Lecture 9: Accelerating SGD with variance reduction and averaging.

CS4787 — Principles of Large-Scale Machine Learning Systems

Recall: In last week's homework, we looked at a case for which SGD can still converge asymptotically to the global optimum even with a constant step size. This happened when the magnitude of the gradient samples $\nabla f_{\tilde{i}}(w_t)$ was going to zero as our iterates w_t approached the optimum point. In today's lecture, we'll look at ways we can modify our SGD algorithm to make this happen automatically. First, recall that (for both convex and non-convex optimization), if the largest eigenvalue of the second derivative is always bounded in magnitude by L, then for SGD we had

$$\mathbf{E}\left[f(w_{t+1})\right] = \mathbf{E}\left[f(w_t - \alpha_t \nabla f_{\tilde{i}_t}(w_t))\right].$$

Next, we apply Taylor's theorem. Since a few people have asked about it in office hours, I'm going to go back and give a bit more detail about these Taylor's theorem results. Taylor's theorem with the Lagrange form of the remainder says that for any function $f : \mathbb{R} \to \mathbb{R}$ continuously differentiable up to order $k+1$,

$$h(a+x) = h(a) + x \cdot h'(a) + \frac{1}{2}x^2 \cdot h''(a) + \cdots + \frac{1}{i!}x^i \cdot h^{(i)}(a) + \cdots + \frac{1}{k!}x^k \cdot h^{(k)}(a) + \frac{1}{(k+1)!}x^{k+1} \cdot h^{(k+1)}(b)$$

for some b in the open interval between a and $a+x$. In particular, for $k=1$,

$$h(a+x) = h(a) + x \cdot h'(a) + \frac{1}{2}x^2 \cdot h''(b).$$

If we define $h(\alpha_t) = f(w_{t+1}) = f(w_t - \alpha_t \nabla f_{\tilde{i}_t}(w_t))$, then applying Taylor's theorem to this by letting x be α_t and a be 0 gives us

$$\begin{aligned} h(\alpha_t) &= h(0) + \alpha_t \cdot h'(0) + \frac{1}{2}\alpha_t^2 \cdot h''(b) \\ &= f(w_t) + \alpha_t \cdot \left(-\nabla f_{\tilde{i}_t}(w_t)^T \nabla f(w_t)\right) + \frac{1}{2}\alpha_t^2 \left(\nabla f_{\tilde{i}_t}(w_t)^T \nabla^2 f(w_t - b\nabla f_{\tilde{i}_t}(w_t)) \nabla f_{\tilde{i}_t}(w_t)\right) \\ &= f(w_t) - \alpha_t \cdot \nabla f_{\tilde{i}_t}(w_t)^T \nabla f(w_t) + \frac{1}{2}\alpha_t^2 \nabla f_{\tilde{i}_t}(w_t)^T \nabla^2 f(\zeta_t) \nabla f_{\tilde{i}_t}(w_t) \end{aligned}$$

where for simplicity we define $\zeta_t = w_t - b\nabla f_{\tilde{i}_t}(w_t)$. Now taking the expected value of both sides and using our bound on the largest eigenvalue of the second derivative, and if we require that $\alpha_t L \le 1$, we get

$$\begin{aligned} \mathbf{E}\left[f(w_{t+1})\right] &= \mathbf{E}\left[f(w_t)\right] - \alpha_t \cdot \mathbf{E}\left[\nabla f_{\tilde{i}_t}(w_t)^T \nabla f(w_t)\right] + \frac{1}{2}\alpha_t^2 \cdot \mathbf{E}\left[\nabla f_{\tilde{i}_t}(w_t)^T \nabla^2 f(\zeta_t) \nabla f_{\tilde{i}_t}(w_t)\right] \\ &\le \mathbf{E}\left[f(w_t)\right] - \alpha_t \cdot \mathbf{E}\left[\left\|\nabla f(w_t)\right\|^2\right] + \frac{\alpha_t^2 L}{2} \cdot \mathbf{E}\left[\left\|\nabla f_{\tilde{i}_t}(w_t)\right\|^2\right] \\ &= \mathbf{E}\left[f(w_t)\right] - \alpha_t \cdot \mathbf{E}\left[\left\|\nabla f(w_t)\right\|^2\right] + \frac{\alpha_t^2 L}{2} \cdot \mathbf{E}\left[\left\|\nabla f(w_t)\right\|^2\right] + \frac{\alpha_t^2 L}{2} \cdot \mathbf{E}\left[\left\|\nabla f_{\tilde{i}_t}(w_t) - \nabla f(w_t)\right\|^2\right] \\ &= \mathbf{E}\left[f(w_t)\right] - \alpha_t \left(1 - \frac{\alpha_t L}{2}\right) \cdot \mathbf{E}\left[\left\|\nabla f(w_t)\right\|^2\right] + \frac{\alpha_t^2 L}{2} \cdot \mathbf{E}\left[\left\|\nabla f_{\tilde{i}_t}(w_t) - \nabla f(w_t)\right\|^2\right] \\ &= \mathbf{E}\left[f(w_t)\right] - \frac{\alpha_t}{2} \cdot \mathbf{E}\left[\left\|\nabla f(w_t)\right\|^2\right] + \underbrace{\frac{\alpha_t^2 L}{2} \cdot \mathbf{E}\left[\left\|\nabla f_{\tilde{i}_t}(w_t) - \nabla f(w_t)\right\|^2\right]}_{\text{second-order variance/error term}}. \end{aligned}$$

