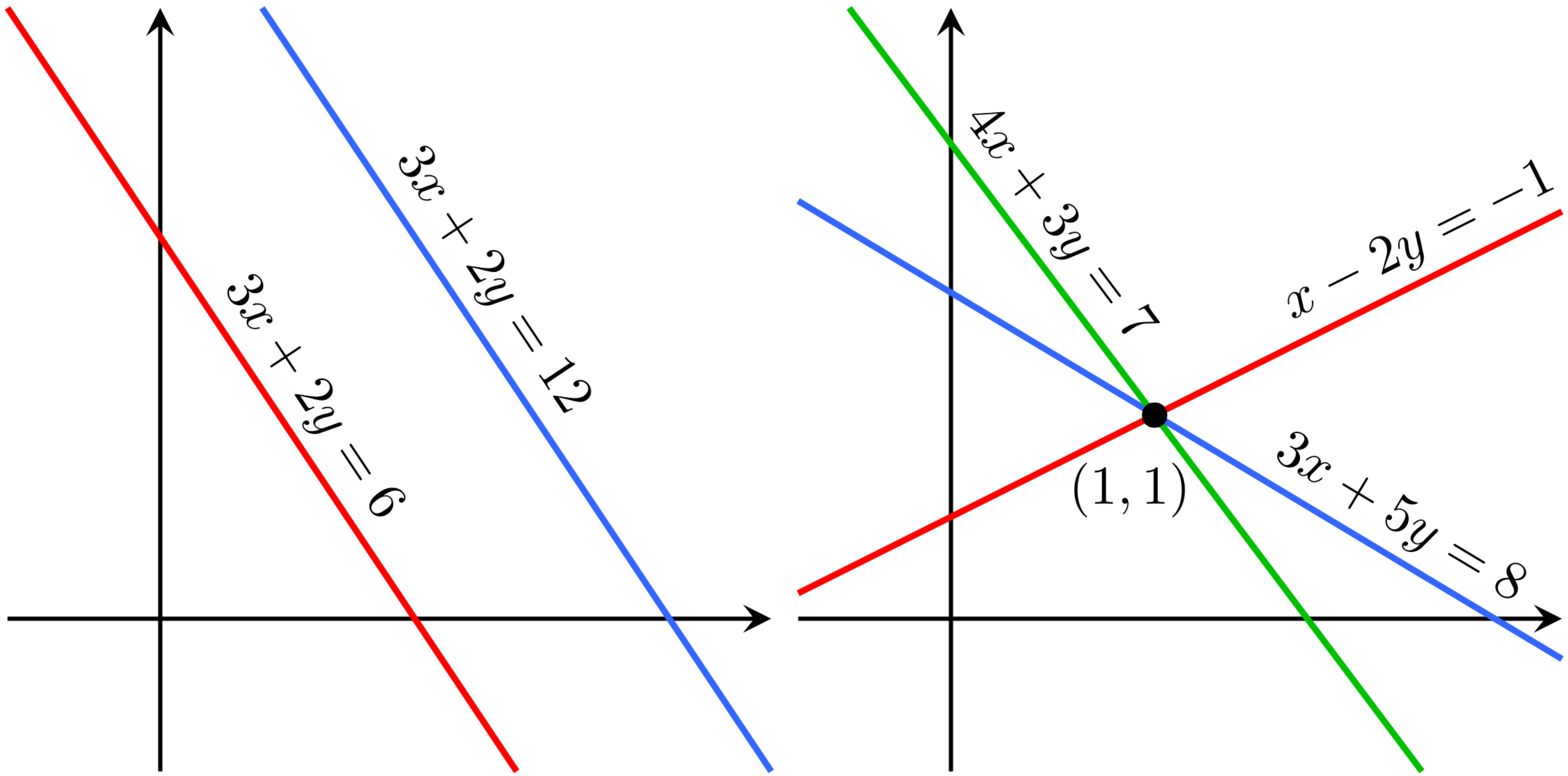
Consider $A\boldsymbol{x} = \mathbf{0}$: Any pair that fulfills $x_1 + 2x_2 = 0$ is a solution.
This line is the **one-dimensional nullspace** $N(A)$.

Choose one point on this line as a "special" solution
$\rightsquigarrow$ all points on the line are multiples.

Let $\boldsymbol{x}_p$ be a particular solution and $\boldsymbol{x}_n \in N(A)$:
The solutions to all linear equations have the form $\boldsymbol{x} = \boldsymbol{x}_p + \boldsymbol{x}_n$.

Proof: $A\boldsymbol{x}_p = \boldsymbol{b}$ and $A\boldsymbol{x}_n = \mathbf{0}$ produce $A(\boldsymbol{x}_p + \boldsymbol{x}_n) = \boldsymbol{b}$.

Inconsistent equations and linear dependency



The equations $3x + 2y = 6$ and $3x + 2y = 12$ are **inconsistent**:
$\boldsymbol{b}$ is **not** in the $C(A) \rightsquigarrow$ no solution exists!

$x - 2y = -1$, $3x + 5y = 8$, and $4x + 3y = 7$ are **linearly dependent**:
$\boldsymbol{b} \in C(A) \rightsquigarrow$ solution exists, but two equations are sufficient.

Linear Dependence

The vectors $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n\}, \boldsymbol{v}_i \in V$, are **linearly dependent**, if there exist a finite number of **distinct vectors** $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_k$
and scalars $a_1, a_2, \ldots, a_k$, **not all zero**, such that

$$a_1\boldsymbol{v}_1 + a_2\boldsymbol{v}_2 + \cdots + a_k\boldsymbol{v}_k = \mathbf{0}.$$

Linear dependence:

Not all of the scalars are zero $\rightsquigarrow$ at least one is non-zero (say a_1):

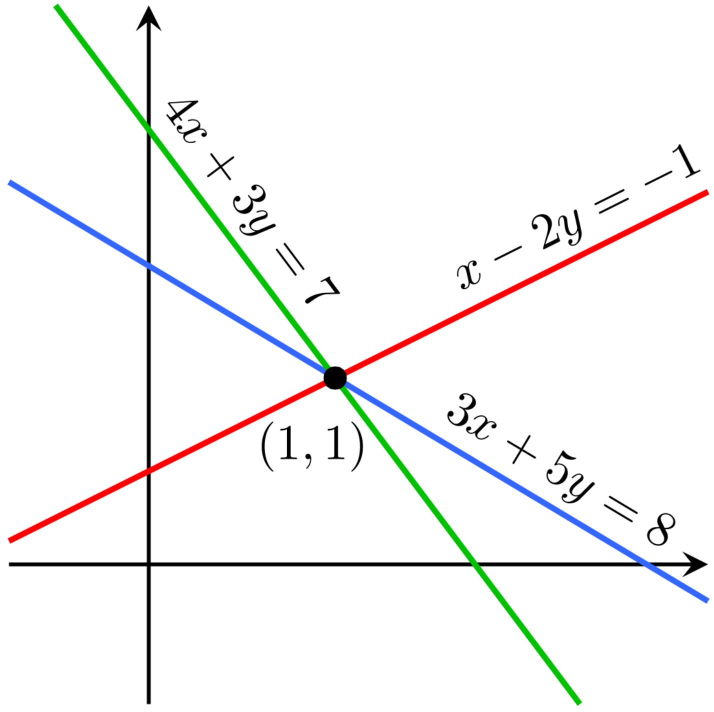
$$\boldsymbol{v}_1 = \frac{-a_2}{a_1}\boldsymbol{v}_2 + \cdots + \frac{-a_k}{a_1}\boldsymbol{v}_k.$$

Thus, v_1 is a **linear combination of the remaining vectors.**

Linear Dependence Example

Matrix form: $A\boldsymbol{x} = \boldsymbol{b}$

$$\begin{bmatrix} 1 & -2 \\ 3 & 5 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -1 \\ 8 \\ 7 \end{bmatrix}$$



Row vectors of A are linearly dependent

$$\begin{bmatrix} 1 \\ -2 \end{bmatrix} + \begin{bmatrix} 3 \\ 5 \end{bmatrix} - \begin{bmatrix} 4 \\ 3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Linear Independence

The vectors $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n\}$ are **linearly independent** if the equation

$$a_1\boldsymbol{v}_1 + a_2\boldsymbol{v}_2 + \cdots + a_n\boldsymbol{v}_n = \mathbf{0}$$

can **only** be satisfied by $a_i = 0$ for $i = 1, \ldots, n$.

- This implies that **no vector in the set can be represented as a linear combination of the remaining vectors** in the set.

- In other words: A set of vectors is linearly independent if the only representations of $\mathbf{0}$ as a linear combination of the vectors is the **trivial representation** in which all scalars a_i are zero.

- Any set of $n > m$ vectors in $\mathbb{R}^m$ must be linearly dependent.

Span and Basis

A set of vectors **spans a space** if their linear combinations fill the space.

Special case: the columns of a matrix A span its **column space** $C(A)$. They might be **independent** $\rightsquigarrow$ **basis of $C(A)$.**

A basis for a vector space is a sequence of vectors such that:
(i) the basis vectors are linearly independent, and
(ii) they span the space.

Immediate consequence: There is **one and only one way** to write an element of the space as a combination of the basis vectors.

The dimension of a space is the number of vectors in every basis.

The dimension of $C(A)$ is called the (column-)**rank** of A.

The dimension of $N(A)$ is called the **nullity** of A.

Nullspace and Independence

Example: The columns of this triangular matrix are linearly independent:

$$A = \begin{bmatrix} 3 & 4 & 2 \\ 0 & 1 & 5 \\ 0 & 0 & 2 \end{bmatrix}.$$

Why? Solving $A\boldsymbol{x} = \mathbf{0} \rightsquigarrow$ look for combination of the columns that produces $\mathbf{0}$:

$$c_1 \begin{bmatrix} 3 \\ 0 \\ 0 \end{bmatrix} + c_2 \begin{bmatrix} 4 \\ 4 \\ 0 \end{bmatrix} + c_3 \begin{bmatrix} 2 \\ 5 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Independence: show that c_1, c_2, c_3 are all forced to be zero.

Last equation $\rightsquigarrow c_3 = 0$. Next equation gives $c_2 = 0$, substituting into 1st eq.: $c_1 = 0$.

The nullspace of A contains only the zero vector $c_1 = c_2 = c_3 = 0$.

The columns of A are independent exactly when $N(A) = \{\mathbf{0}\}$.

Then, the dimension of the column space (the rank) is n.

We say that the matrix has **full rank**.

Chapter 1

Linear Systems of Equations

Gauss-Jordan Elimination

The invertible case: Gauss-Jordan elimination

Assume A is invertible $\rightsquigarrow$ a solution is guaranteed to exist: $\boldsymbol{x} = A^{-1}\boldsymbol{b}$.
Sometimes we also want to find the inverse itself.
Then **Gauss-Jordan elimination** is the method of choice.

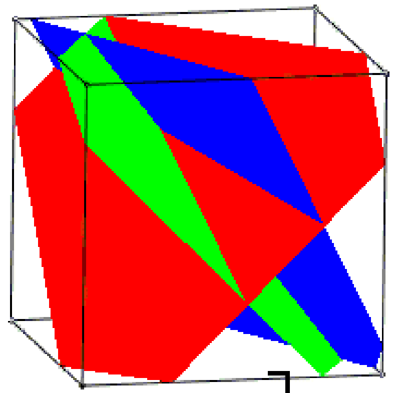
- **PRO**
  - produces both the solution(s), for (multiple) $\boldsymbol{b}_j$, **and the inverse** A^{-1}
  - numerically stable if **pivoting** is used $\rightsquigarrow$ will be discussed later...
  - **straightforward, understandable** method

- **CON**
  - all right hand sides $\boldsymbol{b}_j$ must be known before the elimination starts.
  - **three times slower** than alternatives when inverse is not required

The invertible case: Gauss-Jordan elimination

- **Augmented matrix** $A' = [A, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_j, I_n]$

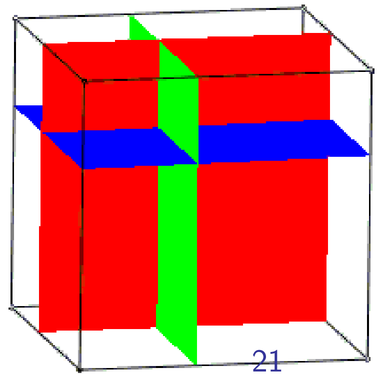
- Idea:
  Define $B = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_j] \quad X = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_j]$
  $[A, B, I] \Rightarrow A^{-1}[A, B, I] = [I X A^{-1}]$.



- Example:

$$A = \begin{bmatrix} 1 & 3 & -2 \\ 3 & 5 & 6 \\ 2 & 4 & 3 \end{bmatrix}, \quad B = \begin{bmatrix} 5 \\ 7 \\ 8 \end{bmatrix} \Rightarrow [A, B, I] = \begin{bmatrix} 1 & 3 & -2 & 5 & 1 & 0 & 0 \\ 3 & 5 & 6 & 7 & 0 & 1 & 0 \\ 2 & 4 & 3 & 8 & 0 & 0 & 1 \end{bmatrix}$$

$$\Rightarrow [I, X, A^{-1}] = \begin{bmatrix} 1 & 0 & 0 & -15 & \frac{9}{4} & \frac{17}{4} & -7 \\ 0 & 1 & 0 & 8 & -\frac{3}{4} & -\frac{7}{4} & 3 \\ 0 & 0 & 1 & 2 & -\frac{1}{2} & -\frac{1}{2} & 1 \end{bmatrix}$$



Gauss-Jordan: Simplest Form

Main idea: Cycle through columns of A ($\leadsto$ pivot column) and select entry on diagonal ($\leadsto$ pivot element).

Then normalize pivot row and introduce zeros below and above.

Pivot column: 1, pivot element $= 1$. Divide pivot row by pivot element

$$\begin{bmatrix} \color{red}{1} & 3 & -2 & 5 & 1 & 0 & 0 \\ 3 & 5 & 6 & 7 & 0 & 1 & 0 \\ 2 & 4 & 3 & 8 & 0 & 0 & 1 \end{bmatrix}$$

For all other rows: (i) store element in pivot column,
(ii) subtract pivot row multiplied with this element

$$\begin{bmatrix} 1 & 3 & -2 & 5 & 1 & 0 & 0 \\ \color{blue}{0} & \color{red}{-4} & 12 & -8 & -3 & 1 & 0 \\ \color{blue}{0} & -2 & 7 & -2 & -2 & 0 & 1 \end{bmatrix}$$

Proceed to pivot column 2 with pivot element $=$ -4

Gauss-Jordan: Simplest Form

Proceed to pivot column 2 with pivot element = -4

$$\begin{bmatrix} 1 & 0 & 7 & -1 & -1.25 & 0.75 & 0 \\ 0 & 1 & -3 & 2 & 0.75 & -0.25 & 0 \\ 0 & 0 & 1 & 2 & -0.5 & -0.5 & 1 \end{bmatrix}$$

After elimination in column 3 with pivot = 1

$$\begin{bmatrix} 1 & 0 & 0 & -15 & 2.25 & 4.25 & -7 \\ 0 & 1 & 0 & 8 & -0.75 & -1.75 & 3 \\ 0 & 0 & 1 & 2 & -0.5 & -0.5 & 1 \end{bmatrix}$$

Now we have transformed A to the identity matrix I.

