Basics in Applied Mathematics – Part I: Numerics

Lecture 1: Introduction, floating-point arithmetic, conditioning and norms

JProf. Dr. Diyora Salimova

Department of Applied Mathematics, University of Freiburg

20 October 2026

The course and the team

Part	Lecturer	Weeks	Dates
I Numerics	JProf. Dr. Diyora Salimova	1–5	20.10.–19.11.2026
II Stochastics	Prof. Dr. Peter Pfaffelhuber	6–10	24.11.–22.12.2026
III Optimization	Dr. Simon Buchwald	11–15	12.01.–11.02.2027

Assistants: Samuel Ayomide Adeosun, Ilkhom Mukhammadiev Tutor: Anas Achmit

- **Lectures:** Tuesday and Thursday, 10–12, HS II, Albertstr. 23b
- **Tutorials:** Friday, 12–14 and 14–16, HS II (two groups)
- **Course page:** `pfaffelh.github.io/hp/2026WS_BAM.html`

Studienleistung and exam

Written tests

- two per part, 30 minutes each
- **one of the two must be passed** in each part
- Numerics: **13 Nov** and **27 Nov 2026**, 13:45–14:15
- make-up test (Numerics): 12 Feb 2027, 13:45–14:15 — for students who joined too late for the regular tests

Oral exam

- all three parts are prepared; the exam covers two of them

Practical exercise (after 13 Feb 2027)

- groups of two assigned a topic
- report and presentation

Exercise sheets and kick-off

Exercise sheets (Numerics)

- five sheets, handed out on Thursdays: 22.10., 29.10., 5.11., 12.11., 19.11.
- optional, not graded
- discussed in the tutorial of the following week
- the best preparation for the tests

New to programming? "Python for Data Analysis", Thursday 14–16.

Kick-off: hands-on introduction to Lytris AI

Friday, 23 October 2026 at 12:30, HS II

- one joint session for both groups
- **bring your laptop**
- Zoom link on the course page

Material

- **Lecture notes** on the course page, updated during the course.
  Asterisk * = not presented in the lecture, not part of the testable core.
- **Literature:** Bartels, *Numerical Mathematics* 3×9 (reference text) · Quarteroni–Sacco–Saleri, *Numerical Mathematics* · Trefethen–Bau, *Numerical Linear Algebra* · Higham, *Accuracy and Stability of Numerical Algorithms*

1 floating point, norms	4 SVD	7 nonlinear equations	10 differential equations
2 linear systems	5 eigenvalues	8 interpolation, splines	
3 least squares, QR	6 iterative methods	9 integration	

Today

After this lecture you can

(i) name the four sources of error in a numerical computation,

(ii) explain how a computer stores real numbers, why `0.1 + 0.2 == 0.3` is `False`, and what cancellation is,

(iii) distinguish a well-conditioned problem from a stable algorithm,

(iv) compute $\|A\|_1$, $\|A\|_\infty$ and $\|A\|_2$.

What numerics is about

Algorithms that compute *approximate* solutions with finitely many operations on numbers of finite precision.

Problems in this part: $Ax = b$ · least squares · eigenvalues, singular values · $f(x) = 0$ · interpolation · integration · differential equations

Three questions for every method

1. **Accuracy:** how far is the computed result from the exact solution?
2. **Cost:** how many operations, how much memory, as a function of the size n?
3. **Robustness:** does the method survive perturbed data (measurement errors, rounding)?

Four sources of error

- **model error** — the equations do not describe reality exactly
- **data error** — the input is known only approximately
- **method error** — an exact problem is replaced by a computable one (series $\to$ finite sum, limit $\to$ iterate)
- **rounding error** — the computer stores numbers with finite precision

Example 1.1 (two formulas for one function)

$$f(x) = \frac{1}{x} - \frac{1}{x+1} \qquad \text{and} \qquad g(x) = \frac{1}{x(x+1)} \qquad (x > 0)$$

are the same function. Evaluate both in double precision for $x = 10^k$.

