Sep 8, 2016

M2 optimization

Fall 2016
Ecole Polytechnique

by Grégoire Allaire
Antonin Chambolle
Thomas Wick

Exercise (TD) today:
Ampli Curie
13.30 - 15.30

Web: www.cmap.polytechnique.fr/~wick
↳ Teaching → M2 → Infos
→ *.pdf with outline of this class
→ Weekly exercises

plus more literature!

Contents
1. Examples and basic algorithms in $\mathbb{R}^n$
2. Convex analysis, existence of minimizers
3. Optimality conditions
4. Algorithms

Sheet of paper to gather information / Exam in TD 2-6 (30 min.)

Topics today
1. Example
2. Problem statement for unconstrained optimization
3. Algorithms
↳ Gradient descent, conjugate gradient, Newton)

(1) Example

→ p. 16

(2) Problem statement, unconstrained optimization, in $\mathbb{R}^n$

$n \hat{=}$ dimension

Let $X \subset \mathbb{R}^n$ be an admissible set

and $f: X \to \mathbb{R}$ the so-called target or cost functional

Task:

$$\min_{x \in \mathbb{R}^n} f(x) \tag{2.1}$$

where $x \in \mathbb{R}^n$ $\left[x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \right]$

Remark:

$\max \tilde{f}(x)$ is equivalent to $f := -\tilde{f}$

Further notation for later:

$\|\cdot\| := \|\cdot\|_2$ Euclidian norm

with $$\|x\| = \sqrt{x^T x} = \left(\sum_{i=1}^{n} x_i^2 \right)^{1/2}, \quad x \in \mathbb{R}^n$$

Gradient $\nabla f(x) = (\partial_{x_1} f, \dots, \partial_{x_n} f)^T \in \mathbb{R}^n$

Hesse $H = \nabla^2 f(x) = \left(\partial_i \partial_j f \right)_{i,j} \in \mathbb{R}^{n \times n}$ if $f \in C^2 \Rightarrow H = H^T$

Local minimum

A point $\bar{x} \in \mathbb{R}^n$ is called

1. local minimum of (2.1) if $\bar{x} \in X$ and $\varepsilon > 0$ exists with $f(x) \geq f(\bar{x}) \; \forall x \in X \cap B$, where $B := \{x \in \mathbb{R}^n : \|x - \bar{x}\| < \varepsilon\}$
2. Similar for maximum

Stationary points

A stationary point $\bar{x}$ of f, which is neither a minimum nor a maximum is called a saddle point

↳ Optimality conditions (curvature) to determine classification

Remark

Convexity plays an outstanding role since then local minima are also global minima

(3.) Algorithms

Computing the minimum in practice

Key question: How to obtain minima in practice?

→ analytical solution only possible in idealized, academic test cases.

Idea: Choose starting point x_0 and generate a sequence of iterates

$$\{x_k\}_{k=0}^{\infty}$$

Question: How to generate $\{x_k\}$ and how to move from x_k to x_{k+1}?

There are <u>two</u> fundamental strategies:

1. Line search (this class!):

   Given x_k, and given a search direction p_k:

   Find a step length α such that

   $$\min_{\alpha>0} f(x_k + \alpha p_k)$$

2. Trust region: Gather information about f around current iterate x_k and construct model function m_k such that at x_k: $m_k \approx f$

   Seek minimizer of m_k: Find p

   such that

   $$\min_{p} m_k(x_k + p)$$

   where $x_k + p$ inside the trust region. Usually $\|p\|_2 \leq \Delta$, $\Delta > 0$ trust region radius

and

$$m_k(x_k + p) = f(x_k) + p^T \nabla f(x_k) + \frac{1}{2} p^T B_k p$$

Differences between line-search and trust region:

direction and distance

Line-search: Given direction, find step length α

Trust-region: Choose maximal distance, seek direction p

Overview algorithms (NW, p. 20 (42))

a) Gradient / steepest gradient descent

$$p = p_k = -\nabla f(x_k), \quad \text{or} \quad p_k = \frac{-\nabla f(x_k)}{\|\nabla f(x_k)\|}$$

Properties: only needs 1st derivative,
but is only 1st order and slow

b) Conjugate gradient

$$p = p_k = -\nabla f(x_k) + \beta_{k-1} p_{k-1}$$

Properties:
- β_{k-1} is chosen such that p_k and p_{k-1} are A-conjugate (later more)
- only needs 1st derivative
- still slow but converges in at most n steps to $\bar{x}$

c) Newton:

$$p = p_k = (-\nabla^2 f(x_k))^{-1} \nabla f(x_k)$$

Properties:

