Optimization

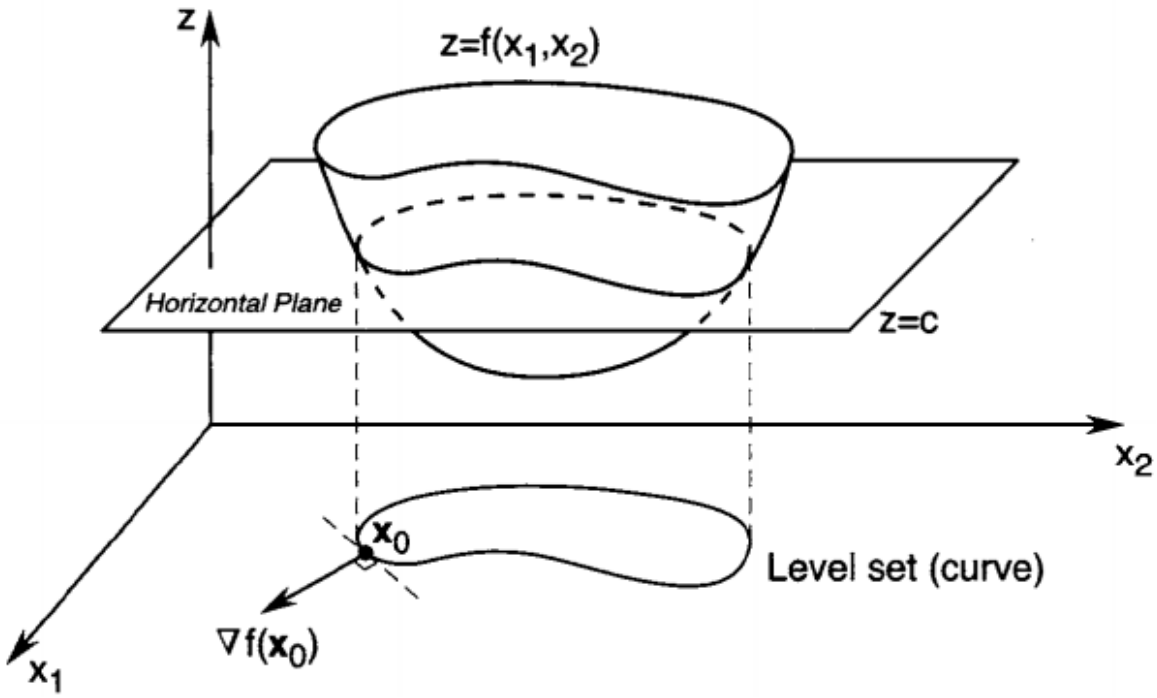
Màster de Fonaments de Ciència de Dades

Lecture IX. Subgradient methods for convex problems

Gerard Gómez

Background. Gradient methods

Recall that **gradient methods** (such as steepest descent or conjugate gradient), are used for the computation of an extremum of a **unconstrained minimization** of a **continuously differentiable function**



Background. Gradient methods

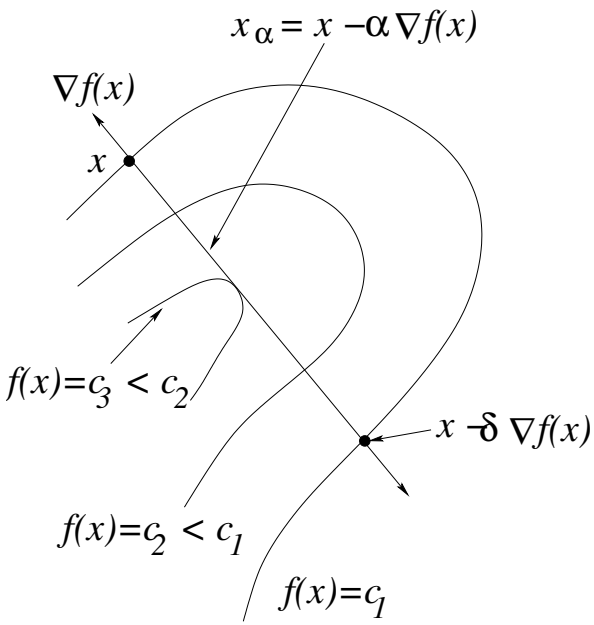
Gradient methods are based in the following equation

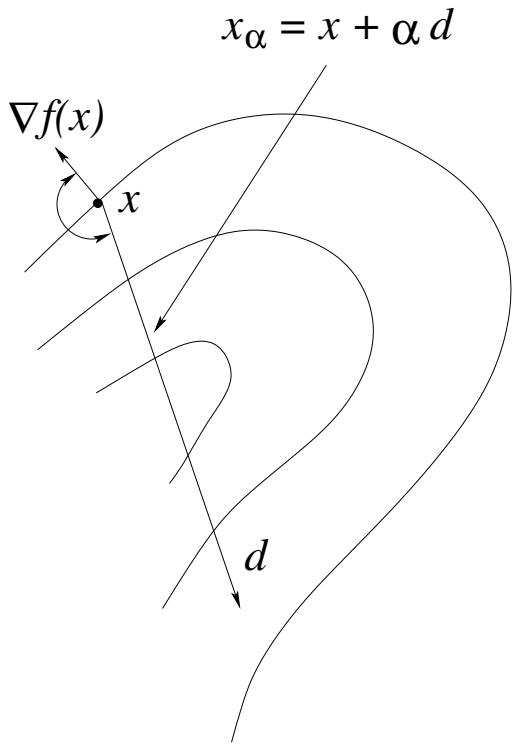
$$\boldsymbol{x}_\alpha = \boldsymbol{x} - \alpha \nabla f(\boldsymbol{x}), \quad \alpha \geq 0$$

that can be generalised to

$$\boldsymbol{x}_\alpha = \boldsymbol{x} + \alpha \boldsymbol{d}, \quad \alpha \geq 0$$

where $\alpha \in \mathbb{R}$ is the **stepsize** and the **descent direction**, $\boldsymbol{d} \in \mathbb{R}^n$ makes an angle with $\nabla f(\boldsymbol{x})$ greater than 90^o $(\nabla f(\boldsymbol{x})^T \boldsymbol{d} < 0)$



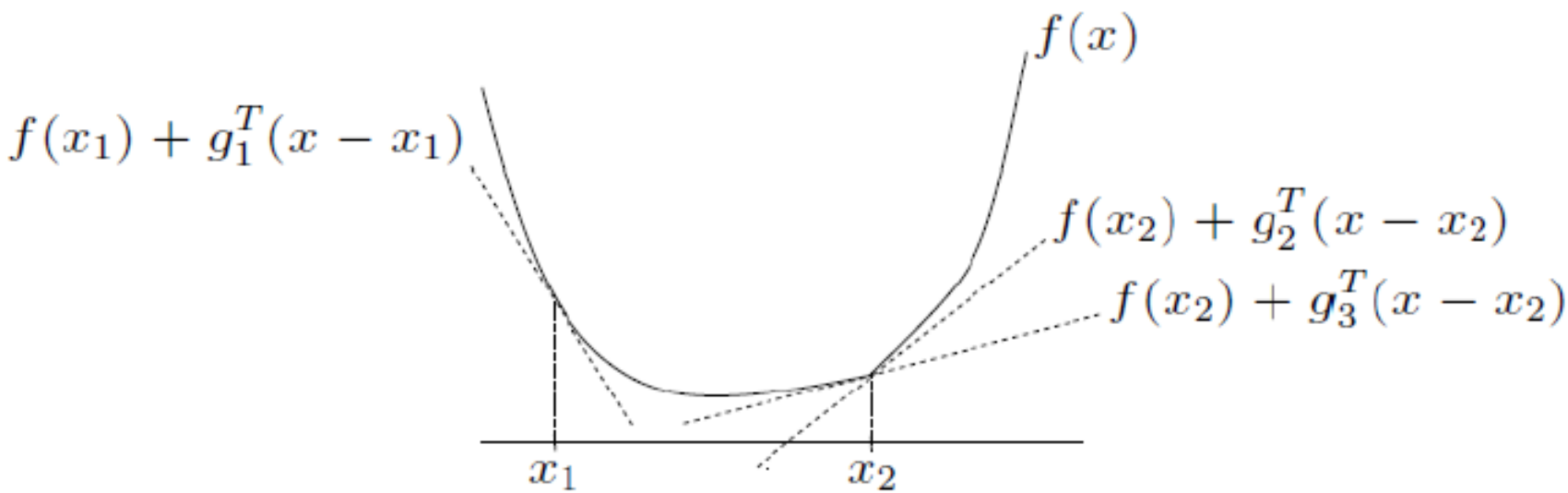


Background. Differential properties of convex functions

Recall that a **subgradient** of a convex function f at a point $x \in \mathbb{R}^n$, is any vector $g \in \mathbb{R}^n$ such that

$$f(y) \geq f(x) + g^T(y - x)$$

for every $y \in \mathbb{R}^n$

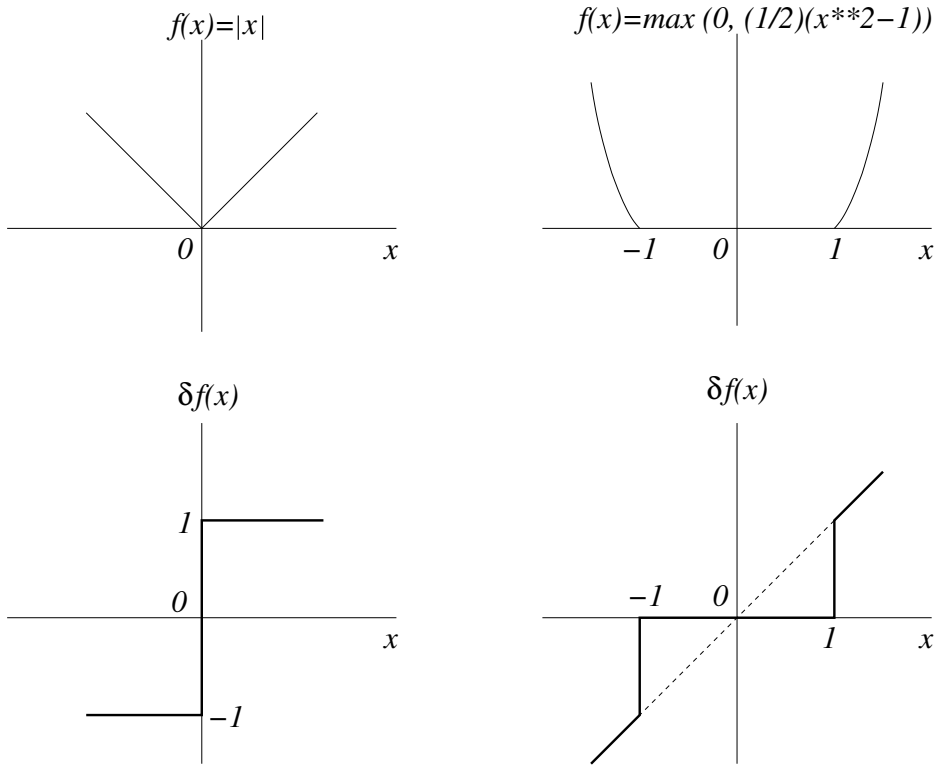


In the case of differentiable convex functions the **subgradient** of a **convex function**, is related to the **ordinary gradient** and the **partial derivatives**

