# Optimization for Machine Learning

September 27, 2023

Today's roadmap:

- Gradient descent and convex optimization
- Accelerated methods

⇒ Notes + illustration notebook available on the course webpage after the class

# GRADIENT METHODS AND CONVEX OPTIMIZATION

Problem of interest:

$$\underset{w \in \mathbb{R}^d}{\text{minimize}} \quad f(w) \qquad f : \mathbb{R}^d \to \mathbb{R}$$

convex

Assumption: $f$ belongs to the class of $C_L^{1,1}$ functions, i.e.

- $f$ is $C^1$ (continuously differentiable)

  At every $w \in \mathbb{R}^d$, $\exists\, \nabla f(w) \in \mathbb{R}^d$ (gradient of $f$ at $w$) that represents how the function varies locally

- The gradient mapping $\nabla f : \mathbb{R}^d \to \mathbb{R}^d$, $w \mapsto \nabla f(w)$ is $L$-Lipschitz continuous where $L > 0$.

$$\Downarrow$$

$$\forall (w, v) \in (\mathbb{R}^d)^2, \quad \| \nabla f(w) - \nabla f(v) \| \leq L \| w - v \|$$

Examples of $C_L^{1,1}$ functions:

- → linear least squares
- → logistic regression objective "logistic loss"
- → quadratic functions

Remark: $C_L^{1,1}$ assumption is a simplifying assumption

- → Functions are not always $C^1$ (see next week!)
- → Functions/gradients are not always Lipschitz continuous on the entire space $\mathbb{R}^d$

  => Possible to use local Lipschitz constants

↳ Sometimes $C^{1,1}_L$ functions are called $L$-smooth

## Two key properties

If $f$ is $C^{1,1}_L$ and convex, then

$\forall w \in \mathbb{R}^d, \forall v \in \mathbb{R}^d$,

(1) $$f(v) \leq f(w) + \nabla f(w)^T (v-w) + \frac{L}{2}\|v-w\|^2$$

(constant in $v$; linear function in $v$; quadratic function of $v$)

$\|v-w\|^2 = \sum_{i=1}^{d} (v_i - w_i)^2$

and

(2) $$f(v) \geq f(w) + \nabla f(w)^T (v-w)$$

(linear function of $v$)

(1) $\Rightarrow$ Upper bound on $f(v)$ with a quadratic function of $v$

(2) $\Rightarrow$ Lower bound on $f(v)$ with a linear function of $v$

((2) is actually a characterization of convexity for $C^1$ functions)

## Towards gradient descent

Main idea behind the gradient descent method

- Suppose that you are at $w \in \mathbb{R}^d$ and you know $f(w), \nabla f(w)$
- If $\|\nabla f(w)\| = 0$, then $w$ is a global minimum ($f$ convex)
- When $\|\nabla f(w)\| \neq 0$, we can find $v$ such that

$$f(w) + \nabla f(w)^T (v-w) + \frac{L}{2}\|v-w\|^2 < f(w)$$

(Need this $< 0$)

implying that $f(v) \overset{(1)}{\leq} f(w) + \nabla f(w)^T (v-w) + \frac{L}{2}\|v-w\|^2 < f(w)$

- We can then replace $w$ by $v$ and repeat the process

# (1) Gradient descent ($f \in C^{2,1}$, convex)

## Gradient descent (GD) algorithm

$\{w_k\}_k$ : iterates

Initialization : Choose $w_0 \in \mathbb{R}^d$.

For $k = 0, 1, \dots$

(i) Evaluate $\nabla f(w_k)$. If $\|\nabla f(w_k)\| = 0$, terminate and output $w_k$.

(ii) Compute a stepsize $\alpha_k > 0$.

(iii) Define $w_{k+1} = w_k - \alpha_k \nabla f(w_k)$

"Gradient descent iteration"

End For

● : In an implementation, the for loop is replaced by a while loop involving a convergence criterion and a budget criterion

Ex) While $[\|\nabla f(w_k)\| \neq 0]$ and $[k \leq K]$

:

End while

$\|\nabla f(w_k)\| \neq 0$ ← convergence criterion

$k \leq K$ ← budget criterion.

→ Number of iterations

→ Number of function/gradient evaluations

→ CPU time

### Other convergence criteria

- $\|\nabla f(w_k)\| < \varepsilon$ where $\varepsilon$ is a small tolerance/precision (e.g. $10^{-4}$, $10^{-5}$, $10^{-16}$)
- $f(w_k) - \min_{w \in \mathbb{R}^d} f(w) < \varepsilon$

"Close to the best possible objective"

⚠ Most of the time the optimal value is unknown!

