1.5 Linear systems and matrices

Very many applications in imaging and visualization reduce to solving linear systems after mathematical modeling. This is why we will spend considerable time on ways to solve them.

Definition (Linear system)**.** *A* ***linear equation*** *has the form*

$$a_1x_1 + \ldots a_nx_n = b,$$

where $x_1, \ldots, x_n \in \mathbb{R}$ *are the unknowns/variables and* $a_1, \ldots, a_n, b \in \mathbb{R}$ *are constants.*
A ***linear system*** *is a finite set of* m *linear equations with variables* $x_1, \ldots, x_n \in \mathbb{R}$*:*

$$\begin{array}{ccc} a_{11}x_1 + \ldots + a_{1n}x_n & = & b_1 \\ \vdots \qquad\qquad \vdots & & \vdots \\ a_{m1}x_1 + \ldots + a_{mn}x_n & = & b_m \end{array}$$

or in matrix notation

$$\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}.$$

In short

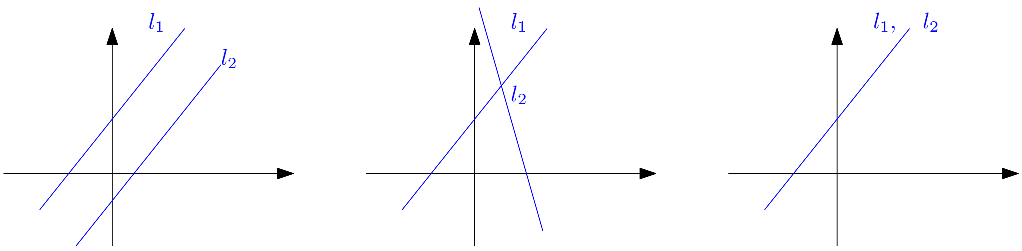
$$Ax = b,$$

with $A = (a_{ji})$ *and* $b = (b_j)$*.*

Property: A linear system always has either no solution, exactly one solution or infinitely many solutions.

This property is illustrated by this simple example of a pair of straight lines in $\mathbb{R}^2$, which can be modeled as a linear equation each:

$$\begin{array}{lll} (l_1) & a_1x + b_1y = c_1 & (a_1, b_1) \neq (0,0) \\ (l_2) & a_2x + b_2y = c_2 & (a_2, b_2) \neq (0,0) \end{array}$$



no solution   exactly one solution   infinite many solutions

The case of no solution corresponds here to parallel lines, the case of exactly one solution to the lines intersecting, and the case of infinitely many solutions to the case of completely overlapping lines.

One of the easiest ways to solve linear systems for the unknown variables is to know the inverse of the matrix.

Definition (Inverse matrix)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a square matrix. If there exists $B \in \mathbb{R}^{n\times n}$ with*

$$AB = BA = I_n,$$

*(where I_n denotes the $n\times n$ identity matrix), then A is called **invertible**, and we call $B := A^{-1}$ the **inverse** of A.*

For small matrices there are explicit formulas to compute the inverse (for bigger matrices, there are unfortunately no explicit formulas). Here is the example of a 2×2 matrix: Let $A = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$. If $ad - bc \neq 0$, then A is invertible with

$$A^{-1} = \frac{1}{ad - bc}\begin{pmatrix} d & -b \\ -c & a \end{pmatrix}.$$

Numerical example: set $A = \left(\begin{smallmatrix} 1 & 2 \\ 3 & 4 \end{smallmatrix}\right)$, then $A^{-1} = \frac{1}{-2}\left(\begin{smallmatrix} 4 & -2 \\ -3 & 1 \end{smallmatrix}\right) = \left(\begin{smallmatrix} -2 & 1 \\ \frac{3}{2} & -\frac{1}{2} \end{smallmatrix}\right)$. We can check that this is indeed the inverse:

$$AA^{-1} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}\begin{pmatrix} -2 & 1 \\ \frac{3}{2} & -\frac{1}{2} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = A^{-1}A.$$

There are some useful properties of invertible matrices:

- If the inverse of a matrix exists, it is unique.
- $A, B \in \mathbb{R}^{n\times n}$ invertible $\iff$ AB invertible and $(AB)^{-1} = B^{-1}A^{-1}$.
- $A \in \mathbb{R}^{n\times n}$ invertible $\iff$ $Ax = 0$ has only the trivial solution $x = 0$
  $\iff$ $Ax = b$ has exactly one solution for every $b \in \mathbb{R}^n$, $x = A^{-1}b$.

The latter in particular is one of the easiest ways of solving a linear system $Ax = b$. If A is invertible and the inverse A^{-1} is known, just compute $x = A^{-1}b$.