Subgradients of a convex function $f(x)$

For a convex function f it is possible that, at some point $\boldsymbol{x}$, and for all $\boldsymbol{y} \in \mathbb{R}^n$:

1. No vector $\boldsymbol{g} \in \mathbb{R}^n$ satisfying $f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{y} - \boldsymbol{x})$ exists
2. There is a unique vector $\boldsymbol{g} \in \mathbb{R}^n$ satisfying $f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{y} - \boldsymbol{x})$
3. There is more than one vector $\boldsymbol{g} \in \mathbb{R}^n$ satisfying $f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{y} - \boldsymbol{x})$



The set of all subgradients of a convex function f at $\boldsymbol{x}$ is denoted by $\partial f(\boldsymbol{x})$, and is called **subdifferential** of f

Differential properties of convex functions

Some basic properties of subgradients are

- The subdifferential $\partial f(\boldsymbol{x})$ of a convex function f at $\boldsymbol{x}$, is a closed convex set
- The set $\partial f(\boldsymbol{x})$ contains a single vector $\boldsymbol{g} \in \mathbb{R}^n$ if and only if the convex function f is differentiable in the ordinary sense at $\boldsymbol{x}$
- If f is differentiable in the ordinary sense at $\boldsymbol{x}$, then $\boldsymbol{g} = \nabla f(\boldsymbol{x})$, that is

$$g_j = \frac{\partial f(\boldsymbol{x})}{\partial x_j}, \quad j = 1, ..., n$$

- $\boldsymbol{x}^*$ is a minimizer of a convex f if and only if $0 \in \partial f(\boldsymbol{x}^*) \neq \emptyset$

Subgradients can be characterized by the directional derivatives, according to the following theorem:

Theorem

A vector $\boldsymbol{g} \in \mathbb{R}^n$ is a subgradient of a convex function f at a point $\boldsymbol{x}$ where $f(\boldsymbol{x})$ is finite if and only if

$$D^+ f(\boldsymbol{x}; \boldsymbol{z}) \geq \boldsymbol{g}^T \boldsymbol{z}$$

for every direction $\boldsymbol{z}$

Calculus of subgradients

Some properties of subgradients

- *Nonnegative scaling*

$$\partial(\alpha f)(\boldsymbol{x}) = \alpha \partial f(\boldsymbol{x}), \quad \forall \alpha \geq 0$$

- *Addition*

  If $f = f_1 + \ldots + f_m$, with all the f_i convex, then

$$\partial f(\boldsymbol{x}) = \partial f_1(\boldsymbol{x}) + \ldots + \partial f_m(\boldsymbol{x})$$

- *Affine transformation of domain*

  If f is convex, and $h(\boldsymbol{x}) = f(A\boldsymbol{x} + \boldsymbol{b})$, then

$$\partial h(\boldsymbol{x}) = \partial f(A\boldsymbol{x} + \boldsymbol{b}) = A^T \partial f(A\boldsymbol{x} + \boldsymbol{b})$$

- *Pointwise max*

  If $f_1, \ldots, f_m$ are convex, and $f(\boldsymbol{x}) = \max_{i=1,\ldots,m} f_i(\boldsymbol{x})$, then

$$\partial f(\boldsymbol{x}) = \partial \left(\max_{i=1,\ldots,m} f_i(\boldsymbol{x}) \right) = \text{convex hull } \{\partial f_i(\boldsymbol{x}) \mid f_i(\boldsymbol{x}) = f(\boldsymbol{x})\}$$

Calculus of subgradients

Example 1

Consider

$$f(\boldsymbol{x}) = \max_{i=1,\ldots,m} (\boldsymbol{a}_i^T \boldsymbol{x} + \boldsymbol{b}_i)$$

- Let $f_i(\boldsymbol{x}) = \boldsymbol{a}_i^T \boldsymbol{x} + \boldsymbol{b}_i$, then $\partial f_i(\boldsymbol{x}) = \{\boldsymbol{a}_i\}$
- Let

$$\mathcal{J}(\boldsymbol{x}) = \left\{ j \mid \boldsymbol{a}_j^T \boldsymbol{x} + \boldsymbol{b}_j = \max_{i=1,\ldots,m} (a_i^T \boldsymbol{x} + \boldsymbol{b}_i) \right\},$$

  then, according to the pointwise max property

$$\partial f(\boldsymbol{x}) = \text{convex hull} \left\{ \bigcup_{j \in \mathcal{J}(\boldsymbol{x})} \{\boldsymbol{a}_j\} \right\}$$

- In particular, when $\mathcal{J}(\boldsymbol{x}) = \{k\}$ we have $\partial f(\boldsymbol{x}) = \{\boldsymbol{a}_k\}$

Calculus of subgradients

Example 2

Consider the case $f = f_1 + ... + f_n$, for instance

$$f(\boldsymbol{x}) = \|\boldsymbol{x}\|_1 = |x_1| + ... + |x_n| \equiv f_1(\boldsymbol{x}) + ... + f_n(\boldsymbol{x}).$$

Then

$$\begin{aligned} \partial f(\boldsymbol{x}) &= \partial f_1(\boldsymbol{x}) + ... + \partial f_m(\boldsymbol{x}) \\ &= \{\boldsymbol{g} \mid g_i = 1 \text{ if } x_i > 0,\ g_i = -1 \text{ if } x_i < 0,\ g_i \in [-1, 1] \text{ if } x_i = 0\} \end{aligned}$$

Note that

$$\boldsymbol{g} = \text{sign}(\boldsymbol{x}) \in \partial f(\boldsymbol{x})$$

Remark. The signum function of a vector $\boldsymbol{x} = (x_1, x_2, ...x_n)$ is defined as

$$\text{sign}(\boldsymbol{x}) = (\text{sign}(x_1), \text{sign}(x_2), ..., \text{sign}(x_1))$$

where the signum function of a real number x is defined as

$$\text{sign}(x) := \begin{cases} -1 & \text{if } x < 0 \\ 0 & \text{if } x = 0 \\ 1 & \text{if } x > 0 \end{cases}$$

The subgradient methods. General properties

- **Subgradient methods** are a simple algorithms for minimizing a **non-differentiable convex function**
- The methods were developed in the former Soviet Union in the 60's and 70's by Shor and others
- Subgradient methods are used only for problems in which very high accuracy is not required, typically around 10%
- The methods look very much like the ordinary gradient method for differentiable functions, except that
  - The step lengths are not chosen via a line search, as in the ordinary gradient method. In the most common cases, **the step lengths are fixed ahead of time**
  - Unlike the ordinary gradient method, the subgradient method **is not a descent method**; the objective function value can (and often does) increase
- The subgradient methods are readily extended to handle problems with constraints

Subgradient methods. Advantages and disadvantages

- Subgradient methods are **first-order methods** that can be (very) **slow** in convergence. The subgradient methods are far slower than Newton's method, but are much simpler and can be applied to a far wider variety of problems
- Combining a subgradient method with **primal or dual decomposition techniques**, it is sometimes possible to develop a simple **distributed algorithm** for a problem
- Subgradient methods can be used to **decouple or decompose a large problem** into many smaller ones. This has played a significant role in internet optimization, network utility maximization,...
- The **memory requirement** of subgradient methods can be **much smaller than an interior-point method** (penalty method for constrained optimization) or Newton method, which means it **can be used for extremely large problems** for which interior-point or Newton methods cannot be used

The subgradient method

- Let $f : \mathbb{R}^n \to \mathbb{R}$ be a **convex** function
- To minimize f, the **subgradient method** uses the iteration

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \boldsymbol{g}_k$$

  where $\boldsymbol{x}_k$ is the k-th iterate, $\boldsymbol{g}_k$ is any subgradient of f at $\boldsymbol{x}_k$, and $\alpha_k > 0$ is the k-th step size

- At each iteration of the method we take a step in the direction of a negative subgradient
- Recall that a subgradient of f at $\boldsymbol{x}$ is any vector $\boldsymbol{g}$ that satisfies the inequality

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{y} - \boldsymbol{x}), \quad \forall \boldsymbol{y}$$

- When f is differentiable, the only possible choice for $\boldsymbol{g}_k$ is $\nabla f(\boldsymbol{x}_k)$, and the subgradient method then reduces to the gradient method, except for the choice of step size α_k

The subgradient method

- Since the subgradient method is not a descent method, it is common to keep track of the best point found so far, i.e., the one with smallest function value. We take $f_{best}^0 = +\infty$, and at each step, we set

$$f_{best}^k = \min\{f_{best}^{k-1}, f(\boldsymbol{x}_k)\}$$

and

$$i_{best}^k = k \quad \text{if} \quad f(\boldsymbol{x}_k) = f_{best}^k$$

so, $\boldsymbol{x}_k$ is the best point found so far

- Then, we have

$$f_{best}^k = \min\{f(\boldsymbol{x}_1), ..., f(\boldsymbol{x}_n)\}$$

so f_{best}^k is the best objective value found in $n \geq k$ iterations

- Since f_{best}^k is decreasing, it has a limit (which can be $-\infty$)

- In a usual descent method there is no need to recall at each step the "best"point, because the current point is always the best one so far

Step size rules

Several different types of **step size rules** are used

- **Constant step size**. $\alpha_k = h$ is a constant, independent of k
- **Constant step length.** $\alpha_k = \dfrac{h}{\|\boldsymbol{g}_k\|_2}$. Since $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \boldsymbol{g}_k$, this means that $\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\|_2 = h$
- **Square summable but not summable.** The step sizes satisfy

$$\sum_{k=1}^{\infty} \alpha_k^2 < \infty, \qquad \sum_{k=1}^{\infty} \alpha_k = \infty$$

  A typical example is: $\alpha_k = \dfrac{a}{b+k}$, whith $a > 0$ and $b \geq 0$
- **Nonsummable diminishing.** The step sizes satisfy

$$\lim_{k \to \infty} \alpha_k = 0, \qquad \sum_{k=1}^{\infty} \alpha_k = \infty$$

  Step sizes that satisfy this condition are called **diminishing step size rules**. A typical example is: $\alpha_k = \dfrac{a}{\sqrt{k}}$, whith $a > 0$

