Numerical Methods I: Numerical optimization

Georg Stadler
Courant Institute, NYU
stadler@cims.nyu.edu

Oct 19, 2017

Optimization problems

Main source: Nocedal/Wright: *Numerical Optimization*, Springer 2006.

Different optimization problems:

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

where $f:\mathbb{R}^n\to\mathbb{R}$. Often, one additionally encounters constraints of the form

$$g(\boldsymbol{x}) = 0 \qquad \text{(equality constraints)}$$

$$h(\boldsymbol{x}) \geq 0 \qquad \text{(inequality constraints)}$$

- Often used: “programming” $\equiv$ optimization
- continuous optimization ($\boldsymbol{x}\in\mathbb{R}^n$) versus discrete optimization (e.g., $\boldsymbol{x}\in\mathbb{Z}^n$)
- nonsmooth (e.g., f is not differentiable) versus smooth optimization (we assume $f\in C^2$)
- convex optimization vs. nonconvex optimization (convexity of f)

Continuous unconstrained optimization

Assumptions

We assume that $f(\cdot) \in C^2$, and assume unconstrained minimization problems, i.e.:

$$\min_{\boldsymbol{x} \in \mathbb{R}^n} f(\boldsymbol{x}).$$

A point $\boldsymbol{x}^*$ is a global solution if

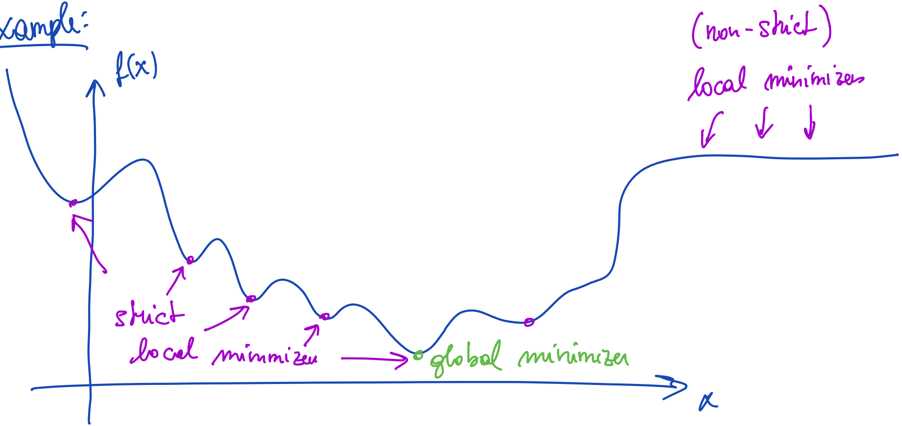
$$f(\boldsymbol{x}) \geq f(\boldsymbol{x}^*) \tag{1}$$

for all $\boldsymbol{x} \in \mathbb{R}^n$, and a local solution if (1) for all $\boldsymbol{x}$ in a neighborhood of $\boldsymbol{x}^*$.

Strict (local/global) minimizers satisfy (1) with a "$>$" instead of a "$\geq$" in a neighborhood of the point.

Continuous unconstrained optimization

Example:



Continuous unconstrained optimization

Necessary conditions

At a local minimum $\boldsymbol{x}^*$ holds the first-order necessary condition

$$\mathbb{R}^n \ni \nabla f(\boldsymbol{x}^*) = 0$$

and the second-order (necessary) sufficient condition

$$\mathbb{R}^{n\times n} \ni \nabla^2 f(\boldsymbol{x}^*) \quad \text{is positive (semi-) definite.}$$

Proof that at the minimum x^* holds $\nabla f(x^*) = 0$ if f is continuously diff'able:

Suppose $\nabla f(x^*) \neq 0$, choose $p = -\nabla f(x^*)$

$p^T \nabla f(x^*) = -\|\nabla f(x^*)\|^2 < 0$, Since f is $C^1 \Rightarrow \exists T > 0$:

$p^T \nabla f(x^* + tp) < 0$ for all $t \in (0, T]$

Taylor: $\bar{t} \in [0, T]$: $f(x^* + \bar{t}p) = f(x^*) + \bar{t}\, \underbrace{p^T \nabla f(x^* + tp)}_{<0}$, $t \in (0, \bar{t})$

$< f(x^*) \rightarrow$ contradiction!

To find a candidate for a minimum, we can thus solve the nonlinear equation for a stationary point:

$$G(\boldsymbol{x}) := \nabla f(\boldsymbol{x}) = 0,$$

for instance with Newton's method. Note that the Jacobian of $G(\boldsymbol{x})$ is $\nabla^2 f(\boldsymbol{x})$.

