Chapter 5: Subgradient

Recall gradient descent

$$\min_{x} f(x)$$

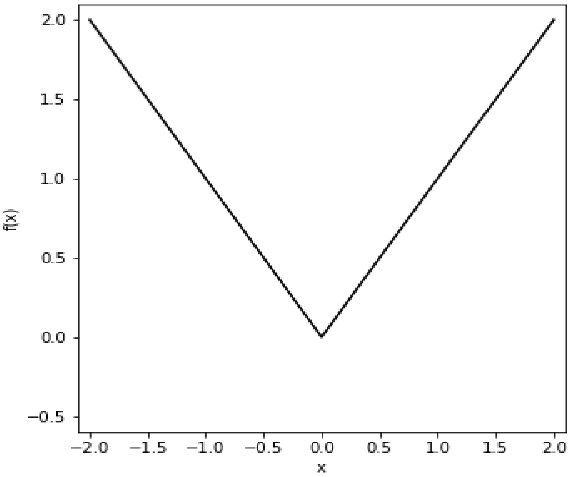
where f is convex and differentiable, $\operatorname{dom}(f) = \mathbb{R}^n$.
Review of **Gradient Descent**

- Choose initial $x^{(0)} \in \mathbb{R}^n$
- Repeat:

$$x^{(k)} = x^{(k-1)} - t_k \cdot \nabla f(x^{(k-1)})$$

where step size t_k is chosen to be fixed and small.

However, not every function is differentiable on its domain.



Subgradient

Recall for convex and differentiable f,

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

for all x, y.

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

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

for all y.

> - Always exists.
> - If f differentiable at x, then $g = \nabla f(x)$ uniquely.
> - Same defination works for nonconvex f if $f(y) \geq f(x) + g^T(y-x)$ for all y in a local neighborhood of x.

Subdifferential: the set of subgradients

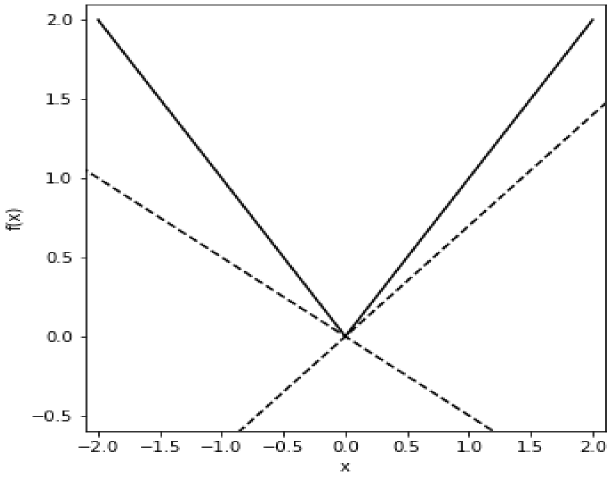
Set of all subgradients of convex f is called the **subdifferential**.

$$\partial f(x) = \{g \in \mathbb{R}^n : g \text{ is a subgradient of } f \text{ at } x\}$$

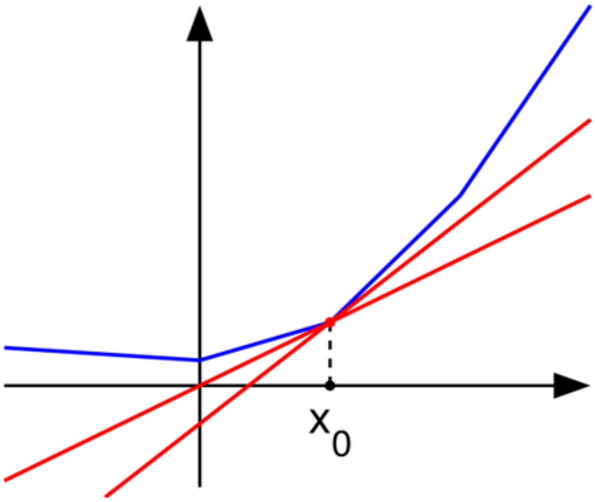
> - Nonempty (For convex f).
> - $\partial f(x)$ is closed and convex (even for non convex f).
> - If f is differentiable and convex at x, then $\partial f(x) = \{\nabla f(x)\}$.
> - If f is convex, $\partial f(x) = \{g\}$, then f is a differentiable at x and $\nabla f(x) = g$
> - If $n = 1$, then $\partial f = [a, b]$ a closed interval, where $a = \lim\limits_{x \to x_0^-} \frac{f(x) - f(x_0)}{x - x_0}$ and $b = \lim\limits_{x \to x_0^+} \frac{f(x) - f(x_0)}{x - x_0}$.