This is a special case of the **reduced row Echelon form** (more on this later).

The **solution vector** is the 4th column $\boldsymbol{x} = (-15, 8, 2)^t$.

Note that we have overwritten the original $\boldsymbol{b} \rightsquigarrow$ no need to allocate further memory.

The **inverse A^{-1}** is the right 3×3 block.

Gauss-Jordan elimination

Elementary operations (they **do not change** the solution):

1. **Replace a row** by a linear combination of itself and any other row(s).
2. **Interchange two rows**.
3. **Interchange two columns** and **corresponding rows** of $\boldsymbol{x}$.

Basic G-J elimination uses **only operation #1** but...
Elimination **fails mathematically** when a **zero pivot** is encountered
$\leadsto$ pivoting is essential to avoid **total failure** of the algorithm.

Example: Try $A\boldsymbol{x} = \boldsymbol{b}$ with

$$A = \begin{bmatrix} 2 & 4 & -2 & -2 \\ 1 & 2 & 4 & -3 \\ -3 & -3 & 8 & -2 \\ -1 & 1 & 6 & -3 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} -4 \\ 5 \\ 7 \\ 7 \end{bmatrix}$$

Chapter 1

Linear Systems of Equations

Linear Systems: Numerical Issues

The need for pivoting

- Elimination **fails mathematically** when a **zero pivot** is encountered

- and **fails numerically** with a **too-close-to-zero pivot** (we will see why in a minute...)

- The fix is **partial pivoting**
  - use **operation #2** to place a **desirable** pivot entry in the current row
  - **usually sufficient** for stability

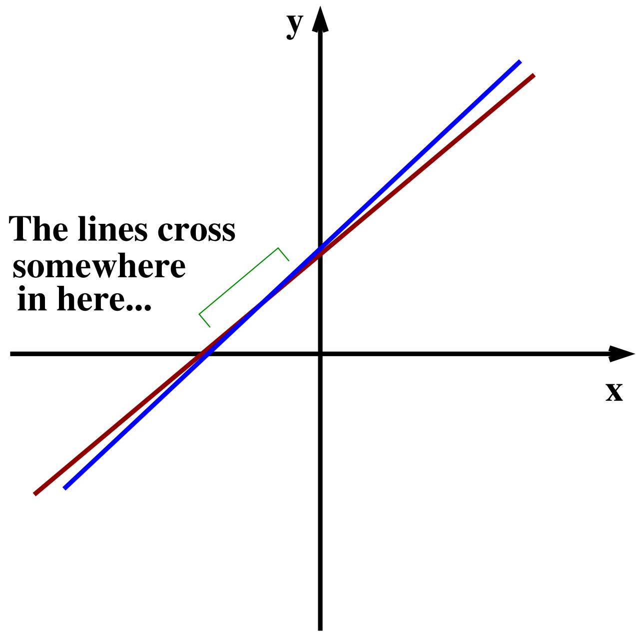
- Using **operation #3** as well gives **full pivoting**

Linear systems: numerical issues

If a system is **too close to linear dependence**

- an algorithm may **fail altogether** to get a solution

- **round off errors** can produce apparent linear dependence at some point in the solution process
  $\rightsquigarrow$ **accumulated roundoff errors**
  can dominate in the solution
  $\rightsquigarrow$ an algorithm may still work
  but **produce nonsense**.



When is sophistication necessary?

- Sophisticated methods can **detect and correct** numerical pathologies

- Rough guide for a **"not-too-singular"** $n \times n$ system:
  - $n < 20 \ldots 50$ **single** precision
  - $n < 200 \ldots 300$ **double** precision
  - $n = 1000$ OK if equations are **sparse**
    (special techniques take advantage of sparsity)

- **Close-to-singular** can be a problem even for **very small** systems

- But...what is the underlying reason for these numerical problems?

Floating Point Numbers: float, double

- `float` similar to **scientific notation**
  $\pm$ D.DDDD $\times 10^{\mathsf{E}}$
  - D.DDDD has leading mantissa digit $\neq 0$
  - D.DDDD has **fixed** number of mantissa digits.
  - E is **signed integer**.

- **Precision varies:** precision of 1.000×10^{-2} is 100 times higher than precision of 1.000×10^{0}.

- The **bigger the number**, the **less precise**:
  $1.000 \times 10^{4} + 1.000 \times 10^{0} = 1.000 \times 10^{4}$!!!

Simple Data Types: float, double (2)

Technical Realization (IEEE Standard 754)

- 32 bit (`float`) or 64 bit (`double`)

- `float`:

  1 bit *sign* $(s \in \{0, 1\})$
  8 bit *exponent* $(e \in \{0, 1, \ldots, 255\})$ (like before, but basis 2!)
  23 bit *mantissa* $(m \in \{0, 1, \ldots, 2^{23} - 1\})$

S	EEEEEEEE	MMMMMMMMMMMMMMMMMMMMMMM

- `double`: 1 bit sign, 11 bit exponent, 52 bit mantissa

S	EEEEEEEEEEE	MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM

Floating Point Arithmetic: Problems

- **Fixed number of mantissa bits $\Rightarrow$ limited precision:**
  If `a` $\gg$ `b` $\Rightarrow$ `a+b = a`.

- Iterated addition of small numbers (like `a=a+b` with `a` $\gg$ `b`) can lead to a **huge error**: at some point, `a` does not increase anymore, **independent of the number of additions**.

- `double` is better, but needs **two times more memory**.

- **Machine epsilon** (informal definition): The smallest number ϵ_m which when added to 1 gives something different than 1.
  Float (23 mantissa bits): $\epsilon_m \approx 2^{-23} \approx 10^{-7}$,
  Double (52 mantissa bits): $\epsilon_m \approx 2^{-52} \approx 10^{-16}$.

Chapter 1

Linear Systems of Equations

Elementary Matrices

Elementary matrices: Row-switching transformations

Switches row i and row j. Example:

$$R_{35}A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{0} & 0 & \mathbf{1} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{1} & 0 & \mathbf{0} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \\ a_{41} & a_{42} \\ a_{51} & a_{52} \\ a_{61} & a_{62} \\ a_{71} & a_{72} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{51} & a_{52} \\ a_{41} & a_{42} \\ a_{31} & a_{32} \\ a_{61} & a_{62} \\ a_{71} & a_{72} \end{bmatrix}$$

The inverse of this matrix is itself: $R_{ij}^{-1} = R_{ij}$

Elementary matrices: Row-multiplying transformations

Multiplies all elements on row i by $m \neq 0$.

$$R_i(m) = \begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & m & & & \\ & & & & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{bmatrix}$$

The inverse of this matrix is: $R_i(m)^{-1} = R_i(1/m)$.

Elementary matrices: Row-addition transformations

Subtracts row j multiplied by m from row i.

$$R_{ij}(m) = \begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & \ddots & & & \\ & & -m & & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{bmatrix}$$

The inverse of this matrix is: $R_{ij}(m)^{-1} = R_{ij}(-m)$.

Row operations

- Elementary row operations correspond to **left-multiplication** by elementary matrices:

$$
\begin{aligned}
A \cdot \boldsymbol{x} &= \boldsymbol{b} \\
(\cdots R_3 \cdot R_2 \cdot R_1 \cdot A) \cdot \boldsymbol{x} &= \cdots R_3 \cdot R_2 \cdot R_1 \cdot \boldsymbol{b} \\
(I_n) \cdot \boldsymbol{x} &= \cdots R_3 \cdot R_2 \cdot R_1 \cdot \boldsymbol{b} \\
\boldsymbol{x} &= \cdots R_3 \cdot R_2 \cdot R_1 \cdot \boldsymbol{b}
\end{aligned}
$$

- $\boldsymbol{x}$ can be built-up **in stages** since the R matrices are multiplied in the **order of acquisition**.

- **Inverse** matrix A^{-1} and **solution** $\boldsymbol{x}$ can be built up in the storage locations of A and $\boldsymbol{b}$ respectively.

Column operations

Elementary column operations correspond to **right-multiplication**: transform rows of A^t, then transpose: $(RA^t)^t = AR^t = AC \rightsquigarrow C = R^t$. Note that $(AB)^t = B^t A^t$.

$$\begin{aligned}
A \cdot \boldsymbol{x} &= \boldsymbol{b} \\
A \cdot C_1 \cdot C_1^{-1} \cdot \boldsymbol{x} &= \boldsymbol{b} \\
A \cdot C_1 \cdot C_2 \cdot C_2^{-1} \cdot C_1^{-1} \cdot \boldsymbol{x} &= \boldsymbol{b} \\
(A \cdot C_1 \cdot C_2 \cdot C_3 \cdots) \cdot (\cdots C_3^{-1} \cdot C_2^{-1} \cdot C_1^{-1}) \cdot \boldsymbol{x} &= \boldsymbol{b} \\
(I_n) \cdot (\cdots C_3^{-1} \cdot C_2^{-1} \cdot C_1^{-1}) \cdot \boldsymbol{x} &= \boldsymbol{b} \\
\boldsymbol{x} &= C_1 \cdot C_2 \cdot C_3 \cdots \cdot \boldsymbol{b}
\end{aligned}$$

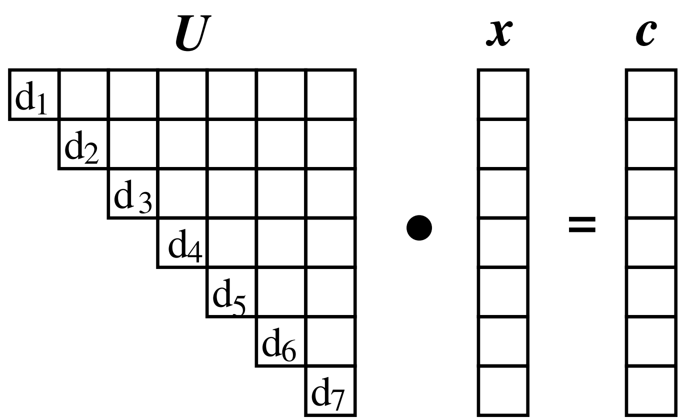
The C matrices must be **stored until the last step:** they are applied to $\boldsymbol{b}$ in the **reverse order of acquisition.**

Gaussian Elimination with Backsubstitution

- Like Gauss-Jordan, but (i) don't normalize pivot row, and (ii) introduce zeros only in rows **below the current pivot element.**

- Example: a_{22} is current **pivot** element
  $\rightsquigarrow$ use pivot row to zero only $a_{32}, a_{42}, \ldots$

- Suppose we use partial pivoting (never change columns)
  $\rightsquigarrow$ Original system $A\boldsymbol{x} = \boldsymbol{b}$ transformed to
  upper triangular system $U\boldsymbol{x} = \boldsymbol{c}$.
  $\rightsquigarrow$ Pivots $d_1, \ldots, d_n$ on diagonal of U.



- Solve with **backsubstitution**.

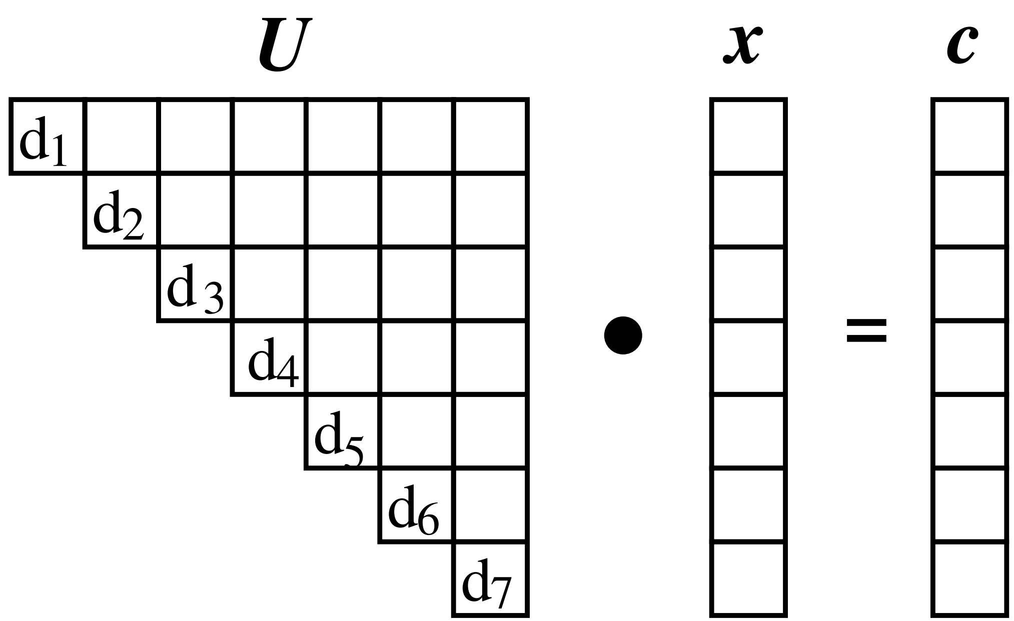
- **Triangular** systems are **computationally and numerically straightforward.**

Gaussian Elimination with Backsubstitution

$$A\boldsymbol{x} = \boldsymbol{b}$$

$$R_1 A\boldsymbol{x} = R_1\boldsymbol{b}$$

$$\underbrace{(\cdots R_2 \cdot R_1) A}_{U}\boldsymbol{x} = \underbrace{(\cdots R_2 \cdot R_1)\boldsymbol{b}}_{\boldsymbol{c}}$$



The invertible case: Summary

- $\boldsymbol{b}$ is in the column space of $A_{n\times n}$, the columns of A are a basis of $\mathbb{R}^n$ (so $C(A) = \mathbb{R}^n$), the rank of A is n.
- G-J: $A \to I$ by multiplication with elementary row matrices:

$$(\cdots R_3 \cdot R_2 \cdot R_1) \cdot A = I = R_E.$$

$R_E = rref(A)$ is the **reduced row Echelon matrix**,
and $A\boldsymbol{x} = \boldsymbol{b} \to R_E\boldsymbol{x} = \boldsymbol{d} \Leftrightarrow \boldsymbol{x} = (\cdots R_3 \cdot R_2 \cdot R_1)\boldsymbol{b}$.

- A invertible $\rightsquigarrow R_E = I \rightsquigarrow$ columns are standard basis of $\mathbb{R}^n$.
- Gaussian elim.: Zeros only below diagonal: $A\boldsymbol{x} = \boldsymbol{b} \to U\boldsymbol{x} = \boldsymbol{c}$.
- Representation of floating-point numbers $\rightsquigarrow$ numerical problems $\rightsquigarrow$ round-off errors $\rightsquigarrow$ nonsense results possible.
- Solution: Partial (rows) and full pivoting (columns).

Chapter 1

Linear Systems of Equations

Singular Systems

The singular case

Recall: Let $\boldsymbol{x}_p$ be a particular solution and $\boldsymbol{x}_n \in N(A)$.
The solutions to all linear equations have the form $\boldsymbol{x} = \boldsymbol{x}_p + \boldsymbol{x}_n$.

How to find $\boldsymbol{x}_p$ and $\boldsymbol{x}_n$? $\rightsquigarrow$ Elimination. Start with the nullspace.

Example: $A_{3\times 4}$: 4 columns, but how many pivots?

$$A = \begin{bmatrix} 1 & 2 & 2 & 2 \\ 2 & 4 & 6 & 8 \\ 3 & 6 & 8 & 10 \end{bmatrix}$$

Initial observations:

- 2nd column is a multiple of first one
- 1st and 3rd column are linearly independent.