Unfortunately, except in very simple matrices, it is not easy to determine if a matrix is invertible. There are, however, several tools that can help. One of them is the determinant:

Definition (Determinant)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a square matrix, $A = (a_{ji})$. The **determinant** of A is defined as*

$$\det(A) := \sum_{\sigma\in S_n} sgn(\sigma)\prod_{i=1}^{n} a_{i,\sigma(i)},$$

where S_n is the set of all permutations of $\{1, \ldots, n\}$.

This is the so-called Leibniz formula for determinants, which is one of the shortest ways to introduce the determinant. Unfortunately, it is really impractical to use to actually compute determinants. As before, for small matrices, there are easy-to-use schemes to compute the determinant of a matrix $A \in \mathbb{R}^{n\times n}$:

- $n = 2$:

  $\oplus \qquad \ominus$

  $$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$$

  $$\implies \quad \det(A) = a_{11}a_{22} - a_{12}a_{21}.$$

- $n = 3$:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{matrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{matrix}$$

$$\begin{aligned} \implies \quad \det(A) &= a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} \\ &\quad - a_{13}a_{22}a_{31} - a_{11}a_{23}a_{32} - a_{12}a_{21}a_{33}. \end{aligned}$$

In case your matrix is not that small, there are some useful properties that can help your computations. Let $A \in \mathbb{R}^{n\times n}$.

- If $A = (a_{ji})$ is a triangular, or diagonal matrix, then

$$\det(A) = a_{11}a_{22}\cdots a_{nn}.$$

- $\det(A) = \det(A^T)$.
- $\det(\lambda A) = \lambda^n \det(A)$ for $\lambda \in \mathbb{R}$.
- $A, B \in \mathbb{R}^{n\times n}$, then $\det(AB) = \det(A)\det(B)$.
- A invertible $\iff \det(A) \neq 0$.
- A invertible, then $\det(A^{-1}) = \frac{1}{\det(A)}$.

In particular, we see that a matrix with non-zero determinant is invertible.

We now introduce more properties of matrices:

Definition (Image)**.** *Let $f : \mathbb{R}^n \to \mathbb{R}^m$ be a mapping. The **image** of f is defined as*

$$Im(f) := \big\{ b \in \mathbb{R}^m : \ \exists x \in \mathbb{R}^n : \ f(x) = b \big\}.$$

*Correspondingly, let $A \in \mathbb{R}^{m\times n}$, then the **image** of A is*

$$Im(A) = \big\{ b \in \mathbb{R}^m : \ \exists x \in \mathbb{R}^n : \ Ax = b \big\}.$$

The image of A is generated by linear combinations of the columns of A,

$$b = Ax = \big(col_1(A) \quad \cdots \quad col_n(A) \big)\, x = col_1(A)x_1 + \cdots + col_n(A)x_n.$$

*Hence we also call $Im(A)$ the **column space** of A, the subspace of $\mathbb{R}^m$ spanned by $\{col_i(A)\}_{i=1}^n$. Correspondingly, the **row space** of A is the subspace of $\mathbb{R}^n$ spanned by $\{row_j(A)\}_{j=1}^m$.*

This allows us to formulate some properties. Let $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$.

- A linear system $Ax = b$ has a solution $\iff$ b is contained in column space of A, i.e., $\mathrm{Im}(A)$.
- The column and row space of A always have the same dimension.

This latter feature is remarkable, and allows us to define the rank of a matrix:

Definition (Rank)**.** *Let $A \in \mathbb{R}^{m\times n}$. The dimension of the column and row space of A is called* ***rank*** *of A, rank(A).*

Closely related to the rank of a matrix is its kernel:

Definition (Kernel)**.** *Let $A \in \mathbb{R}^{m\times n}$. The set*

$$\ker(A) := \big\{x \in \mathbb{R}^n : \; Ax = 0\big\}$$

is a subspace of $\mathbb{R}^n$ and is called ***null space*** *of A or* ***kernel*** *of A.*

The rank and the kernel of a matrix are important properties of a matrix. Let $A \in \mathbb{R}^{m\times n}$. Then:

- $\mathrm{rank}(A) = 0 \quad \Longleftrightarrow \quad A = 0$.
- $\mathrm{rank}(A) \le \min(m, n)$.
- $n - \mathrm{rank}(A) = \dim\ker(A)$.
- If $m = n$, i.e. the matrix is square, then:
  A invertible $\quad\Longleftrightarrow\quad \mathrm{rank}(A) = n \quad$ ("A has full rank")
  $\quad\Longleftrightarrow\quad \dim\ker(A) = 0 \quad (\Longleftrightarrow \ker(A) = \{0\})$.