Examples of subgradients

Consider $f : \mathbb{R} \to \mathbb{R}$, $f(x) = |x|$.

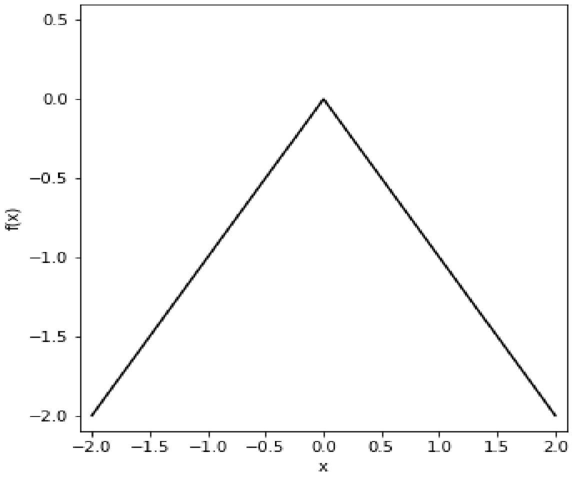


> - For $x \neq 0$, unique subgradient $g = \text{sign}(x)$.
> - For $x = 0$, subdifferential $\partial f(x) = [-1, 1]$.



> A convex function (blue) and subgradients at x_0 (red).

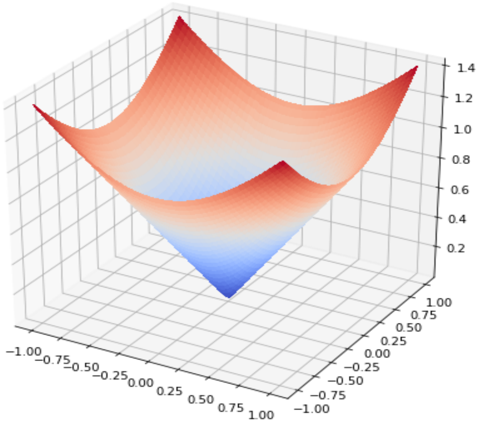
What about the subgradient of (non-convex) $f : \mathbb{R} \to \mathbb{R}$, $f(x) = -|x|$.



> The subdifferential of $-|x|$ at 0 is $\varnothing$.

L_2 norm

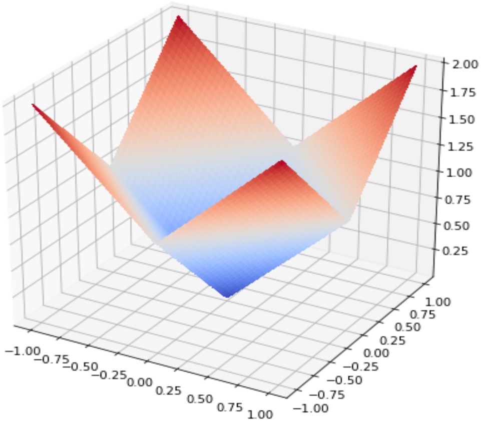
Consider $f : \mathbb{R}^n \to \mathbb{R}$, $f(x) = \|x\|_2$.



> - For $x \neq 0$, unique subgradient $g = \frac{x}{\|x\|_2}$.
> - For $x = 0$, subgradient g is any element of $\{x : \|x\|_2 \leq 1\}$.
>   *Proof*: homework.

L_1 norm

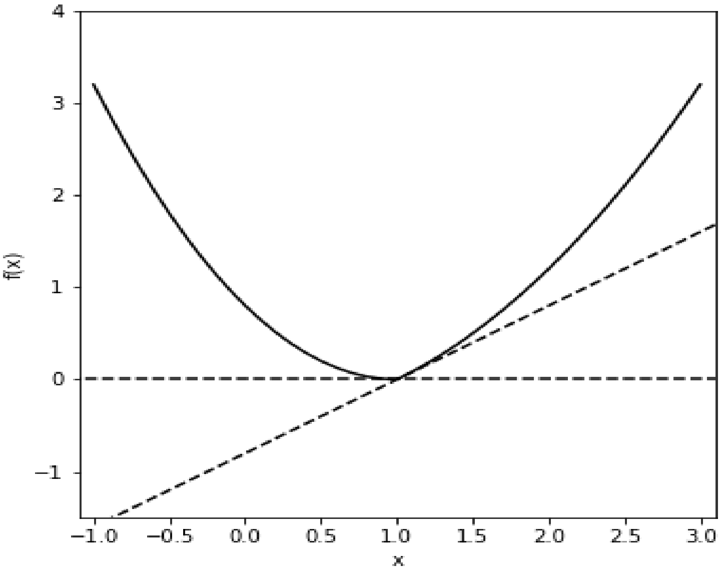
Consider $f : \mathbb{R}^n \to \mathbb{R}$, $f(x) = \|x\|_1$.