$\rightsquigarrow$ **We expect to find pivots for column 1 and 3.**

- 3rd row is linear combination of other rows.

The singular case

$$A = \begin{bmatrix} \color{red}{1} & 2 & 2 & 2 \\ 2 & 4 & 6 & 8 \\ 3 & 6 & 8 & 10 \end{bmatrix} \to \begin{bmatrix} \color{red}{1} & 2 & 2 & 2 \\ 0 & \mathbf{0} & \color{red}{2} & 4 \\ 0 & \mathbf{0} & 2 & 4 \end{bmatrix} \to \begin{bmatrix} \color{red}{1} & 2 & 2 & 2 \\ 0 & 0 & \color{red}{2} & 4 \\ 0 & 0 & 0 & \mathbf{0} \end{bmatrix} = U$$

U is called the **Echelon** (staircase) form of A.

Note that elimination uses only elementary operations that do not change the solutions, so **$A\boldsymbol{x} = \mathbf{0}$ exactly when $U\boldsymbol{x} = \mathbf{0}$.**

U Gives us important information about A:

- 2 pivots, associated with columns 1, 3
  $\rightsquigarrow$ **pivot columns** (not combinations of earlier columns.)
- 2 free columns (these are combinations of earlier columns)
  $\rightsquigarrow$ can assign x_2, x_4 to arbitrary values.

The Reduced Row Echelon Form

Idea: Simplify U further: Elimination also above the pivots.

$$U = \begin{bmatrix} 1 & 2 & 2 & -2 \\ 0 & 0 & 2 & 4 \\ 0 & 0 & 0 & 0 \end{bmatrix} \to \begin{bmatrix} 1 & 2 & 0 & -2 \\ 0 & 0 & 2 & 4 \\ 0 & 0 & 0 & 0 \end{bmatrix} \to \begin{bmatrix} 1 & 2 & 0 & -2 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix} = R_E.$$

A, U and R_E all have 2 independent columns:
$pivcol(A) = pivcol(U) = pivcol(R_E) = (1, 3)$ $\rightsquigarrow$ same rank 2.

Obviously, the **rank equals the number of pivots!** This is equivalent to the algebraic definition **rank** $=$ **dim**$(C(A))$, but maybe more intuitive.

Pivot cols: independent, span the column space $\rightsquigarrow$ basis of $C(A)$.
Pivot rows: independent, span row space $\rightsquigarrow$ basis of $C(A^t)$.

The special solutions

Solutions to $A\boldsymbol{x} = \mathbf{0}$ and $R_E\boldsymbol{x} = \mathbf{0}$ can be obtained by setting the free variables to arbitrary values and solving for the pivot variables.

"Special" solutions are linear independent:
set one free variable equal to 1, and all other free variables to 0.

$$R_E\boldsymbol{x} = \begin{bmatrix} 1 & 2 & 0 & -2 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \mathbf{0}. \quad s_1 = \begin{bmatrix} x_1 \\ 1 \\ x_3 \\ 0 \end{bmatrix}, \quad s_2 = \begin{bmatrix} x_1 \\ 0 \\ x_3 \\ 1 \end{bmatrix}$$

Set 1st free variable $x_2 = 1$, with $x_4 = 0 \rightsquigarrow x_1 + 2 = 0, x_3 = 0$.
Pivot variables are $x_1 = -2, x_3 = 0 \rightsquigarrow s_1 = (-2, 1, 0, 0)^t$.

2nd special solution has $x_2 = 0, x_4 = 1 \rightsquigarrow x_1 - 2 = 0, x_3 + 2 = 0$
$\rightsquigarrow s_2 = (2, 0, -2, 1)^t$.

The nullspace matrix

The nullspace matrix N contains the two special solutions in its columns, so $AN = 0$.

$$R_E = \begin{bmatrix} 1 & 2 & 0 & -2 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad N = \begin{bmatrix} -2 & 2 \\ 1 & 0 \\ 0 & -2 \\ 0 & 1 \end{bmatrix}$$

The linear combinations of these two columns give all vectors in the nullspace $\rightsquigarrow$ basis of null-space $\rightsquigarrow$ complete solution to $A\boldsymbol{x} = \mathbf{0}$.

Consider the dimensions: $n = 4, r = 2$. One special solution for every free variable. r columns have pivots $\rightsquigarrow$ $n - r = 2$ free variables:

$A\boldsymbol{x} = \mathbf{0}$ has r pivots and $n - r$ free variables. The nullspace matrix N contains the $n - r$ special solutions, and $AN = R_E N = 0$.

General form

General form: Suppose that the fist r columns are the pivot columns:

$$R_E = \begin{bmatrix} I & F \\ 0 & 0 \end{bmatrix} \begin{matrix} r & \text{pivot rows} \\ m-r & \text{zero rows} \end{matrix}$$

The **upper left block** is the $r \times r$ **identity matrix**.
There are $n - r$ **free columns**
$\rightsquigarrow$ **upper right block** F has dimension $r \times (n - r)$

Nullspace matrix:

$$N = \begin{bmatrix} -F \\ I \end{bmatrix} \begin{matrix} r & \text{pivot variables} \\ n-r & \text{free variables} \end{matrix}$$

From this definition, we directly see that $R_E N = I(-F) + FI = 0$.

The Complete Solution

- So far: $A\boldsymbol{x} = \mathbf{0}$ converted by elimination to $R_E\boldsymbol{x} = \mathbf{0}$
  $\rightsquigarrow$ solution $\boldsymbol{x}$ is in the nullspace of A.

- Now: $\boldsymbol{b}$ nonzero $\rightsquigarrow$ consider column-augmented matrix $[A\boldsymbol{b}]$.
  We will reduce $A\boldsymbol{x} = \boldsymbol{b}$ to $R_E\boldsymbol{x} = \boldsymbol{d}$.

- Example:

$$\begin{bmatrix} 1 & 3 & 0 & 2 \\ 0 & 0 & 1 & 4 \\ 1 & 3 & 1 & 6 \end{bmatrix} \boldsymbol{x} = \begin{bmatrix} 1 \\ 6 \\ 7 \end{bmatrix} \quad \Rightarrow \quad \begin{bmatrix} 1 & 3 & 0 & 2 & 1 \\ 0 & 0 & 1 & 4 & 6 \\ 1 & 3 & 1 & 6 & 7 \end{bmatrix} = [A\boldsymbol{b}]$$

$$\text{Elimination:} \quad \begin{bmatrix} 1 & 3 & 0 & 2 & 1 \\ 0 & 0 & 1 & 4 & 6 \\ 0 & 0 & 1 & 4 & 6 \end{bmatrix} \quad \Rightarrow \quad \begin{bmatrix} 1 & 3 & 0 & 2 & 1 \\ 0 & 0 & 1 & 4 & 6 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} = [R_E\boldsymbol{d}]$$

The Complete Solution

- Particular solution $\boldsymbol{x}_p$: set free variables $x_2 = x_4 = 0$
  $\rightsquigarrow \boldsymbol{x}_p = (1, 0, 6, 0)^t$. By definition, $\boldsymbol{x}_p$ solves $A\boldsymbol{x}_p = \boldsymbol{b}$.

- The $n - r$ special solutions $\boldsymbol{x}_n$ solve $A\boldsymbol{x}_n = \mathbf{0}$.

- The complete solution is

$$\boldsymbol{x} = \boldsymbol{x}_p + \boldsymbol{x}_n = \begin{bmatrix} 1 \\ 0 \\ 6 \\ 0 \end{bmatrix} + x_2 \begin{bmatrix} -3 \\ 1 \\ 0 \\ 0 \end{bmatrix} + x_4 \begin{bmatrix} -2 \\ 0 \\ -4 \\ 1 \end{bmatrix}$$

Chapter 1

Linear Systems of Equations

Linear Algebra II: The Fundamental Theorem

The Four Fundamental Subspaces

Assume A is $(m \times n)$.

1. The **column space is $C(A)$**, a subspace of $\mathbb{R}^m$.
   It is spanned by the columns of A or R_E.
   Its dimension is the rank $r = \#(\text{independent columns}) = \#(\text{pivots})$.
2. The **row space is $C(A^t)$**, a subspace of $\mathbb{R}^n$. It is spanned by the rows of A or R_E. There is one nonzero row in R_E for every pivot
   $\rightsquigarrow$ dimension is also r.
3. The **nullspace is $N(A)$**, a subspace of $\mathbb{R}^n$.
   It is spanned by the $n-r$ special solutions (one for every free variable), they are independent $\rightsquigarrow$ they form a basis
   $\rightsquigarrow$ dimension of $N(A)$ ("nullity") is $n-r$.
4. The **left nullspace is $N(A^t)$**, a subspace of $\mathbb{R}^m$.
   It contains all vectors $\boldsymbol{y}$ such that $A^t\boldsymbol{y} = \mathbf{0}$. Its dimension is $m-r$.

The Fundamental Theorem of Linear Algebra (I)

1.), 2.) and 3.) are part one of the

Fundamental Theorem of Linear Algebra.
For any $m \times n$ matrix A:

- Column space and row space both have dimension r.
  In other words: column rank $=$ row rank $=$ rank.
- Rank $+$ Nullity $= r + (n-r) = n$.

4.) additionally defines the "left nullspace": it contains any left-side row vectors $\boldsymbol{y}^t$ that are mapped to the zero (row-)vector: $\boldsymbol{y}^t A = \boldsymbol{0}^t$.
$A^t := B$ is a $(n \times m)$ matrix

$$\rightsquigarrow \quad \underbrace{\dim(C(B))}_{r} + \underbrace{\dim(N(B))}_{m-r} = m.$$

$\rightsquigarrow$ **Rank + "Left Nullity" $= m$.**

The Fundamental Theorem of Linear Algebra (II)

Part two of the Fundamental Theorem of Linear Algebra concerns orthogonal relations between the subspaces. Two definitions:

Two vectors $\boldsymbol{v}, \boldsymbol{w} \in V$ are **perpendicular** if their scalar product is zero.
The **orthogonal complement** $V^\perp$ of a subspace V contains **every** vector that is perpendicular to V.

The nullspace is the orthogonal complement of the row space.

Proof: Every $\boldsymbol{x}$ perpendicular to the rows satisfies $A\boldsymbol{x} = \mathbf{0}$.
Reverse is also true: If $\boldsymbol{v}$ is orthogonal to $N(A)$, it must be in the row space. Otherwise we could add $\boldsymbol{v}$ as an extra independent row of the matrix (thereby increasing the rank) without changing the nullspace
$\rightsquigarrow$ row space would grow, contradicting $r + \dim(N(A)) = n$.

The Fundamental Theorem of Linear Algebra (II)

Same reasoning holds true for the left nullspace:

Part two of the Fundamental Theorem of Linear Algebra:

- $N(A)$ is the orthogonal complement of $C(A^t)$ (in $\mathbb{R}^n$).
- $N(A^t)$ is the orthogonal complement of $C(A)$ (in $\mathbb{R}^m$).

Immediate consequences:

Every $\boldsymbol{x} \in \mathbb{R}^n$ can be split into $\boldsymbol{x} = \boldsymbol{x}_{\text{row}} + \boldsymbol{x}_{\text{nullspace}}$.

Thus, the action of A on $\boldsymbol{x}$ is as follows:

$$A\boldsymbol{x}_n = \mathbf{0},$$

$$A\boldsymbol{x}_r = A\boldsymbol{x}$$

The 4 subspaces

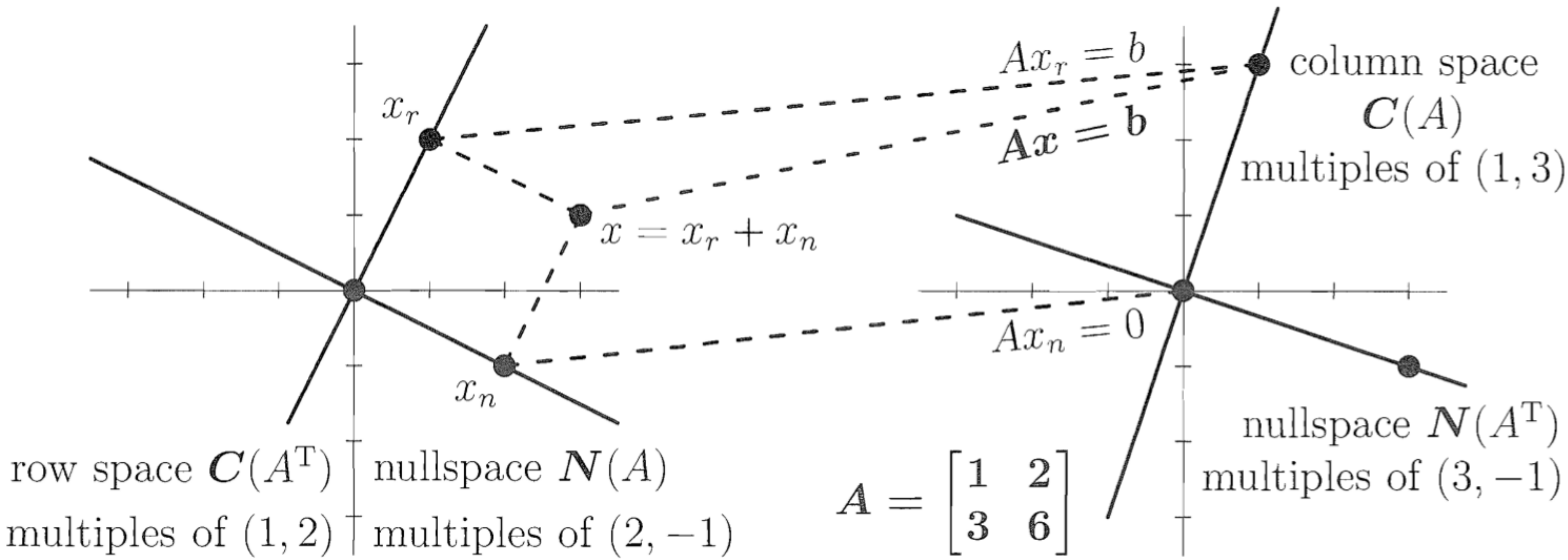


Figure 2.5: The four fundamental subspaces (lines) for the singular matrix A.

Fig. 2.5 in Gilbert Strang: Linear Algebra and Its Applications

Invertible part of a matrix

Every vector in $C(A)$ comes from **one and only one vector** in the row space. Every vector in $C(A^t)$ comes from **one and only one vector** in the column space.

Proof (first assertion):
(i) $A\boldsymbol{x}_r = A\boldsymbol{x}'_r \;\Rightarrow\; A(\boldsymbol{x}_r - \boldsymbol{x}'_r) = \mathbf{0} \;\Rightarrow\; \boldsymbol{\delta} := \boldsymbol{x}_r - \boldsymbol{x}'_r \in N(A)$.
(ii) $\boldsymbol{x}_r \in C(A^t), \boldsymbol{x}'_r \in C(A^t) \;\Rightarrow\; \boldsymbol{\delta} \in C(A^t)$.
But $N(A)$ and $C(A^t)$ are orthogonal $\Rightarrow \boldsymbol{\delta} = \mathbf{0}$.

Conclusion: From the row space to the column space, A is invertible.
In other words: **There is a $r \times r$ invertible matrix "hidden" inside A.**

This will be explored later in this course in the context of the **pseudoinverse** and the **SVD**.