- 2nd derivative (Hessian $\nabla^2 f(x_k)$)
- fast (local) convergence (2nd order)
- "natural" step length associated $\alpha = 1$
- problem when $\nabla^2 f(x_k)$ is not positive definite
- Comp. of Hessian often cumbersome
  ↳ Quasi-Newton methods

In more detail

3.1) Gradient descent / Steepest gradient descent

Let $f \in C^1$, $f: \mathbb{R}^n \to \mathbb{R}$

Algorithm 1 (General descent)

1. Choose a starting point $x_0 \in \mathbb{R}^n$

For $k = 0, 1, 2$

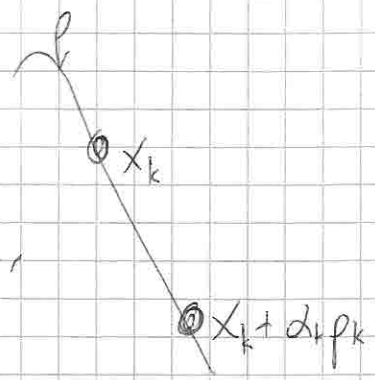
2. Check if x_k is a stationary point

3. Compute a descent direction $p_k \in \mathbb{R}^n$ with

$$\nabla f(x_k)^T p_k < 0$$

4. Determine a step length $\alpha_k > 0$ such that

$$f(x_k + \alpha_k p_k) < f(x_k)$$



and the descent is sufficiently large,

i.e.,

$$f(x_k) - f(x_k + \alpha_k p_k) \gg 0$$

5. Set $x_{k+1} = x_k + \alpha_k p_k$

Crucial idea: descent directions

The vector $p = p_k \in \mathbb{R}^n \setminus \{0\}$ is a descent direction in x if the slope of f in direction p is negative

The slope is given by

$$\lim_{t \to 0^+} \frac{f(x + tp) - f(x)}{\|tp\|} = \frac{\nabla f(x)^T p}{\|p\|}$$

It holds:

Definition

The vector $p \in \mathbb{R}^n \setminus \{0\}$ is a descent direction of $f \in C^1$, $f: \mathbb{R}^n \to \mathbb{R}$ in x if

$$\nabla f(x)^T p < 0$$

3.1.1) Steepest gradient descent

It is obvious that the descent direction should be the steepest descent:

$$p_k = -h \nabla f(x_k), \quad h > 0 \quad \text{(special case } h = 1\text{)}$$

So far, we have computed the search direction. It remains to determine the step length α.

3.1.2) Armijo step length procedure

- Easy to implement
- Basic feature in many software packages

Let $\beta \in (0,1)$ (e.g. $\beta = 0.5$) and $\gamma \in (0,1)$ be fixed parameters.

Go into Step 4 of Algo 1

$$f(x_k + \alpha_k p_k) - f(x_k) < 0$$

$$\Rightarrow f(x_k + \alpha_k p_k) - f(x_k) \leq \alpha_k \gamma \underbrace{\nabla f(x_k) p_k}_{<0} < 0 \tag{8.1}$$

Determine the largest $\alpha_k \in \{1, \beta, \beta^2, \dots\}$
(backtracking line search)

such that (8.1) holds true.

The larger $|\alpha_k \gamma \nabla f(x_k) p_k|$, the better because then a huge decrease can be achieved:

$$f(x_{k+1}) \ll f(x_k)$$

Remark

For instance the Powell-Wolfe step rule makes sure in each step that sufficient decrease of $f(x_k)$ is achieved.

However, we have to ensure that this step is always admissable:

Theorem

Let $U \subset \mathbb{R}^n$ be open, $f \in C^1(U)$ and let $\gamma \in (0,1)$ be given. Let $x \in U$ and $p \in \mathbb{R}^n$ a descent direction of f in x, then it exists an $\bar{\alpha} > 0$ with

$$f(x + \alpha p) - f(x) \leq \alpha \gamma \nabla f(x)^T p \quad \forall \alpha \in [0, \bar{\alpha}]$$

Proof.

