OPTIMIZATION FOR MACHINE LEARNING

October 2, 2023

Outline: Lab on Acceleration (done)
Lecture part: Stochastic gradient

⚠ Thursday: 10.15 am - 1.30 pm (then 1 month break)

STOCHASTIC GRADIENT METHODS

↳ So far we looked at generic problems

$$\underset{w \in \mathbb{R}^d}{\text{minimize}} \; f(w) \qquad f \in C^1$$

and then solving using gradient descent

↳ In ML, most optimization problems have a specific structure that can be exploited to design better algorithms

Typical form of an optimization problem in ML

$$\underset{w \in \mathbb{R}^d}{\text{minimize}} \; f(w) := \frac{1}{n} \sum_{i=1}^{n} f_i(w)$$

↳ Finite sum structure

where n is the number of data points (or samples)
and $f_i : \mathbb{R}^d \to \mathbb{R}$ is the loss function that depend on data point i in the dataset

Usually (especially in deep learning), but not always, n is very large

(Ex) Linear least squares $\qquad f(w) = \frac{1}{2n} \|Xw - y\|^2 = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2} (x_i^T w - y_i)^2$

Data set : $\{(x_1, y_1), \dots, (x_n, y_n)\}$

$f_i(w) = \frac{1}{2} (x_i^T w - y_i)^2$

↳ Assuming that the f_is are C^1, then f is C^1 and we can apply gradient descent

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

$$= w_k - \alpha_k \times \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(w_k)$$

↑ Gradient of f_i with respect to w

Observations

- One iteration of GD involves computing $\nabla f(w_k)$, which itself involves computing $\nabla f_1(w_k), \dots, \nabla f_n(w_k)$
- Each $\nabla f_i(w_k)$ depends on data point i in the dataset

$\Rightarrow$ Computing $\nabla f_i(w_k)$ involves one access to that data point

- Thus, one iteration of GD involves accessing the entire dataset

$\Rightarrow$ Expensive when n is large

↳ In what follows, we assume that n is too large for GD to be practical

(1) Stochastic gradient method

Recall: GD iteration $w_{k+1} = w_k - \alpha_k \nabla f(w_k)$ $\qquad \alpha_k > 0$

$$= w_k - \frac{\alpha_k}{n} \sum_{i=1}^{n} \nabla f_i(w_k)$$

⋮

Stochastic Gradient (SG) iteration

$$w_{k+1} = w_k - \alpha_k \nabla f_{i_k}(w_k)$$

where i_k is drawn randomly in $\{1, 2, \dots, n\}$

$\alpha_k > 0$

Motivation: → One iteration of SG only accesses 1 data point

⇒ Clearly cheaper than 1 iteration of GD (n times cheaper)

→ If the dataset is very large and there is an underlying data distribution, maybe seeing all data points at every iteration is not necessary for optimization

Does this work?

- SG does not always converge because it is a randomized algorithm (depends on a random draw → $i_0, i_1, \dots$)
- But possible to show that SG works on average
- And in practice, the performance is really close to the average (even better)

Randomized algorithm: Running the method twice (on the same problem) gives different results

GD: deterministic, always converges (see previous lectures), expensive for large n

SG: randomized, can fail to converge, cheap

(2) Basic SG theory

Assumptions:

- $f \in C^{1,1}_L$, convex, every f_i C^1 convex
- the random indices $\{i_k\}_{k \in \mathbb{N}}$ are drawn at random

so that:

(a) i_k is independent of $i_0, \dots, i_{k-1}$

(b) $\mathbb{E}_{i_k}[\nabla f_{i_k}(w_k)] = \nabla f(w_k)$

(c) $\mathbb{E}_{i_k}[\|\nabla f_{i_k}(w_k)\|^2] \leq \|\nabla f(w_k)\|^2 + \sigma^2$ $\sigma \geq 0$

Remarks

(a) guarantees that every data point gets accessed infinitely often

(b) on average, the stochastic gradient is the true gradient

$\mathbb{E}_{i_k}[\cdot]$: expected value/mean of a quantity with respect to the random index i_k

$$\mathbb{E}_{i_k}\left[\nabla f_{i_k}(w_k)\right] = \sum_{i=1}^{n} \mathbb{P}(i_k = i) \times \nabla f_i(w_k)$$

($\mathbb{P}(i_k = i)$: probability that i_k takes the value i)

(c) The variance of $\|\nabla f_{i_k}(w_k)\|$ is bounded

"the stochastic gradient $\nabla f_{i_k}(w_k)$ doesn't vary too much from the true gradient $\nabla f(w_k)$"

Example: If at every iteration k, i_k is drawn uniformly at random, i.e. $\mathbb{P}(i_k = i) = \frac{1}{n}$ $\forall i = 1 \dots n$

then (a), (b) and (c) are satisfied.

↳ Analysis:

- Recall that the theory of GD is based on the fundamental inequality

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

- For SG, the value of w_{k+1} is random (depends on the value of the random index i_k) $\Rightarrow$ To remove the randomness, we take the

expectation $\mathbb{E}_{i_k}[\cdot]$ on both sides of the inequality

Theorem: Consider the kth iteration of SG $w_{k+1} = w_k - \alpha_k \nabla f_{i_k}(w_k)$.

Then

$$\mathbb{E}_{i_k}[f(w_{k+1})] \leq f(w_k) - \alpha_k \nabla f(w_k)^T \mathbb{E}_{i_k}[\nabla f_{i_k}(w_k)] + \frac{L}{2}\alpha_k^2 \mathbb{E}_{i_k}[\|\nabla f_{i_k}(w_k)\|^2]$$

- $f(w_k)$, $\nabla f(w_k)$: independent of i_k by (a)
- $\mathbb{E}_{i_k}[\nabla f_{i_k}(w_k)] = \nabla f(w_k)$ by (b)
- $\mathbb{E}_{i_k}[\|\nabla f_{i_k}(w_k)\|^2] \leq \|\nabla f(w_k)\|^2 + \sigma^2$ by (c)

↳ With this inequality, we can find good choices for α_k that guarantee decrease in f on average

↳ By combining these inequalities, we obtain convergence rates for SG

One example of CV rate (for convex f)

→ With appropriate choice of α_k, $\forall K \geq 1$,

$$\underbrace{\mathbb{E}_{i_0,\dots,i_{K-1}}}_{\text{expected value w.r.t. } i_0,\dots,i_{K-1}}\left[f(w_K) - \min_{w \in \mathbb{R}^d} f(w) \right] \leq O\left(\frac{1}{\sqrt{K}}\right)$$

	guarantee after K iterations	number of data points accessed in K iterations
GD	$f(w_K) - \min_{w \in \mathbb{R}^d} f(w) \leq O\left(\frac{1}{K}\right)$	nK
SG	$\mathbb{E}\left[f(w_K) - \min_{w \in \mathbb{R}^d} f(w)\right] \leq O\left(\frac{1}{\sqrt{K}}\right)$	K

→ In terms of iterations, GD is best

→ But if we take the cost of every iteration into account, then SG is better!