Optimization and Data Science

Lecture 9: Gradient and General Descent Methods

Prof. Dr. Thomas Slawig

Kiel University - CAU Kiel
Dep. of Computer Science

Summer 2020

Contents

Descent methods

- What are descent methods?
  Class of iterative optimization algorithms
  Here: for unconstrained problems (can be extended to constrained ones)
- Why are we studying these methods?
  Most important class, a variety of methods, convergence result available
  First choice: gradient method, the easiest descent method
- How does it work?
  Finding a direction where the cost is reduced (search direction)
  Going an appropriate step in this direction (line search)
- What if we can use it?
  Applicable to every problem
  Easy to implement
  Convergence speed known under some assumptions

Contents

Descent directions

Theorem (Generalization of first order necessary condition)

Let $f : \mathbb{R}^n \to \mathbb{R}$ have continuous partial derivatives for all k in the local minimizer $x^* \in \mathbb{R}^n$. Then $\nabla f(x^*)^\top d \geq 0$ for all directions $d \in \mathbb{R}^n$.

$\rightsquigarrow$ If (at $x \in \mathbb{R}^n$) we find some $d \in \mathbb{R}^n$ with

$$\nabla f(x)^\top d = \lim_{h \to 0} \frac{f(x + hd) - f(x)}{h} < 0,$$

we have

$$f(x + hd) < f(x) \text{ for } h > 0 \text{ small enough.}$$

$\rightsquigarrow$ x is not a minimizer.

Definition

Let $f : \mathbb{R}^n \to \mathbb{R}$ and $x \in \mathbb{R}^n$ be a point where all partial derivatives exist. A direction $d \in \mathbb{R}^n$ is called **descent direction (in x)** if $\nabla f(x)^\top d < 0$.

General form of a descent method

Algorithm (General descent method):

1. Choose initial guess $x_0 \in \mathbb{R}^n$.
2. For $k = 0, 1, \dots$:
   1. Choose a descent direction $d_k \in \mathbb{R}^n$.
   2. Choose a step-size $\rho_k > 0$ that satisfies $f(x_k + \rho_k d_k) < f(x_k)$.
   3. Set $x_{k+1} = x_k + \rho_k d_k$.

   until a stopping criterion is satisfied.

- **Notation:** Here $x_k \in \mathbb{R}^n, k = 0, \dots,$ denotes the $k-$th iterate with components $x_{ki} \in \mathbb{R}, i = 1, \dots, n$.
- What are reasonable stopping criteria?
  - $\|\nabla f(x_k)\| < \epsilon_1 \rightsquigarrow$ 1st order necessary condition is satisfied
  - $\rho_k < \epsilon_2 \rightsquigarrow$ step-size too small
  - $\|x_{k+1} - x_k\| = \rho_k \|d_k\| < \epsilon_3 \rightsquigarrow$ step too small.

Contents

Gradient method

- If $\nabla f(x) \neq 0$, the negative gradient is a descent direction in $x \in \mathbb{R}^n$ ...
- ... since for $d = -\nabla f(x)$ we obtain

$$\nabla f(x)^\top d = -\nabla f(x)^\top \nabla f(x) = -\|\nabla f(x)\|_2^2 < 0.$$

- We thus obtain a first descent method, the ...

Algorithm (Gradient method or **Method of steepest descent)**:

1. Choose initial guess $x_0 \in \mathbb{R}^n$.
2. For $k = 0, 1, \ldots :$
    1. Compute the negative gradient $d_k = -\nabla f(x_k)$.
    2. Choose a step-size $\rho_k > 0$ that satisfies $f(x_k + \rho_k d_k) < f(x_k)$.
    3. Set $x_{k+1} = x_k + \rho_k d_k$.

   until a stopping criterion is satisfied.

How to compute or approximate the gradient?

- Exactly/analytically (for simple functions) by hand or ...
- ... symbolically
- ... or algorithmically using some software.
- Approximately by

$$g_i := \frac{f(x + he_i) - f(x)}{h}, \quad i = 1, \ldots, n,$$

  with some fixed $h > 0$. Then

$$g := (g_i)_{i=1}^n \approx \nabla f(x).$$

  How many function evaluations are necessary if the gradient is approximated like this?
- Can use this approximation even if f is only directional differentiable in x in direction $\pm e_i$
- ... consider $f(x) = |x|$ at $x = 0$ with $d = \pm 1$.

