Khoury College of Computer Sciences
Northeastern University
CS 7870: Seminar in Theoretical Computer Science

Fall 2023
14 November 2023
Scribe: Matthew Casey

Lecture 19 Outline:

- Gradient Descent
- Online convex optimization
- Online Gradient Descent

This lecture introduces the offline unconstrained and constrained gradient descent problems as well as the online gradient descent problem. We begin by presenting an algorithm for unconstrained gradient descent and proving that it converges to the correct answer with a rate $O(\frac{1}{\sqrt{T}})$. We then show how this algorithm can be extended to the constrained version of the problem. Finally, we introduce the online gradient descent problem and prove that a very similar algorithm achieves a regret $\leq \frac{3}{2}GD\sqrt{T}$. In a future lecture we will show this is optimal up to constants. The lecture is partially based on [Haz16, Chapters 2,3].

1 Gradient Descent

As a preliminary we give the definition of a convex function and a convex body.

Definition 1. A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if $\forall x, y \in \mathbb{R}^n, \;\; f(y) - f(x) \geq (\nabla f(x))^\top (y - x)$

Definition 2. A set K is convex if $\forall \alpha \in [0,1], \forall x, y \in K, \;\; \alpha x + (1-\alpha)y \in K$

The objective of gradient descent is to find the minimum value and the minimizer of a convex function, so find $\min_x f(x)$ and $\text{argmin}_x f(x)$. We can do this with or without constraints; in the constrained case, we restrict our search space to points in some convex set. The algorithms for each are similar, and give us similar guarantees. In this lecture we will mainly focus on the unconstrained setting.

1.1 Unconstrained convex optimization

Algorithm 1: Unconstrained Gradient Descent

```
input: $f$, initial point $x_1$, $T$, and set of step sizes $\{\eta_t\}$
for $t = 1$ to $T$ do
    $x_{t+1} = x_t - \eta_t \nabla_t$ where $\nabla_t = \nabla f(x_t)$
return $\bar{x} = \underset{x_t}{\text{argmin}}\, f(x_t)$
```

We will specify what the $\{\eta_t\}$ should be when we do the analysis. T is some parameter that we input that specifies how many iterations of gradient descent we will do. The interesting question to ask is: how close is $\bar{x}$ to x^*, where $x^* = \underset{x}{\operatorname{argmin}} f(x)$? Ideally, we would like to figure out how this difference varies with T, as this will tell us how fast our method converges to the optimal.

We need some preliminaries before the analysis. In order to establish the convergence of gradient descent, it helps to assume that the gradient of any point is bounded by some value G, $\forall x, ||\nabla f(x)|| \leq G$. Now we define the following terms, and set η_t to what is referred to as the Polyak step-size.

$$\begin{aligned} d_t &= ||x_t - x^*|| \\ h_t &= f(x_t) - f(x^*) \\ \eta_t &= \frac{h_t}{||\nabla_t||^2} \end{aligned} \tag{1}$$

So we see that in order to run the algorithm we need to have a good idea of what $f(x^*)$, although crucially this does not require us to know what x^* is.

Theorem 1. If we run Algorithm 1 and get result $\bar{x}$ then

$$f(\bar{x}) - f(x^*) \leq \frac{Gd_1}{\sqrt{T}}$$

Proof.

$$\begin{aligned} f(\bar{x}) &\leq \frac{1}{T}\sum_{t=1}^{T} f(x_t) \\ \implies f(\bar{x}) - f(x^*) &\leq \frac{1}{T}\sum_{t=1}^{T}(f(x_t) - f(x^*)) \\ \implies f(\bar{x}) - f(x^*) &\leq \frac{1}{T}\sum_{t=1}^{T} h_t \end{aligned}$$

Next by using the Cauchy-Shwarz inequality $\langle u, v\rangle^2 \leq \langle u, u\rangle \cdot \langle v, v\rangle$ on the two vectors $u = (1, ...1)$ and $v = (h_1, ..., h_T)$ we see that

$$(\sum_{t=1}^{T} h_t)^2 \leq T\sum_{t=1}^{T} h_t^2$$

Combining with the first part we get

$$f(\bar{x}) - f(x^*) \leq \frac{1}{\sqrt{T}}\sqrt{\sum_{t=1}^{T} h_t^2} \tag{2}$$

Next we derive another useful inequality

$$
\begin{aligned}
d_{t+1}^2 - d_t^2 &= ||x_{t+1} - x^*||^2 - ||x_t - x^*||^2 \\
&= ||x_t - x^* - \eta_t \nabla_t||^2 - ||x_t - x^*||^2 \\
&= ||x_t - x^*||^2 + ||\eta_t \nabla_t||^2 - 2(x_t - x^*)^\top \eta_t \nabla_t - ||x_t - x^*||^2 \\
&\leq ||\eta_t \nabla_t||^2 - 2\eta_t (f(x_t) - f(x^*)) \text{ by the definition of convex function} \\
&= \frac{h_t^2}{||\nabla_t||^2} - 2\frac{h_t^2}{||\nabla_t||^2} \\
&= \frac{-h_t^2}{||\nabla_t||^2}
\end{aligned}
$$

Thus we conclude that

$$\frac{h_t^2}{||\nabla_t||^2} \leq d_t^2 - d_{t+1}^2 \tag{3}$$

Combining with equation (2) we get

$$
\begin{aligned}
f(\bar{x}) - f(x^*) &\leq \frac{1}{\sqrt{T}} \sqrt{\sum_{t=1}^{T} ||\nabla_t||^2 (d_t^2 - d_{t+1}^2)} \\
&\leq \frac{G}{\sqrt{T}} \sqrt{d_1^2} \\
&= \frac{G d_1}{\sqrt{T}}
\end{aligned}
$$

□

Now we state some further results without proof. For strongly convex functions the convergence rate is $O(\frac{1}{T})$, for β-smooth functions the converge rate is also $O(\frac{1}{T})$, and finally for well-conditioned functions the convergence rate is $e^{-\Omega(T)}$.