Chapter 1

Linear Systems of Equations

Alternatives to Gaussian Elimination

Further Methods for Linear Systems

- **Direct solution methods**
  - Gauss-Jordan elimination with pivoting
  - Matrix factorization (LU, Cholesky)
  - **Predictable number of steps**

- **Iterative solution methods**
  - Jacobi, Newton etc.
  - **converge in as many steps as necessary**

- **Combination**
  - direct solution, then improved by iterations
  - useful for close-to-singular systems

Factorization methods

- **Disadvantage of Gaussian elimination:** all righthand sides $\boldsymbol{b}_j$ must be known in advance.
- LU decomposition keeps track of the steps in Gaussian elimination $\rightsquigarrow$ The result can be applied to **any future $\boldsymbol{b}$** required.
- A is **decomposed** or factorized as $A = LU$:
  - L lower triangular,
  - U upper triangular.
- **Example:** For a 3×3 matrix, this becomes:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}.$$

LU factorization

- $A = LU$, L lower triangular, U upper triangular.

- $A\boldsymbol{x} = \boldsymbol{b}$ becomes $LU\boldsymbol{x} = \boldsymbol{b}$. Define $\boldsymbol{c} = U\boldsymbol{x}$.
  $L\boldsymbol{c} = \boldsymbol{b}$ solved by **forward-substitution**, followed by
  $U\boldsymbol{x} = \boldsymbol{c}$ solved by **back-substitution**.

- The two interim systems are **trivial to solve** since both are triangular.

- Work effort goes into the **factorization** steps to get L and U.

- U can be computed by **Gaussian elimination**,
  L records the information necessary to **undo the elimination steps**.

LU factorization: the book-keeping

- Steps in Gaussian elimination involve **pre-multiplication** by elementary R-matrices $\rightsquigarrow$ These are trivially invertible.

$$
\begin{aligned}
A &= (R_1^{-1} \cdot R_1) \cdot A = \cdots = \\
&= (R_1^{-1} \cdot R_2^{-1} \cdot R_3^{-1} \cdots R_3 \cdot R_2 \cdot R_1) \cdot A \\
&= \underbrace{(R_1^{-1} \cdot R_2^{-1} \cdot R_3^{-1} \cdots)}_{L} \cdot \underbrace{(\cdots R_3 \cdot R_2 \cdot R_1 \cdot A)}_{U}
\end{aligned}
$$

- Entries for L are the inverses (i.e. negatives) of the multipliers in the row transformation for each step: $R_{ij}(m)$ Subtracts row j multiplied by m from row i. **Inverse:** $R_{ij}(m)^{-1} = R_{ij}(-m)$.

$$
\underbrace{\begin{bmatrix} 2 & 1 \\ 6 & 8 \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} 1 & 0 \\ 3 & 1 \end{bmatrix}}_{R^{-1}} \underbrace{\begin{bmatrix} 1 & 0 \\ -3 & 1 \end{bmatrix}}_{R} \underbrace{\begin{bmatrix} 2 & 1 \\ 6 & 8 \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} 1 & 0 \\ 3 & 1 \end{bmatrix}}_{L} \underbrace{\begin{bmatrix} 2 & 1 \\ 0 & 5 \end{bmatrix}}_{U}
$$

LU factorization via Gaussian elimination

- **LU is not unique:**
  - Decomposition is multiplicative
    $\rightsquigarrow$ factors can be re-arranged between L and U.
- LU **may not exist at all**, if there is a **zero pivot**. **Pivoting:**
  - Can factorize as $A = P^{-1}LU = P^tLU$.
  - P records the effects of row permutations, so $PA = LU$.
    Need to **keep track of permutations** in P.

$$\text{Permutation } \pi = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 4 & 2 & 5 & 3 \end{pmatrix} \Rightarrow P_\pi = \begin{bmatrix} \boldsymbol{e}_1^t \\ \boldsymbol{e}_4^t \\ \boldsymbol{e}_2^t \\ \boldsymbol{e}_5^t \\ \boldsymbol{e}_3^t \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}.$$

Crout's algorithm

- Alternative method to find the L and U matrices
- Write out $A = LU$ with unknowns for the non-zero elements of L, U.
- Equate entries in the $n \times n$ matrix equation
  $\rightsquigarrow$ n^2 equations in $n^2 + n$ unknowns.
- **Underdetermined** $\rightsquigarrow$ n unknowns are arbitrary
  (shows that the LU decomposition is **not unique**)
  $\rightsquigarrow$ choose the diagonal entries $l_{ii} = 1$.
- **Crout's algorithm:**
  - re-write the n^2 equations in a carefully chosen order so that elements of L and U can be found **one-by-one**.

Crout's algorithm

$$\begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Multiplying out gives:

$$\color{red}{u_{11}} = a_{11}$$

$$\color{red}{l_{21}} u_{11} = a_{21}$$

$$\color{red}{l_{31}} u_{11} = a_{31}$$

$$\color{red}{u_{12}} = a_{12}$$

$$l_{21} u_{12} + \color{red}{u_{22}} = a_{22}$$

$$l_{31} u_{12} + \color{red}{l_{32}} u_{22} = a_{32}$$

$$\color{red}{u_{13}} = a_{13}$$

$$l_{21} u_{13} + \color{red}{u_{23}} = a_{23}$$

$$l_{31} u_{13} + l_{32} u_{23} + \color{red}{u_{33}} = a_{33}$$

Red indicates where an element is used for the first time.

Only one red entry in each equation!

Crout's method fills in the combined matrix

$$\begin{bmatrix} u_{11} & u_{12} & \color{green}{u_{13}} & u_{14} & \cdots \\ \color{green}{l_{21}} & u_{22} & \color{red}{u_{23}} & u_{24} & \cdots \\ l_{31} & l_{32} & u_{33} & u_{34} & \cdots \\ l_{41} & l_{42} & l_{43} & u_{44} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

by columns from left to right, and from top to bottom.

A small example

$$\begin{bmatrix} 4 & 3 \\ 6 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ l_{21} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} \\ 0 & u_{22} \end{bmatrix}$$

Solve the linear equations:

$$\begin{aligned} u_{11} &= 4 \\ l_{21} \cdot u_{11} &= 6 \\ u_{12} &= 3 \\ l_{21} \cdot u_{12} + u_{22} &= 3 \end{aligned}$$

Substitution yields: $\begin{bmatrix} 4 & 3 \\ 6 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1.5 & 1 \end{bmatrix} \begin{bmatrix} 4 & 3 \\ 0 & -1.5 \end{bmatrix}$

Chapter 1

Linear Systems of Equations

Positive Definite Matrices and the Cholesky Decomposition

Positive definite matrices

An $n \times n$ **symmetric real matrix** A is positive-definite if $\boldsymbol{x}^t A \boldsymbol{x} > 0$ for all vectors $\boldsymbol{x} \neq \mathbf{0}$.

Simple tests for positive definiteness?

- A positive definite matrix A has **all positive entries** on the main diagonal (use $\boldsymbol{x}^t A \boldsymbol{x} > 0$ with vectors $(1, 0, ..., 0)^t$, $(0, 1, 0, ..., 0)^t$ etc.)
- A is **diagonally dominant** if $|a_{ii}| > \sum_{i \neq j} |a_{ij}|$.
- A diagonally dominant matrix is positive definite if it is **symmetric** and has **all main diagonal entries positive**. Follows from the **Gershgorin circle theorem** (details will follow...). Note that the **converse is false**.

There are many applications of pos. def. matrices:

- **Linear regression** models ($\rightsquigarrow$ chapter 2).
- Solution of **partial differential equations** $\rightsquigarrow$ heat conduction, mass diffusion, wave equation etc.

Example: Heat equation

- $u = u(x,t)$ is **temperature as a function of space and time**. This function will change over time as heat spreads throughout space.

- $u_t := \frac{\partial u}{\partial t}$ is the **rate of change** of temperature at a point over time.

- $u_{xx} := \frac{\partial^2 u}{\partial x^2}$ is the second spatial derivative of temperature.

- **Heat equation:** $u_t \propto u_{xx}$. **The rate of change of temperature over time is proportional to the local difference of temperature**. Proportionality constant: diffusivity of the (isotropic) medium.

- **Discretization** $u_j^{(m)} = u(x_j, t_m)$ at grid points x_j and time points t_m:

$$x_j := j \cdot \underbrace{h}_{\text{spatial step size}} \quad \text{and} \quad t_m := m \cdot \underbrace{\tau}_{\text{temporal step size}}$$

- Assume $h = \tau = 1$, and also diffusivity $= 1$.

Example: Heat equation

- Approximate derivative on grid ($\rightsquigarrow$ **finite differences**):

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}.$$

- Second order (**central difference approximation**):

$$f''(x) \approx \frac{\frac{f(x+h)-f(x)}{h} - \frac{f(x)-f(x-h)}{h}}{h} = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}.$$

- Approximate equation $u_t = u_{xx}$ by (we assumed step size =1)

$$\underbrace{u_j^{(m+1)} - u_j^{(m)}}_{\text{rate of change over time}} = \underbrace{u_{j-1}^{(m+1)} - 2u_j^{(m+1)} + u_{j+1}^{(m+1)}}_{\text{local temperature difference}}$$

Example: Heat equation

- Solve this **implicit scheme** for $u^{(m+1)}$:

$$(1{+}2)u_j^{(m+1)} - u_{j-1}^{(m+1)} - u_{j+1}^{(m+1)} = u_j^{(m)}, \quad \text{for } j = 1, \ldots, n{-}1, \text{ and } m \geq 0.$$

- With $A =$ tri-diagonal with $(a_{j,j-1}, a_{j,j}, a_{j,j+1}) = (-1, 2, -1)$:

$$(I + A)u^{(m+1)} = \begin{bmatrix} . & . & . & . & . & . & . \\ 0 & -1 & \mathbf{3} & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & \mathbf{3} & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & \mathbf{3} & -1 & 0 \\ . & . & . & . & . & . & . \end{bmatrix} u^{(m+1)} = u^{(m)}$$

- $(I + A)$ is diagonal dominant and symmetric, and has positive diagonal entries $\rightsquigarrow$ **positive definite**! It is also **sparse** $\rightsquigarrow$ efficient elimination possible: per column only 1 zero needs to be produced below the pivot.

Cholesky LU decomposition

- **The Cholesky LU factorization of a pos. def. matrix A is $A = LL^t$.**

- Use it to solve a pos. def. system $A\boldsymbol{x} = \boldsymbol{b}$.

- Cholesky algorithm: Partition matrices in $A = LL^t$ as

$$\begin{pmatrix} a_{11} & \boldsymbol{a}_{21}^t \\ \boldsymbol{a}_{21} & A_{22} \end{pmatrix} = \begin{pmatrix} l_{11} & 0 \\ \boldsymbol{l}_{21} & L_{22} \end{pmatrix} \begin{pmatrix} l_{11} & \boldsymbol{l}_{21}^t \\ 0 & L_{22}^t \end{pmatrix} = \begin{pmatrix} l_{11}^2 & l_{11}\boldsymbol{l}_{21}^t \\ l_{11}\boldsymbol{l}_{21} & \boldsymbol{l}_{21}\boldsymbol{l}_{21}^t + L_{22}L_{22}^t \end{pmatrix}$$

  Recursion:

  - step 1: $l_{11} = \sqrt{a_{11}}, \quad \boldsymbol{l}_{21} = \frac{1}{l_{11}}\boldsymbol{a}_{21}$.
  - step 2: compute L_{22} from $S := A_{22} - \boldsymbol{l}_{21}\boldsymbol{l}_{21}^t = L_{22}L_{22}^t$.
    This is a Cholesky factorization of $S_{(n-1)\times(n-1)}$.

Cholesky: Proof

Proof that the algorithm works for positive definite $A_{n \times n}$ **by induction:**

1. If A is positive definite then $a_{11} > 0$,
   $\rightsquigarrow$ $l_{11} = \sqrt{a_{11}}$ **and** $\boldsymbol{l}_{21} = \frac{1}{l_{11}} \boldsymbol{a}_{21}$ **are well-defined.**
2. If A is positive definite, then
   $S = A_{22} - \boldsymbol{l}_{21} \boldsymbol{l}_{21}^t = A_{22} - \frac{1}{a_{11}} \boldsymbol{a}_{21} \boldsymbol{a}_{21}^t$ **is positive definite.**

   Proof: take any $(n-1)$ vector $\boldsymbol{v} \neq 0$ and $w = -(1/a_{11}) \boldsymbol{a}_{21}^t \boldsymbol{v} \in \mathbb{R}$.

$$\boldsymbol{v}^t S \boldsymbol{v} = \begin{pmatrix} w & \boldsymbol{v}^t \end{pmatrix} \begin{pmatrix} a_{11} & \boldsymbol{a}_{21}^t \\ \boldsymbol{a}_{21} & A_{22} \end{pmatrix} \begin{pmatrix} w \\ \boldsymbol{v} \end{pmatrix} = \begin{pmatrix} w & \boldsymbol{v}^t \end{pmatrix} A \begin{pmatrix} w \\ \boldsymbol{v} \end{pmatrix} > 0.$$

- **Induction step:** Algorithm works for $n = k$ if it works for $n = k - 1$.
- **Base case:** It obviously works for $n = 1$; **therefore it works for all n.**

Chapter 1

Linear Systems of Equations

Iterative Methods

Iterative improvement

- Floating point arithmetic **limits the precision of calculated solutions**.

- For large systems and "close-to-singular" small systems, precision is generally **far worse than machine precision** ϵ_m.
  - Direct methods **accumulate roundoff errors**.
  - **Loss of some significant digits** isn't unusual even for well-behaved systems.

- **Iterative improvement:** Start with direct solution method (Gauss, LU, Cholesky etc.), followed by some post-iterations.
  It will get your solution **back to machine precision** efficiently.