> - For $x_i \neq 0$, unique ith component $g_i = \text{sign}(x_i)$.
> - For $x = 0$, ith component g_i is any element of $[-1, 1]$.
> - $\partial f(0) = [-1, 1]^n$.
>   Proof: homework.

$\max$-**fun**

Consider $f(x) = \max\{f_1(x), f_2(x)\}$, for $f_1, f_2 : \mathbb{R}^n \to \mathbb{R}$ convex and differentiable.



> - For $f_1(x) > f_2(x)$, unique subgradient $g = \nabla f_1(x)$.
> - For $f_2(x) > f_1(x)$, unique subgradient $g = \nabla f_2(x)$.
> - For $f_1(x) = f_2(x)$, subgradient g is any point on line segment (i.e., convex combination) between $\nabla f_1(x)$ and $\nabla f_2(x)$.

General conclusion for $f(x) = \max_{i=1,\ldots,m} f_i(x)$,

$$\partial f(x) = \operatorname{conv}\left(\bigcup_{i: f_i(x) = f(x)} \partial f_i(x) \right)$$

Connection to convex geometry

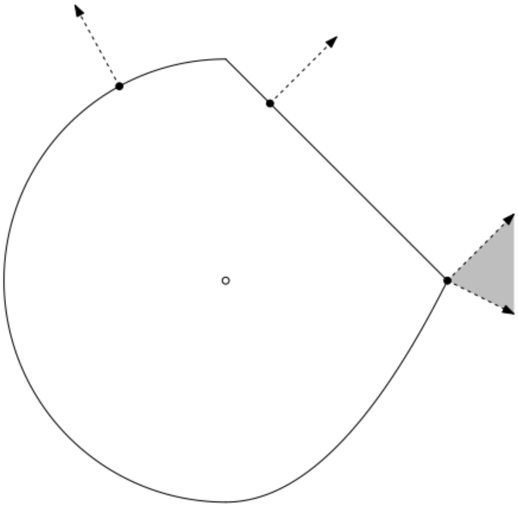
Convex set $C \subset \mathbb{R}^n$, consider ***indicator function*** $I_C : \mathbb{R}^n \to \mathbb{R}$,

$$I_C(x) = I\{x \in C\} = \begin{cases} 0 & \text{if } x \in C \\ \infty & \text{if } x \notin C \end{cases}$$

For $x \in C, \partial I_C(x) = \mathcal{N}_C(x)$, the ***normal cone*** of C at x is

$$\mathcal{N}_C(x) = \{g \in \mathbb{R}^n : g^T x \geq g^T y \text{ for any } y \in C\}$$

> By definition of subgradient g,
>
> $$I_C(y) \geq I_C(x) + g^T(y - x), \text{ for all } y$$
>
> - For $y \notin C, I_C = \infty$.
> - For $y \in C, 0 \geq g^T(y - x)$.



Basic rules

- Scaling:

$$\partial(\alpha f)=\alpha \partial f \qquad \forall \alpha>0$$

- Summation:

$$\partial\left(f_1+f_2\right)=\partial f_1+\partial f_2$$

- Affine transformation:
  If $h(x)=f(Ax+b)$, then

$$\partial h(x)=A^{\top} \partial f(Ax+b)$$

Homework

- Calculate the subdifferential of $\|x\|_{\infty} \triangleq \max_{1 \leq i \leq n}\left|x_i\right|$ at $x=\mathbf{0}$.
- Calculate the subdifferential of $f(x)=\left|x_1\right|+2\left|x_2\right|$ at the point $x=\left(x_1, x_2\right)=(-1,0)$.

Optimal condition: Subgradient

For any f (convex or not)

$$f\left(x^{\star}\right)=\min_x f(x) \Leftrightarrow \mathbf{0} \in \partial f\left(x^{\star}\right)$$

That is, $x^{\star}$ is a minimizer **if and only if** 0 is a subgradient of f at $x^{\star}$. This is called the **subgradient optimality condition**.