E.g. Ulbrich, p. 21

3.1.3 Convergence of steepest gradient

Algorithm 2 (Steepest gradient descent)

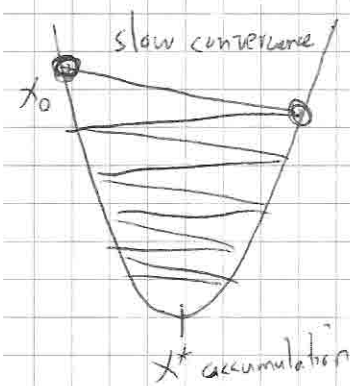
1. Set $\beta \in (0,1)$, $\gamma \in (0,1)$ and a starting point $x_0 \in \mathbb{R}^n$

For $k = 0, 1, 2, \ldots$

2. If $\nabla f(x_k) = 0$ or $\|\nabla f(x_k)\| < TOL \rightarrow$ Stop
3. Set $p_k = -\nabla f(x_k)$
4. Determine step length $\alpha_k > 0$ with Armijo rule
5. Set $x_{k+1} = x_k + \alpha_k p_k$

Theorem (convergence steepest gradient descent)

Let $f \in C^1$, Algo 2 terminates in a finite number of steps with the stationary point $\bar{x} := x_k$ or an infinite sequence $(x_k)_{k \in \mathbb{N}}$ is obtained with the following properties:



1. $\forall k$, we have $f(x_{k+1}) < f(x_k)$
2. Each accumulation point of $(x_k)_{k \in \mathbb{N}}$ is a stationary point of f

Remark 1

The difference between steepest gradient descent and conjugate gradient is that the latter one terminates in at most n steps.

Remark 2

The convergence is rather slow, with rate 1, and therefore Newton methods are more attractive since a "better" search direction p_k is obtained.

3.2) Conjugate gradient

→ Still rule ↑, but faster than steepest descent

→ Strong relationship with solving linear systems $Ax = b$

Definition (A-conjugacy)

Let A be a symmetric, positive definite matrix.

The set $\{p_0, \dots, p_k\}$ is called A-conjugate if

$$p_i^T A p_j = 0 \quad \forall i \neq j$$

Theorem

Let $\{p_0, \dots, p_m\}$ A-conjugate, x_0 the starting value and the iteration given by

$$x_{k+1} = x_k + \alpha_k p_k \,, \quad \alpha_k = - \frac{\nabla f(x_k)^T p_k}{p_k^T A p_k}$$

Then: For $k = 0, \dots, m$

- $f(x_{k+1}) = \min f\left(x_k + \sum_{i=0}^{k} w_i p_i\right)$ with arbitrary $w_0, \dots, w_k$
- $\nabla f(x_{k+1})^T p_i = 0 \quad \forall i = 0, \dots, k$

Corollary

Let $f(x)$ be quadratic, i.e. $f(x) = \frac{1}{2} x^T A x - b^T x$ and A s.p.d. and

$$x_{k+1} = x_k + \alpha_k p_k$$

where $\{p_0, \dots, p_k\}$ are A-conjugate and α_k is obtained through exact line search. Then

$$x^* = A^{-1} b \quad (\text{equ the minimum of } f(x))$$

is found in at most n steps

Remark

This theorem makes the difference between CG and steepest gradient

Question: How to find and construct the conjugate directions p_k?

Answer: E.g. Gram-Schmidt

Lemma

Let $\{p_0, \dots, p_{k-1}\}$ A-conjugate vectors. Then:

$$p_k = -\nabla f(x_k) + \sum_{i=0}^{k-1} \frac{\langle \nabla f(x_k), p_i \rangle_A}{\langle p_i, p_i \rangle_A} p_i \tag{12.1}$$

$\uparrow \langle x, y \rangle_A = x^T A y$

is conjugate to

$$\{p_0, \dots, p_{k-1}\}$$

Remark

It holds $\langle \nabla f(x_k), p_j \rangle_A = 0 \quad \forall j = 0, \dots, k-2$

Then (12.1) yields:

$$p_k = -\nabla f(x_k) + \beta_{k-1} p_{k-1}, \quad \beta_{k-1} = \frac{\langle \nabla f(x_k), p_{k-1} \rangle_A}{\langle p_{k-1}, p_{k-1} \rangle_A}$$

Theorem

Let $f(x)$ be quadratic, $f \in C^2$ and A s.p.d. and all further assumptions.