Contents

Step-size control: the naive way

- Simplest choice of step-size ρ_k: Choose fixed value $\rho_k = \rho$ for all k.
- What to do if

$$f(x_k + \rho_k d_k) < f(x_k) \tag{1}$$

  is not satisfied for the chosen step-size?
- $\rightsquigarrow$ step-size too big
- Choose "smaller" step-size $\rightsquigarrow$ how small?
- One idea: Use smaller step-size in every step.
- But: We need "more" than just a step-size ρ_k that satisfies (1).
- Step-size could be also too small, see next example.

Possible problem: step-size becomes too small

- $f(x) = x^2$, minimizer is $x^* = 0$. Derivative: $\nabla f(x) = f'(x) = 2x$.
- Take $x_0 = 2.5$ as initial guess for an optimization with $d_k = -1$ for all k.
- This is a descent direction for all $x_k > 0$ since

$$\nabla f(x_k)^\top d_k = -f'(x_k) = -2x_k < 0.$$

- We use step-sizes $\rho_k = \left(\frac{1}{2}\right)^k$. This gives

$$x_{k+1} = x_k + \rho_k d_k = x_k - \rho_k = x_0 - \sum_{i=0}^{k} \rho_i = x_0 - \sum_{i=0}^{k} \left(\frac{1}{2}\right)^i = x_0 - \frac{1 - \left(\frac{1}{2}\right)^{k+1}}{1 - \frac{1}{2}}$$

$$= x_0 - 2\left(1 - \left(\frac{1}{2}\right)^{k+1}\right) = \frac{1}{2} + \left(\frac{1}{2}\right)^k \to \frac{1}{2}.$$

- We obtain $x_k \to \frac{1}{2} \neq 0 = x^*$ (minimizer).
- Note: This was **not** the gradient method. Why?

Contents

Formal condition (on the step-size/line search) for convergence

Definition

For

- a sequence $(x_k)_{k\in\mathbb{N}} \subset \mathbb{R}^n$ of iterates with $\nabla f(x_k) \neq 0$ for all $k \in \mathbb{N}$
- and a sequence of search directions $(d_k)_{k\in\mathbb{N}} \subset \mathbb{R}^n \setminus \{0\}$

the sequence of step-sizes $(\rho_k)_{k\in\mathbb{N}} \subset \mathbb{R}_{>0}$ is called **efficient**, if there exists $c_S > 0$ such that

$$f(x_k + \rho_k d_k) \leq f(x_k) - c_S \left(\frac{\nabla f(x_k)^\top d_k}{\|d_k\|} \right)^2 \text{ for all } k \in \mathbb{N}.$$

$\rightsquigarrow$ There has to be "enough" descent in the cost function.

- Important: The constant c_S has to be independent of k.
- The step-size in the above example does **not** satisfy this condition!
- But how to realize this condition?

Contents

Armijo line search

- Most used line search algorithm.
- Based on a step-size halvening ...
- ... and checking one condition.
- Idea: Find biggest step-size

$$\rho \in \left\{2^{-j} : j \in \mathbb{Z}\right\}$$

that satisfies

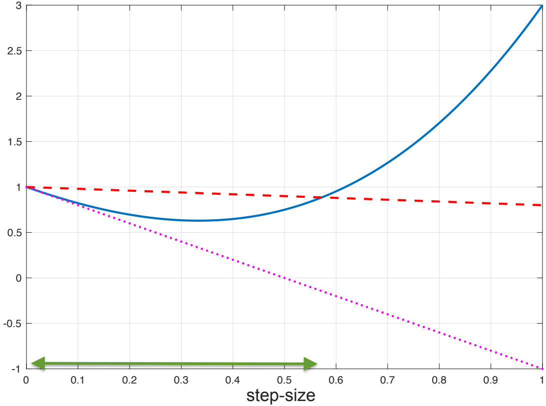
$$f(x + \rho d) \leq f(x) + \rho\delta\nabla f(x)^\top d. \tag{2}$$

- Gives step-size that is neither too big nor too small,
- ... but an efficient step-size (sufficient for convergence result).

