OPTIMIZATION FOR MACHINE LEARNING

September 28, 2023

Today 8.30 - 10.00 : Lecture (Gradient descent)
10.15 - 11.45 : Lab tutorial

Next week Monday: 8.30 - 10.00 : Lab tutorial
10.15 - 11.45 : Lecture

- Thursday 10.15 - 11.45 : Lecture
12.00 - 13.30 : Tutorial

GRADIENT DESCENT (AGAIN!)

⇒ what we can prove about that method?

⇒ Is it the best we can do?

(1) Setup

Pb: minimize $f(w)$, $w \in \mathbb{R}^d$

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

f bounded below $\Leftrightarrow \exists f_{low} \in \mathbb{R}$ such that $f(w) \geq f_{low}$ $\forall w \in \mathbb{R}^d$

Key assumption: f is $C^{1,1}_L$ ($L > 0$), where $C^{1,1}_L$ is the class of functions that satisfy:

i) $f \in C^1$ ($\forall w \in \mathbb{R}^d$, $\exists \nabla f(w) \in \mathbb{R}^d$)

ii) $\nabla f : \mathbb{R}^d \to \mathbb{R}^d$ is L-Lipschitz continuous, i.e.

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

Ex) $f : w \mapsto \frac{1}{2m}\|Xw - y\|^2$ where $X \in \mathbb{R}^{m \times d}$, $y \in \mathbb{R}^m$

is $C^{1,1}_L$ with $L = \frac{\|X^T X\|}{m} = \left\| \frac{X^T X}{m} \right\|$

$\forall w$ $\nabla f(w) = \frac{X^T X}{m} w - \frac{X^T y}{m}$.

$\forall (v,w) \in (\mathbb{R}^d)^2$, $\|\nabla f(v) - \nabla f(w)\| = \left\| \frac{X^T X}{m} v - \frac{X^T X}{m} w \right\|$

$= \left\| \frac{X^T X}{m} (v - w) \right\| \leq \underbrace{\left\| \frac{X^T X}{m} \right\|}_{L} \|v - w\|$

NB: In practice, we do not know L in general, but the theory and the algorithms below can be adapted to work with approximations of L

Key inequality: Suppose that f is $C^{1,1}_L$ (for $L>0$). Then, for any $w \in \mathbb{R}^d$ and $v \in \mathbb{R}^d$,

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

↳ Interest: Given $w \in \mathbb{R}^d$, if we find $v \in \mathbb{R}^d$ such that

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

then by (1), $f(v) < f(w)$

(2) Connection to gradient descent

Gradient descent iteration:

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

(w_k: iterate for iteration k; α_k: stepsize; where $\alpha_k > 0$)

(start from $w_0 \in \mathbb{R}^d$)

Theorem: Suppose that $f \in C^{1,1}_L$ and consider the k-th iteration of gradient descent. Suppose that $\|\nabla f(w_k)\| \neq 0$, and that $\alpha_k = \frac{1}{L}$. Then

$$f(w_{k+1}) \leq f(w_k) - \frac{1}{2L}\|\nabla f(w_k)\|^2 < f(w_k)$$

↳ Guaranteed decrease in the function value

Proof: Apply (1) with $v = w_{k+1}$ and $w = w_k$:

$$f(w_{k+1}) \leq f(w_k) + \nabla f(w_k)^T (w_{k+1} - w_k) + \frac{L}{2} \|w_{k+1} - w_k\|^2$$

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

$$\begin{aligned} f(w_{k+1}) &\leq f(w_k) + \nabla f(w_k)^T (w_k - \alpha_k \nabla f(w_k) - w_k) + \frac{L}{2} \|w_k - \alpha_k \nabla f(w_k) - w_k\|^2 \\ &= f(w_k) + \nabla f(w_k)^T (-\alpha_k \nabla f(w_k)) + \frac{L}{2} \|-\alpha_k \nabla f(w_k)\|^2 \\ &= f(w_k) - \alpha_k \underbrace{\nabla f(w_k)^T \nabla f(w_k)}_{= \|\nabla f(w_k)\|^2} + \frac{L}{2} \alpha_k^2 \|\nabla f(w_k)\|^2 \end{aligned}$$

$u^T v = \sum u_i v_i$

$\|v\| = \sqrt{\sum v_i^2}$

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

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

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

↳ We have shown that the GD produces w_{k+1} such that

$f(w_{k+1}) < f(w_k)$ (any $\alpha_k \in \left(0, \frac{2}{L}\right)$ works !)

↳ We would like to show that $f(w_k)$ converges to $\min_{w \in \mathbb{R}^d} f(w)$

⇒ Possible when f is convex ⇒ Focus today

⇒ when f is nonconvex, we can show that $\|\nabla f(w_k)\| \to 0$ ($k \to \infty$)

↓ see next week

"converges to"

Convergence rates for gradient descent

- Suppose that f is $C_L^{1,1}$ and convex.
  After $K \geq 1$ iterations of gradient descent with $\alpha_k = 1/L$, we have:

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

where $O\left(\frac{1}{K}\right)$ means $C \times \frac{1}{K}$, where $C > 0$ does not depend on K.

We say that GD has a convergence rate of $\frac{1}{K}$ for convex problems

$$K \leftarrow 10K \quad \Rightarrow \quad f(w_{10K}) - \min_{w \in \mathbb{R}^d} f(w) \leq O\left(\frac{1}{10K}\right)$$

- Suppose that f is $C_L^{1,1}$ and μ-strongly convex (unique global minimum)
  Apply GD for $K \geq 1$ iterations with $\alpha_k = \frac{1}{L}$. Then

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

and

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

where w^* is the global minimum of f.

→ Same algorithm
→ Different function class (strong convexity)
⇒ Better convergence rates

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

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

⇒ Faster convergence to the minimum value

ii) In the strongly convex setting, the iterates converge to the minimum at the same rate as the function values.

Takeaways

- GD is a general purpose method (it works on a large class of functions)
- Depending on the function class, GD can have different guarantees (convex vs strongly convex)
- Guarantees like convergence rates are used to compare algorithms, and they are connected to practical performance, especially on convex problems

Other algorithms VS GD (see Monday's lab session)

Algorithm	Convergence rate for $C_L^{1,1}$ convex functions	Convergence rate for $C_L^{1,1}$ μ-strongly convex functions
Gradient descent $w_{k+1} = w_k - \frac{1}{L}\nabla f(w_k)$ (~1800s)	$O\left(\frac{1}{K}\right)$	$O\left(\left(1-\frac{\mu}{L}\right)^K\right)$
Nesterov's method / Accelerated gradient (1983)	$O\left(\frac{1}{K^2}\right)$ optimal convergence rate	$O\left(\left(1-\sqrt{\frac{\mu}{L}}\right)^K\right)$ optimal

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

Nesterov: $w_{k+1} = w_k - \alpha_k \nabla f\left(w_k + \beta_k(w_k - w_{k-1})\right) + \beta_k (w_k - w_{k-1})$

Gradient descent + Previous step $(w_k - w_{k-1})$

$\Rightarrow$ Use only 1 gradient per iteration