Example of square summable but not summable series

$$a_k = \frac{1}{k}$$

$$\begin{aligned}
\sum_{k=1}^{\infty} \frac{1}{k} &= 1 + \frac{1}{2} + \left[\frac{1}{3} + \frac{1}{4}\right] + \left[\frac{1}{5} + \frac{1}{6} + \frac{1}{7} + \frac{1}{8}\right] + \cdots \\
&> 1 + \frac{1}{2} + \left[\frac{1}{4} + \frac{1}{4}\right] + \left[\frac{1}{8} + \frac{1}{8} + \frac{1}{8} + \frac{1}{8}\right] + \cdots \\
&= 1 + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \cdots \\
&= \infty
\end{aligned}$$

Example of square summable but not summable series

$$a_k = \frac{1}{k}$$

$$\begin{aligned}
\sum_{k=1}^{\infty} \frac{1}{k^2} &= 1 + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \cdots \\
&\leq 1 + \frac{1}{2 \cdot 1} + \frac{1}{3 \cdot 2} + \frac{1}{4 \cdot 3} + \cdots \\
&= 1 + \left(\frac{1}{1} - \frac{1}{2}\right) + \left(\frac{1}{2} - \frac{1}{3}\right) + \left(\frac{1}{3} - \frac{1}{4}\right) + \cdots \\
&= 1 + 1 - \frac{1}{2} + \frac{1}{2} - \frac{1}{3} + \frac{1}{3} - \frac{1}{4} + \cdots \\
&= 2
\end{aligned}$$

So

$$\sum_{k=1}^{\infty} \frac{1}{k^2} \leq 2$$

Example of square summable but not summable series

From the Fourier expansion of $f(x) = x^2$, $x \in [-1, 1]$:

$$f(x) = x^2 \sim \frac{1}{3} + \sum_{k \geq 1} (-1)^k \frac{4}{\pi^2 k^2} \cos(k\pi x).$$

Evaluating the series for $x = 1$

$$\begin{aligned} 1 &= \frac{1}{3} + \sum_{k \geq 1} (-1)^k \frac{4}{\pi^2 k^2} \cos(k\pi) \\ &= \frac{1}{3} + \sum_{k \geq 1} (-1)^k \frac{4}{\pi^2 k^2} (-1)^k \\ &= \frac{1}{3} + \frac{4}{\pi^2} \sum_{k \geq 1} \frac{1}{k^2}. \end{aligned}$$

So

$$\sum_{k \geq 1} \frac{1}{k^2} = \frac{\pi^2}{6} = 1.644934.... \leq 2$$

Convergence results of the subgradient method

- For **constant step size** and **constant step length**, the subgradient algorithm is guaranteed to **converge to within some range of the optimal value**

$$\lim_{k \to \infty} (f_{best}^k - f^*) < \epsilon$$

where ϵ is a certain number and f^* denotes the optimal value of the problem. The number ϵ is a function of the step size parameter h and decreases with it

This implies that, in these cases, the subgradient method finds an **ϵ-suboptimal point within a finite number of steps**

- When the function f **is differentiable**, the subgradient method with **constant step size** yields **convergence** to the optimal value, **provided the step h is small enough**

- For the **nonsumable diminishing step size rule**, and also for the **square summable but not summable step size rule**, the algorithm is guaranteed to **converge to the optimal value**

$$\lim_{k \to \infty} f_{best}^k = f^*$$

Convergence proof

For the convergence proof, we will assume that

- There is a minimizer of f, say $\boldsymbol{x}^*$
- The norm of the subgradients is bounded, i.e., there is a G such that

$$\|\boldsymbol{g}\|_2 \leq G, \quad \forall \boldsymbol{g} \in \partial f(\boldsymbol{x}) \quad \text{and} \quad \forall \boldsymbol{x}$$

  This is equivalent to assume that f is Lipschitz continuous with constant $G > 0$

$$|f(\boldsymbol{x}) - f(\boldsymbol{y})| \leq G\|\boldsymbol{x} - \boldsymbol{y}\|_2, \quad \forall \boldsymbol{x}, \boldsymbol{y}$$

  (see next slide for the proof)

- The initial point $\boldsymbol{x}_1$ satisfies

$$\|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2 \leq R$$

Recall that for the gradient descent method, the convergence proof is based on the function value decreasing at each step. In the **subgradient method**, the **key quantity** is not the function value (which often increases), it is the **Euclidean distance to the optimal set**

Lipschitz vs bounded equivalence proof

Proof of the equivalence

- Assume $\|\boldsymbol{g}\|_2 \leq G$ for any subgradient at any point. Let $\boldsymbol{g}_x \in \partial f(\boldsymbol{x})$, $\boldsymbol{g}_y \in \partial f(\boldsymbol{y})$, then

$$\boldsymbol{g}_y \in \partial f(\boldsymbol{y}) \;\Rightarrow\; f(\boldsymbol{x}) \geq f(\boldsymbol{y}) + \boldsymbol{g}_y^T(\boldsymbol{x}-\boldsymbol{y}) \;\Rightarrow\; f(\boldsymbol{x}) - f(\boldsymbol{y}) \geq \boldsymbol{g}_y^T(\boldsymbol{x}-\boldsymbol{y})$$

$$\boldsymbol{g}_x \in \partial f(\boldsymbol{x}) \;\Rightarrow\; f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{g}_x^T(\boldsymbol{y}-\boldsymbol{x}) \;\Rightarrow\; f(\boldsymbol{y}) - f(\boldsymbol{x}) \geq \boldsymbol{g}_x^T(\boldsymbol{y}-\boldsymbol{x})$$

$$\Rightarrow \quad f(\boldsymbol{x}) - f(\boldsymbol{y}) \leq \boldsymbol{g}_x^T(\boldsymbol{x}-\boldsymbol{y})$$

So

$$\boldsymbol{g}_x^T(\boldsymbol{x}-\boldsymbol{y}) \geq f(\boldsymbol{x}) - f(\boldsymbol{y}) \geq \boldsymbol{g}_y^T(\boldsymbol{x}-\boldsymbol{y})$$

and by the Cauchy-Schwarz inequality[1]

$$G\|\boldsymbol{x}-\boldsymbol{y}\|_2 \geq |f(\boldsymbol{x})-f(\boldsymbol{y})| \geq -G\|\boldsymbol{x}-\boldsymbol{y}\|_2 \quad \Rightarrow \quad |f(\boldsymbol{x})-f(\boldsymbol{y})| \leq G\|\boldsymbol{x}-\boldsymbol{y}\|_2$$

- Assume $\|\boldsymbol{g}\|_2 > G$ for some $\boldsymbol{g} \in \partial f(\boldsymbol{x})$. Take $\boldsymbol{y} = \boldsymbol{x} + \dfrac{\boldsymbol{g}}{\|\boldsymbol{g}\|_2}$, then

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{y}-\boldsymbol{x}) = f(\boldsymbol{x}) + \|\boldsymbol{g}\|_2 > f(\boldsymbol{x}) + G$$

so $|f(\boldsymbol{x}) - f(\boldsymbol{y})| > G$ (and f cannot be Lipschitz)

[1] $|\mathrm{u}^T\mathrm{v}| \leq \|\mathrm{u}\|\|\mathrm{v}\|$.

The basic inequality

We want to proof the following inequality

$$f_{best}^k - f^* \leq \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2 \sum_{i=1}^k \alpha_i} \leq \frac{R^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2 \sum_{i=1}^k \alpha_i}$$

The basic inequality (cont.)

If x^* is an optimal point, and according to the subgradient method

$$x_{k+1} = x_k - \alpha_k g_k$$

with $g_k \in \partial f(x_k)$, we get

$$\begin{aligned}
\|x_{k+1} - x^*\|_2^2 &= \|x_k - \alpha_k g_k - x^*\|_2^2 \\
&= [(x_k - x^*) - \alpha_k g_k]^T [(x_k - x^*) - \alpha_k g_k] \\
&= \|x_k - x^*\|_2^2 - 2\alpha_k (g_k)^T (x_k - x^*) + \alpha_k^2 \|g^k\|_2^2 \\
&\leq \|x_k - x^*\|_2^2 - 2\alpha_k (f(x_k) - f^*) + \alpha_k^2 \|g^k\|_2^2
\end{aligned}$$

whith $f^* = f(x^*)$, and where we have used, in the last inequality, the definition of subgradient

$$f(x^*) \geq f(x_k) + (g_k)^T (x^* - x_k) \quad \Rightarrow \quad f(x_k) - f(x^*) \leq (g_k)^T (x_k - x^*)$$

The basic inequality (cont.)

Applying the inequality above

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|_2^2 \leq \|\boldsymbol{x}_k - \boldsymbol{x}^*\|_2^2 - 2\alpha_k(f(\boldsymbol{x}_k) - f^*) + \alpha_k^2\|\boldsymbol{g}^k\|_2^2$$

recursively, we have

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|_2^2 \leq \|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2^2 - 2\sum_{i=1}^{k}\alpha_i(f(\boldsymbol{x}_i) - f^*) + \sum_{i=1}^{k}\alpha_i^2\|\boldsymbol{g}_i\|_2^2$$

Using $\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|_2^2 \geq 0$, we have

$$2\sum_{i=1}^{k}\alpha_i(f(\boldsymbol{x}_i) - f^*) \leq \|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2^2 + \sum_{i=1}^{k}\alpha_i^2\|\boldsymbol{g}_i\|_2^2$$

The basic inequality (cont.)

Combining the last inequality

$$2\sum_{i=1}^{k}\alpha_i(f(\boldsymbol{x}_i)-f^*)\leq\|\boldsymbol{x}_1-\boldsymbol{x}^*\|_2^2+\sum_{i=1}^{k}\alpha_i^2\|\boldsymbol{g}_i\|_2^2$$

with

$$2\sum_{i=1}^{k}\alpha_i(f(\boldsymbol{x}_i)-f^*)\geq 2\left(\sum_{i=1}^{k}\alpha_i\right)\min_{i=1,\ldots,k}(f(\boldsymbol{x}_i)-f^*)=2\left(\sum_{i=1}^{k}\alpha_i\right)(f_{best}^k-f^*),$$

we get

$$f_{best}^k-f^*\leq\frac{2\sum_{i=1}^{k}\alpha_i(f(\boldsymbol{x}_i)-f^*)}{2\sum_{i=1}^{k}\alpha_i}\leq\frac{\|\boldsymbol{x}_1-\boldsymbol{x}^*\|_2^2+\sum_{i=1}^{k}\alpha_i^2\|\boldsymbol{g}_i\|_2^2}{2\sum_{i=1}^{k}\alpha_i}.$$

The basic inequality (cont.)

Finally, using the assumption $\|\boldsymbol{g}_k\|_2 \leq G$, we obtain the **basic inequality**

$$f_{best}^k - f^* = \min_{i=1,\ldots,k} (f(\boldsymbol{x}_i) - f^*) \leq \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2 \sum_{i=1}^k \alpha_i} \leq \frac{R^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2 \sum_{i=1}^k \alpha_i}.$$

The **basic inequality** can also be written as

$$f_{best}^k - f^* \leq \frac{\text{dist}(\boldsymbol{x}_1, X^*)^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2 \sum_{i=1}^k \alpha_i}.$$

where X^* denotes the optimal set, and $\text{dist}(\boldsymbol{x}_1, X^*)$ is the Euclidean distance of $\boldsymbol{x}_1$ to the optimal set, which is assumed to be bounded by R.