> *Proof*: $g=0$ being a subgradient means that for all y
>
> $$f(y) \geq f\left(x^{\star}\right)+\mathbf{0}^T\left(y-x^{\star}\right)=f\left(x^*\right)$$

Derivation of first-order optimality

Recall the **first-order optimality condition**:

$$\min_x f(x) \text{ subject to } x \in C$$

is solved at $x^{\star}$, for f **convex** and differentiable, if and only if

$$\nabla f(x^\star)^T(y - x^\star) \geq 0 \text{ for all } y \in C$$

Now consider the first-order optimal condition for subgradients.
Recast the constraint problem as

$$\min_x f(x) + I_C(x)$$

$x^\star$ is a minimizer of this problem if and only if

$$\nabla f(x^\star)^T(y - x^\star) \geq 0 \text{ for all } y \in C$$

> Apply subgradient optimality, we have
>
> $$\mathbf{0} \in \partial(f(x^\star) + I_C(x^\star))$$
>
> Observe
>
> $$\begin{aligned} & \mathbf{0} \in \partial\left(f(x^\star) + I_C(x^\star)\right) \\ \Longleftrightarrow & \mathbf{0} \in \{\nabla f(x^\star)\} + \mathcal{N}_C(x^\star) \\ \Longleftrightarrow & -\nabla f(x^\star) \in \mathcal{N}_C(x^\star) \\ \Longleftrightarrow & -\nabla f(x^\star)^T x^\star \geq -\nabla f(x)^T y \text{ for all } y \in C \\ \Longleftrightarrow & \nabla f(x^\star)^T(y - x^\star) \geq 0 \text{ for all } y \in C \end{aligned}$$

Example: lasso optimality conditions

Given $y \in \mathbb{R}^n$, $X \in \mathbb{R}^{n \times p}$, lasso problem canbe parametrized as

$$\min_\beta \frac{1}{2}\|y - X\beta\|_2^2 + \lambda\|\beta\|_1$$

where $\lambda \geq 0$.
Subgradient optimality:

$$\begin{aligned} & \mathbf{0} \in \partial\left(\frac{1}{2}\|y - X\beta\|_2^2 + \lambda\|\beta\|_1\right) \\ \Longleftrightarrow & \mathbf{0} \in -X^T(y - X\beta) + \lambda\partial\|\beta\|_1 \\ \Longleftrightarrow & X^T(y - X\beta) = \lambda v \end{aligned}$$

for some $v \in \partial\|\beta\|_1$, i.e.,

$$v_i \in \begin{cases} \{1\} & \text{if } \beta_i > 0 \\ \{-1\} & \text{if } \beta_i < 0, \quad i = 1, \ldots, p \\ [-1, 1] & \text{if } \beta_i = 0 \end{cases}$$

Write $X_1, \ldots, X_p$ for columns of X. Then our condition reads:

$$\begin{cases} X_i^T(y - X\beta) = \lambda \cdot \operatorname{sign}(\beta_i) & \text{if } \beta_i \neq 0 \\ \left|X_i^T(y - X\beta)\right| \leq \lambda & \text{if } \beta_i = 0 \end{cases}$$

> - Subgradient optimality conditions don't lead to closed-form expression for a lasso solution.
> - However, they do provide a way to *check* lasso optimality.

Example: soft-threshholding

Simplfied lasso problem with $X = I$:

$$\min_\beta \frac{1}{2}\|y - \beta\|_2^2 + \lambda\|\beta\|_1$$

This we can solve directly using subgradient optimality. Solution is $\beta = S_\lambda(y)$, where S_λ is the **soft-thresholding operator:**

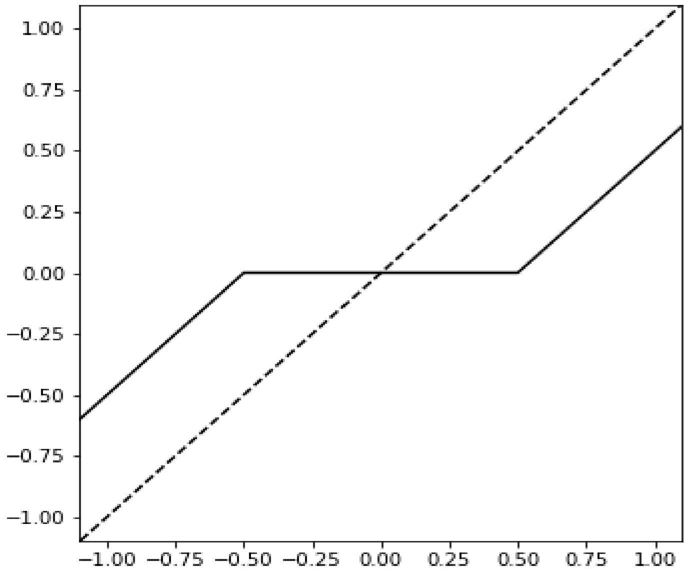
$$[S_\lambda(y)]_i = \begin{cases} y_i - \lambda & \text{if } y_i > \lambda \\ 0 & \text{if } -\lambda \leq y_i \leq \lambda, \\ y_i + \lambda & \text{if } y_i < -\lambda \end{cases} \quad i = 1, \ldots, n$$

