Section 06: Solutions

Solution:

Section Plan

- Conceptual Review of SGD (slides)
- Problems 2b, 3a-c
- Slides (for Problems 1 and 2a)
- Give students time to look over PyTorch intro notebook

1. Gradient Descent

We will now examine gradient descent algorithm and study the effect of learning rate α on the convergence of the algorithm. Recall from lecture that Gradient Descent takes on the form of $x_{k+1} = x_k - \alpha \nabla f(x_k)$, with an initialization x_0.

(a) Assume that $f\colon \mathbb{R}^n \to \mathbb{R}$ is convex and differentiable, and additionally,

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \text{ for any } x, y,$$

i.e., ∇f is Lipschitz continuous with constant $L \geq 0$.
Show that: Gradient descent with fixed step size $\eta \leq \frac{1}{L}$ satisfies

$$f(x_k) - f(x^*) \leq \frac{\|x_0 - x^*\|^2}{2\eta k}, \qquad k \in \mathbb{N}.$$

I.e., gradient descent has convergence rate $O\left(\frac{1}{k}\right)$.

Hints:

(i) ∇f is Lipschitz continuous with constant $L \geq 0 \implies f(y) \leq f(x) + \nabla f(x)(y - x) + \frac{L}{2}\|y - x\|^2$ for all x, y.

(ii) f is convex $\implies f(x) \leq f(x^*) + \nabla f(x)(x - x^*)$, where x^* is the local minima that the gradient descent algorithm is converging to.

(iii) $2\eta \nabla f(x)(x - x^*) - \eta^2 \|\nabla f(x)\|^2 = \|x - x^*\|^2 - \|x - \eta \nabla f(x) - x^*\|^2$.

Solution:

Proof. For any positive integer k, $x_k = x_{k-1} - \eta \nabla f(x)$, according to the gradient descent algorithm.

Following hint (1), we have

$$\begin{aligned}
f(x_k) &\le f(x_{k-1}) + \nabla f(x_{k-1})(x_k - x_{k-1}) + \frac{L}{2}\|(x_k - x_{k-1}\|^2 \\
&= f(x_{k-1}) - \eta \nabla f(x_{k-1})^2 + \frac{L}{2}\eta^2 \nabla f(x_{k-1})^2 \\
&\le f(x_{k-1}) + \left(-\eta + \frac{\eta}{2}\right) \nabla f(x_{k-1})^2 \qquad \text{(Since } \eta \le \frac{1}{L}\text{)} \\
&= f(x_{k-1}) - \frac{\eta}{2} \nabla f(x_{k-1})^2 \\
&\le f(x^*) + \nabla f(x_{k-1})(x_{k-1} - x^*) - \frac{\eta}{2} \nabla f(x_{k-1})^2 \qquad \text{(Following hint (2))} \\
&= f(x^*) + \frac{1}{2\eta}(2\eta \nabla f(x_{k-1})(x_{k-1} - x^*) - \eta^2 \nabla f(x_{k-1})^2) \\
&\le f(x^*) + \frac{1}{2\eta}(\|x_{k-1} - x^*\|^2 - \|x_{k-1} - \eta \nabla f(x_{k-1}) - x^*\|^2) \qquad \text{(Following hint (3))} \\
&= f(x^*) + \frac{1}{2\eta}(\|x_{k-1} - x^*\|^2 - \|x_k - x^*\|^2).
\end{aligned}$$

Hence, we have

$$f(x_k) - f(x^*) \le \frac{1}{2\eta}\left(\|x_{k-1} - x^*\|^2 - \|x_k - x^*\|^2\right), \qquad k \in \mathbb{N}.$$