In optimization, one often prefers iterative descent algorithms that take into account the optimization structure.

Example: $f(x_1, x_2) = f(x) = x_1^4 + x_2^2 + x_1 x_2, \quad f: \mathbb{R}^2 \to \mathbb{R}$

necessary cond: $\nabla f(x) = 0 = \begin{pmatrix} 4x_1^3 + x_2 \\ 2x_2 + x_1 \end{pmatrix} \in \mathbb{R}^2$

$\nabla^2 f(x) = \begin{pmatrix} 12x_1^2 & 1 \\ 1 & 2 \end{pmatrix} \in \mathbb{R}^{2\times 2}$

Convex minimization

A function If $f : \mathbb{R}^n \to \mathbb{R}$ is convex if for all $\boldsymbol{x}, \boldsymbol{y}$ holds, for all $t \in [0, 1]$:

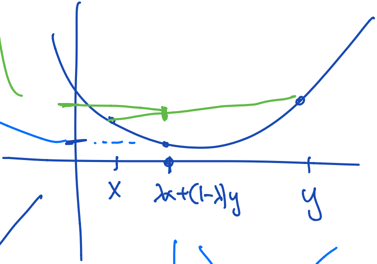
$$f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \leq \lambda f(\boldsymbol{x}) + (1-\lambda) f(\boldsymbol{y})$$

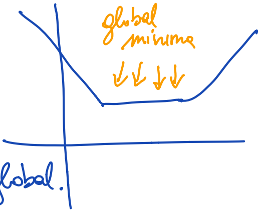
Theorem: If f is convex, then any local minimizer $\boldsymbol{x}^*$ is also a global minimizer. If f is differentiable, then any stationary point $\boldsymbol{x}^*$ is a global minimizer.

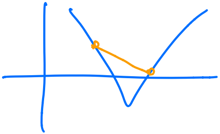
Proof: 1.) Show that any local minimum is also a global minimum.

Proof by contradiction:

Let x^* be a local minimum, but not global.





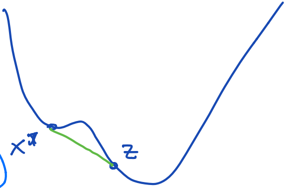


Convex minimization

$\longrightarrow \exists\, z \in \mathbb{R}^n : f(z) < f(x^*)$

Consider line segment between x^* and z.

The convexity implies that for all $\lambda \in (0,1)$



$$f(\lambda x^* + (1-\lambda) z) \leq \lambda f(x^*) + (1-\lambda) f(z) < f(x^*) \quad \text{for all } \lambda \in (0,1)$$

$\longrightarrow$ every neighborhood of x^* contains points that have a function value less than $f(x^*)$ $\rightarrow$ contradiction!

2.) Let x^* with $\nabla f(x^*) = 0$, but x^* is not global minimizer.

$$0 = \nabla f(x^*)(z - x^*) = \frac{d}{d\lambda} f(x^* + \lambda(z - x^*))\Big|_{\lambda=0}$$

$$= \lim_{\lambda \downarrow 0} \frac{f(x^* + \lambda(z - x^*)) - f(x^*)}{\lambda}$$

$$\leq \lim_{\lambda \downarrow 0} \frac{\lambda f(z) + (1-\lambda) f(x^*) - f(x^*)}{\lambda} = f(z) - f(x^*) < 0$$

Contradiction!

Descent algorithm

Basic descent algorithm:

1. Initialize starting point $\boldsymbol{x}^0$, set $k = 1$.
2. For $k = 0, 1, 2, \ldots$, find a descent direction $\boldsymbol{d}^k$
3. Find a step length $\alpha_k > 0$ for the update

$$\boldsymbol{x}^{k+1} := \boldsymbol{x}^k + \alpha_k \boldsymbol{d}^k$$

such that $f(\boldsymbol{x}^{k+1}) < f(\boldsymbol{x}^k)$. Set $k := k + 1$ and repeat.

Descent algorithm

Idea: Instead of solving an n-dim. minimization problem, (approximately) solve a sequence of 1-dim. problems:

- Initialization: As close as possible to $\boldsymbol{x}^*$.
- Descent direction: Direction in which function decreases locally.
- Step length: Want to make large, but not too large steps.
- Check for descent: Make sure you make progress towards a (local) minimum.

Descent algorithm

Initialization: Ideally close to the minimizer. Solution depends, in general, on initialization (in the presence of multiple local minima).