Check: from last slide, subgradient optimality conditions are

$$\begin{cases} y_i - \beta_i = \lambda \cdot \operatorname{sign}(\beta_i) & \text{if } \beta_i \neq 0 \\ |y_i - \beta_i| \leq \lambda & \text{if } \beta_i = 0 \end{cases}$$

Now plug in $\beta = S_\lambda(y)$ and check these are satisfied:

- When $y_i > \lambda$, $\beta_i = y_i - \lambda > 0$, so $y_i - \beta_i = \lambda = \lambda \cdot 1$.
- When $y_i < -\lambda$, argument is similar.
- When $|y_i| \leq \lambda$, $\beta_i = 0$, and $|y_i - \beta_i| = |y_i| \leq \lambda$.



Soft-thresholding in one veriable

Subgradient method

Now consider f convex, having $\operatorname{dom}(f) = \mathbb{R}^n$, but not necessarily differentiable.

Subgradient method

- Initialize $x^{(0)}$.
- Repeat:

$$x^{(k)} = x^{(k-1)} - t_k \cdot g^{(k-1)}, \quad k = 1, 2, 3, \ldots$$

where $g^{(k-1)} \in \partial f\left(x^{(k-1)}\right)$, any ***subgradient*** of f at $x^{(k-1)}$.

- Subgradient method is not necessarily a descent method, thus we keep track of best iterate $x_{\text{best}}^{(k)}$ among $x^{(0)}, \ldots, x^{(k)}$

$$f\left(x_{\text{best}}^{(k)}\right) \triangleq \min_{i=0,\ldots,k} f\left(x^{(i)}\right)$$

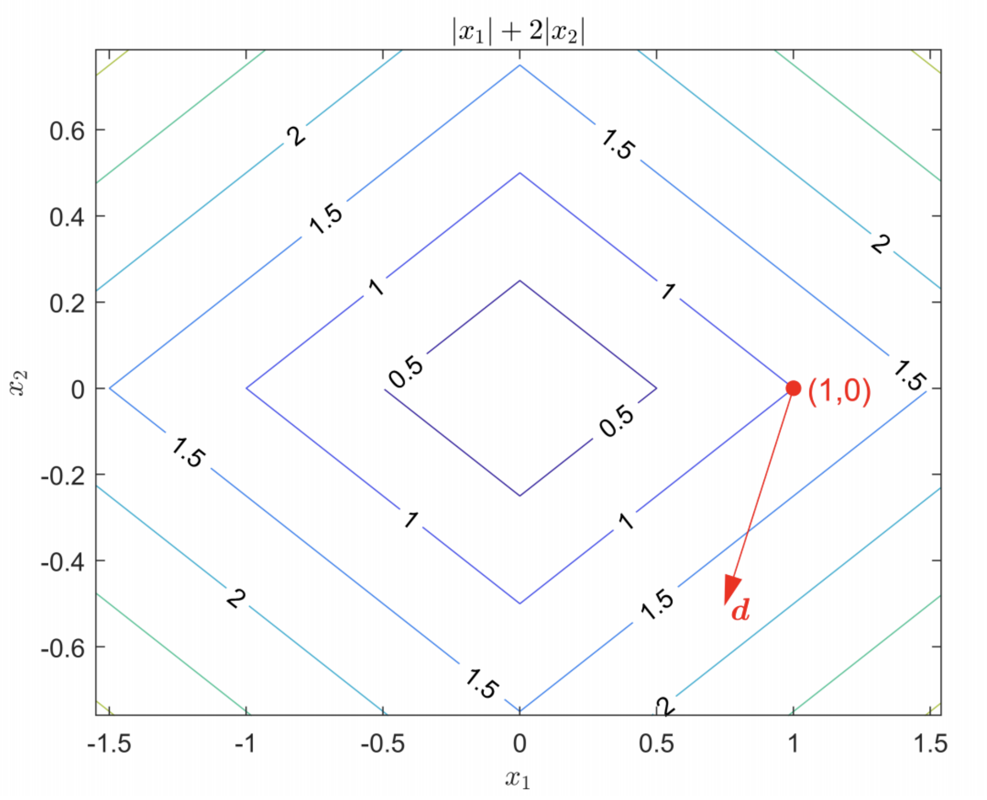
Equivalently,

$$f_{\text{best}}^{(k)} \triangleq \min\left\{f_{\text{best}}^{(k-1)}, f\left(x^{(k)}\right)\right\}$$

Negative subgradients are not necessarily descent directions

Example: $f(x) = |x_1| + 2\,|x_2|$ at $x = (1,0)$
$\boldsymbol{g}_1 = (1,0) \in \partial f(\boldsymbol{x})$, and $-\boldsymbol{g}_1$ is a descent direction
$\boldsymbol{g}_2 = (1,2) \in \partial f(\boldsymbol{x})$, but $-\boldsymbol{g}_2 = (-1,-2)$ is not a descent direction

> Proof: exercise.
> **Reason**: lack of continuity - one can change directions significantly without violating the validity of subgradients. (Taylor approximation fails for non-differential function.)