Before you see the numbers: which formula would you trust for large x — and why?

Example 1.1: two formulas, two results

k	x	f(x)	g(x)	rel. difference
1	1e+01	9.0909090909e-03	9.0909090909e-03	3.82e-16
4	1e+04	9.9990001000e-09	9.9990001000e-09	6.39e-14
8	1e+08	1.0000000003e-16	9.9999999000e-17	1.03e-08
12	1e+12	1.0000484359e-24	1.0000000000e-24	4.84e-05
16	1e+16	0.0000000000e+00	1.0000000000e-32	1.00e+00

rel. difference $= |\hat{f} - \hat{g}|/|\hat{g}|$ for the computed values $\hat{f}$, $\hat{g}$

- f: up to k of the 16 digits lost for $x = 10^k$ (worst-case rule of thumb; at $x = 10^{16}$, $x + 1$ rounds to x and $f = 0$); g: correct to all displayed digits.
- Reason: two reciprocals $\approx 10^{-k}$, each with error $\lesssim 10^{-16-k}$; their difference is only 10^{-2k}.
- Not a defect of f as a formula — **cancellation** (Section 1.2.3).

Definition 1.2

Integers $\beta \geq 2$ (base), $t \geq 1$ (precision), $e_{\min} \leq e_{\max}$:

$$\mathbb{F} = \left\{ \pm\, \beta^e \sum_{i=1}^{t} d_i \beta^{-i} \;:\; d_i \in \{0, \ldots, \beta-1\},\ d_1 \neq 0,\ e_{\min} \leq e \leq e_{\max} \right\} \cup \{0\}.$$

$m = \sum_{i=1}^{t} d_i \beta^{-i} \in [\beta^{-1}, 1)$ mantissa, e exponent;
$d_1 \neq 0$ (normalisation) makes the representation unique.

IEEE 754 double precision: $\beta = 2$, $t = 53$, $e_{\min} = -1021$, $e_{\max} = 1024$

- 64 bits: sign, 11 exponent bits, 52 mantissa bits ($d_1 = 1$ is not stored)
- largest double $\approx 1.8 \cdot 10^{308}$, smallest positive normalised $\approx 2.2 \cdot 10^{-308}$
- also: subnormal numbers, $\pm\infty$, `NaN`; single precision $t = 24$

Our mantissa lies in $[\frac{1}{2}, 1)$; with the convention $m \in [1, 2)$ the exponent range reads $-1022, \ldots, 1023$.

Two properties of $\mathbb{F}$

- $\mathbb{F}$ is **finite** ($\approx 1.8 \cdot 10^{19}$ normalised doubles) $\Rightarrow$ not closed under $+, -, \cdot, /$
- $\mathbb{F}$ is **not equally spaced**: spacing β^{e-t} in $[\beta^{e-1}, \beta^e)$; doubles: 2^{-52} in $[1, 2)$, 2^{k-52} in $[2^k, 2^{k+1})$ for $-1022 \leq k \leq 1023$
  relative spacing $2^{k-52}/x \in (2^{-53}, 2^{-52}]$: nearly constant

```
spacing of doubles near    1e+00: 2.220e-16   (relative: 2.220e-16)
spacing of doubles near    1e+03: 1.137e-13   (relative: 1.137e-16)
spacing of doubles near    1e+15: 1.250e-01   (relative: 1.250e-16)
0.1 + 0.2 == 0.3 ? False    0.1+0.2 = 0.30000000000000004
```

$0.1 \notin \mathbb{F}$ (periodic binary expansion): the literal `0.1` is already rounded, and so is the sum.

Definition 1.3

$\mathrm{fl}\colon [-x_{\max}, x_{\max}] \to \mathbb{F}$, $x \mapsto$ a nearest element of $\mathbb{F}$ $(x_{\max} = (1-\beta^{-t})\beta^{e_{\max}})$.
Unit roundoff $u := \frac{1}{2}\beta^{1-t}$; doubles: $u = 2^{-53} \approx 1.11 \cdot 10^{-16}$.