Descent algorithm

d descent direction if $\nabla f(x^k)^T d < 0$

Directions, in which the function decreases (locally) are called descent directions.

pos. def: $(x, Bx) > 0 \quad \forall x$

- Steepest descent direction:

$$\boldsymbol{d}^k = -\nabla f(\boldsymbol{x}^k)$$

- When $B_k \in \mathbb{R}^{n \times n}$ is positive definite, then

$$\boldsymbol{d}^k = -B_k^{-1} \nabla f(\boldsymbol{x}^k)$$

  is the quasi-Newton descent direction.

- When $H_k = H(\boldsymbol{x}^k) = \nabla^2 f(\boldsymbol{x}^k)$ is positive definite, then

$$\boldsymbol{d}^k = -H_k^{-1} \nabla f(\boldsymbol{x}^k)$$

  is the Newton descent direction. At a local minimum, $H(\boldsymbol{x}^*)$ is positive (semi)definite.

Descent algorithm

Why is the negative gradient the steepest direction?

$\alpha \in \mathbb{R},\ p \in \mathbb{R}^n$

$$g(\alpha) = f(x^k + \alpha p) = f(x^k) + \alpha p^T \nabla f(x^k) + \frac{\alpha^2}{2} p^T \nabla^2 f(x + tp) p$$

rate at which this function changes depends on $p^T \nabla f(x^k)$ $\quad t \in (0, \alpha)$ $\quad (f \in C^2)$

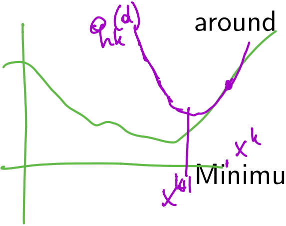
$\longrightarrow$ choose $p = -\nabla f(x^k)$ "steepest descent direction"

Thus: normalized negative gradient direction $-\frac{\nabla f(x^k)}{\|\nabla f(x^k)\|}$ is solution to

$$\min_p p^T \nabla f(x^k), \quad \|p\| = 1$$

$|p^T \nabla f(x^k)| \leq \|p\| \, \|\nabla f(x^k)\|$

$\longrightarrow -\frac{\nabla f(x^k)}{\|\nabla f(x^k)\|}$ makes inequality an equality $\longrightarrow$ optimal



Idea behind Newton's method in optimization: Instead of finding minimum of f, find minimum of quadratic approximation of f around current point:

$$q_k(\boldsymbol{d}) = f(\boldsymbol{x}^k) + \nabla f(\boldsymbol{x}^k)^T \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^T \nabla^2 f(\boldsymbol{x}^k)\boldsymbol{d}$$

Minimum is (provided $\nabla^2 f(\boldsymbol{x}^k)$ is spd):

$$\boldsymbol{d} = -\nabla^2 f(\boldsymbol{x}^k)^{-1} \nabla f(\boldsymbol{x}^k).$$

is the Newton search direction. Since this is the minimum of the quadratic approximation, $\alpha_k = 1$ is the "optimal" step length.

Descent algorithm

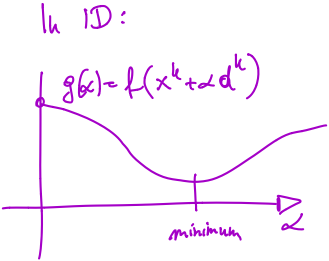
Step length: Need to choose step length $\alpha_k > 0$ in

$$x^{k+1} := x^k + \alpha_k d^k$$

Ideally: Find minimum α of 1-dim. problem

$$\min_{\alpha>0} f(x^k + \alpha d^k).$$

It is not necessary to find the exact minimum.



Descent algorithm

Step length (continued): Find α_k that satisfies the Armijo condition:

$$f(\boldsymbol{x}^k + \alpha_k \boldsymbol{d}^k) \leq f(\boldsymbol{x}^k) + c_1 \alpha_k \nabla f(\boldsymbol{x}^k)^T \boldsymbol{d}^k, \tag{2}$$

where $c_1 \in (0,1)$ (usually chosen rather small, e.g., $c_1 = 10^{-4}$).

