CS 49/149: 21st Century Algorithms (Fall 2018): Lecture 5

Date: 13th September, 2018

Topic: Gradient Descent, Online Convex Optimization, Perceptron.

Scribe: Chongyang Bai

Disclaimer: These notes have not gone through scrutiny and in all probability contain errors. Please email errors to cy@cs.dartmouth.edu.

---

1 Gradient Descent in Convex Optimization

Convex Optimization can be described as follows:

$$\begin{array}{ll} \min\limits_{x} & f(x) \\ \text{s.t.} & x \in S \\ \text{where} & f(x) \text{ is a convex function, } S \text{ is a convex set.} \end{array} \tag{1}$$

Let x_* be the optimal solution of Problem 1, then our goal is to get $\hat{x}$ and a small ϵ, s.t.

$$f(\hat{x}) \leq f(x_*) + \epsilon \tag{2}$$

Unconstrained Convex Optimization. When $S = \mathbb{R}^n$, Problem 1 becomes unconstrained convex optimization (UCO), the whole gradient descent algorithm is simply as follows.

UCO GRADIENT DESCENT

- $x_1 =$ "an arbitrary point" .
- $x_{t+1} = x_t - \eta_t \nabla f(x_t)$

Remark: *η_t is the "step size" that can change according to time t. If f is not differentiable, ∇f can be replaced by a subgradient of f at x.*

Projected Gradient Descent When $S \neq \mathbb{R}^n$we need to "project" the updated x to S if $x \notin S$. Let the updated point be z, we replace z by the nearest point in S to z. The algorithm is as follows.

PROJECTED GRADIENT DESCENT

- $x_1 =$ "an arbitrary point"
- $z_{t+1} = x_t - \eta_t \nabla f(x_t)$
- $x_{t+1} = \pi_s(z_{t+1}) := \operatorname{argmin}\limits_{p \in S} \|z_{t+1} - p\|_2$

Let's look at some examples of projections.

1. $S \equiv$ unit ball $\equiv \{v : \sum_j v_j^2 \leq 1\}, \pi_S(x) = \frac{x}{\|x\|^2}$
2. $S = [-1,1]^n$, for example: $n = 3, x = (\frac{1}{2}, 3, -1) \Rightarrow \pi_S(x) = (\frac{1}{2}, 1, -1)$. Basically, $\forall x_i, x_i \notin [-1,1]$, $\pi_S(x)_i = \underset{u \in [-1,1]}{\text{argmin}} |x_i - u|$.
3. $S = \{\vec{p} : p_i \geq 0 \text{ for } \forall i, \text{ and } \sum_i p_i = 1\}, \pi(x) = \frac{x}{\sum_i x}$ is not true in Euclidean distance, but is easy to compute.

[Comments] To Do!

1. Why choosing the closest point $\pi(x)$? We will show 2 good properties it holds and use them to do error analysis of gradient descent.
2. Projection may not be easy to compute.

Fact 1. $\forall v \notin S, u \in S, \|v - \pi_S(v)\|_2 \leq \|v - u\|_2$ (by definition of projection)

Fact 2. $\forall u \in S, (v - \pi_S(v))^T (u - \pi_S(v)) \leq 0$
Proof of Fact 2: Denote $p = \pi_S(v)$, if Fact 2 is not correct, i.e., $\exists u \in S, (v-p)^T(u-p) > 0$, and let $q = p + \epsilon(u - p)$, we have

$$\begin{aligned} \|q - v\|_2^2 &= \|(p - v) + \epsilon(u - p)\|_2^2 \\ &= \|p - v\|_2^2 + \epsilon^2 \|u - p\|_2^2 - 2\epsilon(v - p)^T(u - p) \end{aligned}$$

when $\epsilon < \frac{2(v-p)^T(u-p)}{\|u-p\|_2^2}$, we get

$$\epsilon^2 \|u - p\|_2^2 - 2\epsilon(v - p)^T(u - p) < 0 \tag{3}$$

so $\|q - v\|_2^2 < \|p - v\|_2^2$, but when ϵ is very small, q can be in S, this contradicts with Fact 1!