Iterative improvement

- Suppose $\boldsymbol{x}$ is the (unknown) **exact solution** of $A\boldsymbol{x} = \boldsymbol{b}$ and $\boldsymbol{x} + \delta\boldsymbol{x}$ is a calculated (**inexact**) solution with **unknown error** $\delta\boldsymbol{x}$.
- Substitute calculated solution in original equation:

$$A(\boldsymbol{x} + \delta\boldsymbol{x}) = \boldsymbol{b} + \delta\boldsymbol{b}, \tag{1}$$

- Subtract $A\boldsymbol{x}$ (or $\boldsymbol{b}$) from both sides:

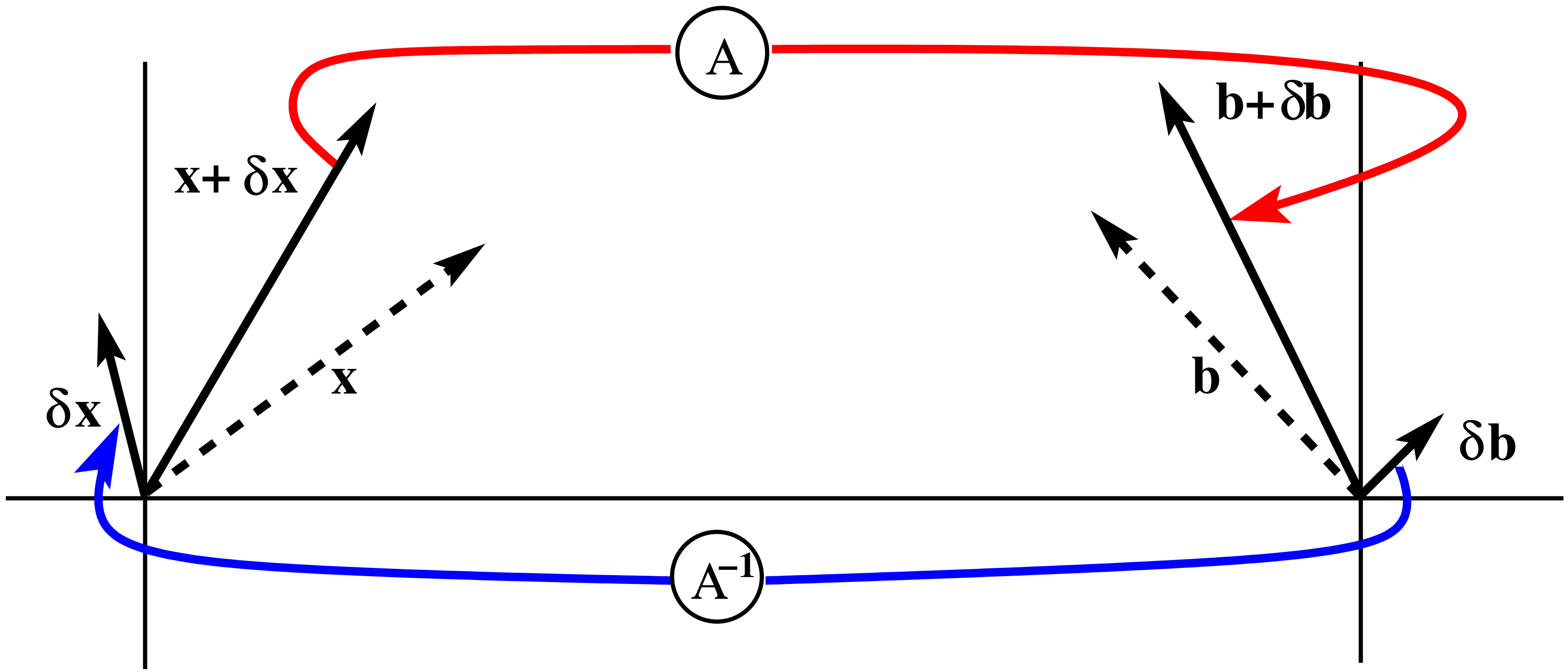
$$A\delta\boldsymbol{x} = \delta\boldsymbol{b}. \tag{2}$$

- Eqn. (1) gives:

$$\delta\boldsymbol{b} = A \underbrace{(\boldsymbol{x} + \delta\boldsymbol{x})}_{\text{calculated solution}} - \boldsymbol{b}. \tag{3}$$

- Right hand side of eqn. (3) is known
  $\leadsto$ get $\delta\boldsymbol{b}$ and use this in (2) to solve for $\delta\boldsymbol{x}$.

Iterative improvement



Iterative improvement: first guess $\boldsymbol{x} + \delta\boldsymbol{x}$ is multiplied by A to produce $\boldsymbol{b} + \delta\boldsymbol{b}$.
Known vector $\boldsymbol{b}$ is subtracted $\rightsquigarrow \delta\boldsymbol{b}$.
Inversion gives $\delta\boldsymbol{x}$ and subtraction gives an improved solution $\boldsymbol{x}$.

LU **factorization of** A **can be used to solve** $A\delta x = LU\delta x = \delta \boldsymbol{b}$ **to get** δx.

Repeat until $\|\delta\boldsymbol{x}\| \approx \epsilon_m$.

Iterative methods: Jacobi

- Assume that all diagonal entries of A are nonzero.
- Write $A = D + L + U$

where $D = \begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn} \end{bmatrix}$ and $L+U = \begin{bmatrix} 0 & a_{12} & \cdots & a_{1n} \\ a_{21} & 0 & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & 0 \end{bmatrix}$

- So $A\boldsymbol{x} = \boldsymbol{b} \quad \rightsquigarrow \quad (L + D + U)\boldsymbol{x} = \boldsymbol{b}$.
- The solution is then obtained iteratively via

$$D\boldsymbol{x} = \boldsymbol{b} - (L + U)\boldsymbol{x}.$$

Iterative methods: Jacobi

- The solution is obtained iteratively via

$$D\boldsymbol{x} = \boldsymbol{b} - (L+U)\boldsymbol{x}. \tag{4}$$

- Given $\boldsymbol{x}_{(i)}$ obtain $\boldsymbol{x}_{(i+1)}$ by solving (4) with $\boldsymbol{x} = \boldsymbol{x}_{(i)}$:

$$\boldsymbol{x}_{(i+1)} = -D^{-1}(L+U)\boldsymbol{x}_{(i)} + D^{-1}\boldsymbol{b}.$$

- Define $J = D^{-1}(L+U)$ as the **iteration matrix.**
  $\rightsquigarrow \boldsymbol{x}_{(i+1)} = -J\boldsymbol{x}_{(i)} + D^{-1}\boldsymbol{b}.$
- From (4): $D^{-1}\boldsymbol{b} = \boldsymbol{x} + D^{-1}(L+U)\boldsymbol{x} = \boldsymbol{x} + J\boldsymbol{x}$
  $\Rightarrow \; \boldsymbol{x}_{(i+1)} = -J\boldsymbol{x}_{(i)} + \boldsymbol{x} + J\boldsymbol{x}$.
- $(i+1)$-th error term: $\boldsymbol{\epsilon}_{(i+1)} = \boldsymbol{x}_{(i+1)} - \boldsymbol{x} = -J(\boldsymbol{x}_{(i)} - \boldsymbol{x}) = -J\boldsymbol{\epsilon}_{(i)}$.
- **Convergence guaranteed** if J is “contracting”.

Calculating the error, revisited

- Error in $(i+1)$-th iteration: $\boldsymbol{\epsilon}_{(i+1)} = -J\boldsymbol{\epsilon}_{(i)}$.
- $\boldsymbol{\epsilon}_{(i+1)} = -J(-J\boldsymbol{\epsilon}_{(i-1)}) = J^2\boldsymbol{\epsilon}_{(i-1)} = \cdots = (-1)^{i+1}J^{i+1}\boldsymbol{\epsilon}_{(0)}$.
- So if $J^i \to 0$ (zero matrix) for $i \to \infty$ then $\boldsymbol{\epsilon}_{(i)} \to \mathbf{0}$.
- The key to understanding this condition is the **eigenvalue decomposition** $J = V\Lambda V^{-1}$ (details next section)
  - the columns of V consist of **eigenvectors** of J and
  - Λ is a diagonal matrix of **eigenvalues** of J.
- Then $J^2 = V\Lambda V^{-1}V\Lambda V^{-1} = V\Lambda^2 V^{-1} \rightsquigarrow J^n = V\Lambda^n V^{-1}$.

  If all the eigenvalues of J have magnitude < 1,
  then $\Lambda^n \to 0$ and consequently $J^n \to 0 \rightsquigarrow$ convergence.
- **A diagonally dominant $\rightsquigarrow$ Jacobi method converges.**
  Follows from the **Gershgorin circle theorem**.

Chapter 1

Linear Systems of Equations

Linear Algebra III: Eigenvalues and Eigenvectors

Eigenvalues and eigenvectors

- Consider a square matrix A. A vector $\boldsymbol{v}$ for which $A\boldsymbol{v} = \lambda\boldsymbol{v}$ for some (possibly complex) scalar λ is an **eigenvector** of A, and λ is the associated **eigenvalue.**
- **The eigenvectors span the nullspace** of $(A - \lambda I)$: They are the solutions of $(A - \lambda I)\boldsymbol{v} = \mathbf{0}$.
- A **non-zero solution** $\boldsymbol{v} \neq \mathbf{0}$ exists if and only if the matrix $(A - \lambda I)$ is **not invertible**:

  otherwise we could invert $(A - \lambda I)$ and get the unique solution $\boldsymbol{v} = (A - \lambda I)^{-1}\mathbf{0} = \mathbf{0}$, i.e. only the zero solution.
- Equivalently we have **non-zero eigenvectors** if and only if the rank of $(A - \lambda I) < n$.
- Equivalently we want: $det(A - \lambda I) = 0$. Why?

Determinants

- The **determinant of a square matrix** is a single number.
  It contains a lot of information about the matrix.
- But is is not a "simple" function...
  Explicit formulas are complicated, but its properties are simple.

Three rules completely determine the number $det(A)$:

1. **The determinant of the identity matrix is 1: $det(I) = |I| = 1$.**
2. **The determinant changes sign when two rows are exchanged.**
3. **The determinant is a linear function in each row separately** (all other rows stay fixed): 2d-example for first row

$$\begin{vmatrix} ta & tb \\ c & d \end{vmatrix} = t \begin{vmatrix} a & b \\ c & d \end{vmatrix}$$

$$\begin{vmatrix} a + a' & b + b' \\ c & d \end{vmatrix} = \begin{vmatrix} a & b \\ c & d \end{vmatrix} + \begin{vmatrix} a' & b' \\ c & d \end{vmatrix}$$

Determinants

Further rules can be deduced:

4. **If two rows of A are equal, then $det(A) = 0$.**
   Rule 2: Exchange of the equal rows $\rightsquigarrow$ $det(A)$ changes sign.
   But matrix stays the same, so det cannot change $\rightsquigarrow$ $det(A) = 0$.
5. **Subtracting a multiple of one row from another row leaves the same determinant.**

$$\begin{vmatrix} a - lc & b - ld \\ c & d \end{vmatrix} = \begin{vmatrix} a & b \\ c & d \end{vmatrix} - l \underbrace{\begin{vmatrix} c & d \\ c & d \end{vmatrix}}_{=0 \text{ (rule 4)}}$$

   Usual elimination steps do not affect the determinant!

6. **If A has a row of zeros, then $det(A) = 0$.**
   Add some other row to zero row $\rightsquigarrow$ $det(A)$ is unchanged (rule 5).
   But now there are two identical rows $\rightsquigarrow$ $det(A) = 0$ by rule 4.

Determinants

7. **If A is triangular then $det(A) = \prod_i a_{ii}$.**
   Suppose the diagonal entries are nonzero. Then elimination can remove all the off-diagonal entries, without changing $det(A)$ (rule 5).
   Factoring out the diagonal elements gives
   $det(A) = \prod_i a_{ii} \cdot det(I) = \prod_i a_{ii}$ (rules 3 and 1).
   Zero diagonal entry $\rightsquigarrow$ elimination produces a zero row.
   Rule 5: elimination steps do not change $det(A)$.
   Rule 6: zero row $\rightsquigarrow$ $det(A) = 0$.
8. **If A is singular, then $det(A) = 0$. If A is invertible, $det(A) \neq 0$.**
   A singular: Elimination $\rightsquigarrow$ zero row in U $\rightsquigarrow$ $det(A) = det(U) = 0$.
   A nonsingular: Elimination puts the nonzero pivots $d_1, \ldots, d_n$ on the diagonal. Sign depends on whether the number of row exchanges is even or odd: $det(A) = \pm det(U) = \pm \prod_i d_i \neq 0$.

Determinants

9. **The determinant of AB is the product $det(A) \cdot det(B)$.**
   Proof sketch: When $|B| \neq 0$, consider ratio $D(A) := |AB|/|B|$.
   Check that this ratio has properties
   1: $A = I$ implies $D(A) = 1$,
   2: exchange of two rows of A gives a sign reversal of $D(A)$,
   3: linearity in each row
   $\rightsquigarrow$ $D(A)$ must be the determinant of A: $D(A) = |A| = |AB|/|B|$.
10. **Formula for 2×2 case:**

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} = LU = \underbrace{\begin{bmatrix} 1 & 0 \\ c/a & 1 \end{bmatrix}}_{det=1} \underbrace{\begin{bmatrix} a & b \\ 0 & (ad-bc)/a \end{bmatrix}}_{det=ad-bc}$$

Eigenvalues and eigenvectors

- $det(A - \lambda I) = 0$ is the **characteristic polynomial** of A.
  - it's a **polynomial of degree** n for $A_{(n \times n)}$,
  - its solutions give **all the eigenvalues** λ_i.

Example: $$\begin{vmatrix} a-\lambda & b \\ c & d-\lambda \end{vmatrix} = (a-\lambda)(d-\lambda) - bc = 0.$$

- Once we know all the $\lambda_1, \lambda_2, \ldots, \lambda_n$ we take each one in turn and find the **corresponding eigenvectors** $\boldsymbol{v}_i$ by solving the **linear system**

$$(A - \lambda_i I)\boldsymbol{v}_i = \mathbf{0}.$$

- All eigenvectors fulfill $A\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i$. In matrix form: $AV = V\Lambda$, where $\boldsymbol{v}_i$ is the i-th column of V and Λ is the diagonal matrix $\Lambda = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}$

Eigenvalues, pivots and determinants

- Suppose that $\lambda_1, \dots \lambda_2$ are eigenvalues of A. Then the λ_i are the roots of the characteristic polynomial, and this polynomial of degree n always separates into n factors involving the (possibly complex) eigenvalues (**fundamental theorem of algebra**), i.e.

$$det(A - \lambda I) = (\lambda_1 - \lambda)(\lambda_2 - \lambda) \cdots (\lambda_n - \lambda).$$

- Holds for every $\lambda \rightsquigarrow$ can set $\lambda = 0$:

$$det(A) = \prod_i \lambda_i$$

- We already showed that $det(A) = \pm det(U) = \pm \prod_i d_i$, so **Determinant $=$ $\pm$ (product of pivots) $=$ product of eigenvalues.**

Diagonalization

- **Not all** linear operators can be represented by diagonal matrices with respect to some basis.

- A square matrix A for which there is some (invertible) P so that $P^{-1}AP = D$ is a diagonal matrix is called **diagonalizable**.

Theorem. *Suppose that $A_{(n\times n)}$ has n **linearly independent** eigenvectors $v_1, \dots, v_n$, arranged as columns in the matrix V. Then*

$$V^{-1}AV = \Lambda = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}$$

Proof. $Av_i = \lambda_i \boldsymbol{v}_i \;\Rightarrow\; AV = V\Lambda \;\Rightarrow\; V^{-1}AV = \Lambda$ □

Diagonalization

Theorem. *Eigenvectors corresponding to distinct (all different) eigenvalues are linearly independent.*

Proof. Suppose $c_1\boldsymbol{v}_1 + c_2\boldsymbol{v}_2 = \mathbf{0}$.
Then $A(c_1\boldsymbol{v}_1 + c_2\boldsymbol{v}_2) = c_1\lambda_1\boldsymbol{v}_1 + c_2\lambda_2\boldsymbol{v}_2 = \mathbf{0}$.
Also $\lambda_2(c_1\boldsymbol{v}_1 + c_2\boldsymbol{v}_2) = c_1\lambda_2\boldsymbol{v}_1 + c_2\lambda_2\boldsymbol{v}_2 = \mathbf{0}$.
Subtraction gives:

$$(\lambda_1 - \lambda_2)c_1\boldsymbol{v}_1 = \mathbf{0} \;\Rightarrow\; c_1 = 0 \text{ , since } \boldsymbol{v}_1 \neq \mathbf{0} \text{ and } \lambda_1 \neq \lambda_2.$$

Similarly, $c_2 = 0$. Thus, no other combination $c_1\boldsymbol{v}_1 + c_2\boldsymbol{v}_2 = \mathbf{0}$, and the eigenvectors must be independent. Proof directly extends to any number of eigenvectors. □

Orthogonal Diagonalization

- If P is also orthogonal ($PP^t = I$), A is **orthogonally** diagonalizable.