Soft-thresholding in one veriable

Step size choices

- Fixed step sizes: $t_k = t$, for all $k = 1, 2, \ldots$
- Fixed length: $t_k = \tilde{t}_k / \|g^{(k-1)}\|$ where $\tilde{t}_k$ can be a fixed small number so that $\|x^{(k)} - x^{(k-1)}\|$ is a constant.

> Equivalently, one may consider the following slightly modified scheme
>
> $$x^{(k)} = x^{(k-1)} - \ell_k \cdot g^{(k-1)} / \|g^{(k-1)}\|, \quad k = 1, 2, 3, \ldots$$
>
> But for smooth function f where $g(x^\star) = \nabla f(x^\star) = 0$, this is not recommended.

- **Diminishing step sizes:**
  Choose $\{t_k\}$ that meet conditions

$$\lim_{k\to\infty} t_k = 0, \quad \sum_{k=1}^{\infty} t_k = \infty$$

> example $t_k \propto \sqrt{1/k}$ or $t_k \propto 1/k$.

- **square summable but not summable.**

$$\sum_{k=1}^{\infty} t_k^2 < \infty, \quad \sum_{k=1}^{\infty} t_k = \infty$$

  Step sizes go to zero, but not too fast. For example.

$$t_k \propto 1/k$$

> There are several other options too.
> Key difference to gradient descent: step sizes are pre-specified, not adaptively computed (no line search).

Convergence analysis

Assume that f convex, $\operatorname{dom}(f) = \mathbb{R}^n$, and also that f is Lipschitz continuous with constant $G > 0$, i.e.,

$$|f(x) - f(y)| \leq G\|x - y\|_2 \quad \text{for all } x, y$$

Theorem: For a fixed step size t, subgradient method satisfies

$$\lim_{k\to\infty} f\left(x_{\text{best}}^{(k)}\right) \leq f^{\star} + G^2 t/2$$

Theorem: For diminishing step sizes, subgradient method satisfies

$$\lim_{k\to\infty} f\left(x_{\text{best}}^{(k)}\right) = f^{\star}$$

> Proof:
> First, let’s prove $\|g\|_2$ is bounded by G. According to the defination of subgradient
>
> $$g^T(y - x) \leq f(y) - f(x)$$

> Since
>
> $$|f(x)-f(y)| \leq G\|x-y\|_2$$
>
> We have
>
> $$g^T(y-x) \leq G\|x-y\|_2$$
>
> $$g^T\frac{y-x}{\|x-y\|_2} \leq G$$
>
> where $\frac{y-x}{\|x-y\|_2}$ is an unit vector. So
>
> $$\|g\|_2 \leq G$$
>
> Consider the defination of subgradient method
>
> $$\begin{aligned} & \left\|x^{(k)}-x^{\star}\right\|_2^2 \\ = & \left\|x^{(k-1)}-t_k g^{(k-1)}-x^{\star}\right\|_2^2 \\ \leq & \left\|x^{(k-1)}-x^{\star}\right\|_2^2-2t_k\left(f\left(x^{(k-1)}\right)-f\left(x^{\star}\right)\right)+t_k^2\left\|g^{(k-1)}\right\|_2^2 \end{aligned}$$
>
> Likewise, for every $i=1,2,\ldots,k$,
>
> $$\left\|x^{(i)}-x^{\star}\right\|_2^2 \leq \left\|x^{(i-1)}-x^{\star}\right\|_2^2-2t_i\left(f\left(x^{(i-1)}\right)-f\left(x^{\star}\right)\right)+t_i^2\left\|g^{(i-1)}\right\|_2^2$$
>
> So we have
>
> $$\left\|x^{(k)}-x^{\star}\right\|_2^2 \leq \left\|x^{(0)}-x^{\star}\right\|_2^2-2\sum_{i=1}^k t_i\left(f\left(x^{(i-1)}\right)-f\left(x^{\star}\right)\right)+\sum_{i=1}^k t_i^2\left\|g^{(i-1)}\right\|_2^2$$
>
> Let
>
> $$R=\left\|x^{(0)}-x^{\star}\right\|_2,$$
>
> recall $\|g\|_2 \leq G$,
>
> $$\begin{aligned} 0 \leq R^2-2\sum_{i=1}^k t_i\left(f\left(x^{(i-1)}\right)-f\left(x^{\star}\right)\right)+G^2\sum_{i=1}^k t_i^2 \\ 2\sum_{i=1}^k t_i\left(f\left(x_{best}^k\right)-f\left(x^{\star}\right)\right) \leq R^2+G^2\sum_{i=1}^k t_i^2 \\ f\left(x_{best}^k\right)-f\left(x^{\star}\right) \leq \frac{R^2+G^2\sum_{i=1}^k t_i^2}{2\sum_{i=1}^k t_i} \end{aligned}$$