Again, the latter feature is important for the solution of linear systems. If the rank is full or the kernel just contains the zero vector, then the linear system with matrix A has exactly one solution.

Numerical example: We can use the properties introduced earlier (determinant, rank, kernel) to determine whether a matrix is invertible.

- $A = \left(\begin{smallmatrix}1 & 2\\ 3 & 4\end{smallmatrix}\right)$, then $\quad \det(A) = 4 - 6 = -2 \quad\Longrightarrow\quad A$ invertible, and $\mathrm{rank}(A) = 2$.
- $A = \left(\begin{smallmatrix}1 & 2\\ 2 & 4\end{smallmatrix}\right)$ has linearly dependent columns, e.g. $2\left(\begin{smallmatrix}1\\ 2\end{smallmatrix}\right) = \left(\begin{smallmatrix}2\\ 4\end{smallmatrix}\right) \quad\Longrightarrow\quad \mathrm{rank}(A) = 1 \quad\Longrightarrow$ A not invertible. In addition: $\det(A) = 4 - 4 = 0$, which also implies non-invertibility.

1.6 Solving linear systems: Computed Tomography

As one of the prime application examples in this course we consider Transmission X–ray Computed Tomography (X-ray CT). Here the goal is to recover a function $f : V \to \mathbb{R}$ that maps a volume of interest $V \subset \mathbb{R}^n$ to some real–valued property (in X-ray CT that is the X–ray attenuation coefficient of the object that is measured).

One typical approach to this problem is the so-called "series expansion" approach outlined below:

- **Step 1:** Discretize f.

Choose a finite set of "basis functions" $b_i : V \to \mathbb{R}$, $i \in I$, such that f can be approximated by a linear combination $\hat{f}$ of "basis functions" b_i. (The b_i can be a real basis, but that is not necessary.) That is:

$$\hat{f}(\cdot) = \sum_{i \in I} x_i b_i(\cdot), \qquad \{x_i\}_{i \in I} \subset \mathbb{R},$$

such that $\|f - \hat{f}\| < \varepsilon$, where $\varepsilon > 0$ small.

The vector $x = (x_i)_{i \in I}$ is the discretized version of the quantity to be reconstructed, f. If the b_i are linearly independent, the mapping $\hat{f} \mapsto x$ is bijective.

Standard choice for 2D case: $k \times k$ pixel grid, that is $I = \{1, \dots, k^2\}$ and

$$b_i(c_1, c_2) := \begin{cases} 1 & \text{if } (c_1, c_2) \text{ is inside } i\text{–th pixel} \\ 0 & \text{else.} \end{cases}$$

- **Step 2:** Decide on a measurement model $\mathcal{M}_j$.

  Let $m = (m_j)_{j \in J} \subset \mathbb{R}$ denote the finite set of measured values of the detector. We need a model of the measurement process

  $$\mathcal{M}_j : \big(f : V \to \mathbb{R}\big) \longrightarrow \mathbb{R},$$

  which describes how the property f could generate our measured signals m, such that $\mathcal{M}_j f = m_j$ for all $j \in J$. Assuming $\mathcal{M}_j$ is linear, we have

  $$\mathcal{M}_j f \approx \mathcal{M}_j \hat{f} = \mathcal{M}_j \left(\sum_{i \in I} x_i b_i \right) = \sum_{i \in I} x_i \; \mathcal{M}_j b_i.$$

  In our example of X-ray CT a simple model is the X-ray transform: $\mathcal{M}_j f = \int_{L_j} f(x)\, dx$, which is just a line integral along the line L_j the X-ray took from source to detector.

  This allows us to form a system equation

  $$Ax = m$$

  using the system matrix $A = (a_{ji})$ with $a_{ji} = \mathcal{M}_j b_i$.

- **Step 3:** Solve the system equation $Ax = m$.

  Compute a solution $\hat{x}$ (or approximation to solution) such that $\hat{f}^* = \sum_{i \in I} \hat{x}_i b_i$ is the desired reconstruction of f.

Each of the three steps has many choices for implementation. Of particular interest for us right now is step 3, which involves solving a linear system.

In the following we present a method for solving such a linear system (Kaczmarz's method), which requires one more mathematical tool:

Orthogonal projections. To project a vector $x \in \mathbb{R}^n$ orthogonally onto a unit vector $u \in \mathbb{R}^n$ (a unit vector has length 1, i.e. $\|u\| = 1$) we use the projection operator

$$P_u : \mathbb{R}^n \to \mathbb{R}^n, \qquad P_u(x) = \langle x, u \rangle u$$

P_u has the following useful properties:

- We have $P_u(u) = \langle u, u\rangle u = u$.
- For $x = x_\| + x_\perp$, where
  - $x_\|$ is the part of x which is parallel to u and
  - $x_\perp$ is the part that is orthogonal to u

  we have
$$P_u(x) = \langle x_\|, u\rangle u + \langle x_\perp, u\rangle u = \|x_\|\| u + 0u = x_\|.$$

Now we can introduce a method for solving linear systems.