Theorem 1.4 (rounding error)

If $x \neq 0$ satisfies

$$\beta^{e_{\min}-1} \le |x| \le x_{\max} \qquad \text{(no underflow, no overflow)}, \tag{1.1}$$

then

$$\frac{|\,\mathrm{fl}(x) - x|}{|x|} \le u, \qquad \text{i.e.} \qquad \mathrm{fl}(x) = x\,(1+\delta),\ |\delta| \le u.$$

The standard model

IEEE: the four basic operations are *correctly rounded*, $a \circ b \;\mapsto\; \mathrm{fl}(a \circ b)$ for $a, b \in \mathbb{F}$.

Standard model of floating-point arithmetic

$$\mathrm{fl}(a \circ b) = (a \circ b)(1 + \delta), \qquad |\delta| \le u \tag{1.2}$$

whenever $a \circ b \neq 0$ satisfies (1.1) ($\circ \in \{+, -, \cdot, /\}$; $b \neq 0$ for $/$).
If $a \circ b = 0$, the result is exact.

Under these assumptions every single operation is accurate to a relative error $\le u$. The trouble starts when operations are combined.

Floating-point addition is not associative

```
(1 + 1e-16) - 1 = 0.0              1 + (1e-16 - 1) = 1.1102230246251565e-16
```

- spacing in $[1, 2)$ is 2^{-52}: $1 + 10^{-16}$ is rounded to 1
- spacing in $[\frac{1}{2}, 1)$ is 2^{-53}: $10^{-16} - 1$ is rounded to $-(1 - 2^{-53})$, result 2^{-53}
- exact value 10^{-16}: the first result has $100\,\%$ error, the second $11\,\%$ — with every operation correctly rounded

Cancellation

Operands with errors: $\tilde{a} = a(1+\delta_a)$, $\tilde{b} = b(1+\delta_b)$, $|\delta_a|, |\delta_b| \le u$, $a \neq b$. Even for an exact subtraction,

$$\frac{|(\tilde{a}-\tilde{b}) - (a-b)|}{|a-b|} \;\le\; u\,\frac{|a|+|b|}{|a-b|}. \tag{1.3}$$

Cancellation: leading digits agree and cancel; the trailing digits, which carry the errors, become leading.

Example 1.5: the quadratic equation

$x^2 + px + q = 0$, $p \neq 0$, $p^2/4 - q > 0$:

$$x_{\pm} = -\frac{p}{2} \pm \sqrt{\frac{p^2}{4} - q}$$

- $p^2 \gg |q|$: $\sqrt{p^2/4 - q} \approx |p|/2$, the root of smaller modulus is a difference of nearly equal numbers
- remedy: $x_1 = -\frac{p}{2} - \operatorname{sgn}(p)\sqrt{\frac{p^2}{4} - q}$ (same sign: no cancellation), $x_2 = q/x_1$ (Vieta)

Example 1.5: $p = 10^8$, $q = 1$

Exact: $x_1 = -10^8 + 10^{-8} + \ldots$, $x_2 = -10^{-8} - 10^{-24} - \ldots$
"smaller" and "larger" in the output refer to the modulus.

```
direct formula: smaller root = -7.4505805969e-09   larger root = -1.0000000000e+08
stable        : smaller root = -1.0000000000e-08   larger root = -1.0000000000e+08
```

- direct formula: $25\,\%$ error — near $5 \cdot 10^7$ the doubles are $2^{-27} \approx 7.45 \cdot 10^{-9}$ apart
- stable procedure: correct to all displayed digits

Rule

Avoid subtracting nearly equal quantities that carry rounding errors — rewrite if you can.

Conditioning of a problem

Is the *problem* sensitive to perturbations of its data? (independent of any algorithm)

Idea: relative change of the output $\approx \kappa_{\mathrm{rel}}(x) \cdot$ relative change of the input.

