Introduction to Numerical Optimization

Oswin Krause, NO, 2022

KØBENHAVNS UNIVERSITET

Course Organisation

This is a flipped-classroom course. Reading material must be read before course starts Monday.

- On Mondays (Lecture hall):
  - Short lectures most of time,
  - General question sessions,
  - Weekly creation of random work groups,
  - Start on group study work / study questions.
- On Wednesdays (exercise classes, hopefully)
  - Group work on study group questions,
- Weekly assignment handed in on Fridays, 22:00 latest.

The Team

- Oswin Krause, course responsible, co-teacher.
- François Lauze, co-teacher.
- Niels-Christian Borbjerg, TA, Live
- Ziheng Liu, TA, Live
- Robin Bruneau, TA, Correction
- Nikolin Prenga, TA, Correction

New Service this year: Nikolin will help you on Fridays 13:00-16:00 in the DIKU Kanteen!

Goal of the course

- Students should be able to
  1. Implement an optimization algorithm
  2. Evaluate correctness of an implementation
  3. Benchmark algorithms against each other
  4. Understand the ideas behind the algorithms
  5. Follow theoretical derivations
- These all involve making decisions
  - Meaningful decisions require uncertainty.

  $\rightarrow$ There are lots of uncertainties in this course.
  - We help you to evaluate your decisions.

Role of Lectures

- This course used to be lecture free
- Lectures won't cover everything
- Not lectured $\neq$ unimportant
- Teachers assumption: You have read the text.
- Study Group Questions:
    - Part of lecture / reading material
    - The answer to some of them can save an hour of working on an assignment.
    - At least quickly check whether you know how to answer them!

Role of Exercise Classes

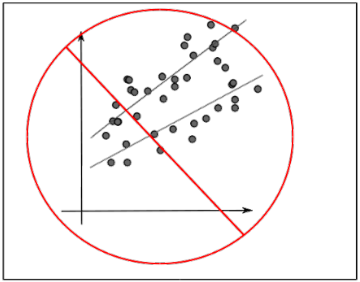
- Group work on assignments.
- A group that has not read the text is inefficient.
- Teachers / TAs
  - Walk around. They help you, when you get stuck.
  - We can not answer questions you have not asked.
  - Everyone struggle(s/d) with the material. There are no dumb questions.
- Random Groups
  - Are annoying, but very efficient
  - Explaining stuff to other students improves learning outcome
  - Don't get stuck in a role "theory"/"practice"

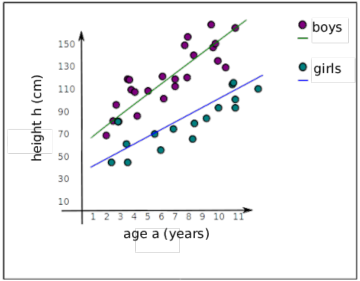
Assignment – Formalities

- Theoretical and Programming exercises.
- All programming exercises are mandatory, unless mentioned explicitly.
- Mandatory theory part: Choose which theoretical exercises you want to hand in.
- No code in report! This is not a programming learning course!
- Report is limited to 5 pages. LaTeX is much nicer than Word!
- Format your report: title, brief introduction (2 or 3 lines), content, short conclusion (2 or 3 lines).
    - Introduction: Structure of report/goal setting.
    - Conclusion: something interesting you have learned? What is important?
- Reports are individual (but can share code, figures, derivations with the group)

Figures

- Always comment and label your graphs and figures, tables...
- If your peers can't understand your figure, we can't either.
- Remember: a picture is worth a thousand words, and you are limited to 5 pages!