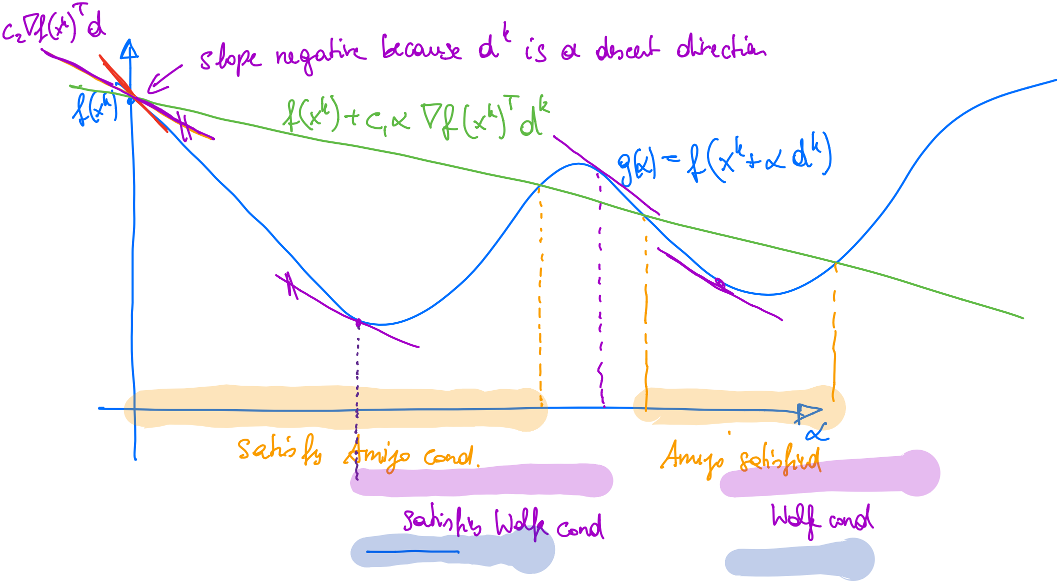
Additionally, one often uses the gradient condition

$$\nabla f(\boldsymbol{x}^k + \alpha_k \boldsymbol{d}^k)^T \boldsymbol{d}^k \geq c_2 \nabla f(\boldsymbol{x}^k)^T \boldsymbol{d}^k \tag{3}$$

with $c_2 \in (c_1, 1)$.

The two conditions (2) and (3) are called Wolfe conditions.

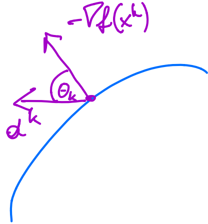
Armijo/Wolfe conditions



Descent algorithm

Convergence of line search methods

Denote the angle between $\boldsymbol{d}^k$ and $-\nabla f(\boldsymbol{x}^k)$ by Θ_k:



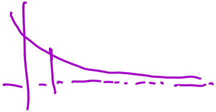
$$\cos(\Theta_k) = \frac{-\nabla f(\boldsymbol{x}^k)^T \boldsymbol{d}^k}{\|\nabla f(\boldsymbol{x}^k)\| \|\boldsymbol{d}^k\|}.$$

Assumptions on $f : \mathbb{R}^n \to \mathbb{R}$: continuously differentiable, derivative is Lipschitz-continuous, f is bounded from below.
Method: descent algorithm with Wolfe-conditions.
Then:

$$\sum_{k\geq 0} \cos^2(\Theta_k) \|\nabla f(\boldsymbol{x}^k)\|^2 < \infty.$$

In particular: If $\cos(\Theta_k) \geq \delta > 0$, then $\lim_{k\to\infty} \|\nabla f(\boldsymbol{x}^k)\| = 0$.

Note that this does not imply that $\boldsymbol{x}^k$ converges.



Descent algorithm

Alternative to Wolfe step length: Find α_k that satisfies the Armijo condition:

$$f(\boldsymbol{x}^k + \alpha_k \boldsymbol{d}^k) \leq f(\boldsymbol{x}^k) + c_1 \alpha_k \nabla f(\boldsymbol{x}^k)^T \boldsymbol{d}^k, \tag{4}$$

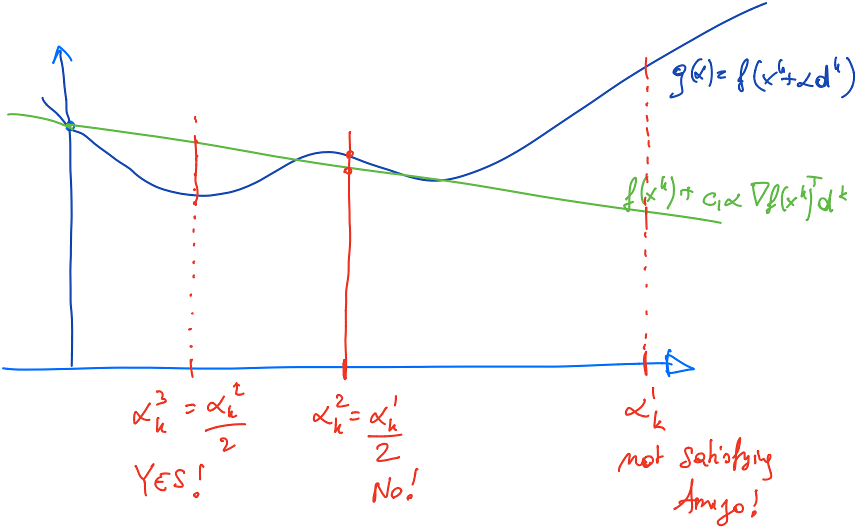
where $c_1 \in (0, 1)$.