- Columns of P = linearly independent eigenvectors of A.

- Diagonal entries of D are the corresponding eigenvalues.

Theorem. *If a matrix is orthogonally diagonalizable, then it is symmetric*

Proof. We assume that $P^tAP = D$ holds, with $P^t = P^{-1}$.
Thus, $A = PDP^t$ and $A^t = (PDP^t)^t = PDP^t = A$. □

Orthogonal Diagonalization

Theorem. *Eigenvectors of a symmetric matrix corresponding to different eigenvalues are orthogonal.*

Proof. Let $A^t = A$ have eigenvectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ for eigenvalues $\lambda_1 \neq \lambda_2$.

$$(A\boldsymbol{v}_1)^t\boldsymbol{v}_2 = \boldsymbol{v}_1^t(A\boldsymbol{v}_2) = \lambda_1\boldsymbol{v}_1^t\boldsymbol{v}_2 = \lambda_2\boldsymbol{v}_1^t\boldsymbol{v}_2.$$

Since $\lambda_1 \neq \lambda_2$, we must have $\boldsymbol{v}_1^t\boldsymbol{v}_2 = 0$. □

More general version (without explicit proof): Spectral Theorem

Theorem. *Suppose the $n \times n$ matrix A is symmetric.*
Then it has n orthogonal eigenvectors with real eigenvalues.

A square matrix A is orthogonally diagonalizable
if and only if it is symmetric.

Chapter 1

Linear Systems of Equations

Matrix Powers and Markovian Matrices

Matrix Powers

- Consider a square matrix A with eigenvector decomposition $A_{(n \times n)} = V \Lambda V^{-1}$.

- What are the eigenvectors of $A^2 = AA$?
  Substitution gives: $A^2 = V \Lambda V^{-1} V \Lambda V^{-1} = V \Lambda^2 V^{-1}$.
  So A^2 **has the same eigenvectors and squared eigenvalues.**

- General form: $A^n = V \Lambda^n V^{-1}$.

- When does $A^k \to 0$ (zero matrix)?
  All $|\lambda_i| < 1$ (cf. convergence analysis of Jacobi iterations).

- Are there other interesting applications of matrix powers? Yes, **many!**

Matrix Powers: Fibonacci numbers

- The Fibonacci sequence $0, 1, 1, 2, 3, 5, 8, 13, \ldots$ comes from $F_{k+2} = F_{k+1} + F_k$.

- Assume you want to compute F_{100}. **Can it be done directly?** Yes, with the help of matrix powers...

- Define $\boldsymbol{u}_k = \begin{bmatrix} F_{k+1} \\ F_k \end{bmatrix}$ and the **transition matrix** $A = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$.

- The rule $\begin{array}{rcl} F_{k+2} & = & F_{k+1} + F_k \\ F_{k+1} & = & F_{k+1} \end{array}$ is $\boldsymbol{u}_{k+1} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \boldsymbol{u}_k$.

- After 100 steps we reach $\boldsymbol{u}_{100} = A^{100}\boldsymbol{u}_0$, with $\boldsymbol{u}_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$.

Matrix Powers: Fibonacci numbers

1. Find **eigenvectors** $\boldsymbol{v}_1, \boldsymbol{v}_2$ and associated **eigenvalues** of A.

2. Express $\boldsymbol{u}_0$ as **combination of eigenvectors:**
$\boldsymbol{u}_0 = c_1\boldsymbol{v}_1 + c_2\boldsymbol{v}_2 \rightsquigarrow \boldsymbol{c} = V^{-1}\boldsymbol{u}_0$.

3. Now $A^{100}\boldsymbol{u}_0 = V\Lambda^{100}V^{-1}\boldsymbol{u}_0 = V\Lambda^{100}\boldsymbol{c}$. Thus, multiply each eigenvector $\boldsymbol{v}_i$ with λ_i^{100} and **add up the results with weights c_i.**

In our case:

$$A - \lambda I = \begin{bmatrix} 1-\lambda & 1 \\ 1 & -\lambda \end{bmatrix} \rightsquigarrow det(A - \lambda I) = \lambda^2 - \lambda - 1 \overset{!}{=} 0$$

$$\rightsquigarrow \lambda = \tfrac{1}{2} \pm \sqrt{\tfrac{1}{4} + 1} \rightsquigarrow \lambda_1 = \tfrac{1+\sqrt{5}}{2} \approx 1.618, \quad \lambda_2 = \tfrac{1-\sqrt{5}}{2} \approx -0.618.$$

$$(A - \lambda I)\boldsymbol{v} = \boldsymbol{0} \rightsquigarrow \boldsymbol{v}_1 = \begin{bmatrix} \lambda_1 \\ 1 \end{bmatrix}, \quad \boldsymbol{v}_2 = \begin{bmatrix} \lambda_2 \\ 1 \end{bmatrix}.$$

Matrix Powers: Fibonacci numbers

Weights: $c_1 = 1/(\lambda_1 - \lambda_2), \quad c_2 = -1/(\lambda_1 - \lambda_2)$

After 100 steps: $\boldsymbol{u}_{100} = c_1\lambda_1^{100}\boldsymbol{v}_1 + c_2\lambda_2^{100}\boldsymbol{v}_2 = \frac{\lambda_1^{100}\boldsymbol{v}_1 - \lambda_2^{100}\boldsymbol{v}_2}{\lambda_1 - \lambda_2}$.

- We want F_{100} = second component of $\boldsymbol{u}_{100}$.
  Second components of eigenvectors are 1, and $\lambda_1 - \lambda_2 = \sqrt{5}$. Thus,

$$F_{100} = \frac{\lambda_1^{100} - \lambda_2^{100}}{\lambda_1 - \lambda_2} = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{100} - \left(\frac{1-\sqrt{5}}{2}\right)^{100}\right] \approx 3.54 \cdot 10^{20}.$$

- **Note:** $\lambda_2^k/(\lambda_1 - \lambda_2) < 1/2$ and result must be an integer,
  so $F_k = \frac{\lambda_1^k - \lambda_2^k}{\lambda_1 - \lambda_2}$ must be the **nearest integer to** $\frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^k$**.**

- The ratio $\frac{F_{k+1}}{F_k}$ approaches the **golden ratio** $\frac{1+\sqrt{5}}{2} \approx 1.618$ for large k.

Matrix Powers: Markov matrices

- A matrix is a **Markov matrix** iff the following holds:
  1. Every entry is non-negative, 2. Every column adds to 1.

- A Markov matrix is called **column-stochastic:**
  entries in every column can be interpreted as probabilities.

- Suppose A is Markovian, and start with probability vector $\boldsymbol{u}_0$.

- **Observation:** if we make a sequence of update steps, $\boldsymbol{u}_k = A^k \boldsymbol{u}_0$, we will approach a **steady state** for $k \to \infty$, and this steady state **does not depend on the starting vector u_0!**

- **Asymptotic loss of memory:** Markov chain "forgets" where it started. The question is why...

Matrix Powers: Markov matrices

- **Intuition:** Since the eigenvalues of A are raised to larger and larger powers, a non-trivial steady state can only occur for $\lambda = 1$.
  The **steady state equation $A\boldsymbol{u}_\infty = \boldsymbol{u}_\infty$** then makes $\boldsymbol{u}_\infty$ an eigenvector of A with eigenvalue $\lambda = 1$.

Theorem. *A positive Markov matrix (entries $a_{ij} > 0$) has one eigenvalue $\lambda_1 = 1$, all other eigenvalues have $|\lambda| < 1$.*

We will not formally prove this theorem here. But the existence of $\lambda_1 = 1$ easily follows from this observation:

- Consider $A = \begin{bmatrix} p_1 & q_1 \\ p_2 & q_2 \end{bmatrix}$. A is column-stochastic, so every column of $A - 1I$ adds to $1 - 1 = 0 \rightsquigarrow$ the **row vectors add up to zero**, $(p_1 - 1, q_1)^t + (p_2, q_2 - 1)^t = (0, 0)^t$, so they are **linearly dependent** $\rightsquigarrow det(A - 1I) = 0 \rightsquigarrow$ **$\lambda = 1$ is an eigenvalue of A.**

Matrix Powers: Markov matrices

- **A^2 is also a Markov matrix:**

$$A^2 = \begin{bmatrix} p_1^2 + p_2 q_1 & p_1 q_1 + q_1 q_2 \\ p_1 p_2 + p_2 q_2 & p_2 q_2 + q_2^2 \end{bmatrix}$$

  Note that this matrix is also column-stochastic: Sum of first column is

$$p_1^2 + p_2 q_1 + p_1 p_2 + p_2 q_2 = p_1(p_1 + p_2) + p_2(q_1 + q_2) = p_1 + p_2 = 1.$$

- By induction, **all matrices A^k are Markov matrices!**
  $\rightsquigarrow$ they all have the eigenvalue $\lambda = 1$.

- This argument holds true for any $n \times n$ Markov matrix A.

Markov matrices: Rental cars example

Rental cars in Denver. Every month, 80% of the Denver cars stay in Denver, 20% leave. 5% of outside cars come in, 95% stay outside. Fraction of cars in Denver starts at $1/50 = 0.02 \rightsquigarrow \boldsymbol{u}_0 = (0.02, 0.98)^t$.

First month: $\boldsymbol{u}_1 = \begin{bmatrix} 0.8 & 0.05 \\ 0.2 & 0.95 \end{bmatrix} \begin{bmatrix} 0.02 \\ 0.98 \end{bmatrix} = \begin{bmatrix} 0.065 \\ 0.935 \end{bmatrix}$

k-th month: $\boldsymbol{u}_k = A^k \boldsymbol{u}_0 = V \Lambda^k V^{-1} \boldsymbol{u}_0$

Eigenvalues and eigenvectors:

$$A \begin{bmatrix} 0.2 \\ 0.8 \end{bmatrix} = 1 \begin{bmatrix} 0.2 \\ 0.8 \end{bmatrix}, \quad A \begin{bmatrix} -1 \\ 1 \end{bmatrix} = 0.75 \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

Weights: $\boldsymbol{u}_0 = c_1 \boldsymbol{v}_1 + c_2 \boldsymbol{v}_2 = \begin{bmatrix} 0.02 \\ 0.98 \end{bmatrix} = 1 \begin{bmatrix} 0.2 \\ 0.8 \end{bmatrix} + 0.18 \begin{bmatrix} -1 \\ 1 \end{bmatrix}$

Markov matrices: Rental cars example

After k months: $\boldsymbol{u}_k = A^k \boldsymbol{u}_0 = V \Lambda^k \boldsymbol{c} = 1^k \cdot 1 \begin{bmatrix} 0.2 \\ 0.8 \end{bmatrix} + (0.75)^k \cdot 0.18 \begin{bmatrix} -1 \\ 1 \end{bmatrix}$

Thus, the eigenvector $\boldsymbol{v}_1 = \begin{bmatrix} 0.2 \\ 0.8 \end{bmatrix}$ with $\lambda_1 = 1$ is the steady state,

i.e. in the limit, **20% of the cars are in Denver** and 80% outside.

Initial vector **$\boldsymbol{u}_0$ is asymptotically irrelevant.**

Other eigenvector v_2 disappears because $|\lambda_2| < 1$.

Magnitude of λ_2 controls the **speed of convergence** to the steady state.

Markov matrices: Google example

Idea: for n websites, columns in $A_{n\times n}$ contain pairwise transition probabilities from one website to all other ones, computed from the **number of links between the sites.**

Then find $\boldsymbol{u}_\infty$ by a **random walk** that follows links (i.e. random surfing).

This steady state vector gives the **limit fraction of time at each site.**

The **ranking** of sites is then based on $\boldsymbol{u}_\infty$.

According to Google, the Markov matrix A has $2.7 \cdot 10^9$ rows and cols. Probably the largest eigenvalue problem ever solved!

Chapter 1

Linear Systems of Equations

Differential Equations and Matrix Exponentials

Applications to Differential Equations

Main Idea: **Convert constant-coefficient DEs into linear algebra.**

- One equation: $\frac{du}{dt} = \lambda u$ has solutions $u(t) = ce^{\lambda t}$.
- **Initial conditions:** Choose $c = u(0)$ (since $e^0 = 1$).
- n equations: $\frac{d\boldsymbol{u}}{dt} = A\boldsymbol{u}$, starting from $\boldsymbol{u}(0)$ at $t = 0$.
- **Equations are linear:**
  If $\boldsymbol{u}(t)$ and $\boldsymbol{v}(t)$ are solutions $\rightsquigarrow c\boldsymbol{u}(t) + d\boldsymbol{v}(t)$ is solution.
- Here, A is a constant matrix
  $\rightsquigarrow$ compute eigen-vectors and -values satisfying $A\boldsymbol{v} = \lambda\boldsymbol{v}$.
- Substitute $\boldsymbol{u}(t) = e^{\lambda t}\boldsymbol{v}$ into $\frac{d\boldsymbol{u}}{dt} = A\boldsymbol{u}$:

$$\rightsquigarrow \lambda e^{\lambda t}\boldsymbol{v} = Ae^{\lambda t}\boldsymbol{v} \Leftrightarrow A\boldsymbol{v} = \lambda\boldsymbol{v}.$$

First Order Equations

- All components of this **special solution** $\boldsymbol{u}(t) = e^{\lambda t}\boldsymbol{v}$ share the same time-dependent scalar $e^{\lambda t}$.
- Real eigenvalues: $\lambda > 0 \rightsquigarrow$ solution grows, $\lambda < 0 \rightsquigarrow$ solution decays.
- **Complex eigenvalues:** Real part describes **growth/decay**, imaginary part ω gives **oscillation** like a sine wave: $e^{i\omega t} = \cos(\omega t) + i\sin(\omega t)$.
- Complete solution is **linear combination of special solutions** for each $(\boldsymbol{v}, \lambda)$-pair. Coefficients are determined by initial conditions.
- **Recipe** (assuming no repeated eigenvalues $\rightsquigarrow n$ eigenvectors):
  - Write $\boldsymbol{u}(0)$ as combination of eigenvectors $c_1\boldsymbol{v}_1 + \cdots + c_n\boldsymbol{v}_n$
  - Multiply $\boldsymbol{v}_i$ by $e^{\lambda_i t}$
  - Solution is $\boldsymbol{u}(t) = c_1 e^{\lambda_1 t}\boldsymbol{v}_1 + \cdots + c_n e^{\lambda_n t}\boldsymbol{v}_n$.

Second Order Equations

- **Mechanics** is dominated by

$$\overset{\text{mass}}{m} \underset{\text{acceleration}}{\ddot{y}} + \underset{\text{damping}}{b\dot{y}} + \underset{\text{restoring force}}{ky} = 0$$

Linear second-order equation with constant coefficients m, b, k.