1.1 Error Analysis of Gradient Descent

If f is convex, according to definition of subgradient, for any y, we get

$$f(y) \geq f(x) + (y - x)^T \nabla f(x) \tag{4}$$

Let $y = x_*, x = x_t$, denote $err(t) := f(x_t) - f(x_*)$,we get

$$\begin{aligned} f(y) &\geq f(x) + (y - x)^T \nabla f(x) \\ \Rightarrow (x_t - x_*)^T \nabla f(x_t) &\geq f(x_t) - f(x_*) \geq 0 \end{aligned} \tag{5}$$

Remark: *Equation 5 indicates that the angle between gradient direction $\nabla f(x_t)$ and optimal moving direction $x_* - x_t$ is acute.*

Denote $D_t := \|x_t - x_*\|_2^2$ to be the square of distance between point at time t and the optimal point. We put two useful equations here (cosine rules) for later use.

$$\|u - v\|_2^2 = \|u\|_2^2 + \|v\|_2^2 - 2u^t v \tag{6}$$
$$\|u + v\|_2^2 = \|u\|_2^2 + \|v\|_2^2 + 2u^t v \tag{7}$$

Besides, we give two reasonable assumptions.

Assumption 1. $\|x_1 - x_*\|_2 \leq D$

Assumption 2. $\|\nabla f(x)\|_2 \leq \rho$
Now let's jump into the case of unconstrained gradient descent

Unconstrained Gradient Descent

$$\begin{aligned}
err(t) &\leq (x_t - x_*)^T \nabla f(x_t) \\
&= \frac{1}{\eta_t}(x_t - x_*)^T (x_t - x_{t+1}) \\
&= \frac{1}{2\eta_t}(\|x_t - x_*\|_2^2 + \|x_t - x_{t+1}\|_2^2 - \|x_{t+1} - x_*\|_2^2) \qquad (8) \\
&= \frac{1}{2\eta_t}(D_t^2 - D_{t+1}^2) + \frac{\eta_t}{2}\|\nabla f(x_t)\|_2^2 \qquad (9) \\
&\leq \frac{1}{2\eta_t}(D_t^2 - D_{t+1}^2) + \frac{\eta_t}{2}\rho^2 \qquad (10)
\end{aligned}$$

where the first line is by Equation 5, the second line is by replacing $\nabla f(x_t)$ according to the gradient descent update rule, Equation 8 is by applying $x_t - x_*$ to u and $x_t - x_{t+1}$ to v in Equation 6, the last equality is by replacing $x_t - x_{t+1}$ according to gradient update rule, and the last line is by assumption 2.

Sum over t from 1 to T, we get

$$\begin{aligned}
\sum_{t=1}^{T} err(t) &\leq \frac{1}{2\eta_t}(D_1^2 - D_{T+1}^2) + \frac{\eta_t}{2}\rho^2 T \\
&\leq \frac{1}{2\eta_t}D^2 + \frac{\eta_t}{2}\rho^2 T \\
\text{divided by } T \Rightarrow \frac{1}{T}\sum_{t=1}^{T} err(t) &\leq \frac{1}{2\eta_t T}D^2 + \frac{\eta_t}{2}\rho^2 \qquad (11)
\end{aligned}$$

Set $\eta_t = \eta = \frac{D}{\rho}\frac{1}{\sqrt{T}}$, RHS of Equation 11 reaches the min value of $\frac{D\rho}{\sqrt{T}}$. Since we want $\frac{D\rho}{\sqrt{T}} \leq \epsilon$, when $T = \frac{D^2\rho^2}{\epsilon^2}$,$\eta = \frac{D}{\rho}\frac{1}{\sqrt{T}} = \frac{\epsilon}{\rho^2}$, we finally get

$$\frac{1}{T}\sum_{t=1}^{T} f(x_t) \leq f(x_*) + \epsilon \tag{12}$$

Finally, the algorithm can return

- $\hat{x} = \underset{t}{\text{argmin}} f(x_t)$
- $\hat{x} = \frac{1}{T}\sum_{t=1}^{T} x_t$

In both cases, we have $f(\hat{x}) \leq \frac{1}{T}\sum_{t=1}^{T} f(x_t) \leq f(x_*) + \epsilon$

Projected Gradient Descent We'll show how to get to Equation 9, and the rest analysis are eactly the same as Unconstrained case.

$$\begin{aligned} err(t) \leq & (x_t - x_*)^T \nabla f(x_t) \\ = & \frac{1}{\eta_t}(x_t - x_*)^T (x_t - z_{t+1}) \\ = & \frac{1}{2\eta_t}(\|x_t - x_*\|_2^2 + \|x_t - z_{t+1}\|_2^2 - \|z_{t+1} - x_*\|_2^2) \end{aligned} \tag{13}$$