Summing up from 1 to k, we get

$$\begin{aligned}
&\sum_{i=1}^{k} [f(x^{(i)}) - f(x^*)] \le \sum_{i=1}^{k} \frac{1}{2\eta}(\|x^{(i-1)} - x^*\|^2 - \|x^{(i)} - x^*\|^2) \\
\implies &\sum_{i=1}^{k} f(x^{(i)}) - k f(x^*) \le \frac{1}{2\eta}(\|x^{(0)} - x^*\|^2 - \|x_k - x^*\|^2) \le \frac{1}{2\eta}\|x^{(0)} - x^*\|^2.
\end{aligned}$$

Since $f(x_k) \le f(x_{k-1})$, we have $f(x_k) \le \frac{1}{k}\sum_{i=1}^{k} f(x_i)$, and thus

$$f(x_k) - f(x^*) \le \frac{1}{2k\eta}(\|x^{(0)} - x^*\|^2). \qquad \square$$

2. Stochastic Gradient Descent

Consider minimizing an average of functions:

$$\min_w \frac{1}{n}\sum_{i=1}^{n} \ell_i(w),$$

where w is a $d-$dimensional vector (or the feature dimension is d). The minimization of the negative of a log-likelihood function can serve as an example. Recall that the (full) gradient descent step is given by

$$w^{(t+1)} = w^{(t)} - \eta \cdot \frac{1}{n}\sum_{i=1}^{n} \nabla \ell_i\big(w^{(t)}\big).$$

The computational cost of a single step here is $\mathcal{O}(dn)$. To reduce cost, one idea is to just use a subset of all samples to approximate the full gradient. Specifically, consider revising the gradient descent step as follows:

$$w^{(t+1)} = w^{(t)} - \eta \cdot \nabla \ell_{I_t}(w^{(t)}),$$

where I_t is chosen randomly within $\{1, 2, ..., n\}$ with equal probability. This is called **stochastic gradient descent (SGD)**, and the computational cost of a single step now reduces to $\mathcal{O}(d)$.

(a) The following two results provide intuitions or foundations for why SGD works.

- $\mathbb{E}_{I_t}\left[\nabla \ell_{I_t}(w^{(t)})\right] = \frac{1}{n}\sum_{i=1}^{n} \nabla \ell_i\big(w^{(t)}\big)$, which is the full gradient. Hence the estimate of gradient is unbiased.
- Let $\ell(w) = \frac{1}{n}\sum_i \ell_i(w)$ and $w^* = \arg\min_w \ell(w)$. Assume $\|w^{(1)} - w^*\|_2^2 \le R$ and $\sup_w \max_i \|\nabla \ell_i(w)\|_2^2 \le G$. Then

$$\mathbb{E}[\ell(\bar{w}) - \ell(w^*)] \le \sqrt{\frac{RG}{T}},$$

where $\bar{w} \coloneqq \frac{1}{T}\sum_{t=1}^{\top} w^{(t)}$. Therefore, the expected error over T iterations is $\mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$. (The proof of this result is provided in the solution part for reference.)

Solution:

We first consider deriving an upper bound for $\mathbb{E}\left[\ell(w^{(t)}) - \ell(w^*)\right]$:

$$\begin{aligned}
\mathbb{E}\Big[\|w^{(t+1)} - w^*\|_2^2\Big] &= \mathbb{E}\Big[\|w^{(t)} - \eta \nabla \ell_{I_t}(w^{(t)}) - w^*\|_2^2\Big] \\
&= \mathbb{E}\Big[\|w^{(t)} - w^*\|_2^2\Big] - 2\eta\,\mathbb{E}\Big[\nabla \ell_{I_t}(w^{(t)})^\top (w^{(t)} - w^*)\Big] + \eta^2\,\mathbb{E}\Big[\|\nabla \ell_{I_t}(w^{(t)})\|_2^2\Big] \\
&\le \mathbb{E}\Big[\|w^{(t)} - w^*\|_2^2\Big] - 2\eta\,\mathbb{E}\Big[\nabla \ell_{I_t}(w^{(t)})^\top (w^{(t)} - w^*)\Big] + \eta^2 G \\
&\le \mathbb{E}\Big[\|w^{(t)} - w^*\|_2^2\Big] - 2\eta\,\mathbb{E}\Big[\ell(w^{(t)}) - \ell(w^*)\Big] + \eta^2 G,
\end{aligned}$$