Convergence of the subgradient method for constant step size h

- If $\alpha_k = h$, we have

$$\sum_{i=1}^{k} \alpha_i = kh, \quad \sum_{i=1}^{k} \alpha_i^2 = kh^2$$

and

$$f_{best}^k - f^* \leq \frac{\text{dist}(x_1, X^*)^2 + G^2 h^2 k}{2hk} = \frac{\text{dist}(x_1, X^*)^2}{2hk} + \frac{G^2 h}{2},$$

and the righthand side converges to $G^2 h/2$ as $k \to \infty$

- Thus, for the subgradient method with fixed step size h, f_{best}^k converges within $G^2 h/2$ of optimal
- We can also say that

$$f(x_k) - f^* \leq G^2 h$$

within a finite number of steps

Convergence of the subgradient method for constant step length $\alpha_k = h/\|\boldsymbol{g}_k\|_2$

- If $\alpha_k = \dfrac{h}{\|\boldsymbol{g}_k\|_2} \geq h/G$, then the basic inequality becomes

$$f_{best}^k - f^* \leq \frac{\text{dist}(\boldsymbol{x}_1, X^*)^2 + h^2 k}{2\sum_{i=1}^k \alpha_i}$$

- By assmpution, we have $\alpha_k \geq h/G$. Applying this to the denominator of the above inequality gives

$$f_{best}^k - f^* \leq \frac{\text{dist}(\boldsymbol{x}_1, X^*)^2 + h^2 k}{2hk/G} = \frac{G\ \text{dist}(\boldsymbol{x}_1, X^*)^2}{2hk} + \frac{Gh}{2}$$

- The righthand side converges to $Gh/2$ as $k \to \infty$, so in this case the subgradient method for constant step length converges to within $Gh/2$ of optimal

Convergence of the subgradient method for square summable but not summable

- In this case we have

$$\sum_{k=1}^{\infty} \alpha_k^2 < \infty, \qquad \sum_{k=1}^{\infty} \alpha_k = \infty$$

- Then, we have

$$f_{best}^k - f^* \leq \frac{\text{dist}(\boldsymbol{x}_1, X^*)^2 + G^2 \sum_{k=1}^{\infty} \alpha_k^2}{2 \sum_{i=1}^{k} \alpha_i}$$

  When $k \to \infty$, the numerator converges to a finite number and the denominator converges to ∞, so $f_{best}^k - f^*$ converges to zero as $k \to \infty$

- In other words, the subgradient method for square summable but not summable stepsize converges:

$$f_{best}^k \to f^*$$

Convergence of the subgradient method for nonsummable diminishing

- In this case we have that

$$\lim_{k\to\infty} \alpha_k = 0, \qquad \sum_{k=1}^{\infty} \alpha_k = \infty$$

- Then the right hand side of the basic inequality

$$f_{best}^k - f^* \leq \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2\sum_{i=1}^k \alpha_i},$$

converges to zero, which implies the subgradient method converges

To proof the convergence of the right hand side of the inequality, let $\epsilon > 0$. Then

- There exists an integer N_1 such that $\alpha_i \leq \epsilon/G^2$, for all $i > N_1$
- There also exists an integer N_2 such that

$$\sum_{i=1}^{k} \alpha_i \geq \frac{1}{\epsilon}\left(\|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2^2 + G^2 \sum_{i=1}^{N_1} \alpha_i^2\right) \quad \text{for all} \quad k > N_2$$

Convergence of the subgradient method for nonsummable diminishing

Let $N = \max(N_1, N_2)$. Then, for all $k > N$, we have

$$\begin{aligned}
\min_{i=1,\ldots,k} (f(\boldsymbol{x}_i) - f^*) &\leq \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2^2 + G^2 \sum_{i=1}^{N_1} \alpha_i^2}{2 \sum_{i=1}^{k} \alpha_i} + \frac{G^2 \sum_{i=N_1+1}^{k} \alpha_i^2}{2 \sum_{i=1}^{N_1} \alpha_i + 2 \sum_{i=N_1+1}^{k} \alpha_i} \\
&\leq \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2^2 + G^2 \sum_{i=1}^{k} \alpha_i^2}{(2/\epsilon)\left(\|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2^2 + G^2 \sum_{i=1}^{N_1} \alpha_i^2\right)} + \frac{G^2 \sum_{i=N_1+1}^{k} (\epsilon/G^2)\alpha_i}{2 \sum_{i=N_1+1}^{k} \alpha_i} \\
&\leq \epsilon/2 + \epsilon/2 = \epsilon
\end{aligned}$$

Stopping criterion

- **The truth:** there really isn't a good stopping criterion for the subgradient method, but...
- One can terminate the iterations when

$$\frac{R^2 + G^2 \sum_{i=1}^{k} \alpha_i^2}{2 \sum_{i=1}^{k} \alpha_i} \leq \epsilon$$

  Take into account that the convergence can be very slow
- Do an optimal choice of α_i to achieve

$$\frac{R^2 + G^2 \sum_{i=1}^{k} \alpha_i^2}{2 \sum_{i=1}^{k} \alpha_i} \leq \epsilon$$

  for smallest k, for instance

$$\alpha_i = \frac{R/G}{\sqrt{k}}$$

  In this case, the minimum number of steps required is (see also Polyak's step length)

$$k = \left(\frac{RG}{\epsilon} \right)^2$$

The projected subgradient method

One extension of the subgradient method is the **projected subgradient method**, which solves the constrained convex optimization problem

$$\begin{cases} \text{minimize} & f(\boldsymbol{x}) \\ \text{subject to} & \boldsymbol{x} \in \mathcal{C} \end{cases}$$

where $\mathcal{C}$ is a convex set

The projected subgradient method is given by

$$\boldsymbol{x}_{k+1} = P(\boldsymbol{x}_k - \alpha_k \boldsymbol{g}_k)$$

where P is the Euclidean projection on $\mathcal{C}$, and $\boldsymbol{g}_k$ is any subgradient of f at $\boldsymbol{x}_k$

All the step size rules described for the subgradient method can be used here, with similar convergence results

The projected subgradient method. Convergence proofs

Let

$$\boldsymbol{z}_{k+1} = \boldsymbol{x}_k - \alpha_k \boldsymbol{g}_k$$

this is, a standard subgradient update before the projection back onto $\mathcal{C}$

As in the subgradient method, we have

$$\begin{aligned} \|\boldsymbol{z}_{k+1} - \boldsymbol{x}^*\|_2^2 &= \|\boldsymbol{x}_k - \alpha_k \boldsymbol{g}_k - x^*\|_2^2 \\ &= \|\boldsymbol{x}_k - \boldsymbol{x}^*\|_2^2 - 2\alpha_k (\boldsymbol{g}_k)^T (\boldsymbol{x}_k - \boldsymbol{x}^*) + \alpha_k^2 \|\boldsymbol{g}_k\|_2^2 \\ &\leq \|\boldsymbol{x}_k - \boldsymbol{x}^*\|_2^2 - 2\alpha_k (f(\boldsymbol{x}_k) - f^*) + \alpha_k^2 \|\boldsymbol{g}_k\|_2^2 \end{aligned}$$

Now, since when we project a point onto $\mathcal{C}$ we move closer to every point in $\mathcal{C}$, and $\boldsymbol{x}_{k+1} = P(\boldsymbol{z}_{k+1})$, we observe that

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|_2 = \|P(\boldsymbol{z}_{k+1}) - \boldsymbol{x}^*\|_2 \leq \|\boldsymbol{z}_{k+1} - \boldsymbol{x}^*\|_2$$

Combining this with the inequality above we get

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|_2^2 \leq \|\boldsymbol{x}_k - \boldsymbol{x}^*\|_2^2 - 2\alpha_k (f(\boldsymbol{x}_k) - f^*) + \alpha_k^2 \|\boldsymbol{g}_k\|_2^2$$

and the convergence proofs proceed exactly as in the ordinary subgradient method

The projected subgradient method when $\mathcal{C}$ is affine

Assume that $\mathcal{C}$ is affine, i.e., $\mathcal{C} = \{\boldsymbol{x} \mid A\boldsymbol{x} = \boldsymbol{b}\}$, where $A \in \mathbb{R}^{n \times m}$ is fat ($n < m$) and full rank

Let us see first that the projection operator is affine, and given by

$$P(\boldsymbol{z}) = \boldsymbol{z} - A^T(AA^T)^{-1}(A\boldsymbol{z} - \boldsymbol{b})$$

The projection of $\boldsymbol{z}$ is a point $P(\boldsymbol{z}) = \boldsymbol{x}$ that minimizes $f(\boldsymbol{x}) = (1/2)\|\boldsymbol{x} - \boldsymbol{z}\|^2$ and satisfies the constraint $A\boldsymbol{x} = \boldsymbol{b}$

To determine $\boldsymbol{x}$ we construct the Lagrangian

$$L(\boldsymbol{x}, \boldsymbol{\lambda}) = \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{z}\|_2^2 + \boldsymbol{\lambda}^T(A\boldsymbol{x} - \boldsymbol{b}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{z})^T(\boldsymbol{x} - \boldsymbol{z}) + \boldsymbol{\lambda}^T(A\boldsymbol{x} - \boldsymbol{b})$$

Equating to zero de gradient of L with respect to $\boldsymbol{x}$ we get

$$\boldsymbol{x} - \boldsymbol{z} + A^T\boldsymbol{\lambda} = 0$$

Multiplying this equation by A, isolating $\boldsymbol{\lambda}$ and substituting again in the equation, we get

$$A\boldsymbol{x} - A\boldsymbol{z} + AA^T\boldsymbol{\lambda} = 0 \;\Rightarrow\; \boldsymbol{\lambda} = (AA^T)^{-1}(A\boldsymbol{z} - \boldsymbol{b}) \;\Rightarrow\; \boldsymbol{x} = \boldsymbol{z} - A^T(AA^T)^{-1}(A\boldsymbol{z} - \boldsymbol{b})$$

The projected subgradient method when $\mathcal{C}$ is affine

Using

$$P(\boldsymbol{z}) = \boldsymbol{z} - A^T(AA^T)^{-1}(A\boldsymbol{z} - \boldsymbol{b})$$

we can write the subgradient update $\boldsymbol{x}_{k+1} = P(\boldsymbol{x}_k - \alpha_k \boldsymbol{g}_k)$ as