> where $f\left(x_{\text{best}}^{(k)}\right)=\min_{i=0,\ldots,k} f\left(x^{(i)}\right)$

We have the **basic inequality**

$$f\left(x_{best}^{(k)}\right)-f\left(x^{\star}\right) \leq \frac{R^{2}+G^{2} \sum_{i=1}^{k} t_{i}^{2}}{2 \sum_{i=1}^{k} t_{i}}$$

The two theorems above can then be proved accordigly.

> - For a fixed step size $t_k = t$,
>
> $$f\left(x_{best}^{(k)}\right)-f\left(x^{\star}\right) \leq \frac{R^{2}+G^{2} k t^{2}}{2 k t}$$
>
> As $k \rightarrow \infty$,
>
> $$\lim_{k \rightarrow \infty} f\left(x_{best}^{(k)}\right) \leq f\left(x^{\star}\right)+\frac{G^{2} t}{2}$$
>
> - For a diminishing step size t_k, we have
>
> $$\sum_{k=1}^{\infty} t_{k}^{2}<\infty, \quad \sum_{k=1}^{\infty} t_{k}=\infty$$
>
> Thus,
>
> $$\frac{R^{2}+G^{2} \sum_{i=1}^{k} t_{i}^{2}}{2 \sum_{i=1}^{k} t_{i}} \rightarrow 0, \quad k \rightarrow \infty$$
>
> So
>
> $$f\left(x_{best}^{(k)}\right) \rightarrow f(x^{\star})$$

For different step sizes choices, convergence results can be directly obtained from this bound, e.g., previous theorems follow.

Convergence rate of constant step size

Now we fix the total number of step to be N and use *fixed step size* t, the basic inequality gives

$$f\left(x_{\text{best}}^{(k)}\right)-f^{\star}\leq\frac{R^{2}}{2kt}+\frac{G^{2}t}{2},\quad k=1,2,\ldots,N$$

Now we fix the total number of step to be N, then the iteration k is from 1 to N and the final error is

$$f\left(x_{\text{best}}^{(N)}\right)-f^{\star}\leq\frac{R^{2}}{2Nt}+\frac{G^{2}t}{2}.$$

To find the optimal constant step size, we solve

$$\min_{t}\quad\frac{R^{2}}{2Nt}+\frac{G^{2}t}{2}$$

for each integer N. The solutions is

$$t_{*}=\frac{R}{G}\frac{1}{\sqrt{N}}$$

> **Note**: this step size is not adaptive, it depends on how many ***total*** steps (N) to take. If N changes, the step size also changes. So this fixed step size is very incovenient for online and stream data where N may increase dynamically.

and we have

$$f\left(x_{\text{best}}^{(N)}\right)-f^{\star}\leq\frac{R^{2}}{2Nt_{*}}+\frac{G^{2}t_{*}}{2}=\frac{G}{R}\frac{1}{\sqrt{N}}=O(\frac{1}{\sqrt{N}})$$

- For the error to be $\leq\epsilon$, make $\frac{G}{R}\frac{1}{\sqrt{N}}\leq\epsilon$. We can choose the total number of steps

  $$k=\frac{G^{2}}{R^{2}\epsilon^{2}}$$

  and the (optimal constant) step size is determined by

  $$t=\frac{R}{G}\frac{1}{\sqrt{N}}=\frac{R^{2}}{G^{2}}\epsilon$$

- The subgradient method has iteration complexity $O\left(1/\epsilon^{2}\right)$.

> Note that this has a higher compputational cost than $O(1/\epsilon)$ rate of gradient descent.