Armijo Condition 1: Step-size not too big

- Choose a parameter $\delta \in (0,1)$. Then determine $\rho_k > 0$ such that

$$f(x_k + \rho_k d_k) \leq f(x_k) + \rho_k \delta \underbrace{\nabla f(x_k)^\top d_k}_{<0}. \tag{2}$$



$f(x_k + \rho d_k)$

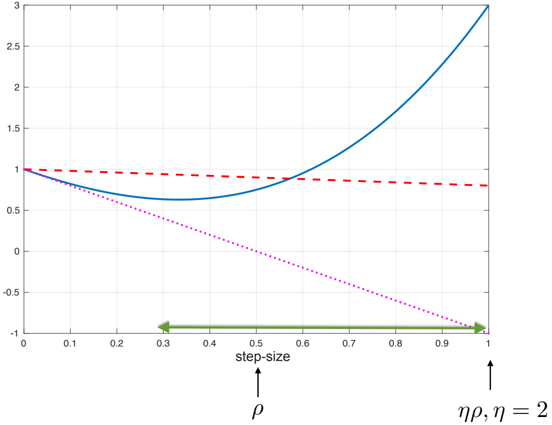
$f(x_k) + \rho\delta \underbrace{\nabla f(x_k)^\top d_k}_{<0}$

$f(x_k) + \rho \underbrace{\nabla f(x_k)^\top d_k}_{<0}$

Armijo condition 2: Step-size not too small

- Choose a second parameter $\eta > 1$ (e.g., $\eta = 2$) and ensure that $\rho_k > 0$ satisfies also

$$f(x_k + \eta\rho_k d_k) \geq f(x_k) + \eta\rho_k\delta\nabla f(x_k)^\top d_k. \quad (3)$$



Armijo condition 2: Step-size not too small

Lemma

The sequence of step-sizes $(\rho_k)_{k\in\mathbb{N}}$ *is efficient if it satisfies* (2) *and there exists* $\alpha > 0$ *with*

$$\rho_k \geq -\alpha \frac{\nabla f(x_k)^\top d_k}{\|d_k\|^2} = \alpha \frac{|\nabla f(x_k)^\top d_k|}{\|d_k\|^2} \quad \textit{for all } k \in \mathbb{N}. \tag{4}$$

Proof.

(2) gives

$$f(x_k + \rho_k d_k) - f(x_k) \leq \rho_k \delta \overbrace{\nabla f(x_k)^\top d_k}^{<0} = -\rho_k \delta |\nabla f(x_k)^\top d_k| \leq -\alpha\delta \left(\frac{\nabla f(x_k)^\top d}{\|d_k\|} \right)^2$$

which is the definition of an efficient step-size (with $c_S = \alpha\delta > 0$). □

Second Armijo condition gives step-size that is not too small

(3):

$$
\begin{aligned}
& & f(x_k+\eta\rho d_k) &\geq f(x_k)+\eta\rho\delta\nabla f(x_k)^\top d_k \\
\iff & & \eta\rho\delta\nabla f(x_k)^\top d_k &\leq f(x_k+\eta\rho d_k)-f(x_k) \\
\iff & & (\delta-1)\eta\rho\nabla f(x_k)^\top d_k &\leq \underbrace{f(x_k+\eta\rho d_k)-f(x_k)}_{=\nabla f(x_k+\theta\eta\rho d_k)^\top\eta\rho d_k} -\eta\rho\nabla f(x_k)^\top d_k \\
& \text{Mean value theorem } \rightarrow & & \\
& & &\leq (\nabla f(x_k+\theta\eta\rho d_k)-\nabla f(x_k))^\top\eta\rho d \quad \text{with some } \theta\in[0,1] \\
& \text{If gradient Lipschitz-continuous } \rightarrow & &\leq \|\nabla f(x_k+\theta\eta\rho d_k)-\nabla f(x_k)\|\|d_k\|\eta\rho \\
& & &\leq L\theta\eta^2\rho^2\|d_k\|^2 \leq L\eta^2\rho^2\|d_k\|^2.
\end{aligned}
$$

Thus, using $\delta\in(0,1)$, we get

$$
\rho \geq \frac{(\delta-1)\nabla f(x_k)^\top d_k}{L\eta\|d_k\|^2} = -\frac{(1-\delta)\nabla f(x_k)^\top d_k}{L\eta\|d_k\|^2}
$$

which gives (4) with $\alpha=(1-\delta)/(L\eta)$.

Algorithm: Armijo step-size

Input: parameter $\delta > 0$ (typical choice is $\delta = 10^{-4}$),
iterate $x \in \mathbb{R}^n$, descent direction $d \in \mathbb{R}^n, d \neq 0$.

Output: Efficient step-size ρ.