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \boldsymbol{g}_k - A^T(AA^T)^{-1}(A\boldsymbol{x}_k - \alpha_k A\boldsymbol{g}_k - \boldsymbol{b}) = \boldsymbol{x}_k - \alpha_k (I - A^T(AA^T)^{-1}A)\boldsymbol{g}_k$$

where we have used $A\boldsymbol{x}_k = \boldsymbol{b}$

The projected subgradient method when $\mathcal{C}$ is affine. Example

Consider the least l_1-norm problem

$$\begin{cases} \text{minimize} & \|\boldsymbol{x}\|_1 = \sum_{i=1}^{n} |x_i| \\ \text{subject to} & A\boldsymbol{x} = \boldsymbol{b} \end{cases}$$

whith $\boldsymbol{x} = (x_1, ..., x_m)^T \in \mathbb{R}^m$, $A \in \mathbb{R}^{n \times m}$ and $\boldsymbol{b} \in \mathbb{R}^n$

Remark: This problem can also be solved using linear programming

Assume that A is fat ($n < m$) and full rank, $rank(A) = n$

As we have already seen, the subgradient of the objective function at $\boldsymbol{x}$ is given by $\boldsymbol{g} = \text{sign}(\boldsymbol{x})$, where $g_i = -1$ if $x_i < 0$, $g_i \in [-1, 1]$ if $x_i = 0$, and $g_i = +1$ if $x_i > 0$

Thus, the projected subgradient update is

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k (I - A^T (AA^T)^{-1} A) \, \text{sign}(\boldsymbol{x}_k)$$

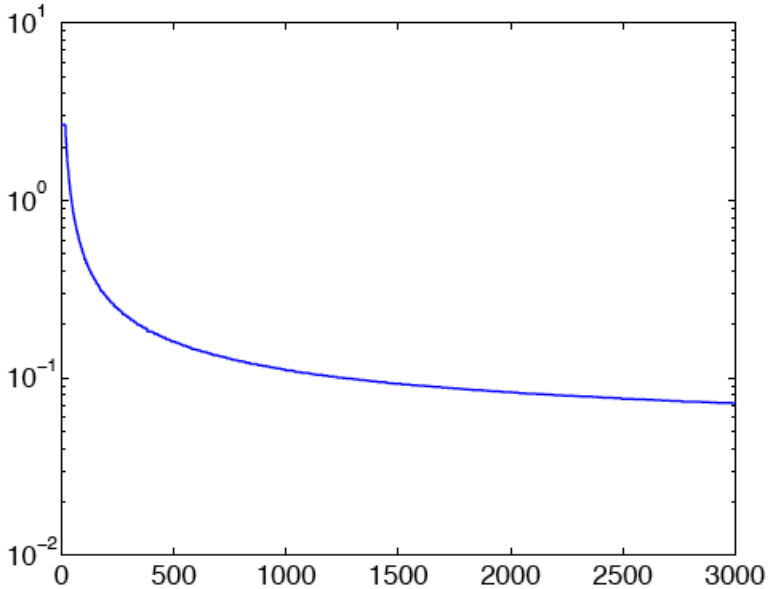
The projected subgradient method when $\mathcal{C}$ is affine. Numerical example

Consider the above problem with $n = 50$ and $m = 1000$, with randomly generated A and $\boldsymbol{b}$

We use the least-norm solution as the starting point:

$$\boldsymbol{x}^1 = A^T(AA^T)^{-1}\boldsymbol{b}$$

The value of $f^* \approx 3.2$ has been computed using linear programming



The figure shows the progress of the projected subgradient method, $f^k_{best} - f^*$ vs k, with the Polyak estimated step size rule $\gamma_k = 100/k$ (to be explained later)

Piecewise linear minimization

Consider the following problem

$$\text{minimize} \quad f(x) = \max_{i=1,\ldots,m} (a_i^T x + b_i)$$

As we have already seen (Example 1, page 8) , finding a subgradient of f is easy: given x, we first find an index j for which

$$a_j^T x + b_j = \max_{i=1,\ldots,m} (a_i^T x + b_i)$$

Then, we can take as subgradient $g = a_j$, and $G = \max_{i=1,\ldots,m} \|a_i\|_2$

The subgradient method update has the form

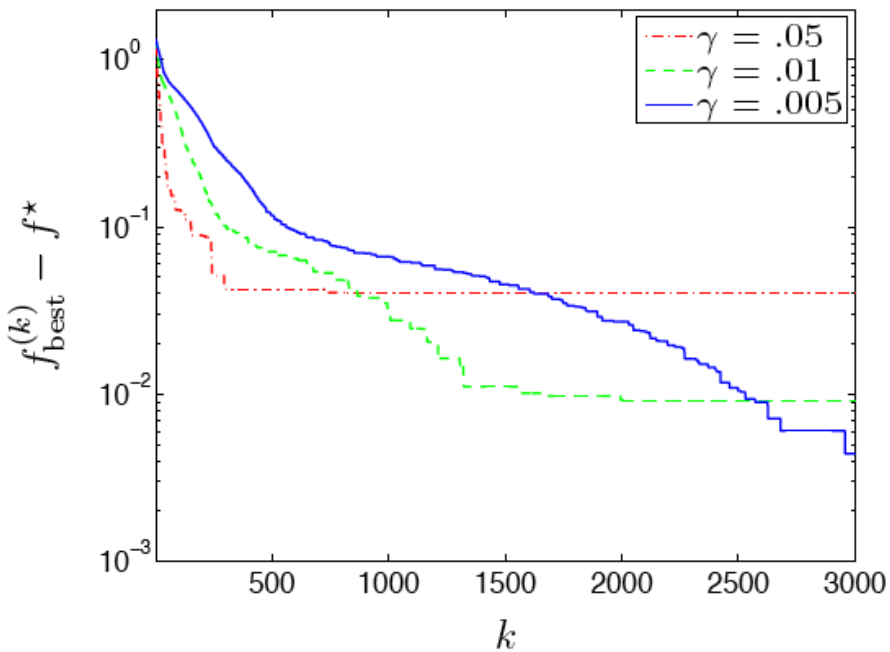
$$x_{k+1} = x_k - \alpha_k a_j$$

Note that to apply the subgradient method, all we need is a way to evaluate $\max_{i=1,\ldots,m}(a_i^T x + b_i)$ to find the right value of j

Piecewise linear minimization. Example

Example: minimize $f(\boldsymbol{x}) = \max_{i=1,\ldots,m}(\boldsymbol{a}_i^T \boldsymbol{x} + \boldsymbol{b}_i)$
with $m = 100$ terms and $n = 20$ variables: $\boldsymbol{a}_i, \boldsymbol{x} \in \mathbb{R}^{20}$, $\boldsymbol{b}_i \in \mathbb{R}^{100}$

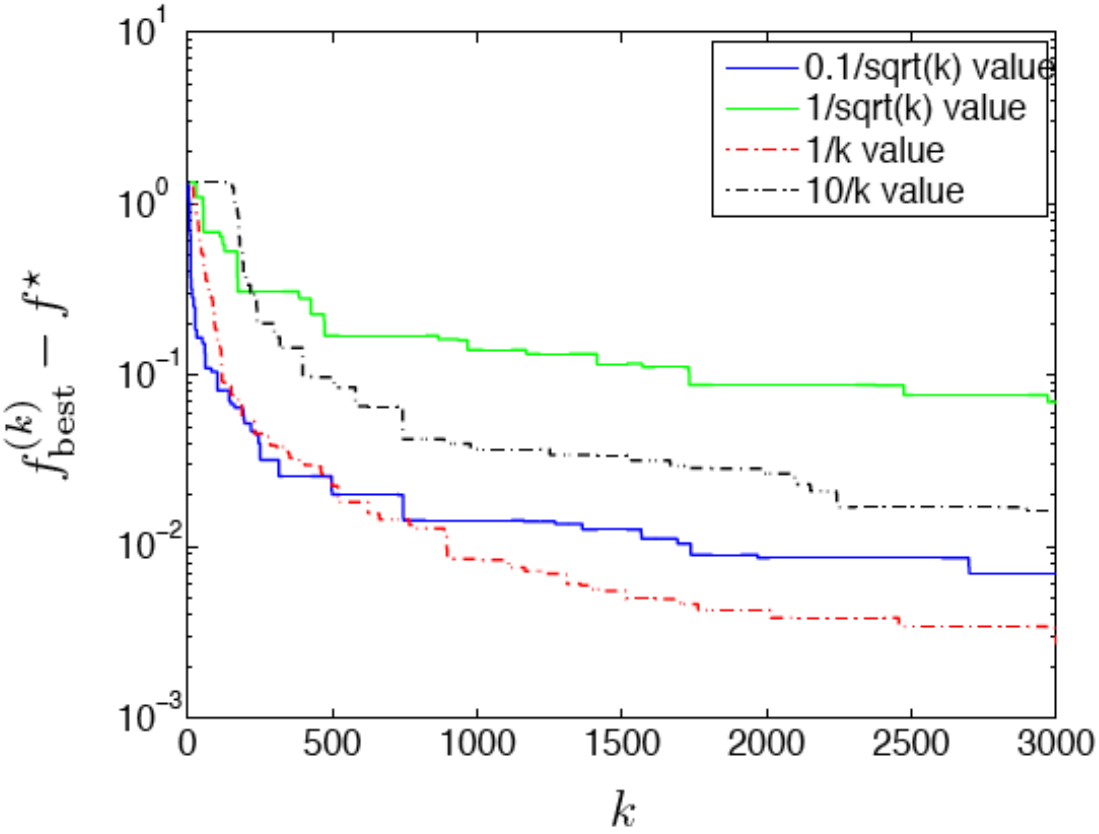
The values of $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$ are chosen randomly. There is no simple way to find a justifiable value for R (a value of R for which we can prove that $\|\boldsymbol{x}_k - \boldsymbol{x}*\|_2 \leq R$), and the value $R = 10$ has been used (in the numerical example: $f^* \approx 1.1$ and $R = 0.91$)



Results for constant step length $\gamma = 0.05, 0.01, 0.005$. Larger γ gives faster convergence but larger final suboptimality

Piecewise linear minimization. Example

The subgradient method is very slow!



Results for dimishing step rules $\alpha_k = 0.1/\sqrt{k}$, $1/\sqrt{k}$, *and square summable step size rules* $\alpha_k = 1/k$, $10/k$

Polyak's step length

Polyak suggests a step size that can be used when the optimal value f^* is known (or when optimal value is estimated), and is in some sense optimal

One can think that f^* is rarely known, but we will see that's not the case

The step size is

$$\alpha_k = \frac{f(\boldsymbol{x}_k) - f^*}{\|\boldsymbol{g}_k\|_2^2}$$

To motivate this step size recall that the subgradient method starts from the basic inequality

$$0 \leq \|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|_2^2 \leq \|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2^2 - 2\sum_{i=1}^{k} \alpha_i (f(\boldsymbol{x}_i) - f^*) + \sum_{i=1}^{k} \alpha_i^2 \|\boldsymbol{g}_i\|_2^2, \quad \Rightarrow$$

$$\Rightarrow \quad 2\sum_{i=1}^{k} \alpha_i (f(\boldsymbol{x}_i) - f^*) \leq \|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2^2 + \sum_{i=1}^{k} \alpha_i^2 \|\boldsymbol{g}_i\|_2^2$$

and we choose α_k to minimize the righthand side.