Convergence rate of diminishing step size $1/\sqrt{k}$

For the diminishing step size $t_k \propto 1/\sqrt{k}, k = 1, 2, \ldots,$

$$\sum_{i=1}^{k} t_i \sim \sum_{i=1}^{k} \frac{1}{\sqrt{k}} \approx \int_1^k \frac{1}{\sqrt{x}} dx \sim \sqrt{k}$$

$$\sum_{i=1}^{k} t_i^2 \sim \sum_{i=1}^{k} \frac{1}{k} \approx \int_1^k \frac{1}{x} dx \sim \log k$$

$$f\left(x_{\text{best}}^{(k)}\right) - f\left(x^\star\right) \leq \frac{R^2 + G^2 \sum_{i=1}^k t_i^2}{2 \sum_{i=1}^k t_i} \sim \frac{R^2 + G^2 \log k}{2\sqrt{k}} = O(\frac{1}{\sqrt{k}})$$

with a negligible $\log k$ term.

Convergence rate of step size $1/k$

For suqare summable step size $t_k \propto 1/k, k = 1, 2, \ldots,$ then

$$\sum_{k=1}^{\infty} \frac{1}{t_k^2} = const < \infty$$

and

$$f\left(x_{\text{best}}^{(k)}\right) - f\left(x^\star\right) \leq \frac{R^2 + G^2 \sum_{i=1}^k t_i^2}{2 \sum_{i=1}^k t_i} = O(\frac{1}{\log k})$$

The rate $O(\frac{1}{\log k})$ is extremely slow. But for strongly convex function, we can improve the convergence rate to $1/k$ with this step size. [1]

Summary of convergence of subgradient methods

	step size rule	**convergence rate**	**iteration complexity**
f convex and Lipschitz	$t_k \propto 1/\sqrt{k}$	$O(\frac{1}{\sqrt{k}})$	$O(\frac{1}{\epsilon^2})$
f **strongly** convex and Lipschitz	$t_k \propto 1/k$	$O(\frac{1}{k})$	$O(\frac{1}{\epsilon})$

Recall: gradient methods for differential convex function

	convergence rate	iteration complexity
f convex and ∇f Lipschitz	$O(\frac{1}{k})$	$O(\frac{1}{\epsilon})$
f **strongly** convex and ∇f Lipschitz	γ^k	$O(\log \frac{1}{\epsilon})$

The stepsize is constant or from backtracking.

Example: regularized Logistic regression

Given $(x_i, y_i) \in \mathbb{R}^p \times \{0,1\}$ for $i = 1, \ldots, n$, the logistic regression loss is

$$f(\beta) = \sum_{i=1}^{n} \left(-y_i x_i^T \beta + \log\left(1 + \exp\left(x_i^T \beta\right)\right)\right)$$

Gradient of this function is

$$\nabla f(\beta) = \sum_{i=1}^{n} \left(-y_i + p_i(\beta)\right) x_i$$

where

$$p_i(\beta) = \frac{\exp\left(x_i^T \beta\right)}{\left(1 + \exp\left(x_i^T \beta\right)\right)}, i = 1, \ldots, n.$$

Considerthe regularized problem:

$$\min_{\beta} f(\beta) + \lambda \cdot P(\beta)$$

where $P(\beta) = \|\beta\|_2^2$ for ridge penalty, $P(\beta) = \|\beta\|_1$ for lasso penalty.

Summary: subgradient method

For convex and Lipchitz continuous and non-differentiable functions:

- Pros:
    - Handles ***general*** nondifferentiable convex problem.
    - Often leads to *very simple algorithms*.
- Cons:
    - Convergence can be very slow.
    - No good stopping criterion.
    - Theoretical complexity: $O\left(1/\epsilon^2\right)$ iterations to find ϵ -suboptimal point.
    - An “optimal” first-order method: $O\left(1/\epsilon^2\right)$ bound cannot be improved.

Improving on the subgradient method

In words, we cannot do better than the $O\left(1/\epsilon^2\right)$ rate of subgradient method (unless we go beyond nonsmooth first-order methods).

The classical subgradient methods have poor performance and are no longer recommended for many use in machine learning. However, they are still used widely in specialized applications because they are simple and they can be easily adapted to general questions.

In machine learning, instead of trying to improve across the board in general, people focus on minimizing composite functions of the form

$$f(x) = g(x) + h(x)$$

where g is convex and differentiable, h is convex and ***nonsmooth*** but “simple”.

For a lot of problems (i.e., functions h), we can recover the $O(1/\epsilon)$ rate of gradient descent with a simple algorithm, having important practical consequences. We discuss this later.

Reference and further reading

- ”Convex optimization, EE364B lecture notes,” S. Boyd, Stanford.
- Yu. Nesterov, Lectures on Convex Optimization (2018), section 3.2.3.
- http://www.seas.ucla.edu/~vandenbe/236C/lectures/sgmethod.pdf

1. http://www.princeton.edu/~yc5/ele522_optimization/lectures/subgradient_methods.pdf ↩