Use backtracking linesearch to find a step length that is large enough:

- Start with (large) step length $\alpha_k^0 > 0$.
- If it satisfies (4), accept the step length.
- Else, compute $\alpha_k^{i+1} := \rho \alpha_k^i$ with $\rho < 1$ (usually, $\rho = 0.5$) and go back to previous step.

This also leads to a globally converging method to a stationary point.

Backtracking



Let us consider a simple case, where f is quadratic:

$$f(\boldsymbol{x}) := \frac{1}{2}\boldsymbol{x}^T Q\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{x},$$

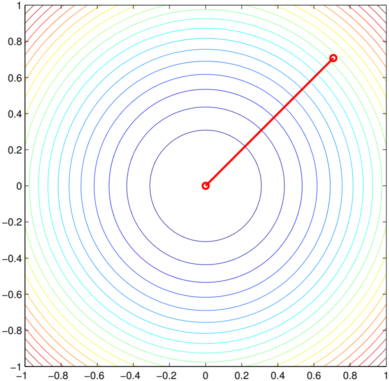
where Q is spd. The gradient is $\nabla f(x) = Q\boldsymbol{x} - b$, and minimizer $\boldsymbol{x}^*$ is solution to $Q\boldsymbol{x} = \boldsymbol{b}$. Using exact line search, the convergence is:

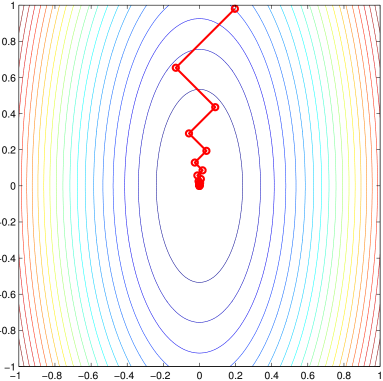
$$\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^*\|_Q^2 \leq \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}}\|\boldsymbol{x}^k - \boldsymbol{x}^*\|_Q^2$$

(linear convergence with rate depending on eigenvalues of Q)

Descent algorithms

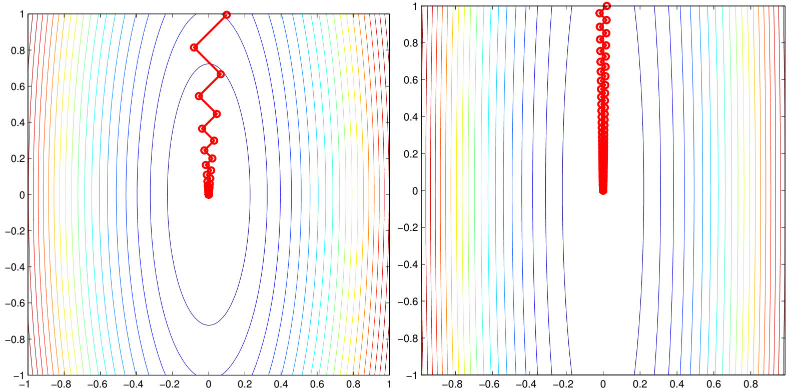
Convergence of steepest descent





Descent algorithms

Convergence of steepest descent



Newton's method: Assumptions on f: 2×differentiable with Lipschitz-continuous Hessian $\nabla^2 f(\boldsymbol{x}^k)$. Hessian is positive definite in a neighborhood around solution $\boldsymbol{x}^*$.

Assumptions on starting point: $\boldsymbol{x}^0$ sufficient close to $\boldsymbol{x}^*$.

Then: Quadratic convergence of Newton's method with $\alpha_k = 1$, and $\|\nabla f(\boldsymbol{x}^k)\| \to 0$ quadratically.

Equivalent to Newton's method for solving $\nabla f(\boldsymbol{x}) = 0$, if Hessian is positive.

How many iterations does Newton need for quadratic problems?

Summary of Newton methods and variants

- Newton to solve nonlinear equation $F(\boldsymbol{x}) = 0$.
- Newton to solve optimization problem is equivalent to solving for the stationary point $\nabla f(\boldsymbol{x}) = 0$, provided Hessian is positive and full steps are used (compare also convergence result).
- Optimization perspective to solve $\nabla f(\boldsymbol{x})$ provided additional information.
- Gauss-Newton method for nonlinear least squares problem is a specific quasi-Newton method.