Polyak's step length

To analyze convergence, we substitute the value of the step size $\alpha_k = (f(\boldsymbol{x}_k) - f^*)/\|\boldsymbol{g}_k\|_2^2$ into the **basic inequality**

$$2\sum_{i=1}^{k} \alpha_i (f(\boldsymbol{x}_i) - f^*) \leq \|\boldsymbol{x}_1 - \boldsymbol{x}^*\|_2^2 + \sum_{i=1}^{k} \alpha_i^2 \|\boldsymbol{g}_i\|_2^2$$

to get

$$2\sum_{i=1}^{k} \frac{(f(\boldsymbol{x}_i) - f^*)^2}{\|\boldsymbol{g}_i\|_2^2} \leq R^2 + \sum_{i=1}^{k} \frac{(f(\boldsymbol{x}_i) - f^*)^2}{\|\boldsymbol{g}_i\|_2^2}$$

So

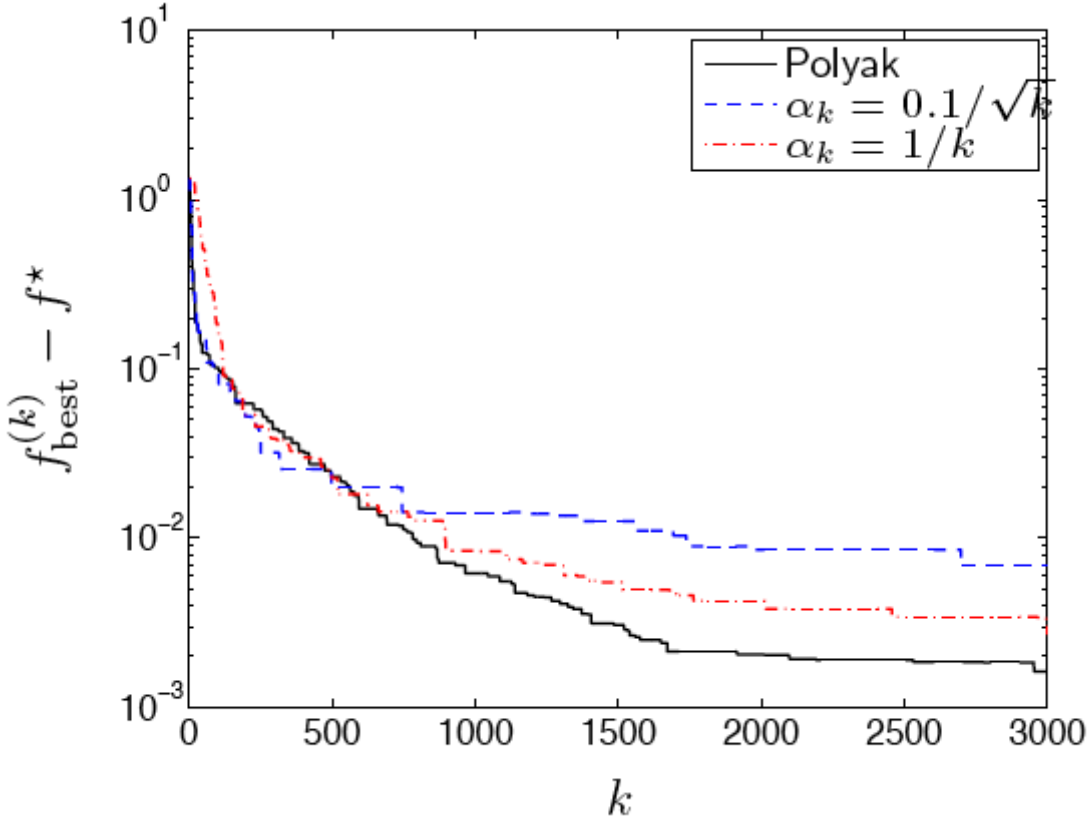
$$\sum_{i=1}^{k} \frac{(f(\boldsymbol{x}_i) - f^*)^2}{\|\boldsymbol{g}_i\|_2^2} \leq R^2$$

Using that $\|\boldsymbol{g}_k\|_2^2 \leq G$, we get

$$\sum_{i=1}^{k} (f(\boldsymbol{x}_i) - f^*)^2 \leq R^2 G^2$$

We conclude that the number of steps needed before we can guarantee suboptimality ϵ is $k = (RG/\epsilon)^2$

Polyak's step length. Example



The figure shows the progress of the subgradient method with Polyak's step size for the same piece-wise linear example

Of course this isn't fair, since we don't know f^* before solving the problem; but even with this unfair advantage in choosing step lengths, the subgradient method is slow

Polyak's step size choice with estimated f^*

The **basic idea** is to estimate the optimal value f^*, as $f^i_{best} - \gamma_i$, where $\gamma_i > 0$, $\gamma_i \to 0$

This gives as step size

$$\alpha_i = \frac{f(\boldsymbol{x}_i) - f^i_{best} + \gamma_i}{\|\boldsymbol{g}_i\|_2^2}$$

Note that γ_k has a simple interpretation: it's our estimate of how suboptimal the current point is

We will also need that $\sum_{i=1}^{\infty} \gamma_i = \infty$, then we have $f^i_{best} \to f^*$

Polyak's step size choice with estimated f^*

To proof this, we consider again the basic inequality

$$2\sum_{i=1}^{k}\alpha_i(f(\boldsymbol{x}_i)-f^*)\leq\|\boldsymbol{x}_1-\boldsymbol{x}^*\|_2^2+\sum_{i=1}^{k}\alpha_i^2\|\boldsymbol{g}_i\|_2^2\leq R^2+\sum_{i=1}^{k}\alpha_i^2\|\boldsymbol{g}_i\|_2^2$$

Substituting the value of α_i, we get

$$\begin{aligned}
R^2 &\geq \sum_{i=1}^{k}\left(2\alpha_i(f(\boldsymbol{x}_i)-f^*)-\alpha_i^2\|\boldsymbol{g}_i\|_2^2\right)\\
&= \sum_{i=1}^{k}\frac{2(f(\boldsymbol{x}_i)-f_{best}^i+\gamma_i)(f(\boldsymbol{x}_i)-f^*)-(f(\boldsymbol{x}_i)-f_{best}^i+\gamma_i)^2}{\|\boldsymbol{g}_i\|_2^2}\\
&= \sum_{i=1}^{k}\frac{(f(\boldsymbol{x}_i)-f_{best}^i+\gamma_i)\left[(f(\boldsymbol{x}_i)-f^*)+(f_{best}^i-f^*)-\gamma_i)\right]}{\|\boldsymbol{g}_i\|_2^2} \qquad (1)
\end{aligned}$$

Polyak's step size choice with estimated f^* (cont.)

Now we can prove convergence. Suppose $f^i_{best} - f^* \geq \epsilon > 0$. Then for $i = 1, ..., k$, $f(\boldsymbol{x}_i) - f^* \geq \epsilon$. Find N for which $\gamma_i \leq \epsilon$ for $i \geq N$ (we assume $k \geq N$). This implies the second term in the numerator is at least ϵ

$$(f(\boldsymbol{x}_i) - f^*) + (f^i_{best} - f^*) - \gamma_i \geq \epsilon$$

In particular it is positive, and so the terms in the sum in (1) for $i \geq N$ are positive. Let S denote the sum up to $i = N - 1$. We then have

$$\sum_{i=N}^{k} \frac{(f(\boldsymbol{x}_i) - f^i_{best} + \gamma_i)\left((f(\boldsymbol{x}_i) - f^*) + (f^i_{best} - f^*) - \gamma_i\right)}{\|\boldsymbol{g}_k\|_2^2} \leq R^2 - S$$

We get a lower bound on the left hand side using and $\|\boldsymbol{g}_k\|_2^2 \leq G^2$ and

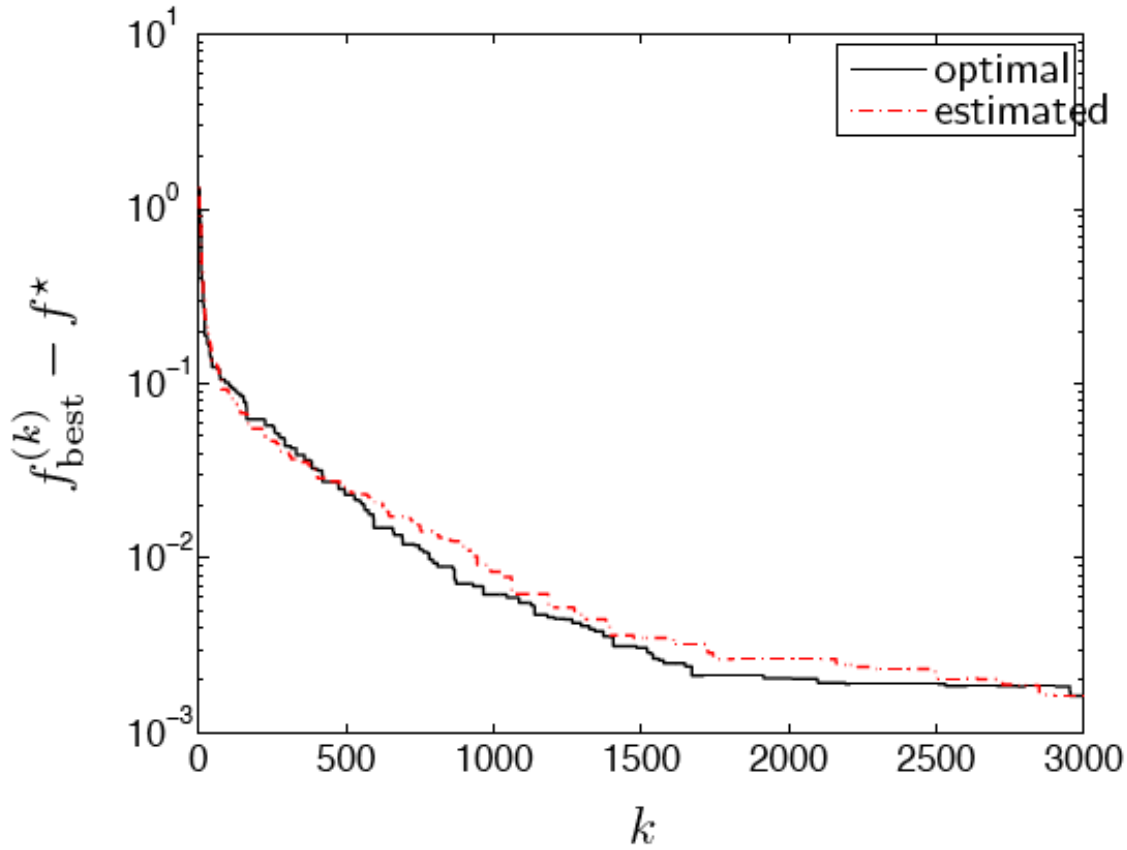
$$f(\boldsymbol{x}_i) - f^i_{best} + \gamma_i \geq \gamma_i$$

along with the inequality above we get

$$\frac{\epsilon}{G^2} \sum_{i=N}^{k} \gamma_i \leq R^2 - S$$

Since the lefthand side converges to ∞ when $k \to \infty$, and righthand side doesn't depend on k, we see that k cannot be too large. So, $f(\boldsymbol{x}_k) - f^* < \epsilon$, if k is large enough