But for convex functions, can provide theoretical bounds on the cost of satisfying this condition

- $\|w_k - w^*\| < \varepsilon$

where $w^* \in \operatorname{argmin}_{w \in \mathbb{R}^d} f(w)$ ($w^*$ solution of the problem)

"close to a solution"

⚠ In general optimal solutions are not known!

⇒ But for certain convex functions can provide bounds on the cost of satisfying $\|w_k - w^*\| < \varepsilon$

→ How do we select $\alpha_k$?

- One choice that works for $C^{1,1}_L$ functions: $\alpha_k = \frac{1}{L}$

**Proposition:** Suppose that we are at iteration $k$ of GD and $\|\nabla f(w_k)\| \neq 0$. Then, if $\alpha_k = 1/L$, we have

$$f\left(w_k - \alpha_k \nabla f(w_k)\right) \leq f(w_k) - \frac{1}{2L}\|\nabla f(w_k)\|^2 < f(w_k)$$

**Proof:** Apply (1) with $v = w_k - \alpha_k \nabla f(w_k)$ and $w = w_k$

$$f(w_k - \alpha_k \nabla f(w_k)) \leq f(w_k) + \nabla f(w_k)^T (w_k - \alpha_k \nabla f(w_k) - w_k) + \frac{L}{2}\|w_k - \alpha_k \nabla f(w_k) - w_k\|^2$$

$$= f(w_k) + \nabla f(w_k)^T(-\alpha_k \nabla f(w_k)) + \frac{L}{2}\|-\alpha_k \nabla f(w_k)\|^2$$

$$= f(w_k) - \alpha_k \nabla f(w_k)^T \nabla f(w_k) + \frac{L}{2}\alpha_k^2 \|\nabla f(w_k)\|^2$$

($\nabla f(w_k)^T \nabla f(w_k) = \|\nabla f(w_k)\|^2$)

$$= f(w_k) - \alpha_k \|\nabla f(w_k)\|^2 + \frac{L}{2}\alpha_k^2 \|\nabla f(w_k)\|^2$$

$$= f(w_k) - \frac{1}{L}\|\nabla f(w_k)\|^2 + \frac{L}{2} \times \frac{1}{L^2}\|\nabla f(w_k)\|^2$$

$$= f(w_k) + \left[-\frac{1}{L} + \frac{1}{2L}\right]\|\nabla f(w_k)\|^2 = f(w_k) - \frac{1}{2L}\|\nabla f(w_k)\|^2$$

$\forall u, v \in \mathbb{R}^d$

$\|v\|^2 = \sum_{i=1}^{d} v_i^2$

$u^T v = \sum_{i=1}^{d} u_i v_i$

$v^T v = \|v\|^2$

More general result: For any $\alpha_k \in (0, \frac{2}{L})$, one has

$$f(w_k - \alpha_k \nabla f(w_k)) < f(w_k)$$

Remarks:
- → Suggests a problem-dependent choice for $\alpha_k$
- → Requires knowledge of the Lipschitz constant (assumed to be known in this lecture)

Q: Under our assumptions, what can we prove about the algorithm?

**Theorem:** Suppose that $f$ is convex and $C^{1,1}_L$.
Suppose that we run GD with $\alpha_k = \frac{1}{L}$ $\forall k$.

a) Convergence rate

For any $K \geq 1$, after $K$ iterations of GD, we have

$$f(w_K) - \min_{w \in \mathbb{R}^d} f(w) \leq O\left(\frac{1}{K}\right)$$

"worst case behavior"

"big-O" notation
A constant times $\frac{1}{K}$ where the constant does not depend on $K$

b) Complexity bound

For any $\varepsilon > 0$, GD reaches an iterate $w_K$ such that

$$f(w_K) - \min_{w \in \mathbb{R}^d} f(w) \leq \varepsilon$$

after at most $O\left(\frac{1}{\varepsilon}\right)$ iterations.

constant that depends on $w_0$, $L$, distance between $w_0$ and argmin $f$

$\varepsilon \Rightarrow \frac{\varepsilon}{10}$

$O(\varepsilon^{-1}) \Rightarrow O(10\varepsilon^{-1})$

↳ CV rates and complexity bounds are used as a guidance / as an indicator of algorithmic performance.
They can be close to practical performance on simple examples and these rates/bounds are attained on some (pathological) examples.