Then:

1. It exists $m \leq n$ such that for

$$k < m \leq n$$

the iterations

$$x_{k+1} = x_k + \alpha_k p_k$$

with p_k from the previous lemma are well-defined and

$$\nabla f(x_m) = 0$$

but $\nabla f(x_k) \neq 0 \quad \forall k < m \leq n$

2. Fletcher-Reeves

$$p_k = -\nabla f(x_k) + \beta_{k-1} p_{k-1}, \quad \beta_{k-1} = \frac{\|\nabla f(x_k)\|_2^2}{\|\nabla f(x_{k-1})\|_2^2}$$

3. $\{p_0, \dots, p_k\}$ is A-conjugate

4. $\{\nabla f(x_0), \dots, \nabla f(x_k)\}$ are pair-wise orthogonal

5. $p_k^T \nabla f(x_k) = -\|\nabla f(x_k)\|_2^2 < 0$, i.e., p_k is a descent direction

6. $\operatorname{span}(p_0, \dots, p_k) = \operatorname{span}(\nabla f(x_0), \dots, \nabla f(x_k))$

$$= \operatorname{span}(\nabla f(x_0), A \nabla f(x_0), \dots, A^k \nabla f(x_0))$$

3.3 Newton

→ "endless" applications (cf. NW, Deuflhard 2011)
etc. etc.

→ allows for fast (local) convergence

Let $f \in C^2$. Consider again

$$\min_x f(x)$$

1st-order necessary condition: $\nabla f(x) = 0$

↳ Gradient $\nabla f: \mathbb{R}^n \to \mathbb{R}^n$ is C^1

Based on Taylor expansion, we have:

$$\nabla f(x_k + p) = \nabla f(x_k) + \nabla^2 f(x_k)\, p + \underbrace{g(p)}_{\to o(p) \text{ small}}$$

Setting:

$$0 = \nabla f(x_k) + \nabla^2 f(x_k) p$$

$$\Rightarrow \underbrace{\nabla^2 f(x_k)}_{\text{Hessian}}\, p = -\underbrace{\nabla f(x_k)}_{\text{gradient}}$$

↑ search direction

Algorithm 3 (Local Newton)

1. Choose starting value $x_0 \in \mathbb{R}^n$

For $k = 0, 1, \ldots$

2. If $\nabla f(x_k) = 0$ or $\|\nabla f(x_k)\| < TOL \rightarrow STOP$
3. Compute $p_k \in \mathbb{R}^n$ via

$$\nabla^2 f(x_k) p_k = -\nabla f(x_k)$$

4. Set $x_{k+1} = x_k + p_k$

Remark:

The step length is $\alpha = 1$

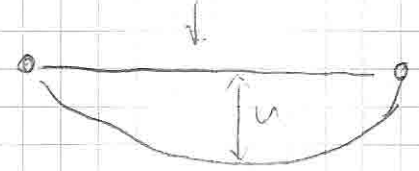
Remarks

1. Newton is only local
   ↳ globalization crucial topic (many possibilities) (e.g. $\alpha < 1$)
2. Local convergence is quadratic
3. Building the Hessian can be cumbersome
   ↳ Quasi-Newton methods: replace $\nabla^2 f(x_k)$ by some "good" approximation
4. Solving the linear system might be expensive
5. Memory of storing matrix $\nabla^2 f(x_k)$ might be problematic
6. Newton is invariant under affine linear coordinate transformations (Deuflhard 2011)
7. Difficulty if $\nabla^2 f(x_k)$ is not s.p.d (NW) 15

1. Two introductory examples

1.1 Dirichlet energy minimization

Deflection of a membrane over Ω



→ many physical processes get into the state of minimal energy

Solve:

$$\min_{v \in H_0^1(\Omega)} f(v) := \frac{1}{2} \int |\nabla v|^2 dx - \int_\Omega f v \, dx$$

⇒ infinite-dimensional problem, but can be transformed into a sequence of finite dimensional subproblems $V_h \subset H_0^1(\Omega)$, $\dim V_h < \infty$

e.g. $\dim V_h = n$

$$\Rightarrow \min_{v \in V_h} f(v)$$

Typical choice for V_h is a triangulation $\mathcal{T}_h$ of Ω into elements T with

$$V_h = \{ v \in C(\bar{\Omega}) : v|_T \in P_n(T), \, v|_{\partial\Omega} = 0 \}$$

1.2. Example 2 (Linear regression)

A procedure (physics, engineering, biology, ...) yields to certain input data $u \in \mathbb{R}^r$ an answer $y \in \mathbb{R}^s$.

The behavior of this system shall be analyzed (through approximated)

$$u \longmapsto g(u,x) \quad , x \in X \subset \mathbb{R}^n$$

For example y_i $(i=1,\dots,n)$ are measurements caused by the input data u_i.

The function $g(u,x)$ should coincide as good as possible with y_i

↓ in a certain norm

In the least squares approach, we work with the Euclidian norm and obtain x as solution of the following minimization problem:

$$\min \sum_{i=1}^{N} \|y_i - g(u_i,x)\|^2 \qquad \text{s.t. } x \in X$$

In case of $X = \mathbb{R}^n$ we have the classical problem of nonlinear regression.