Polyak's step length. Example



Value of $f_{best}^{i} - f^{*}$ versus iteration number k, for the subgradient method with Polyak's step size (solid black line) and the estimated optimal step size (dashed red line)

Finding a point in the intersection of convex sets

Suppose we want to find a point in

$$C = C_1 \cap ... \cap C_m$$

where $C_1, ..., C_m \subseteq \mathbb{R}^n$ are closed and convex, and we assume that C is nonempty

We can do this by **minimizing** the function

$$f(x) = \max\{\text{dist}(x, C_1), ..., \text{dist}(x, C_m)\}$$

which is convex, and has minimum value $f^* = 0$ (since C is nonempty)

Let us see how to find a subgradient g of f at x

Finding a point in the intersection of convex sets

Computation of a subgradient $\boldsymbol{g}$ of

$$f(\boldsymbol{x}) = \max\{\text{dist}(\boldsymbol{x}, C_1), ..., \text{dist}(\boldsymbol{x}, C_m)\}$$

at $\boldsymbol{x}$

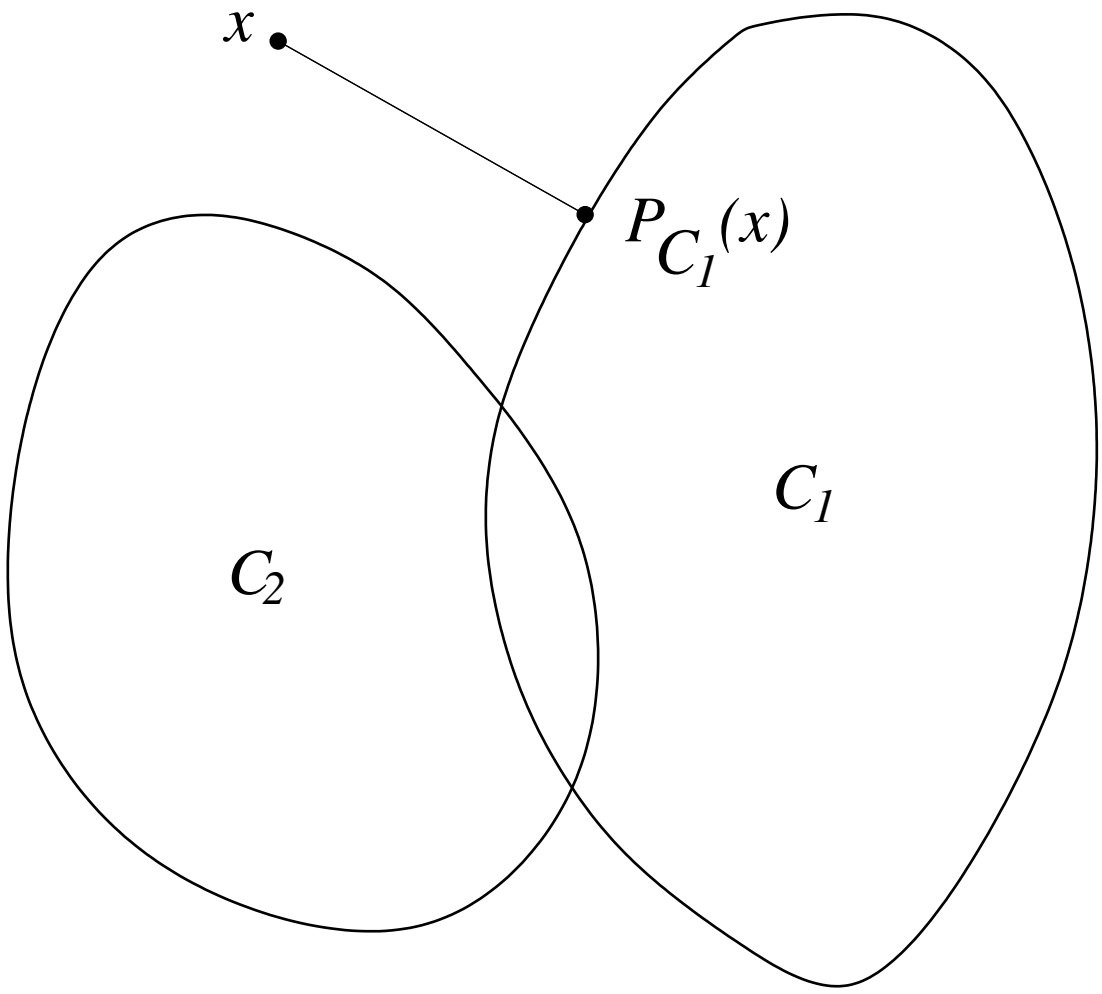
- If $f(\boldsymbol{x}) = 0$, we can take $\boldsymbol{g} = 0$ (which in any case means we are done)
- Otherwise find an index j such that $\text{dist}(\boldsymbol{x}, C_j) = f(\boldsymbol{x})$, i.e., find a set C_j that has maximum distance to $\boldsymbol{x}$, then a subgradient of f is

$$\boldsymbol{g} = \nabla \text{dist}(\boldsymbol{x}, C_j) = \frac{\boldsymbol{x} - P_{C_j}(\boldsymbol{x})}{\|\boldsymbol{x} - P_{C_j}(\boldsymbol{x})\|_2}$$

  where P_{C_j} is Euclidean projection onto C_j

- Note that $\|\boldsymbol{g}\|_2 = 1$, so we can take $G = 1$

Finding a point in the intersection of convex sets



Here

$$f(\boldsymbol{x}) = \max\{\text{dist}(\boldsymbol{x}, C_1), \text{dist}(\boldsymbol{x}, C_2)\}$$

The index j is such that $\text{dist}(\boldsymbol{x}, C_j) = f(\boldsymbol{x})$ is $j = 1$, i.e., the set C_1 that has maximum distance to $\boldsymbol{x}$

Finding a point in the intersection of convex sets

The subgradient algorithm update, with step size rule

$$\alpha_k = \frac{f(\boldsymbol{x}_k) - f^k_{best} + \gamma_k}{\|\boldsymbol{g}_k\|_2^2}$$

and assuming that the index j is one for which $\boldsymbol{x}_k$ has maximum distance to C_j, is given by

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k - \alpha_k \boldsymbol{g}_k \\ &= \boldsymbol{x}_k - f(\boldsymbol{x}_k) \frac{\boldsymbol{x}_k - P_{C_j}(\boldsymbol{x}_k)}{\|\boldsymbol{x}_k - P_{C_j}(\boldsymbol{x}_k)\|_2} \\ &= P_{C_j}(\boldsymbol{x}_k) \end{aligned}$$

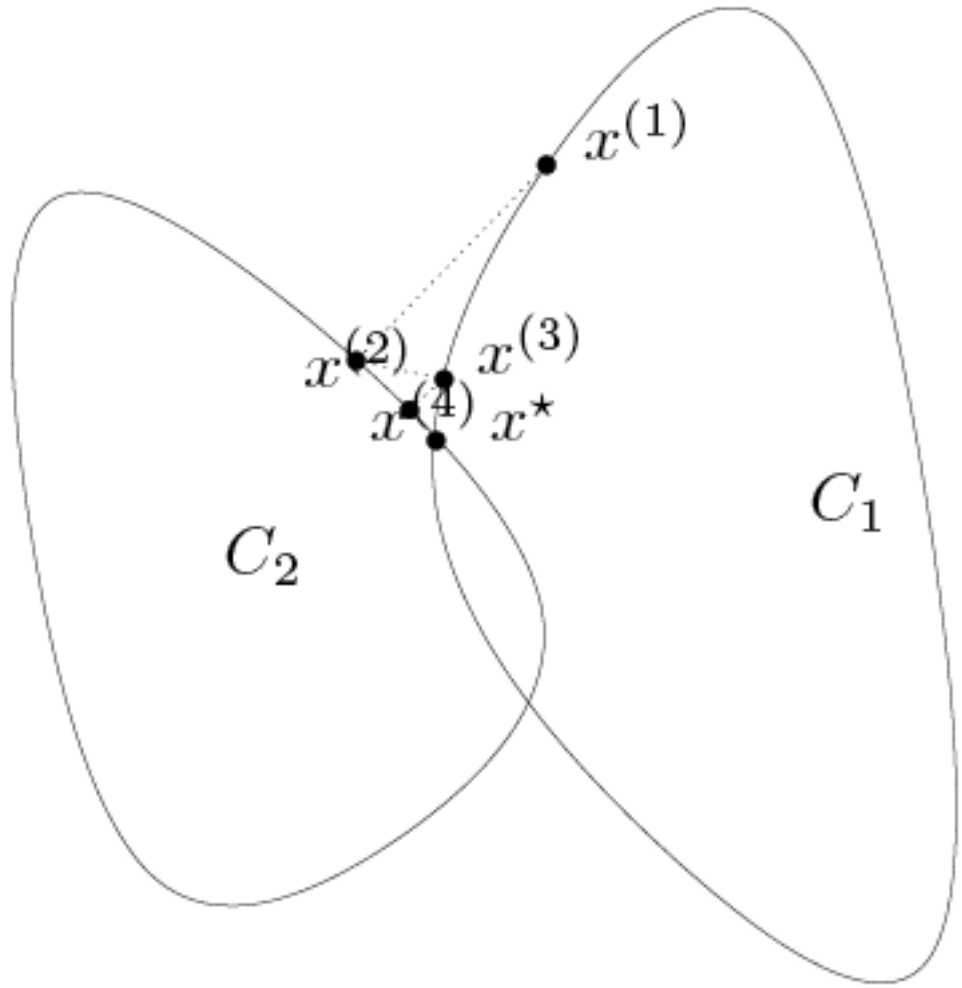
We have used: $\|\boldsymbol{g}_k\|_2 = 1$, $f^* = 0$, and

$$f(\boldsymbol{x}_k) = \text{dist}(\boldsymbol{x}, C_j) = \|\boldsymbol{x}_k - P_{C_j}(\boldsymbol{x}_k)\|_2$$

The algorithm is very simple: at each step, we simply project the current point onto the farthest set

This is an extension of the alternating projections algorithm. (When there are just two sets, then at each step you project the current point onto the other set. Thus the projections simply alternate)