↳ Two ways to get better rates / bounds
- Add assumptions on the function ⇒ strong convexity
- Change the algorithm ⇒ acceleration

## (2) Strongly convex functions

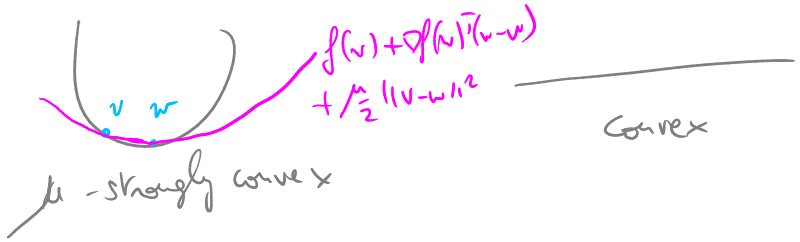
Def: $f : \mathbb{R}^d \to \mathbb{R}$ $C^1$

$f$ is called $\mu$-strongly convex (where $\mu > 0$) if

$$\forall (v,w) \in (\mathbb{R}^d)^2, \quad f(v) \geq f(w) + \nabla f(w)^T (v-w) + \frac{\mu}{2}\|v-w\|^2$$

$f(v) \geq f(w) + \nabla f(w)^T (v-w)$ corresponds to (2)

$\frac{\mu}{2}\|v-w\|^2$: Additional quadratic term compared to (2)




NB: $\mu$-strongly convex ⇒ convex

Property: If $f$ is $C^1$ and $\mu$-strongly convex, then it has a unique global minimum, which is (the unique) solution of $\|\nabla f(w)\| = 0$

### Consequences on GD

↳ Suppose that $f \in \mathcal{F}^{1,1}_{L,\mu}$ $\quad \mathcal{F}^{1,1}_{L,\mu} = \{ C^{1,1}_L$ and $\mu$-strongly convex$\}$

$\Rightarrow L \geq \mu$

$\{w^*\} = \operatorname{argmin}_w f(w)$

Then running GD with $\alpha_k = \frac{1}{L}$ gives the following guarantees:

a) CV rate After $K \geq 1$ iterations

$$\|w_K - w^*\| \leq \left(1 - \frac{\mu}{L}\right)^K \|w_0 - w^*\|$$

$\|w_K - w^*\|$: distance between the last iterate and the solution

$\left(1 - \frac{\mu}{L}\right) \in [0, 1)$

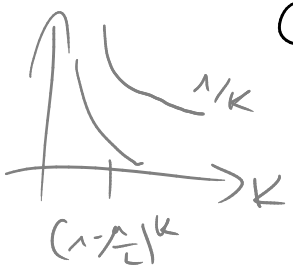
$\|w_0 - w^*\|$: distance between the initial point and the solution

$\left(1 - \frac{\mu}{L}\right)^K \to 0$ as $K \to \infty$

This is called a linear rate of CV or an exponential rate of CV

$\left(1 - \frac{\mu}{L}\right)^K = e^{K \log\left(1 - \frac{\mu}{L}\right)}$

"linear" $\Rightarrow \log(\|w_K - w\|)$ decreases as a linear function of $K$




⊕ Better than the results for convex functions

- Stronger criterion: $\|w_K - w^*\|$ vs $f(w_K) - \min_{w \in \mathbb{R}^d} f(w)$
- Better rate

$\left(1 - \frac{\mu}{L}\right)^K$ goes to zero faster than $1/K$

⊕ In practice, GD is faster on strongly convex functions than on convex ones

For certain strongly convex functions, one can define a better stepsize than $\frac{1}{L}$ (i.e. $\frac{2}{L+\mu}$ for quadratic functions)

b) Complexity bound

$\|w_K - w^*\| \leq \varepsilon$ holds after at most $O\left(\frac{L}{\mu} \log(1/\varepsilon)\right)$ iterations

"Condition number of the function"

$\log(1/\varepsilon) < 1/\varepsilon$ for $\varepsilon$ sufficiently small

# (3) Acceleration

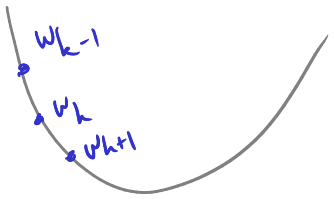
**Question:** Is there an algorithm that has better guarantees than GD while doing the same amount of work per iteration (i.e. evaluating 1 gradient)?

