Statistical Machine Learning

Instructor: Jie Shen

Dept. of Computer Science

February 14, 2020

Gradient Descent

Consider minimization without constraints:

$$\min_{\boldsymbol{w}} \ F(\boldsymbol{w}), \ \ \boldsymbol{w} \in \mathbb{R}^d$$

Gradient Descent:

1. Initialize $\boldsymbol{w}^0$ arbitrarily, e.g. $\boldsymbol{w}^0 = \boldsymbol{0}$
2. For $t = 1, 2, \ldots$

$$\boldsymbol{w}^t = \boldsymbol{w}^{t-1} - \eta_t \nabla F(\boldsymbol{w}^{t-1}) \tag{1}$$

Goal:

- $\boldsymbol{w}^t \to \boldsymbol{w}^*$, where $\boldsymbol{w}^* = \arg\min F(\boldsymbol{w})$
- in few iterations (cheap computation)

Informal Analysis

Why GD “decreases” objective value (under proper conditions)?

Smoothness

Smooth: $F(\boldsymbol{w})$ is smooth if for any $\boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathbb{R}^d$

$$\|\nabla F(\boldsymbol{w}_2) - \nabla F(\boldsymbol{w}_1)\|_2 \leq L \|\boldsymbol{w}_2 - \boldsymbol{w}_1\|_2$$

Examples:

When GD fails to find global optimum

- when it terminates

- when it gets stuck at a non-optimal point

The Big Picture

Convex Optimization

Convex set: $\mathcal{C} \subset \mathbb{R}^d$ is said to be a convex set if for any $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^d$ and any $0 \leq \lambda \leq 1$, $\lambda \boldsymbol{u} + (1-\lambda)\boldsymbol{v} \in \mathcal{C}$

- illustration, examples

Convex Function

Convex function: $F(\boldsymbol{w})$ is said to be a convex function if the set $\mathcal{E} = \{(\boldsymbol{w}, y): \ y \geq F(\boldsymbol{w})\}$ is convex

- $\mathcal{E}$ is called the epigraph of $F(\boldsymbol{w})$
- illustration

Characterization of Convex Functions

Theorem 1

Suppose that $F : \mathbb{R}^d \to \mathbb{R}$ is twice differentiable. The following are equivalent:

1. F *is convex;*
2. $F(\boldsymbol{w}_2) \geq F(\boldsymbol{w}_1) + \langle \nabla F(\boldsymbol{w}_1), \boldsymbol{w}_2 - \boldsymbol{w}_1 \rangle$*;*
3. $\nabla^2 F(\boldsymbol{w})$ *is positive semi-definite.*

Typically use 3 to check the convexity.

Let f and h be convex functions.

- $a \cdot f + b \cdot h$ is convex when $a \geq 0$ and $b \geq 0$
- $f(h)$ may NOT be convex

Convex Program

$$\min_{\boldsymbol{w}} \; F(\boldsymbol{w}), \quad \boldsymbol{w} \in \mathcal{C}.$$

Convex Program: both $F(\boldsymbol{w})$ and $\mathcal{C}$ are convex

- optimality: local optimum $\Longleftrightarrow$ global optimum
- works well
- easy to solve

Convergence Analysis

Theorem 2

Suppose $F(\boldsymbol{w})$ is convex and L-smooth. Pick $0 < \eta \leq 1/L$. Then for all $t \geq 1$,

$$F(\boldsymbol{w}^t) - F(\boldsymbol{w}^*) \leq \frac{\left\|\boldsymbol{w}^0 - \boldsymbol{w}^*\right\|_2^2}{2\eta} \cdot \frac{1}{t}$$

In particular, picking $\eta = 1/L$ gives

$$F(\boldsymbol{w}^t) - F(\boldsymbol{w}^*) \leq \frac{2L\left\|\boldsymbol{w}^0 - \boldsymbol{w}^*\right\|_2^2}{t}.$$

Implications

1. $F(\boldsymbol{w}^t) - F(\boldsymbol{w}^*)$ v.s. $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2$

Implications

2. Iteration complexity

Implications

3. Estimate L

Faster Rate of Convergence

Strongly Convex: for any $\boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathbb{R}^d$

$$\|\nabla F(\boldsymbol{w}_2) - \nabla F(\boldsymbol{w}_1)\|_2 \geq \alpha \|\boldsymbol{w}_2 - \boldsymbol{w}_1\|_2$$

- functions satisfying SC
- not satisfying

Theoretical Guarantee

Theorem 3

Suppose $F(\boldsymbol{w})$ is α-strongly convex and L-smooth. Let $\{\boldsymbol{w}^t\}_{t\geq 1}$ be the iterates generated by GD where $0 < \eta \leq 2/(\alpha + L)$. Then for all $t \geq 1$,

$$\left\|\boldsymbol{w}^t - \boldsymbol{w}^*\right\|_2 \leq \sqrt{1 - \frac{2\eta\alpha L}{\alpha + L}}\left\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\right\|_2.$$

In particular, picking $\eta = 2/(\alpha + L)$ gives

$$\left\|\boldsymbol{w}^t - \boldsymbol{w}^*\right\|_2 \leq \left(1 - \frac{2}{c+1}\right)\left\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\right\|_2$$

where $c \stackrel{def}{=} L/\alpha$ is the condition number.