Kaczmarz's method. (also called ART — Algebraic Reconstruction Technique)
Let
$$Ax = m$$
be a linear system with $A \in \mathbb{R}^{l\times n}$, $A \neq 0$, $m = (m_j) \in \mathbb{R}^l$ and unknowns $x \in \mathbb{R}^n$.

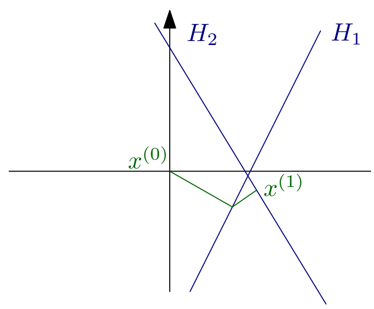
Denote $a_j := \mathrm{row}_j(A)$, then each row $a_j x = m_j$ of the linear system forms a hyperplane in $\mathbb{R}^n$,
$$H_j := \big\{x \in \mathbb{R}^n : \ \langle a_j, x\rangle = m_j\big\}.$$
(A hyperplane of $\mathbb{R}^n$ is a subspace of dimension $n-1$ and divides $\mathbb{R}^n$ in half.) We have
$$x = \bigcap_{j=1}^{l} H_j,$$
that means the solution of the linear system is the intersection of all hyperplanes H_j (think back to our pair of lines example earlier).

Kaczmarz's method is now to compute x by successively projecting onto the hyperplanes H_j using the projection operator introduced above, until we converge to the solution. If P_{H_j} denotes the projection onto H_j, then one iteration of Kaczmarz's method can be written as $\Phi = P_{H_l} \circ \ldots \circ P_{H_1}$. In full, the method reads

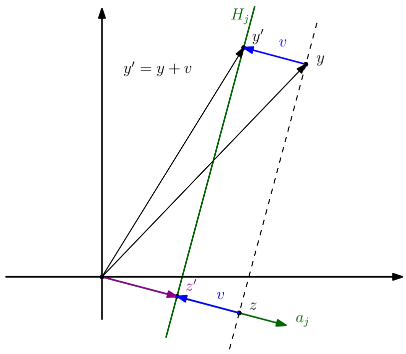
$$\begin{aligned}
&x^{(0)} = 0 && \text{(or any other starting value)}\\
&\text{iterate over } k = 1, \ldots, \hat{k}\\
&\qquad x^{(k+1)} = \Phi\ x^{(k)}
\end{aligned}$$

In case $Ax = m$ has a solution, this method will indeed converge to a solution.

Here is an example illustration of one iteration of the Kaczmarz method in $\mathbb{R}^2$ with two hyperplanes H_1, H_2:



In order to calculate P_{H_j} we denote $y := P_{H_{j-1}} \circ P_{H_{j-2}} \circ \ldots \circ P_{H_1} x^{(k)}$ and $y' := P_{H_j} y$.



In order to obtain $y' = P_{H_j} y$ we try to find a correction v such that $y' = y + v$. As y' is the projection of y onto H_j we can expect that v will be parallel to a_j, because a_j is the normal vector of (and thus orthogonal to) H_j.

At first, we note that v can be obtained as

$$v = z' - z,$$

where z is the projection of y onto $a_j / \|a_j\|_2$ and z' is the projection of z onto H_j. Thus, we can compute y' via the following three steps:

1. Compute z: The projection of y onto $a_j / \|a_j\|_2$ can be obtained by employing the definition of a projection, i.e.

$$z = \left\langle y, \frac{a_j}{\|a_j\|_2} \right\rangle \frac{a_j}{\|a_j\|_2}.$$

2. Compute z': z' has to be parallel to $a_j / \|a_j\|_2$ as well as an element of H_j. Thus, it must be a scaled version of $a_j / \|a_j\|_2$. Choosing $z' = m_j a_j / \|a_j\|_2^2$ yields the desired vector as the following calculation shows:

$$\left\langle z', a_j \right\rangle - m_j = \left\langle m_j \frac{a_j}{\|a_j\|_2^2}, a_j \right\rangle - m_j = m_j \frac{\langle a_j, a_j \rangle}{\|a_j\|_2^2} - m_j = m_j - m_j = 0.$$

3. Compute y': We just have to combine the above made observations:

$$y' = y + v = y + z' - z = y + \frac{m_j - \langle y, a_j \rangle}{\|a_j\|_2^2} a_j.$$