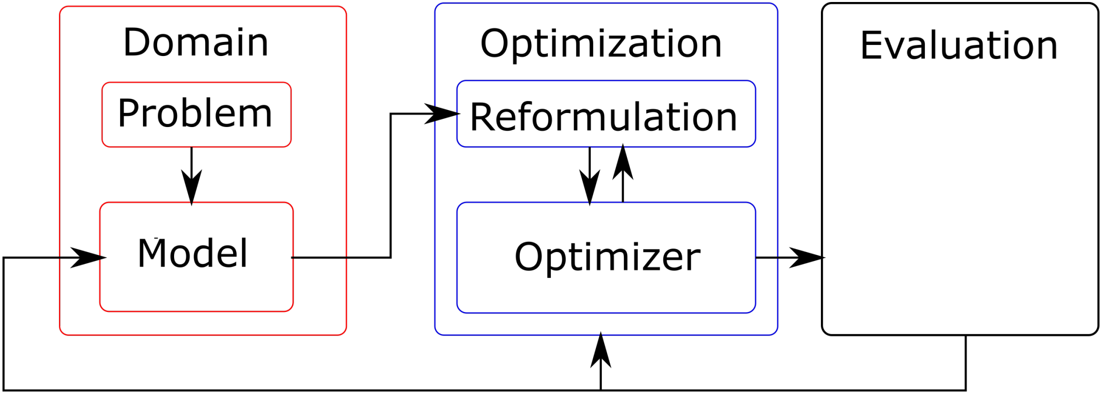




Optimisation

- To optimise is to:
- Search for the *best*, the *cheapest*, *the lowest*...
- ...that needs to fulfill some additional constraints
- In Mathematics, Physics, Economics, Chemistry, Biology, Computer Science...
- Many laws and classical results are formulated as optimisation, e.g., Newton second law!
- Ubiquitous in almost every discipline!

Role of Optimization



Setup in this course

- We are given an *objective function* $f : \mathbb{R}^n \to \mathbb{R}$.
- A *feasible region*, $\mathcal{C} \subseteq \mathbb{R}^n$
- Task: Search for a point $x^* \in \mathcal{C}$ where f is minimal

$$x^* = \arg\min_{x \in \mathcal{C}} f(x)$$

- Our Task: develop an optimizer that can find x^*
- Requires: Intuition, math skills and good numerical implementation

Refresher on Linear Algebra

Norms

Norms measure the length of a vector in different metrics. Let $x \in \mathbb{R}^n$, we will use the following norms::

$$\|x\|_1 = \sum_{i=1}^{n} |x_i|$$
$$\|x\|_2 = \sqrt{x^T x}$$
$$\|x\|_\infty = \max_{i=1}^{n} |x_i|$$

By default, we treat $\|x\| = \|x\|_2$

Orthogonal matrices

Symmetric matrix $Q \in \mathbb{R}^{n \times n}$. Q is called orthogonal if

$$Q^T Q = QQ^T = I_n, \quad (I_n \text{ is the identity matrix})$$

Properties:

- $Q^{-1} = Q^T$
- $\|x\|_2 = \|Qx\|_2$
  - Task (5 Min): Show this.

Symmetric Eigenvalue Decomposition (sum form)

Symmetric matrix $A \in \mathbb{R}^{n\times n}$, $A = A^T$. It holds:

$$A = \sum_{i=1}^{n} \lambda_i q_i q_i^T$$

- $q_i \in \mathbb{R}^n, \|q_i\| = 1, q_i^T q_j = 0,$ for $i \neq j$
- It holds $Aq_i = \lambda_i q_i$
- q_i are called Eigenvectors
- λ_i are called Eigenvalues
  - We assume them to be ordered
  - $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_n$
  - $\sigma_i(A) = \lambda_i$ is the function returning the ith largest eigenvalue of A

Symmetric Eigenvalue decomposition (Matrix form)

Symmetric matrix $H \in \mathbb{R}^{n \times n}$, $H = H^T$ (For example a hessian matrix):

$$H = \sum_{i=1}^{n} \lambda_i q_i q_i^T = Q \Lambda Q^T$$

- $Q \in \mathbb{R}^{n \times n}$, q_i is ith column of Q
- $\Lambda \in \mathbb{R}^{n \times n}$ diagonal matrix with $\Lambda_{ii} = \lambda_i$
- Q orthogonal matrix

Positive definite matrix

Symmetric matrix $A \in \mathbb{R}^{n \times n}$. If:

$$x^T A x \geq 0, \; \forall x \in \mathbb{R}^n$$

Then we call A positive semi-definite(PSD). If

$$x^T A x > 0, \; \forall x \in \mathbb{R}^n \setminus \{0\}$$

Then, A is positive definite (PD).

- Task (8 min):
  Show that a symmetric matrix A is PSD/(PD) if all eigenvalues are non-negative (positive)

Matrix Norms

Every norm induces a matrix norm. Let $A \in \mathbb{R}^{n \times m}$ be a matrix and $x \in \mathbb{R}^m$ We define for any vector-norm $\|\cdot\|$:

$$\|A\| = \max_{\|x\| \leq 1} \frac{\|Ax\|}{\|x\|}$$

For $\|\cdot\|_2$ holds:

$$\|A\|_2 = \sqrt{\sigma_1(AA^T)}$$

If A is symmetric PSD:

$$\|A\|_2 = \sigma_1(A)$$

Conditioning

Conditioning

- All optimization algorithms make small errors
- Instead of x, we obtain perturbed solution $\tilde{x} = x + \epsilon$
- How sensitive is a function $f : \mathbb{R}^n \to \mathbb{R}^m$ to small deviations $x \to \tilde{x}$?
- This is measured by the *Condition number*:

$$\operatorname{cond}(f, x) = \lim_{\delta \to 0} \max_{\substack{\epsilon \\ \|\epsilon\| \leq \delta}} \frac{\overbrace{\dfrac{\|f(x+\epsilon) - f(x)\|}{\|f(x)\|}}^{\text{Relative error in function value}}}{\underbrace{\dfrac{\|\epsilon\|}{\|x\|}}_{\text{Relative error in argument}}}$$