Recall: GD iteration

$$\forall k \geq 0, \quad w_{k+1} = w_k - \alpha_k \nabla f(w_k) \qquad \text{(think } \alpha_k = 1/L\text{)}$$

To define the step $w_{k+1} - w_k = -\alpha_k \nabla f(w_k)$, we only rely on information related to $w_k$

and we ignore what we did in the previous steps $w_k - w_{k-1}, \dots$




**Intuition:** Maybe $w_k - w_{k-1}$ would also be a good step from $w_k$

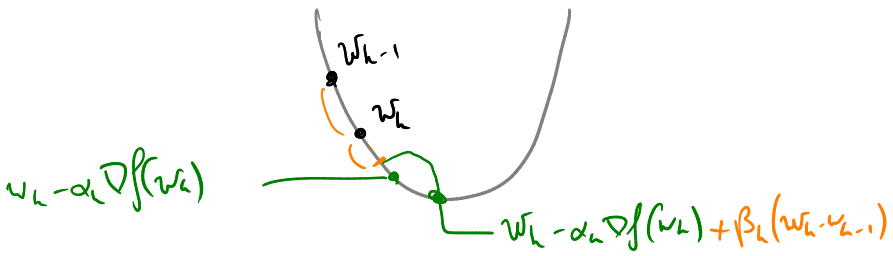
Polyak (1964): Heavy-ball method / Gradient descent with momentum

$$\forall k \geq 0, \quad w_{k+1} = w_k \underbrace{- \alpha_k \nabla f(w_k)}_{\text{GD step}} + \underbrace{\beta_k (w_k - w_{k-1})}_{\text{"Momentum term"}}$$

"Previous step"

with $\beta_0 = 0$ $\quad w_{-1} = w_0$

$\beta_k \geq 0$



↳ This is optimal on strongly convex quadratic functions (better than GD and cannot do better with only 1 gradient evaluation per iteration)

CV rate $\left(1-\sqrt{\frac{\mu}{L}}\right)^K \leq \underbrace{\left(1-\frac{\mu}{L}\right)^K}_{GD}$

↳ But Heavy-ball can fail on strongly convex functions that are not quadratic

Nesterov (1983): Accelerated gradient / Nesterov's method

$$\forall k \geq 0, \quad w_{k+1} = w_k - \alpha_k \nabla f\left(w_k + \beta_k (w_k - w_{k-1})\right) + \beta_k (w_k - w_{k-1})$$

≠ with Heavy ball:
- First compute $w_k + \beta_k (w_k - w_{k-1})$
- then do a gradient step from $w_k + \beta_k (w_k - w_{k-1})$

First do $w_k - \alpha_k \nabla f(w_k)$
then add $\beta_k (w_k - w_{k-1})$

Guarantees for Nesterov's method (optimal: cannot do better)

- If $f \in \mathcal{F}^{1,1}_{L,\mu}$, set $\alpha_k = \frac{1}{L}$ and $\beta_k = \frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}$ ↘ problem dependent

Then

CV rate $\|w_K - w^*\| \leq \left(1-\sqrt{\frac{\mu}{L}}\right)^K \|w_0 - w^*\|$ $\forall K \geq 1$

↑ improves over $\frac{\mu}{L}$ for GD

complexity $\|w_K - w^*\| \leq \varepsilon$ after $O\left(\sqrt{\frac{L}{\mu}} \log(1/\varepsilon)\right)$ iterations

↑ improves over $\frac{L}{\mu}$ for GD

- If $f \in C^{1,1}_L$ and convex

Set $\alpha_k = \frac{1}{L}$ and $\beta_k = \frac{t_k - 1}{t_{k+1}}$ where $\begin{cases} t_0 = 0 \\ t_{k+1} = \frac{1}{2}\left(1+\sqrt{1+4t_k^2}\right) \end{cases}$

independent of the problem (or even $w_0$)

CV rate $\quad f(w_K) - \min_{w \in \mathbb{R}^d} f(w) \leq O\left(\frac{1}{K^2}\right)$

Complexity $\quad f(w_K) - \min_{w \in \mathbb{R}^d} f(w) \leq \varepsilon$ after at most $O(\varepsilon^{-1/2})$ iterations

$\frac{1}{K^2} << \frac{1}{K}$ for large $K$ and $\varepsilon^{-1/2} << \varepsilon^{-1}$ for small $\varepsilon$