- Assume $m = 1$. define $\boldsymbol{u} = (y, \dot{y})^t$. The two eqs.

$$\frac{dy}{dt} = \dot{y} \text{ and } \frac{d\dot{y}}{dt} = -ky - b\dot{y}$$

convert to

$$\frac{d}{dt}\boldsymbol{u} = A\boldsymbol{u} \rightsquigarrow \frac{d}{dt}\begin{bmatrix} y \\ \dot{y} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k & -b \end{bmatrix}\begin{bmatrix} y \\ \dot{y} \end{bmatrix}$$

- **Reduction to first-order system!**

Second Order Equations

- Determinant $|A - \lambda I| = \lambda^2 + b\lambda + k \overset{!}{=} 0$
  $\rightsquigarrow$ two distinct eigenvalues λ_1, λ_2
  $\rightsquigarrow$ two eigenvectors. Here: $\boldsymbol{v}_1 = (1, \lambda_1)^t, \boldsymbol{v}_2 = (1, \lambda_2)^t$.

- Solution:
  $\boldsymbol{u}(t) = c_1 e^{\lambda_1 t} \boldsymbol{v}_1 + c_2 e^{\lambda_2 t} \boldsymbol{v}_2$.
  First component: $y(t)$ (position),
  Second component: $\dot{y}(t)$ (velocity).

The exponential of a matrix

- If there are n independent eigenvectors: Complete solution is **linear combination of special solutions** for each $(\boldsymbol{v}, \lambda)$-pair.
  More general & compact version?
- **Taylor series** of function $f(x)$ is $\sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n$, where $f^{(n)}(a)$ is the n-th derivative of f at point a.
- Exponential function, $a = 0$: $e^x = 1 + x + \frac{1}{2}x^2 + \frac{1}{6}x^3 + \cdots$
- Substitute square matrix At for x:

$$e^{At} = I + At + \frac{1}{2}(At)^2 + \frac{1}{6}(At)^3 + \cdots$$

$$\frac{d}{dt}e^{At} = A + A^2t + \frac{1}{2}A^3t^2 + \cdots = Ae^{At}$$

$\rightsquigarrow$ we immediately see that **$\boldsymbol{u} = e^{At}\boldsymbol{u}(0)$ solves $\frac{d}{dt}\boldsymbol{u} = A\boldsymbol{u}$.**

The exponential of a matrix

Simple case: n indep. eigenvectors $\rightsquigarrow$ A is diagonalizable $\rightsquigarrow$ $A = V\Lambda V^{-1}$:

$$
\begin{aligned}
e^{At} &= I + V\Lambda V^{-1}t + \frac{1}{2}(V\Lambda V^{-1}t)^2 + \cdots \\
&= V\left[I + \Lambda t + \frac{1}{2}(\Lambda t)^2 + \cdots\right]V^{-1} \\
&= Ve^{\Lambda t}V^{-1} = V\begin{bmatrix} e^{\lambda_1 t} & & \\ & \ddots & \\ & & e^{\lambda_n t}\end{bmatrix}V^{-1}.
\end{aligned}
$$

Substitute in general form of solution:

$$
\begin{aligned}
\boldsymbol{u}(t) = e^{At}\boldsymbol{u}(0) &= Ve^{\Lambda t}\underbrace{V^{-1}\boldsymbol{u}(0)}_{=\boldsymbol{c},\ \text{since } V\boldsymbol{c}=\boldsymbol{u}_0} \\
&= c_1 e^{\lambda_1 t}\boldsymbol{v}_1 + \cdots + c_n e^{\lambda_n t}\boldsymbol{v}_n.
\end{aligned}
$$

The exponential of a matrix

What if there are **not enough eigenvectors?** Example:

$$\frac{d}{dt}\boldsymbol{u} = A\boldsymbol{u} \rightsquigarrow \frac{d}{dt}\begin{bmatrix} y \\ \dot{y} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & 2 \end{bmatrix}\begin{bmatrix} y \\ \dot{y} \end{bmatrix}$$

$det(A-\lambda I) = \lambda^2 - 2\lambda + 1 = (\lambda - 1)^2 = 0 \rightsquigarrow$ repeated e.value $\lambda_1 = \lambda_2 = 1$
$\rightsquigarrow$ only one eigenvector $\rightsquigarrow$ **diagonalization not possible.**

Idea: Use Taylor series directly. **Series ends after linear term!**

$$e^{At} = e^{It}e^{(A-I)t} = e^{t}[I + (A-I)t + \frac{1}{2}\underbrace{(A-I)^2}_{0}t^2 + 0 + \cdots]$$

$$\boldsymbol{u}(t) = e^{At}\boldsymbol{u}(0) = e^{t}\left[I + \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} t\right]\boldsymbol{u}(0)$$

First component: $y(t) = e^t\, y(0) - te^t\, y(0) + te^t\, \dot{y}(0)$.

Chapter 1

Linear Systems of Equations

Singular Value Decomposition

Singular value decomposition

- Remember matrix diagonalization: $V^{-1}AV = \Lambda$. Three problems:
  - A must be square.
  - There are not always enough eigenvectors.
  - Only for symmetric matrices, the $\boldsymbol{v}_i$ are orthogonal.

- The SVD solves these problems, but at an additional price: we now have **two sets of singular vectors** $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$.
  Denoting by σ_i the **singular values**, they are related as:

$$A\boldsymbol{v}_i = \sigma_i \boldsymbol{u}_i \quad \text{and} \quad A^t \boldsymbol{u}_i = \sigma_i \boldsymbol{v}_i.$$

- If r is the rank of A, there will be r positive singular values, say $\sigma_1, \ldots, \sigma_r > 0$. All remaining ones will be zero.

Calculating the SVD

- Combine the two equations that define a pair $\boldsymbol{u}, \boldsymbol{v}$:

$$A^t(A\boldsymbol{v}) = A^t(\sigma\boldsymbol{u}) = \sigma(A^t\boldsymbol{u}) = \sigma(\sigma\boldsymbol{v}) = \sigma^2\boldsymbol{v}.$$

- So $A^tA\boldsymbol{v} = \sigma^2\boldsymbol{v}$.
  Singular values: **square roots of the eigenvalues of** A^tA (note that A^tA is positive semi-definite).
  Singular vectors $\boldsymbol{v}$: **eigenvectors of** A^tA.
- We can always choose orthonormal eigenvectors: Orthonormal basis always exists, because A^tA is symmetric $\rightsquigarrow$ orthogonally diagonalizable.
- Given $\boldsymbol{v}_i, \sigma_i$, compute $\boldsymbol{u}_i$ according to $\boldsymbol{u}_i = \sigma_i^{-1}A\boldsymbol{v}_i, \quad i = 1, \ldots, r$.
- Arrange singular values on **diagonal of a matrix** S and singular vectors as the columns of **orthogonal matrices** U **and** V.
- Then we have $AV = US$ and $A^tU = VS$.

Calculating the SVD: starting with U

- So far: Start with eigenvector decomposition of $A^tA \rightsquigarrow V$ and S, then compute $\boldsymbol{u}_i = \sigma_i^{-1} A \boldsymbol{v}_i$.
- Can also start with $AA^t \rightsquigarrow U$ and S:

$$AA^t\boldsymbol{u} = A\sigma\boldsymbol{v} = \sigma(A\boldsymbol{v}) = \sigma(\sigma\boldsymbol{u}) = \sigma^2\boldsymbol{u}.$$

  Singular values are also the **square roots of the eigenvalues of** AA^t, and the **eigenvectors of** AA^t are the columns of U.
- Then $\boldsymbol{v}_i = \sigma_i^{-1} A^t \boldsymbol{u}_i, \quad i = 1, \ldots, r.$
- **BUT:** Don't mix the two methods. Problem: Eigenvectors only determined **up to the direction**: if $\boldsymbol{v}$ is eigenvector of A^tA, then also $-\boldsymbol{v}$ is one: $A^tA\boldsymbol{v} = \lambda\boldsymbol{v} \;\Rightarrow\; A^tA(-\boldsymbol{v}) = \lambda(-\boldsymbol{v})$.
  So if you compute both $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ as eigenvectors of AA^t and A^tA, the signs can be arbitrary $\rightsquigarrow$ not necessarily a correct SVD.

Singular value decomposition

Orthogonality implies $AV = US \rightsquigarrow AVV^t = A = USV^t$.

Economy version of the **singular value decomposition** (SVD) of A:

U is $m \times r$, S is $r \times r$, V is $n \times r$.

$$A = USV^t = \begin{bmatrix} | & | & | & | \\ \boldsymbol{u}_1 & \boldsymbol{u}_2 & \dots & \boldsymbol{u}_r \\ | & | & | & | \end{bmatrix} \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_r \end{bmatrix} \begin{bmatrix} - & \boldsymbol{v}_1^t & - \\ - & \boldsymbol{v}_2^t & - \\ & \vdots & \\ - & \boldsymbol{v}_r^t & - \end{bmatrix}$$

What about the remaining $n - r$ vectors $\boldsymbol{v}_i$ and the $m - r$ vectors $\boldsymbol{u}_i$ with $\sigma_i = 0$? They span the nullspaces of A and A^t:

$$A\boldsymbol{v}_j = \boldsymbol{0}, \quad \text{for } j > r$$

$$A^t\boldsymbol{u}_j = \boldsymbol{0}, \quad \text{for } j > r$$

Singular value decomposition

Full singular value decomposition (SVD) of A:

U is $m \times m$, S is $m \times n$, V is $n \times n$.

$$\begin{bmatrix} | & | & & | & | & & | \\ \boldsymbol{u}_1 & \boldsymbol{u}_2 & \dots & \boldsymbol{u}_r & \boldsymbol{u}_{r+1} & \dots & \boldsymbol{u}_m \\ | & | & | & | & | & | & | \end{bmatrix} \begin{bmatrix} \sigma_1 & & & & & & \\ & \sigma_2 & & & & & \\ & & \ddots & & & & \\ & & & \sigma_r & & & \\ & & & & 0 & & \\ & & & & & \ddots & \\ & & & & & & 0 \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} - & \boldsymbol{v}_1^t & - \\ - & \boldsymbol{v}_2^t & - \\ & \vdots & \\ - & \boldsymbol{v}_r^t & - \\ - & \boldsymbol{v}_{r+1}^t & - \\ & & \\ & \vdots & \\ - & \boldsymbol{v}_n^t & - \end{bmatrix}$$

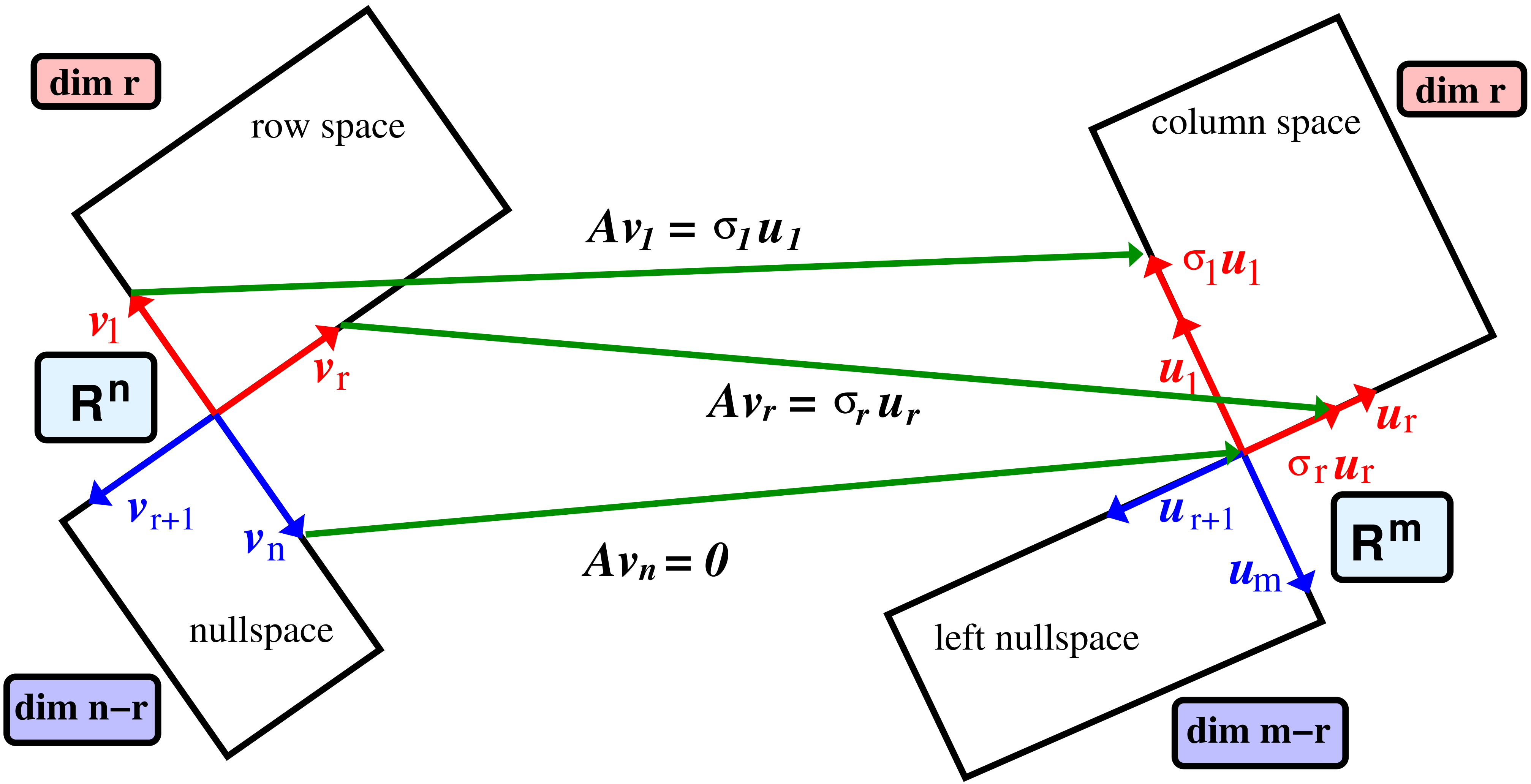
SVD and bases for the 4 subspaces

$$A\boldsymbol{v}_j = \sigma_j \boldsymbol{u}_j, \quad \text{for } j \le r. \qquad A^t\boldsymbol{u}_j = \sigma_j \boldsymbol{v}_j, \quad \text{for } j \le r.$$

$$A\boldsymbol{v}_j = \mathbf{0}, \quad \text{for } j > r. \qquad A^t\boldsymbol{u}_j = \mathbf{0}, \quad \text{for } j > r.$$

- **Columns of V with $\sigma_j > 0$ are an orthonormal basis for $C(A^t)$.** Diagonal elements in S scale the columns in V: $A^t\boldsymbol{y} = VS^tU^t\boldsymbol{y}$, so the columns of V with nonzero σ span the row space.
- **Last $n - r$ columns of V are an orthonormal basis for $N(A)$.**
- **Columns of U with $\sigma_j > 0$ are an orthonormal basis for $C(A)$.** Diagonal elements in S scale the columns in U: $A\boldsymbol{x} = USV^t\boldsymbol{x}$, so the columns of U with nonzero σ span the column space.
- **Last $m - r$ columns of U are an orthonormal basis for $N(A^t)$.**

SVD and bases for the 4 subspaces



SVD and linear systems

Assume A is a $n \times n$ matrix. With the SVD decomposition:

$$
\begin{aligned}
A\boldsymbol{x} &= \boldsymbol{b} \\
USV^t\boldsymbol{x} &= \boldsymbol{b} \\
SV^t\boldsymbol{x} &= U^t\boldsymbol{b} \\
S\boldsymbol{z} &= \boldsymbol{d}, \text{ where } \boldsymbol{z} = V^t\boldsymbol{x} \text{ and } \boldsymbol{d} = U^t\boldsymbol{b}.
\end{aligned}
$$

Written in blocks this is

$$
\begin{bmatrix}
\sigma_1 & & & & & & \\
& \sigma_2 & & & & & \\
& & \ddots & & & & \\
& & & \sigma_r & & & \\
& & & & \sigma_{r+1} {\color{red}= 0} & & \\
& & & & & \ddots & \\
& & & & & & \sigma_n {\color{red}= 0}
\end{bmatrix}
\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_r \\ z_{r+1} \\ \vdots \\ z_n \end{bmatrix}
=
\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_r \\ d_{r+1} \\ \vdots \\ d_n \end{bmatrix}
$$

Solution: $z_i = d_i/\sigma_i, \quad i = 1, \ldots, r$. What about the remaining entries?