where the last inequality holds because of the following:

$$\begin{aligned}
\mathbb{E}\Big[\nabla \ell_{I_t}(w^{(t)})^\top (w^{(t)} - w^*)\Big] &\overset{*}{=} \mathbb{E}\Big[\mathbb{E}\Big[\nabla \ell_{I_t}(w^{(t)})^\top (w^{(t)} - w^*) \;\Big|\; I_1, w^{(1)}, ..., I_{t-1}, w^{(t-1)}\Big]\Big] \\
&= \mathbb{E}\left[\frac{1}{n}\sum_i \nabla \ell_i(w^{(t)})^\top (w^{(t)} - w^*)\right] \\
&= \mathbb{E}\Big[\nabla \ell(w^{(t)})^\top (w^{(t)} - w^*)\Big] \\
&\ge \mathbb{E}\Big[\ell(w^{(t)}) - \ell(w^*)\Big],
\end{aligned}$$

where the last inequality holds from the convexity of ℓ. Furthermore, in the right-hand side of starred equality above, the outer expectation is over the variables $I_1, \ldots, I_{t-1}$ and $w^{(1)}, \ldots, w^{(t-1)}$, and the inner expectation is over $I_t, w^{(t)}$ conditioned on the other variables.

Therefore, we've proved that $\mathbb{E}\big[\|w^{(t+1)} - w^*\|_2^2\big] \le \mathbb{E}\big[\|w^{(t)} - w^*\|_2^2\big] - 2\eta\,\mathbb{E}\big[\ell(w^{(t)}) - \ell(w^*)\big] + \eta^2 G$, which implies (from rearrangement) that

$$\mathbb{E}\Big[\ell(w^{(t)}) - \ell(w^*)\Big] \le \frac{1}{2\eta}\Big(\mathbb{E}\Big[\|w^{(t)} - w^*\|_2^2\Big] - \mathbb{E}\Big[\|w^{(t+1)} - w^*\|_2^2\Big] + \eta^2 G\Big). \tag{1}$$

Now note that the convexity of ℓ and Jensen's inequality ensure that $\ell(\bar{w}) \le \frac{1}{T}\sum_{t=1}^{T} \ell(w^{(t)})$, which implies

$$\mathbb{E}[\ell(\bar{w}) - \ell(w^*)] \le \frac{1}{T}\sum_t \mathbb{E}\Big[\ell(w^{(t)}) - \ell(w^*)\Big]. \tag{2}$$

From (1) and (2), we have

$$\begin{aligned}
\mathbb{E}[\ell(\bar{w}) - \ell(w^*)] &\leq \frac{1}{T}\sum_t \mathbb{E}\Big[\ell(w^{(t)}) - \ell(w^*)\Big] \\
&\leq \frac{1}{T}\sum_t \frac{1}{2\eta}\Big(\mathbb{E}\Big[\|w^{(t)} - w^*\|_2^2\Big] - \mathbb{E}\Big[\|w^{(t+1)} - w^*\|_2^2\Big] + \eta^2 G\Big) \\
&= \frac{1}{2\eta T}\Big(\mathbb{E}\Big[\|w^{(1)} - w^*\|_2^2\Big] - \mathbb{E}\Big[\|w^{(T+1)} - w^*\|_2^2\Big]\Big) + \frac{\eta G}{2} \\
&\leq \frac{1}{2\eta T}\mathbb{E}\Big[\|w^{(1)} - w^*\|_2^2\Big] + \frac{\eta G}{2} \\
&\leq \frac{R}{2\eta T} + \frac{\eta G}{2} \\
&= \sqrt{\frac{RG}{T}},
\end{aligned}$$

where the last equality holds by choosing $\eta = \sqrt{\frac{R}{GT}}$.