Definition 1.6 (condition numbers)

Problem $f\colon \mathbb{R}^n \to \mathbb{R}^m$, input x, output $f(x)$:

$$\kappa_{\mathrm{abs}}(x) = \lim_{\varepsilon \to 0} \sup_{0 < \|\Delta x\| \le \varepsilon} \frac{\|f(x+\Delta x) - f(x)\|}{\|\Delta x\|},$$

$$\kappa_{\mathrm{rel}}(x) = \lim_{\varepsilon \to 0} \sup_{0 < \|\Delta x\| \le \varepsilon} \frac{\|f(x+\Delta x) - f(x)\| / \|f(x)\|}{\|\Delta x\| / \|x\|} \qquad (x \neq 0,\ f(x) \neq 0).$$

well-conditioned: $\kappa_{\mathrm{rel}}(x)$ moderate **ill-conditioned**: $\kappa_{\mathrm{rel}}(x)$ large

What counts: $\kappa_{\mathrm{rel}}(x)\times$ relative input error must be small compared with the accuracy you need.

Lemma 1.7

$f\colon \mathbb{R} \to \mathbb{R}$ differentiable at x (with $x \neq 0$, $f(x) \neq 0$):

$$\kappa_{\mathrm{abs}}(x) = |f'(x)|, \qquad \kappa_{\mathrm{rel}}(x) = \frac{|f'(x)|\,|x|}{|f(x)|}.$$

Proof: the difference quotient converges to $|f'(x)|$ (notes).

Remark 1.9. $\kappa_{\mathrm{rel}}(x)$ is a *worst-case amplification factor*: a small relative input error ε can produce an output error of about $\kappa_{\mathrm{rel}}(x)\,\varepsilon$ — and no algorithm that only sees the perturbed input can repair this.

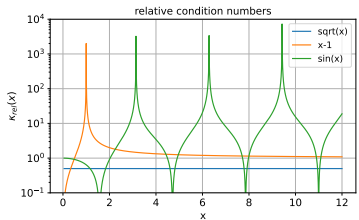
Example 1.8

(i) $f(x) = \sqrt{x}$, $x > 0$: $\kappa_{\text{rel}}(x) = \frac{1}{2}$

(ii) $f(x) = x - 1$: $\kappa_{\text{rel}}(x) = \dfrac{|x|}{|x-1|} \to \infty$ as $x \to 1$

(iii) $f(x) = e^x$: $\kappa_{\text{rel}}(x) = |x|$

(iv) $f(x) = \sin x$: $\kappa_{\text{rel}}(x) = |x \cot x|$, large near $k\pi$, $k \neq 0$ (but $\to 1$ as $x \to 0$)



From the problem to the algorithm

Well-conditioned problem $\Rightarrow$ accurate result?

Example 1.1: the problem is well-conditioned, $\kappa_{\text{rel}}(x) = \dfrac{2x+1}{x+1} \leq 2$ (Example 1.12) — yet the direct formula $f(x) = 1/x - 1/(x+1)$ loses k digits for $x = 10^k$.

So the *algorithm* matters too. What is a good algorithm?

Stability of an algorithm

Algorithm $\tilde{f}$ for f: finitely many operations, each subject to (1.2).

Definition 1.10

With a constant C of moderate size (independent of x and u), $\tilde{f}$ is

(i) **forward stable** if $\dfrac{\|\tilde{f}(x) - f(x)\|}{\|f(x)\|} \le C\,\big(1 + \kappa_{\mathrm{rel}}(x)\big)\,u$ for all admissible x;

(ii) **backward stable** if for every admissible x there is $\tilde{x}$ with

$$\tilde{f}(x) = f(\tilde{x}) \qquad \text{and} \qquad \frac{\|\tilde{x} - x\|}{\|x\|} \le C\,u.$$

Admissible inputs: $x \neq 0$, $f(x) \neq 0$.