We saw that as long as this second-order term is bounded, we can show that SGD with a constant learning rate will converge to a noise ball. In this lecture, we'll look at methods we can use to *reduce* this second-order term, which can sometimes result in faster and more scalable convergence.

Increasing minibatch sizes. One thing we can try to do is use a minibatch size that becomes larger over time. If we use a constant learning rate and a minibatch of size B_t at time t, then the above expression looks like

$$\mathbf{E}\left[f(w_{t+1})\right] \le \mathbf{E}\left[f(w_t)\right] - \frac{\alpha}{2} \cdot \mathbf{E}\left[\left\|\nabla f(w_t)\right\|^2\right] + \underbrace{\frac{\alpha^2 L}{2} \cdot \mathbf{E}\left[\left\|\frac{1}{B_t}\sum_{b=1}^{B_t} \nabla f_{\tilde{i}_{b,t}}(w_t) - \nabla f(w_t)\right\|^2\right]}_{\text{second-order variance/error term}}.$$

If the variance of an individual example gradient is bounded by σ^2 as

$$\frac{1}{n}\sum_{i=1}^{n} \left\|\nabla f_i(w_t) - \nabla f(w_t)\right\|^2 \le \sigma^2,$$

then by the analysis we've already done we know that

$$\mathbf{E}\left[f(w_{t+1})\right] \le \mathbf{E}\left[f(w_t)\right] - \frac{\alpha}{2} \cdot \mathbf{E}\left[\left\|\nabla f(w_t)\right\|^2\right] + \frac{\alpha^2 \sigma^2 L}{2B_t}$$

which implies that

$$\frac{\alpha}{2} \cdot \mathbf{E}\left[\left\|\nabla f(w_t)\right\|^2\right] \le \mathbf{E}\left[f(w_t)\right] - \mathbf{E}\left[f(w_{t+1})\right] + \frac{\alpha^2 \sigma^2 L}{2B_t}.$$

Summing this up over T iterations of SGD, and telescoping the sum as usual, we get

$$\begin{aligned}\frac{\alpha}{2}\sum_{t=0}^{T-1} \mathbf{E}\left[\left\|\nabla f(w_t)\right\|^2\right] &\le \mathbf{E}\left[f(w_0)\right] - \mathbf{E}\left[f(w_T)\right] + \sum_{t=0}^{T-1} \frac{\alpha^2 \sigma^2 L}{2B_t} \\ &\le f(w_0) - f^* + \sum_{t=0}^{T-1} \frac{\alpha^2 \sigma^2 L}{2B_t}\end{aligned}$$

which implies that

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbf{E}\left[\left\|\nabla f(w_t)\right\|^2\right] \le \frac{2(f(w_0) - f^*)}{\alpha T} + \frac{\alpha \sigma^2 L}{T}\sum_{t=0}^{T-1} \frac{1}{B_t}.$$

This means that any increasing batch size scheme is going to converge, and if the sum of the reciprocals of the batch sizes converges, then SGD with this scheme will converge at a rate of $1/T$.

We can do a similar analysis for convex problems...I won't discuss this in class but there's a great analysis in Chapter 5 of "Optimization Methods for Large-Scale Machine Learning" if you are curious.

Polyak averaging. Intuition: SGD is converging to a "noise ball" where the iterates are randomly jumping around some space surrounding the optimum. We can think about these iterates as random samples that approximate the optimum.

What can we do when we have a bunch of random samples that approximate something to improve the precision of our estimate?