(b) What disadvantages can SGD have? How can we balance between the noise in updates and computational cost? **Solution:**

By treating SGD as noise-injected gradient descent:

$$\nabla \ell_{I_t}(w^{(t)}) = \mathbb{E}_{I_t}\Big[\nabla \ell_{I_t}(w^{(t)}\Big] + e_t = \frac{1}{n}\sum_{i=1}^{n} \ell_i(w^{(t)}) + e_t,$$

where e_t represents the noise term and is random, we know that the steps taken towards a minimum can be very noisy because the gradient used in updating involves noise. One way to balance the noise in updates and computational cost is to consider a technique called **mini-batching**, which is employed with SGD.

3. Extensions of SGD

(a) Gradient descent requires the full gradient when updating while (standard) SGD utilizes the gradient of one sample when updating. **Mini-batching** is somewhere between the two extremes. That is, we choose a random subset $I_t \subseteq \{1, ..., n\}$ with size $|I_t| = b \ll n$ in the stochastic gradient descent step:

$$w^{(t+1)} = w^{(t)} - \eta \cdot \frac{1}{b}\sum_{i_t \in I_t} \nabla \ell_{i_t}(w^{(t)}).$$

With mini-batching, we have the following results:

- $\mathbb{E}_{I_t}\big[\frac{1}{b}\sum_{i_t \in I_t} \nabla \ell_{i_t}(w^{(t)})\big] = \frac{1}{n}\sum_{i=1}^{n} \nabla \ell_i(w^{(t)})$: we still have an unbiased estimate of the full gradient.
- Compared to standard SGD, variance of the gradient estimate is reduced approximately by $\frac{1}{b}$.
- Computational cost for each step now becomes $\mathcal{O}(db)$.

Remark: By matrix computations (computing b gradients at a time) and parallelization, we can denoise the estimated gradients without increasing much computational cost (for batch size b that is not large).

(b) How should we choose the batch size? **Solution:**

The choice of the optimal batch size is not an easy question, and there is no standard answer to it. However, we still try to provide some important intuitions regarding the choice of batch size. Firstly, when the objective function (to be minimized) behaves "better" (e.g., Lipschitz continuous, strong convex) than convex functions, the difference in the convergence rates between GD and SGD becomes significant, suggesting a nontrivial gain of having a faster convergence rate and hence we should consider relatively larger batch size. Secondly, a smaller batch size yields less stable gradient estimates, suggesting that we shall employ a fairly small step size/learning rate. An increase in the batch size can be paired with an increase in the step size/learning rate.

(c) Are there other extensions or variants of the basic stochastic gradient descent algorithm?

Solution:

Many improvements, which are listed below, on the basic SGD algorithm have been developed and used.

- Implicit updates (ISGD)
- Momentum
- Averaged stochastic gradient descent
- Adaptive gradient algorithm (AdaGrad)
- Root Mean Square Propagation (RMSProp)
- Adaptive Moment Estimation (Adam)

Basically, these methods consider to fine-tune the step size parameter, take previous update magnitude into account, or introduce the second moments of the gradients when updating. For example, Momentum remembers the previous update magnitude so that $w^{(t)}$ tends to keep traveling in the same direction, preventing oscillations:

$$w^{(t+1)} = w^{(t)} - \eta \nabla \ell_{I_t}(w^{(t)}) + \alpha(w^{(t)} - w^{(t-1)}).$$

Adam, as another example, considers to tune to step size with the second moments of the gradients:

$$w^{(t+1)} = w^{(t)} - \eta G\Big(\nabla \ell^{(t)}, \nabla \ell^{(t-1)}, ..., (\nabla \ell^{(t)})^2, (\nabla \ell^{(t-1)})^2, ...\Big),$$

where $\nabla \ell^{(t)} = \nabla \ell_{I_t}(w^{(t)})$ and G is a function that involves element-wise square of all previous gradients. The paper below provides more details on Adam: https://arxiv.org/pdf/1412.6980.pdf.