SVD and linear systems

Recall: $\boldsymbol{b}$ must be in $C(A)$ (otherwise no solution exists),
last $m-r$ columns of U form basis of orthogonal complement $N(A^t)$.

Right hand side is

$$\boldsymbol{d} = U^t\boldsymbol{b} = \begin{bmatrix} - & \boldsymbol{u}_1^t & - \\ - & \boldsymbol{u}_2^t & - \\ & \vdots & \\ - & \boldsymbol{u}_r^t & - \\ - & \boldsymbol{u}_{r+1}^t & - \\ - & \boldsymbol{u}_{r+2}^t & - \\ & \vdots & \\ - & \boldsymbol{u}_m^t & - \end{bmatrix} \boldsymbol{b} = \begin{bmatrix} d_1 \\ \vdots \\ d_r \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

For $r+1 \le i \le n$: $0 \cdot z_i = 0 \rightsquigarrow$ can choose them arbitrarily.

Pseudoinverse

Alternative formalism for SVD solution:

- Write S^+ to denote the matrix obtained by replacing each σ_k in S^t by its reciprocal, so $S^+S = \begin{bmatrix} I_r & 0 \\ 0 & 0 \end{bmatrix}$
- Then compute:

$$\begin{aligned} USV^t\boldsymbol{x} &= \boldsymbol{b} \\ SV^t\boldsymbol{x} &= U^t\boldsymbol{b} \\ S\boldsymbol{z} &= U^t\boldsymbol{b} \\ \tilde{\boldsymbol{z}} = (\boldsymbol{z}_{[1:r]}, \mathbf{0})^t &= S^+U^t\boldsymbol{b} \\ \boldsymbol{x} = V\tilde{\boldsymbol{z}} &= \underbrace{VS^+U^t}_{A^+}\boldsymbol{b}. \end{aligned}$$

A^+ is the **pseudoinverse** of A: it maps $\boldsymbol{b} \in C(A)$ back to $\boldsymbol{x} \in C(A^t)$.

SVD and linear systems

Homogeneous equations:

- Zero right hand side: $\boldsymbol{b} = \mathbf{0}$
- Columns of V with $\sigma_j = 0$ are an orthonormal basis for the $N(A)$.
- **Solved immediately by SVD:**
  Any column of V whose corresponding $\sigma_j = 0$ yields a solution.

General case:

- Consider arbitrary $\boldsymbol{b}$. **Two cases: does $\boldsymbol{b}$ lie in $C(A)$ or not?**
- If **YES**, there exists a solution $\boldsymbol{x}$; in fact more than one, since any vector in the nullspace can be added to $\boldsymbol{x}$.
- SVD solution $\boldsymbol{x} = A^{+}\boldsymbol{b}$ is the "purest" solution: the one with smallest length $\|\boldsymbol{x}\|^2$. Why? $\boldsymbol{x} \in C(A^t)$, any nonzero component in the orthogonal nullspace would only increase the length.

General case

Consider arbitrary $\boldsymbol{b}$. **Two cases: does b lie in $C(A)$ or not?**

NO: If $\boldsymbol{b}$ is not in $C(A)$, there is **no solution**.

But: can compute **compromise** solution: Among all possible $\boldsymbol{x}$, it will minimize the **sum of squared errors** between left- and right hand side $\rightsquigarrow$ **least-squares methods.**

SVD and Zeroing

The SVD can solve further numerical problems:

- Zero a small singular value if σ_j is (too) close to zero.
- This forces a **zero coefficient** instead of a **random large coefficient** that would scale a vector "close to" the nullspace:

$$\boldsymbol{x} = V\boldsymbol{z} = \underbrace{\boldsymbol{v}_1 \frac{d_1}{\sigma_1} + \cdots + \boldsymbol{v}_r \frac{d_r}{\sigma_r}}_{\text{rowspace}} + \underbrace{\boldsymbol{v}_{r+1} z_{r+1} + \cdots + \boldsymbol{v}_n z_n}_{\text{nullspace}}$$

- Rule of thumb: if the ratio $\sigma_j/\sigma_1 < \epsilon_m$ then **zero the entry in the pseudo-inverse matrix**, since the value is probably **corrupted by roundoff** anyway.

Chapter 1

Linear Systems of Equations

The condition number

Conditioning

- **Conditioning** is a measure of the **sensitivity to perturbations**, due to measurement error, statistical fluctuations in the data analysis process, or caused by roundoff errors.
  These perturbations might affect the numerical values in $\boldsymbol{b}$ and/or A.

- Conditioning describes how this **problem error** $\Delta \boldsymbol{b}, \Delta A$ will affect the **solution error** $\Delta \boldsymbol{x}$.

- A function of the **problem itself**, **independent** of the algorithm used. (In practice, however, this separation between the problem and the algorithm might be less clear as it seems...Example: In elimination, the values in A change after every elimination step.)

Vector and matrix norms

- How to compare **closeness** of two vectors $\boldsymbol{x}$ and $\boldsymbol{x}+\Delta\boldsymbol{x}$?
  Look at relative quantities like $\frac{\|\Delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|} \le \delta$, or $\frac{\|\Delta\boldsymbol{x}\|}{\|\boldsymbol{x}+\Delta\boldsymbol{x}\|} \le \delta$.
- **Vector norm properties:**
  (i) $\|\boldsymbol{x}\| > 0,\ \forall \boldsymbol{x} \neq 0$
  (ii) $\|a\boldsymbol{x}\| = |a|\|\boldsymbol{x}\|$
  (iii) $\|\boldsymbol{x}+\boldsymbol{y}\| \le \|\boldsymbol{x}\| + \|\boldsymbol{y}\|$
- The **vector** p**-norms (**ℓ_p **norms)** are defined by

$$\|\boldsymbol{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}, \quad 1 \le p \le \infty,$$

$$\|\boldsymbol{x}\|_\infty = \max(|x_1|, \cdots |x_n|).$$

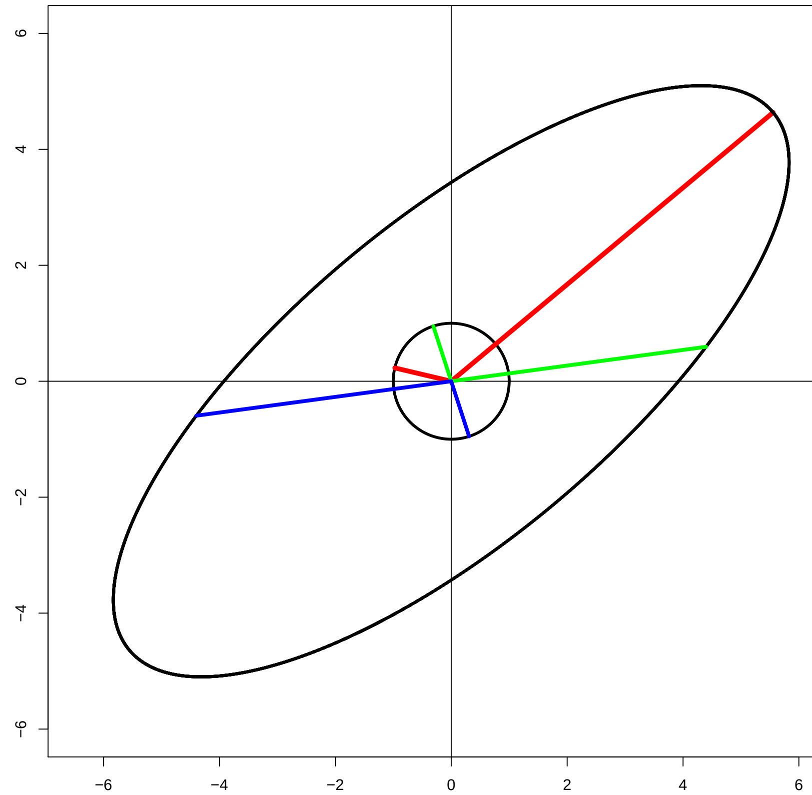
- $\|x\|_2$ is the usual **Euclidean norm.** What about **matrix norms**?

Vector and matrix norms

- $\boldsymbol{y} = A\boldsymbol{x}$ **transforms** vector $\boldsymbol{x}$ into $\boldsymbol{y}$
  $\rightsquigarrow$ A **rotates and/or stretches** $\boldsymbol{x}$.

- Consider the effect of A on a unit vector $\boldsymbol{x}$
  (i.e. $\boldsymbol{x}$ so that $\|\boldsymbol{x}\|_2 = 1$).

- The "largest" $A\boldsymbol{x}$ value is a measure of the **geometric effect of the transformation** A.



- The 2-norm is $\|A\|_2 = \max_{\|\boldsymbol{x}\|_2=1} \|A\boldsymbol{x}\|_2$.

- Also called the **spectral norm** of A,
  because $\|A\|_2 = \sqrt{\max(\lambda_i)}$ where λ_i is an **eigenvalue** of A^tA
  (See handout on matrix norms).

Vector and matrix norms

- Two other **useful and easier-to-calculate** matrix norms:

- $\|A\|_1 = \max_j \sum_{i=1}^{m} |a_{ij}|$ **column sum norm**.

- $\|A\|_\infty = \max_i \sum_{j=1}^{n} |a_{ij}|$ **row sum norm**.

- $\|A\|$ satisfies vector norm properties **PLUS**
  $\|AB\| \leq \|A\|\|B\|$ and, in particular, $\|A\boldsymbol{x}\| \leq \|A\|\|\boldsymbol{x}\|$.

Sensitivity to perturbations

- Original system is $A\boldsymbol{x} = \boldsymbol{b}$. Assume that right hand side is changed to $\boldsymbol{b} + \Delta\boldsymbol{b}$ because of roundoff or measurement error.

- Then the solution is changed to $\boldsymbol{x} + \Delta\boldsymbol{x}$.
  Goal: Estimate the change in the solution from the change $\Delta\boldsymbol{b}$.
  Subtract $A\boldsymbol{x} = \boldsymbol{b}$ from $A(\boldsymbol{x} + \Delta\boldsymbol{x}) = \boldsymbol{b} + \Delta\boldsymbol{b}$
  to find $A(\Delta\boldsymbol{x}) = \Delta\boldsymbol{b} \quad \Leftrightarrow \quad \Delta\boldsymbol{x} = A^{-1}\Delta\boldsymbol{b}$

$$
\begin{aligned}
\Delta\boldsymbol{x} = A^{-1}\Delta\boldsymbol{b} \quad &\Rightarrow \quad \|\Delta\boldsymbol{x}\| \le \|A^{-1}\|\|\Delta\boldsymbol{b}\| \\
A\boldsymbol{x} = \boldsymbol{b} \quad &\Rightarrow \quad \|\boldsymbol{b}\| \le \|A\|\|\boldsymbol{x}\|
\end{aligned}
$$

  Multiplication and division of both sides by $(\|\boldsymbol{b}\|\|\boldsymbol{x}\|)$ gives

$$
\frac{\|\Delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|} \le \underbrace{\|A\|\|A^{-1}\|}_{k(A)}\frac{\|\Delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|}.
$$

Sensitivity to perturbations

- Error can also be in the matrix: we have $A + \Delta A$ instead of the true matrix A.
- Subtract $A\boldsymbol{x} = \boldsymbol{b}$ from $(A + \Delta A)(\boldsymbol{x} + \Delta\boldsymbol{x}) = \boldsymbol{b}$ to find $A(\Delta\boldsymbol{x}) = -(\Delta A)(\boldsymbol{x}+\Delta\boldsymbol{x}) \quad \Leftrightarrow \quad \Delta\boldsymbol{x} = -A^{-1}(\Delta A)(\boldsymbol{x}+\Delta\boldsymbol{x})$

$$\|\Delta\boldsymbol{x}\| \leq \|A^{-1}\|\|\Delta A\|\|\boldsymbol{x} + \Delta\boldsymbol{x}\|$$

$$\frac{\|\Delta\boldsymbol{x}\|}{\|\boldsymbol{x} + \Delta\boldsymbol{x}\|} \leq \underbrace{\|A\|\|A^{-1}\|}_{k(A)}\frac{\|\Delta A\|}{\|A\|}.$$

- Conclusion: Errors can be in the matrix or in the r.h.s.
  This **problem error** is amplified into the **solution error $\Delta\boldsymbol{x}$**.
  Rel. solution error is bounded by $k(A)$ times rel. problem error.

Condition number

- $k(A) = \|A\|\|A^{-1}\|$ is called the **condition number** of A.
- $1 \leq k(A) \leq \infty$.
- An **ill-conditioned problem has a large condition number.**
- Small residual does **not** guarantee accuracy for ill-conditioned problems: $\hat{\boldsymbol{x}}$ is a numerical solution to $A\boldsymbol{x} = \boldsymbol{b}$, and $\Delta\boldsymbol{x} = \boldsymbol{x} - \hat{\boldsymbol{x}}$.
  Define **residual** $\boldsymbol{r}$ to represent the error $\boldsymbol{r} = \boldsymbol{b} - A\hat{\boldsymbol{x}} = \boldsymbol{b} - \hat{\boldsymbol{b}} = \Delta\boldsymbol{b}$:

$$\frac{\|\Delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|} \leq k(A)\frac{\|\boldsymbol{r}\|}{\|\boldsymbol{b}\|}.$$

- $k(A)$ is a **mathematical property of the coefficient matrix** A.
- In **exact** math a singular matrix has $k(A) = \infty$. $k(A)$ indicates how close a matrix is to being **numerically** singular.

Condition number

- $k(A)$ can be measured with any matrix p-norm.
- Spectral norm $\|A\|_2 = \sqrt{\lambda_{\max}(A^t A)} = \sigma_{\max}(A)$

  For an invertible matrix M we have:

$$M\boldsymbol{v} = \lambda \boldsymbol{v} \;\Rightarrow\; \boldsymbol{v} = \lambda M^{-1}\boldsymbol{v} \;\Rightarrow\; M^{-1}\boldsymbol{v} = \lambda^{-1}\boldsymbol{v},$$

  so M^{-1} has the same eigenvectors but inverse eigenvalues, and

$$\|A^{-1}\|_2 = \sqrt{\lambda_{\min}(A^t A)} = \sigma_{\min}(A) \rightsquigarrow \quad k(A) = \frac{\sigma_{\max}}{\sigma_{\min}}.$$

- Can be generalized to singular/rectangular matrices: $k(A) = \|A\|\|A^+\|$ $=$ ratio of largest and smallest positive singular value.