- converges linearly / geometric rate of convergence
- typically the best one can hope

Implications

For all $t \geq 1$,

$$\begin{aligned} \left\| \boldsymbol{w}^t - \boldsymbol{w}^* \right\|_2 &\leq \left(1 - \frac{2}{c+1}\right)^t \left\| \boldsymbol{w}^0 - \boldsymbol{w}^* \right\|_2 \\ &\leq e^{-\frac{2t}{c+1}} \left\| \boldsymbol{w}^0 - \boldsymbol{w}^* \right\|_2 \qquad \text{(by } 1 + x \leq e^x\text{)} \end{aligned}$$

For any pre-defined error $0 < \epsilon < 1$,

$$\underbrace{t = c \log \frac{\left\| \boldsymbol{w}^0 - \boldsymbol{w}^* \right\|_2}{\epsilon}}_{\text{iteration complexity}} \Longrightarrow \left\| \boldsymbol{w}^t - \boldsymbol{w}^* \right\|_2 \leq \epsilon$$

Implications

3. Estimate α

Overall Computational Complexity

Below #Iter hides the dependence on $\left\|\boldsymbol{w}^0 - \boldsymbol{w}^*\right\|_2$, $c = L/\alpha$

Condition	Guarantee	#Iter
α-SC, L-smooth	$\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2 \leq \epsilon$	$c \log(1/\epsilon)$
L-smooth	$F(\boldsymbol{w}^t) - F(\boldsymbol{w}^*) \leq \epsilon$	L/ϵ

- illustration

- GD solves linear regression efficiently

$$d^2(n+d) \quad \text{v.s.} \quad nd \cdot c \log(1/\epsilon)$$

- note on c

Improve Gradient Descent

Program

$$\min_{\boldsymbol{w}} \; F(\boldsymbol{w}), \quad \text{s.t. } \boldsymbol{w} \in \mathbb{R}^d.$$

- $F(\boldsymbol{w}) = \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|_2^2$ for $y \in \mathbb{R}$
- $F(\boldsymbol{w}) = \frac{1}{n}\sum_{i=1}^{n} \max\{1 - y_i \boldsymbol{x}_i \cdot \boldsymbol{w}, 0\} + 0.5\lambda \|\boldsymbol{w}\|_2^2$ for $y \in \{+1, -1\}$

GD:

- $O(nd)$ to evaluate $\nabla F(\boldsymbol{w})$
- always converges to opt

If computational cost is major concern,

can we boost the efficiency?

Explore the problem structure

Investigate Problem Structure

Suppose

$$F(\boldsymbol{w}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\boldsymbol{w})$$

- linear regression

$$F(\boldsymbol{w}) = \frac{1}{n}\left\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\right\|_2^2, \quad f_i(\boldsymbol{w}) = (y_i - \boldsymbol{x}_i \cdot \boldsymbol{w})^2$$

- SVM

$$F(\boldsymbol{w}) = \frac{1}{n}\sum_{i=1}^{n} \max\{1 - y_i\boldsymbol{x}_i \cdot \boldsymbol{w}, 0\} + 0.5\lambda \left\|\boldsymbol{w}\right\|_2^2$$

$$f_i(\boldsymbol{w}) = \max\{1 - y_i\boldsymbol{x}_i \cdot \boldsymbol{w}, 0\} + 0.5\lambda \left\|\boldsymbol{w}\right\|_2^2$$

- any sample-wise loss

Stochastic Gradient Descent

1. Initialize $\boldsymbol{w}^0$, say $\boldsymbol{w}^0 = \mathbf{0}$
2. For $t = 1, 2, \ldots$

   Uniformly draw i_t from $\{1, 2, \ldots, n\}$, and update

$$\boldsymbol{w}^t = \boldsymbol{w}^{t-1} - \eta_t \nabla f_{i_t}(\boldsymbol{w}^{t-1}) \tag{2}$$

- example, intuition
- time cost per iteration is $O(d)$ (GD needs nd)
- total time = cost/iter · #iter

Convergence Rate for SGD

α-SC, Lipschitz

$$\eta_t \leq \frac{1}{\alpha t}, \quad \mathbb{E}\left[\left\| \boldsymbol{w}^t - \boldsymbol{w}^* \right\|_2 \right] \leq \frac{\log t}{t}$$

convex, Lipschitz

$$\eta_t \leq \frac{1}{\sqrt{t}}, \quad \mathbb{E}\left[F(\boldsymbol{w}^t) - F(\boldsymbol{w}^*) \right] \leq \frac{\log t}{\sqrt{t}}$$

We can modify SGD for faster rate (a rich literature).

GD v.s. SGD

Table 1: Overall computational cost to obtain ϵ opt. error

Condition	GD	SGD
SC	$n \log(1/\epsilon)$	$1/\epsilon$
Convex	n/ϵ	$(1/\epsilon)^2$

SGD wins if

- large-scale data

GD wins if

- small data set
- need high accuracy (i.e. ϵ is small)

SGD is used in

and more...

Other Practical Concerns

- Storage

- real-time decision-making

Online Learning

Initialize the model.

for $t = 1, 2, \dots$

- receive $\boldsymbol{x}_t$
- make prediction $\hat{\boldsymbol{y}}_t$
- receive $\boldsymbol{y}_t$
- evaluate loss $\ell(\hat{\boldsymbol{y}}_t, \boldsymbol{y}_t)$
- update model

Compare to SGD

Mid-term

March 12, 18:30 - 21:00 EST

- linear algebra, calculus
- advanced probability
- linear regression
- gradient descent
- stochastic gradient
- online learning
- statistical machine learning (Mar. 5)