1. Set $\rho = 1$.
2. Repeat $\rho := 2\rho$ until (2), i.e.,

$$f(x + \rho d) \leq f(x) + \rho\delta\nabla f(x)^\top d$$

   is violated.
3. Repeat $\rho := \rho/2$ until (2) is satisfied.

Remark:

- This gives the biggest step-size $\rho \in \{2^{-j} : j \in \mathbb{Z}\}$ that satisfies (2).

Contents

Stopping criteria use norms

- Typical criteria

$$
\begin{aligned}
\|x_{k+1} - x_k\| &\leq \epsilon_1 \\
\|\nabla f(x_k)\| &\leq \epsilon_2 \quad \text{with some } \epsilon_1, \epsilon_2 > 0,
\end{aligned}
$$

- involve vector norms:

$$
\begin{aligned}
\|x\|_2 &:= \sqrt{\sum_{i=1}^{n} x_i^2} \quad \text{(Euclidean norm)} \\
\|x\|_1 &:= \sum_{i=1}^{n} |x_i|, \\
\|x\|_\infty &:= \max_{i=1,\dots,n} |x_i| \quad \text{(Maximum norm)}
\end{aligned}
$$

Absolute and relative differences

- But: What does it mean if difference $\|x_{k+1} - x_k\|$ is "small"?
- $\|x_{k+1} - x_k\| < 10^{-4}$ for $\|x_{k+1}\|, \|x_k\| \approx 10^{12} \rightsquigarrow$ "small"?
- $\|x_{k+1} - x_k\| < 10^{-4}$ for $\|x_{k+1}\|, \|x_k\| \approx 10^{-4} \rightsquigarrow$ "small"?
- Better than to check the **absolute difference**

$$\|x_{k+1} - x_k\|$$

- ... is to check the **relative difference**

$$\frac{\|x_{k+1} - x_k\|}{\|x_k\|}$$

- But x_k might tend to or even become zero.
- Define "typical value" $x_{typ} \neq 0$ beforehand (or use $x_{typ} = x_0$) and check

$$\frac{\|x_{k+1} - x_k\|}{\max\{\|x_k\|, \|x_{typ}\|\}}.$$

Criteria for differently scaled optimization variables

- The different optimization variables (i.e., the components x_{ki} of $x_k = (x_{ki})_{i=1}^n$) might be also very different in their magnitude.

$\rightsquigarrow$ We say they are differently (or badly) **scaled**.

- Then it makes sense to use the **component-wise relative difference**

$$\operatorname{reldiff} x_i \quad := \quad \frac{|x_{k+1,i} - x_{ki}|}{\max(|x_{ki}|, x_{typ,i})}, \qquad i = 1, \dots n,$$

and its norm as stopping criterion:

$$\|(\operatorname{reldiff} x_i)_{i=1}^n\| \leq \epsilon.$$

Checking for the relative gradient

- Same thing for the gradient:

$$(\nabla f(x_k))_i := \frac{\partial f}{\partial x_i}(x_k) = \lim_{h \to 0} \frac{f(x_k + he_i) - f(x_k)}{h}, \quad i = 1, \ldots, n.$$

- Use relative numerator and denominator:

$$\lim_{h \to 0} \frac{\dfrac{f(x_k + he_i) - f(x_k)}{f(x_k)}}{\dfrac{h}{x_{ki}}} = \lim_{h \to 0} \frac{f(x_k + he_i) - f(x_k)}{h} \frac{x_{ki}}{f(x_k)} = \frac{(\nabla f(x_k))_i x_{ki}}{f(x_k)}.$$

- Use again typical values of x and function f_{typ} (e.g., $f_{typ} = f(x_0)$)

$$\text{relgrad}_i f(x_k) := \frac{(\nabla f(x_k))_i \max(|x_{ki}|, x_{typ,i})}{\max(|f(x_k)|, f_{typ})}$$

  and its norm as stopping criterion:

$$\|(\text{relgrad}_i f(x_k))_{i=1}^n\| \leq \epsilon.$$

What is important

- Descent methods are a class of iterative optimization methods ...
- ... using a descent direction and a line search until some stopping criterion is satisfied.
- The gradient method (method of steepest descent) is one descent method, using the negative gradient as descent direction.
- The step-size in the line search must neither be too small nor too big.
- For convergence, we require a so-called **efficient** step-size.
- The Armijo rule/algorithm provides such kind of step-size.
- For the stopping criteria, different options are available.
- Taking relative values in the criteria avoid some problems w.r.t. bad scaling.