Example: Linear function

- Calculate conditioning of $f(x) = Ax$
- A invertible
- The condition number is:

$$\mathrm{cond}(f,x) = \lim_{\delta\to 0} \max_{\substack{\epsilon\\ \|\epsilon\|\leq\delta}} \frac{\frac{\|f(x+\epsilon)-f(x)\|}{\|f(x)\|}}{\frac{\|\epsilon\|}{\|x\|}}$$

$$\mathrm{cond}(f,x) = \lim_{\delta\to 0} \max_{\substack{\epsilon\\ \|\epsilon\|\leq\delta}} \frac{\|f(x+\epsilon)-f(x)\|}{\|\epsilon\|}\frac{\|x\|}{\|f(x)\|}$$

$$\mathrm{cond}(f,x) = \lim_{\delta\to 0} \max_{\substack{\epsilon\\ \|\epsilon\|\leq\delta}} \underbrace{\frac{\|A\epsilon\|}{\|\epsilon\|}}_{\text{Constant in} \|\epsilon\|} \frac{\|x\|}{\|Ax\|}$$

$$\mathrm{cond}(f,x) = \underbrace{\max_{\substack{\epsilon\\ \|\epsilon\|\leq 1}} \frac{\|A\epsilon\|}{\|\epsilon\|}} \frac{\|x\|}{\|Ax\|}$$

Condition number $\kappa(A)$

We found for $f(x) = Ax$

$$\mathrm{cond}(f, x) = \|A\| \frac{\|x\|}{\|Ax\|}$$

We define as condition number of A the worst case conditioning over all x

$$\kappa(A) = \max_x \mathrm{cond}(f, x) = \|A\| \max_x \frac{\|x\|}{\|Ax\|}$$
$$= \|A\| \|A^{-1}\| = \frac{\sigma_1(A)}{\sigma_n(A)}$$

Deriving the last step is a group question!

Importance of Condition number $\kappa(x)$

- Assume we want to minimize
$$f(x) = \frac{1}{2}x^T Ax \ ,$$
$A \in \mathbb{R}^{n\times n}$ PD matrix
- We need to find x such, that $\nabla f(x) = 0$
- $\nabla f(x) = Ax$

$\rightarrow$ Condition number of gradient

$$\mathrm{cond}(\nabla f, x) \leq \max_x \mathrm{cond}(\nabla f, x) = \kappa(A)$$

$\kappa(A)$ tells us, how sensitive the gradient is to small perturbations to its argument.

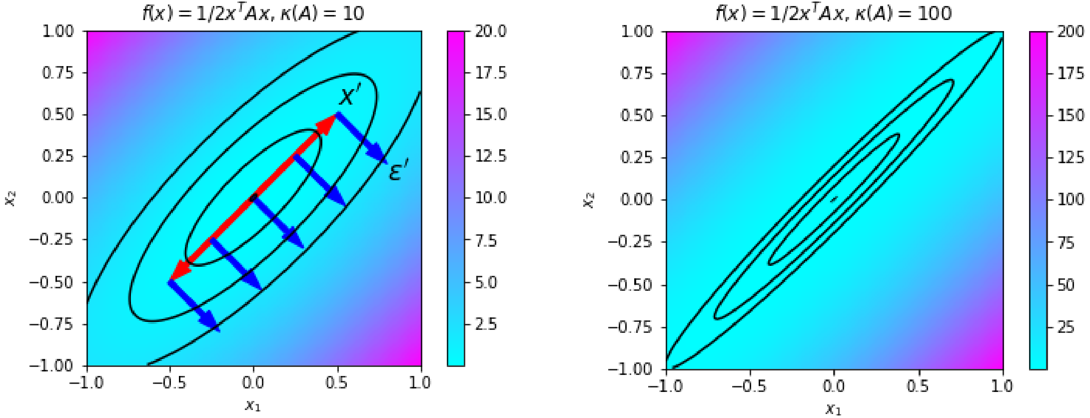
Understanding the Condition Number $\kappa(x)$

We can interpret the terms of the condition calculation as follows

$$\kappa(A) = \max_x \operatorname{cond}(\nabla f, x) = \underbrace{\max_{\substack{x \\ ||x||=1}} \frac{||x||}{||Ax||}}_{\substack{\text{Smallest gradient} \\ \text{direction } x'}} \underbrace{\max_{\substack{\epsilon \\ ||\epsilon||=1}} \frac{||A\epsilon||}{||\epsilon||}}_{\substack{\text{Worst perturbation} \\ \text{direction } \epsilon'}}$$

Let us visualize this

Understanding the Condition Number $\kappa(x)$



At high $\kappa(A)$ small perturbations can dominate function-value and gradient.