Alternating projections. Example



First few iterations of the gradient method that, eventually, converge to a point $x^* \in C_1 \cap C_2$

Subgradient method for inequality constrained optimization

The subgradient algorithm can be extended to solve the inequality constrained problem

$$\begin{array}{ll} \text{minimize} & f_0(\boldsymbol{x}) \\ \text{subject to} & f_i(\boldsymbol{x}) \leq 0, \quad i = 1, \ldots, m \end{array}$$

where f_i are convex. The algorithm takes the same form

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \boldsymbol{g}_k$$

where $\alpha_k > 0$ is a step size, and $\boldsymbol{g}_k$ is a subgradient of the objective or one of the constraint functions at $\boldsymbol{x}_k$. More specifically, we take

$$\boldsymbol{g}_k \in \begin{cases} \partial f_0(\boldsymbol{x}_k), & \text{if } f_i(\boldsymbol{x}_k) \leq 0, \quad i = 1, \ldots, m \\ \partial f_j(\boldsymbol{x}_k), & \text{if } f_j(\boldsymbol{x}_k) > 0 \end{cases}$$

In other words

- if the current point is feasible, we use an objective subgradient, as if the problem were unconstrained
- if the current point is infeasible, we choose any violated constraint, and use a subgradient of the associated constraint function

Subgradient method for inequality constrained optimization

As in the basic subgradient method, we keep track of the best (feasible) point found so far:

$$f_{best}^k = \min\{f_0(x_i) \mid x_i \text{ feasible}, \ i = 1, ..., k\}$$

If none of the points $x_1, ..., x_k$ is feasible, then $f_{best}^k = \infty$

We assume that

- The problem is strictly feasible: there is some point x^{sf} with $f_i(x^{sf}) < 0$, $i = 1, ..., m$
- The problem has an optimal point x^*
- There are numbers R and G with $\|x_1 - x^*\|_2 \leq R$, $\|x^{sf} - x^*\|_2 \leq R$ and $\|g_k\|_2 \leq G$ for all k

We willproof convergence of the generalized subgradient method using diminishing nonsummable α_k. (Similar results can be obtained for other step size rules)

We claim that $f_{best}^k \to f^*$ as $k \to \infty$. This implies in particular that we obtain a feasible iterate within some finite number of steps

Subgradient method for inequality constrained optimization. Convergence

- Assume that $f_{best}^k \to f^*$ does not occur. Then there exists some $\epsilon > 0$ so that $f_{best}^k \geq f^* + \epsilon$ for all k, which in turn means that $f(\boldsymbol{x}_k) \geq f^* + \epsilon$ for all k for which $\boldsymbol{x}_k$ is feasible. We'll show this leads to a contradiction

- First, we need to find a point $\tilde{\boldsymbol{x}}$ and positive number μ that satisfy

$$f_0(\tilde{\boldsymbol{x}}) \leq f^* + \epsilon/2, \quad f_1(\tilde{\boldsymbol{x}}) \leq -\mu, ..., f_m(\tilde{\boldsymbol{x}}) \leq -\mu$$

  Such a point is $(\epsilon/2)$-suboptimal, and also satisfies the constraints with a margin of μ

- Taking $\tilde{\boldsymbol{x}} = (1-\theta)\boldsymbol{x}^* + \theta \boldsymbol{x}^{sf}$, where $\theta \in (0,1)$, we get

$$f_0(\tilde{\boldsymbol{x}}) \leq (1-\theta) f^* + \theta f_0(\boldsymbol{x}^{sf})$$

  so, if we choose $\theta = \min\{1, (\epsilon/2)/(f_0(\boldsymbol{x}^{sf}) - f^*)\}$, we have

$$f_0(\tilde{\boldsymbol{x}}) \leq f^* + \epsilon/2.$$

- We have

$$f_i(\tilde{\boldsymbol{x}}) \leq (1-\theta) f_i(\boldsymbol{x}^*) + \theta f_i(\boldsymbol{x}^{sf}) \leq \theta f_i(\boldsymbol{x}^{sf})$$

  so, we can take

$$\mu = -\theta \min_i f_i(\boldsymbol{x}^{sf})$$

Subgradient method for inequality constrained optimization. Convergence (cont.)

- Consider any index $i \in \{1, ..., k\}$ for which x_i is feasible. Then we have $g_i \in \partial f_0(x_i)$ and also $f_0(x^i) \geq f^* + \epsilon$
  Since $\tilde{x}$ is $(\epsilon/2)$-suboptimal, we have $f_0(x_i) - f_0(\tilde{x}) \geq \epsilon/2$. Therefore

$$\begin{aligned} \|x_{i+1} - \tilde{x}\|_2^2 &= \|x_i - \tilde{x}\|_2^2 - 2\alpha_i (g_i)^T (x_i - \tilde{x}) + \alpha_i^2 \|g_k\|_2^2 \\ &\leq \|x_i - \tilde{x}\|_2^2 - 2\alpha_i (f_0(x_i) - f_0(\tilde{x}))) + \alpha_i^2 \|g_k\|_2^2 \\ &\leq \|x_i - \tilde{x}\|_2^2 - \alpha_i \epsilon + \alpha_i^2 \|g_k\|_2^2 \end{aligned}$$

  In the second line here we have used the usual subgradient inequality

$$f_0(\tilde{x}) \geq f_0(x_i) + (g_i)^T (\tilde{x} - x_i)$$

- Now suppose that $i \in \{1, ..., k\}$ is such that x^i is infeasible, and that $g_i \in \partial f_p(x_i)$ where $f_p(x_i) > 0$. Since $f_p(\tilde{x}) \leq -\mu$, we have $f_p(x_i) - f_p(\tilde{x}) \geq \mu$. Therefore

$$\begin{aligned} \|x_{i+1} - \tilde{x}\|_2^2 &= \|x_i - \tilde{x}\|_2^2 - 2\alpha_i (g_i)^T (x_i - \tilde{x}) + \alpha_i^2 \|g_k\|_2^2 \\ &\leq \|x_i - \tilde{x}\|_2^2 - 2\alpha_i (f_p(x_i) - f_p(\tilde{x}))) + \alpha_i^2 \|g_k\|_2^2 \\ &\leq \|x_i - \tilde{x}\|_2^2 - 2\alpha_i \mu + \alpha_i^2 \|g_k\|_2^2 \end{aligned}$$

Subgradient method for inequality constrained optimization. Convergence (cont.)

- $$\|\boldsymbol{x}_{i+1} - \tilde{\boldsymbol{x}}\|_2^2 \leq \|\boldsymbol{x}_i - \tilde{\boldsymbol{x}}\|_2^2 - \alpha_i \delta + \alpha_i^2 \|\boldsymbol{g}_k\|_2^2$$

  where $\delta = \min\{\epsilon, 2\mu\}$. Applying this inequality recursively for $i = 1, ..., k$, we get

  $$\|\boldsymbol{x}_{k+1} - \tilde{\boldsymbol{x}}\|_2^2 \leq \|\boldsymbol{x}_1 - \tilde{\boldsymbol{x}}\|_2^2 - \delta \sum_{i=1}^{k} \alpha_i + \sum_{i=1}^{k} \alpha_i^2 \|\boldsymbol{g}_k\|_2^2$$

- It follows that

  $$\delta \sum_{i=1}^{k} \alpha_i \leq R^2 + G^2 \sum_{i=1}^{k} \alpha_i^2$$

  which cannot hold for large k since

  $$\frac{R^2 + G^2 \sum_{i=1}^{k} \alpha_i^2}{\sum_{i=1}^{k} \alpha_i}$$

  converges to zero as $k \to \infty$

Subgradient method for inequality constrained optimization. Numerical example

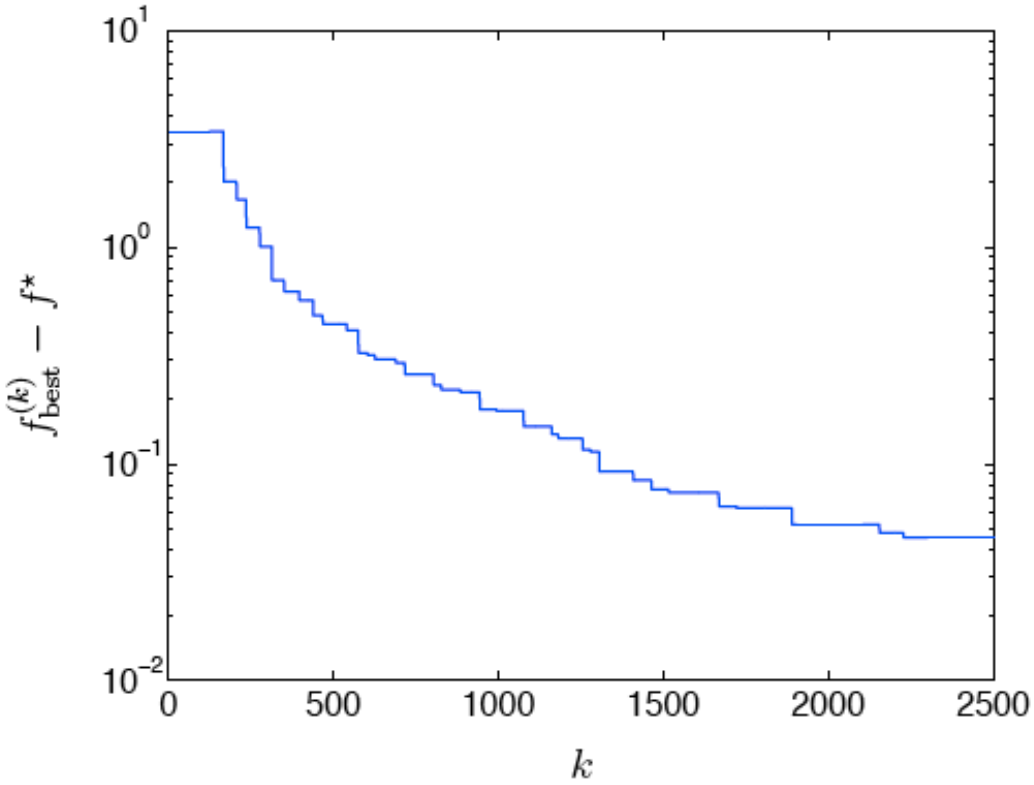
Consider the linear problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, m \end{array}$$

with $x \in \mathbb{R}^n$

- The objective and constraint functions are affine, and so have only one subgradient, independent of x
- For the objective function we have $g = c$, and for the i-th constraint we have $g_i = a_i$
- We solve the problem with $n = 20$ and $m = 200$ using the subgradient method
- The value of $f^* \approx -3.4$ is obtained by other means

Subgradient method for inequality constrained optimization. Numerical example



The figure shows the value of $f_{best}^k - f^*$ versus the iteration number k of the subgradient method using the square summable step size with $\alpha_k = 1/k$ for the optimality update. The objective value only changes for the iterations when $\mathbf{x}_k$ is feasible