- forward stable: as accurate as the conditioning permits
- backward stable: *exact* result for slightly perturbed data

Backward stability is the stronger property

Remark 1.11. Fix x with $\kappa_{\mathrm{rel}}(x) < \infty$. If $\tilde{f}(x) = f(\tilde{x})$ with $\|\tilde{x} - x\|/\|x\| \le Cu$, then

$$\frac{\|\tilde{f}(x) - f(x)\|}{\|f(x)\|} = \frac{\|f(\tilde{x}) - f(x)\|}{\|f(x)\|} \le \big(\kappa_{\mathrm{rel}}(x) + o(1)\big)\, C\, u \qquad \text{as } u \to 0.$$

The rule to keep in mind

$$\text{error of the result} \lesssim \underbrace{\text{conditioning of the problem}}_{\text{given by the problem}} \times \underbrace{\text{backward error of the algorithm}}_{\text{choose a good algorithm}}$$

Vector norms

Definition 1.14

A **norm** on $\mathbb{R}^n$: $\|\cdot\|:\mathbb{R}^n\to[0,\infty)$ with

$$\|x\|=0 \iff x=0, \qquad \|\alpha x\|=|\alpha|\,\|x\|, \qquad \|x+y\|\le\|x\|+\|y\|.$$

$$\|x\|_p=\Big(\sum_{j=1}^{n}|x_j|^p\Big)^{1/p}\ (1\le p<\infty), \qquad \|x\|_\infty=\max_j|x_j|$$

In particular $\|x\|_1=\sum_j|x_j|$, $\|x\|_2=\sqrt{x^\top x}$.

Proposition 1.15. $\|\cdot\|_1$, $\|\cdot\|_2$, $\|\cdot\|_\infty$ are norms. (proof in the notes; 2-norm via Cauchy–Schwarz)

Equivalence of the 1-, 2- and ∞-norm

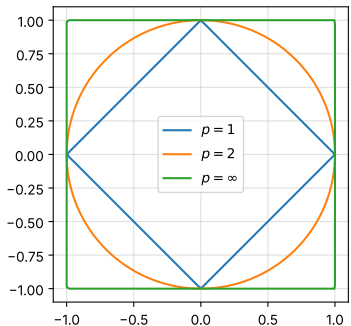
Theorem 1.16

For all $x \in \mathbb{R}^n$,

$$\|x\|_\infty \le \|x\|_2 \le \sqrt{n}\,\|x\|_\infty, \qquad \|x\|_\infty \le \|x\|_1 \le n\,\|x\|_\infty,$$
$$\|x\|_2 \le \|x\|_1 \le \sqrt{n}\,\|x\|_2.$$

Sharp: $x = e_1$ on the left, $x = (1, \dots, 1)^\top$ on the right.

Proof in the notes (elementary).



unit spheres $\{x \in \mathbb{R}^2 : \|x\|_p = 1\}$

Equivalence of norms

Theorem 1.17

For any two norms $\|\cdot\|_a$, $\|\cdot\|_b$ on $\mathbb{R}^n$ there are $0 < c \le C$ with

$$c\,\|x\|_a \le \|x\|_b \le C\,\|x\|_a \qquad \text{for all } x \in \mathbb{R}^n.$$

Proof in the notes: compactness of the unit sphere.

Remark 1.18. Convergence does not depend on the norm — the constants in error estimates do, and they may depend on n.

Frobenius norm $\|A\|_F = \left(\sum_{i,j} a_{ij}^2\right)^{1/2}$. More useful: how much can A stretch a vector?

Definition 1.19 (induced norm)

Vector norms on $\mathbb{R}^n$ and on $\mathbb{R}^m$, both written $\|\cdot\|$; for $A \in \mathbb{R}^{m\times n}$

$$\|A\| = \sup_{x\neq 0} \frac{\|Ax\|}{\|x\|} = \max_{\|x\|=1} \|Ax\|.$$

$$\|Ax\| \le \|A\|\,\|x\| \quad \text{for all } x, \tag{1.4}$$

and $\|A\|$ is the smallest such constant.