Refresher on Calculus

Big O-notation I

Let $f : \mathbb{R} \to \mathbb{R}$, $g : \mathbb{R} \to \mathbb{R}_{>0}$

We write

$$f(x) = O(g(x)), \text{ as } x \to \infty$$

if there exists $M > 0$ and x_0 such, that

$$|f(x)| \leq Mg(x), \forall x > x_0$$

Intuition: f grows asymptotically at most as quickly as g.

Big O-notation II

Let $f : \mathbb{R} \to \mathbb{R}$, $g : \mathbb{R} \to \mathbb{R}_{>0}$

We write

$$f(x) = O(g(x)), \text{ as } x \to a$$

if

$$\limsup_{x \to a} \frac{|f(x)|}{g(x)} < \infty$$

f grows asymptotically at most as quickly as g as x approaches a.

Little O-notation

Let $f : \mathbb{R} \to \mathbb{R}$, $g : \mathbb{R} \to \mathbb{R}_{>0}$.
We write

$$f(x) = o(g(x)), \text{ as } x \to \infty$$

if for each positive ϵ, $|f(x)| \leq \epsilon g(x)$ for all x large enough.

We write

$$f(x) = o(g(x)), \text{ as } x \to a$$

if $g(x) \neq 0$ around a and

$$\lim_{x \to a} \frac{|f(x)|}{g(x)} = 0$$

Intuition: f is dominated by g asymptotically.

Quick exercise

Are the following statements true?

- $x^2 = O(|x^3|)$ as $x \to \infty$
- $x^2 = O(|x^3|)$ as $x \to 0$
- $x^2 = o(|x|)$ as $x \to 0$

Discuss 5 min

Taylor's theorem

The most central theorem for this course.

- 1D. $f : \mathbb{R} \to \mathbb{R}$, $(k+1)$-times continuously differentiable: Then there is a $c \in (x, x+h)$

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \cdots + \frac{h^k}{k!}f^{(k)}(x) + \underbrace{\frac{h^{k+1}}{(k+1)!}f^{(k+1)}(c)}_{o(h^k),\ \text{as}\ h \to 0}$$

- In several variables $f : U \in \mathbb{R}^n \to \mathbb{R}$: usually second order is enough

$$f(x+h) = f(x) + \nabla f(x)^T h + \frac{1}{2}h^T \nabla^2 f(x) h + o(\|h\|^2)$$

- Intuition: for small values $\|h\|$, $g(h) = f(x+h) \approx$ linear/quadratic/... function

Convergence of sequences

Sequence $(x_k)_{k \in \mathcal{N}}$ converges to $x \in \mathbb{R}^n$ if

$$\forall \epsilon > 0, \exists K > 0, \quad \forall k \geq K, \quad \|x_k - x\| < \epsilon$$

Given a ball centred at x, with radius ϵ, starting from a given k, all the members of the sequence will be in that ball.

$(x_k)_{k \in \mathcal{N}}$ is *Cauchy* if

$$\forall \epsilon > 0, \exists K > 0, \quad \forall k, k' \geq K, \quad \|x_k - x_{k'}\| < \epsilon$$

Given a radius ϵ, starting from a given k, all the members of the sequence will be in ball centred at any of the x_k with radius ϵ.

$$\text{Convergent} \iff \text{Cauchy}.$$

But no need to know the limit!

Convergence speed

When designing a minimising sequence, we want to know how fast it converges – from seconds or less to days of calculations!

- Linear convergence:

$$\frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} \le r \in (0, 1)$$

  One at least gains some (fraction of) decimals at each iteration.

- Superlinear convergence:

$$\frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} \to 0$$

  One gains more and more (fractions of) decimals in the process.

Convergence Ferrari

- Quadratic convergence.

$$\frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|^2} \leq M, \quad M \geq 0$$

  No special requirement on M except of being positive. Still the number of decimals roughly "double" with each iteration, i.e., exponential increase of precision. Why?
- Faster? Could have superquadratic etc... but in practice, never required. And there is already a complexity price for quadratic convergence.
- Some complicated objective can not even be optimized in linear rate!