The reasons for the first two lines are the same as Unconstrained Gradient Descent. According to the algorithm, $x_t - z_{t+1} = \eta \nabla f(x_t)$. Since x_{t+1} is the projection of z_{t+1}, $\|z_{t+1} - x_*\| \geq \|x_{t+1} - x_*\| = D_{t+1}$, so Equation 13 $\leq$ Equation 9. ✍

Question: *How to prove* $\|z_{t+1} - x_*\|_2 \geq \|x_{t+1} - x_*\|_2$ *?*

proof:

$$\begin{aligned} \|z_{t+1} - x_*\|_2^2 = & \ \|(z_{t+1} - x_{t+1}) - (x_* - x_{t+1})\|_2^2 \\ = & \ \|z_{t+1} - x_{t+1}\|_2^2 + \|x_* - x_{t+1}\|_2^2 - 2(x_* - x_{t+1})^T (z_{t+1} - x_{t+1}) \\ \geq & \ \|x_{t+1} - x_*\|_2^2 \end{aligned}$$

Due to Fact 2, $(x_* - x_{t+1})^T (z_{t+1} - x_{t+1}) \leq 0$, so the last inequality holds.

1.2 Online Convex Optimization (OCO)

The setting is as follows.

- Space: Convex set S
- At every time t, you play $x_t \in S$
- A convex loss function $f_t : \mathbb{R}^n \mapsto \mathbb{R}$ is fed, so your loss when playing at time t is $f_t(x_t)$
- $\|\nabla f_t(z)\| \leq \rho, \forall z \in S, \forall t = 1, ..., T$

Since $\texttt{alg} = \sum_{t=1}^{T} f_t(x_t)$, $\texttt{opt} = \min_{S} \sum_{t=1}^{T} f_t(x)$. Denote $x_* = \operatorname{argmin}_{x \in S} \sum_{t=1}^{T} f_t(x)$. In the error analysis of gradient descent, replace $f(x)$ by $f_t(x)$, everything still holds. Particularly, if $T = \frac{D^2\rho^2}{\epsilon^2}$ and $\eta = \frac{\epsilon}{\rho^2}$,

$$\text{REGRET} = \frac{1}{T}\sum_{t=1}^{T}(f_t(x_t) - f_t(x_*)) \leq \epsilon \tag{14}$$

Application: Linear Classification Suppose we have data $\{(a_1, b_1), (a_2, b_T), ..., (a_t, b_T)\}$, where $a_t \in \mathbb{R}^n, b_t \in \{-1, 1\}$. We promise that $\exists x_* \in \mathbb{R}^n$, $\|x_*\|_2 = 1$, s.t. $\forall t$, $\frac{b_t(x_*^T a_t)}{\|a_t\|_2} \geq \gamma$. That is, the dataset is linear separable. At everytime t, we 'play' a hyperplane x_t in$\mathbb{R}^n$. we make a mistake $\iff \exists (a_t, b_t) \in$ data, s.t. $\frac{b_t(x_t^T a_t)}{\|a_t\|} \leq 0$. We want to update x_t to x_{t+1}.

Define $f_t(z) = -\frac{b_t(z_t^T a_t)}{\|a_t\|_2}$, then we have

- $f_t(x_t) \geq 0$
- $f_t(x_*) \leq -\gamma$

We update x_t by projected gradient descent,

$$x_{t+1} = \pi_S(x_t - \eta \nabla f_t(x_t))$$

where $S \equiv \{v : \sum_j v_j^2 \leq 1\}$ is the unit ball. Since $\|x_1 - x_*\|_2^2 \leq \|x_1\|_2^2 + \|x_*\|_2^2 = 2$ and $\|\nabla f_t(z)\| = \left\|\nabla(-\frac{b_t(z^T a_t)}{\|a_t\|_2})\right\|_2 = \left\|-\frac{b_t a_t}{\|a_t\|_2}\right\|_2 = 1$, we have $\rho = 1$ and $D = 2$. According to projected gradient descent's error analysis, when $T = \frac{4}{\epsilon^2}$, $\eta = \epsilon$, we get Equation 14.

Since $f_t(x_t) \geq 0$ and $-f_t(x_*) \geq \gamma$, from Equation 14, we have $\gamma \leq$ REGRET $\leq \epsilon$ after making $T = \frac{4}{\epsilon^2}$ classification mistakes. $\gamma \leq \epsilon \Rightarrow T \leq \frac{4}{\gamma^2}$, we get

Theorem 1. The algorithm above cannot bake more than $\frac{4}{\gamma^2}$ mistakes. (PERCEPTRON)

We showed that PERCEPTRON algorithm is an instantiate of online gradient descent.