Properties of induced norms

Proposition 1.20

(i) the induced norm is a norm on $\mathbb{R}^{m\times n}$

(ii) $\|I\| = 1$

(iii) $\|AB\| \le \|A\|\,\|B\|$ (submultiplicativity)

Proof: (iii) apply (1.4) twice.

Remark 1.21. $\|\cdot\|_F$ is submultiplicative too, but not induced for $n \ge 2$: $\|I\|_F = \sqrt{n} \ne 1$.

Explicit formulas

Theorem 1.22

For $A \in \mathbb{R}^{m \times n}$,

$$\|A\|_1 = \max_k \sum_{i=1}^{m} |a_{ik}| \qquad \text{maximal absolute column sum}$$

$$\|A\|_\infty = \max_i \sum_{k=1}^{n} |a_{ik}| \qquad \text{maximal absolute row sum}$$

$$\|A\|_2 = \sqrt{\lambda_{\max}(A^\top A)} \qquad \text{square root of the largest eigenvalue of } A^\top A$$

$\|A\|_1$, $\|A\|_\infty$: $O(mn)$ flops. $\|A\|_2$: eigenvalue problem (Chapter 5) or SVD (Chapter 4).
Proofs for the 1- and the 2-norm in the notes; the ∞-norm on the blackboard.

The 2-norm and the spectral radius

$B \in \mathbb{R}^{n\times n}$: $\rho(B) = \max\{|\lambda| : \lambda \in \mathbb{C} \text{ eigenvalue of } B\}$ · Q orthogonal: $Q^\top Q = I$

Proposition 1.23

(i) $\|A^\top\|_2 = \|A\|_2$

(ii) $\|QA\|_2 = \|A\|_2 = \|AP\|_2$ for orthogonal Q, P

(iii) A symmetric $\Rightarrow \|A\|_2 = \rho(A)$; in general $\|A\|_2 = \sqrt{\rho(A^\top A)}$

Proposition 1.24

$\rho(A) \le \|A\|$ for every $A \in \mathbb{R}^{n\times n}$ and every induced norm.

Proofs in the notes.

Example 1.25

$A = \begin{pmatrix} 1 & 2 \\ -3 & 1 \end{pmatrix}$: $\quad A^{\top}A = \begin{pmatrix} 10 & -1 \\ -1 & 5 \end{pmatrix}$, eigenvalues $\dfrac{15 \pm \sqrt{29}}{2}$; $\quad$ eigenvalues of A: $1 \pm i\sqrt{6}$

```
||A||_1   = 4.0  (max column sum)
||A||_inf = 4.0  (max row sum)
||A||_2   = 3.192582403567252  = sqrt(lambda_max(A^T A))
||A||_F   = 3.872983346207417
spectral radius rho(A) = 2.6457513110645907  <= every induced norm
```

The computer confirms the hand computation; $\|A\|_2 = \sqrt{(15+\sqrt{29})/2}$ and $\rho(A) = \sqrt{7}$ are not visible from the entries.

Summary

- $\mathbb{F}$ is finite and not equally spaced; standard model (1.2): without under-/overflow, every operation is exact up to a relative error $u \approx 1.1 \cdot 10^{-16}$
- **cancellation**: subtracting nearly equal numbers amplifies existing errors by up to $(|a| + |b|)/|a - b|$
- **conditioning** is a property of the problem, **stability** a property of the algorithm; error $\lesssim$ conditioning $\times$ backward error
- **norms**: all equivalent on $\mathbb{R}^n$; $\|Ax\| \le \|A\|\|x\|$, $\|AB\| \le \|A\|\|B\|$, $\rho(A) \le \|A\|$; $\|A\|_1$ max. absolute column sum, $\|A\|_\infty$ max. absolute row sum, $\|A\|_2 = \sqrt{\lambda_{\max}(A^\top A)}$