1.2 Constrained convex optimization

For the constrained version of the problem, we seek to find $\underset{x \in K}{\operatorname{argmin}} f(x)$ for some convex set K and convex function f. We will have to introduce a function $\Pi_k(x)$ which returns the projection of x onto K.

Algorithm 2: Constrained Gradient Descent

```
input: $f$, initial point $x_1$, $T$, and set of step sizes $\{\eta_t\}$
for $t = 1$ to $T$ do
    $y_{t+1} = x_t - \eta_t \nabla_t$ where $\nabla_t = \nabla f(x_t)$
    $x_{t+1} = \Pi_K(y_{t+1})$
return $\bar{x} = \underset{x_t}{\operatorname{argmin}} f(x_t)$
```

This algorithm gives us the same convergence bound as did the unconstrained algorithm, but we leave this without proof. Intuitively this works because for any other point $z \in K$, $||y_t - z|| \geq ||x_t - z||$, so we are still moving towards x^*.

2 Online Gradient Descent

The online gradient descent problem is as follows. At each time step t, there exists a convex function f_t that is not known by the algorithm. The algorithm has to produce some $x_t \in K$, and then is made aware of f_t. Its loss at each time step is $f_t(x_t)$, and the overall performance of the algorithm is $\sum_t f_t(x_t)$. We then calculate the regret of the function as $\sum_t f_t(x_t) - \sum_t f_t(x^*)$ where $x^* = \underset{x \in K}{\operatorname{argmin}} \sum_t f_t(x)$.

```
Algorithm 3: Online Gradient Descent
input: A convex set $K$, a set of step sizes $\{\eta_t\}$
$x_1$ = arbitrary $x \in K$
for $t = 1$ to $T$ do
    Play $x_t$ for $f_t$
    $y_{t+1} = x_t - \eta_t \nabla_t$ where $\nabla_t = \nabla f_t(x_t)$
    $x_{t+1} = \Pi_K(y_{t+1})$
```

Similarly to in the offline setting we assume the gradients are bounded, $||\nabla_t|| \leq G$. We will set the value of η_t in the analysis. Finally we let D be the diameter of K.

Theorem 2. The regret of algorithm 3 is at most $\frac{3}{2}GD\sqrt{T}$ when $\eta_t = \frac{1}{\sqrt{t}}$.

Proof.

$$\text{Regret} = \sum_t (f_t(x_t) - f_t(x^*)) \tag{4}$$

$$f_t(x_t) - f_t(x^*) \leq \nabla_t^\top (x_t - x^*) \text{ by the definition of convex function} \tag{5}$$

$$\begin{aligned}
||x_{t+1} - x^*||^2 &= ||\Pi_K(y_{t+1}) - x^*||^2 \\
&\leq ||y_{t+1} - x^*||^2 \text{ by Pythagorean theorem} \\
&= ||x_t - x^* - \eta_t \nabla_t||^2 \\
&= ||x_t - x^*||^2 + \eta_t^2 ||\nabla_t||^2 - 2\eta_t \nabla_t^\top (x_t - x^*),
\end{aligned}$$

which then gives us that

$$\nabla_t^\top (x_t - x^*) \leq \frac{1}{2\eta_t}(d_t^2 - d_{t+1}^2 + \eta_t^2 ||\nabla_t||^2) \tag{6}$$

Putting together equations (4), (5), and (6) we get that

$$\begin{aligned}
\text{Regret} &\leq \sum_t (\frac{1}{2\eta_t}(d_t^2 - d_{t+1}^2 + \eta_t^2 ||\nabla_t||^2)) \\
&= \frac{1}{2}\sum_t \frac{1}{\eta_t}(d_t^2 - d_{t+1}^2) + \frac{1}{2}\sum_t \eta_t ||\nabla_t||^2 \\
&\leq \frac{1}{2}\sum_t d_t^2 \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right) + \frac{G^2}{2}\sum_t \eta_t \\
&\leq \frac{D^2}{2}\sum_t \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right) + \frac{G^2}{2}\sum_t \eta_t \\
&\leq \frac{D^2}{2\eta_T} + \frac{G^2}{2}\sum_t \eta_t
\end{aligned}$$

Now we try to set these two quantities equal to each other as a proxy for minimizing their sum. Essentially we want to set η_t such that $\sum_t \eta_t \approx \frac{1}{\eta_T}$. We can approximate this using an integral.

$$\begin{aligned}
&\int_1^T \eta_t dt = \frac{1}{\eta_T} \\
&\implies \eta_T = \frac{-1}{\eta_T^2} \cdot \frac{d\eta_T}{dT} \text{ by taking the derivative} \\
&\implies dT \approx \frac{-d\eta_T}{\eta_T^3} \\
&\implies \eta_T \approx \Theta(\frac{1}{\sqrt{T}}) \text{ by integrating}
\end{aligned}$$

Alternatively, we can also show

$$\sum_t \frac{1}{\sqrt{t}} \leq 1 + \int_0^T \frac{1}{\sqrt{t}} dt = 1 + 2\sqrt{T}.$$

So we conclude that we should set $\eta_t = \frac{D}{G\sqrt{t}}$ and subequently derive

$$\text{Regret} \leq \frac{D^2}{2\eta_T} + \frac{G^2}{2}\sum_t \eta_t = \frac{GD\sqrt{T}}{2} + \frac{2GD\sqrt{T}}{2} = \frac{3}{2}GD\sqrt{T}.$$

□

References

[Haz16] Elad Hazan. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.