Technique: run regular SGD and just average the iterates. That is,

$$\begin{aligned}w_{t+1} &= w_t - \alpha_t \nabla f_{\tilde{i}_t}(w_t) \\ \bar{w}_{t+1} &= \frac{t}{t+1} \cdot \bar{w}_t + \frac{1}{t+1} \cdot w_{t+1};\end{aligned}$$

eventually we output the average $\bar{w}_T$ at the end of execution. This is equivalent to writing

$$\bar{w}_T = \frac{1}{T}\sum_{t=1}^{T} w_t.$$

To gain intuition about Polyak averaging, let's look at a simple one-dimensional quadratic...

$$f(w) = \frac{1}{2}w^2$$

with example gradients

$$\nabla f_i(w) = w + u_i$$

where $u \sim \mathcal{N}(0,1)$ is a random normally-distributed random variable with mean 0 and variance σ^2. Suppose that we run SGD on this with a constant learning rate α. Our update step will be

$$w_{t+1} = w_t - \alpha w_t - \alpha u_{\tilde{i}_t} = (1-\alpha)w_t - \alpha u_{\tilde{i}_t}.$$

Applying this recursively, we get

$$w_T = (1-\alpha)^T w_0 - \underbrace{\alpha \sum_{t=0}^{T-1}(1-\alpha)^{T-1-t} u_{\tilde{i}_t}}\text{ noise term.}$$

What is the variance of this noise term? Well, assuming that these normal random variables are independent, we have that

$$\begin{aligned}
\mathbf{E}\left[\left(\alpha \sum_{t=0}^{T-1}(1-\alpha)^{T-1-t} u_{\tilde{i}_t}\right)^2\right] &= \alpha^2 \sum_{t=0}^{T-1} \mathbf{E}\left[(1-\alpha)^{2(T-1-t)} u_{\tilde{i}_t}^2\right] \\
&= \alpha^2 \sum_{t=0}^{T-1}(1-\alpha)^{2(T-1-t)} \cdot \sigma^2 \\
&\leq \alpha^2\sigma^2 \sum_{k=0}^{\infty}(1-\alpha)^{2k} \\
&= \alpha^2\sigma^2 \frac{1}{1-(1-\alpha)^2} = \frac{\alpha\sigma^2}{2-\alpha}.
\end{aligned}$$

On the other hand, if we use averaging, we get

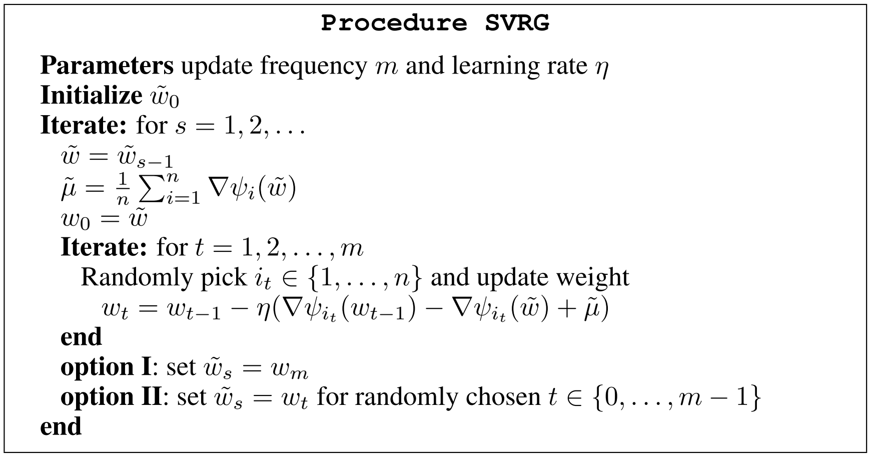
$$\bar{w}_T = \frac{1}{T}\sum_{k=0}^{T-1}\left((1-\alpha)^k w_0 - \alpha \sum_{t=0}^{k-1}(1-\alpha)^{k-1-t} u_{\tilde{i}_t}\right).$$

If we look at the variance of the noise term here, we get

$$\begin{aligned}
\mathbf{E}\left[\left(\frac{1}{T}\sum_{k=0}^{T-1}\left(\alpha\sum_{t=0}^{k-1}(1-\alpha)^{k-1-t}u_{\tilde{i}_t}\right)\right)^2\right] &= \frac{\alpha^2}{T^2}\mathbf{E}\left[\left(\sum_{t=0}^{T-2}\sum_{k=t+1}^{T-1}(1-\alpha)^{k-1-t}u_{\tilde{i}_t}\right)^2\right] \\
&= \frac{\alpha^2}{T^2}\sum_{t=0}^{T-2}\mathbf{E}\left[\left(\sum_{k=t+1}^{T-1}(1-\alpha)^{k-1-t}u_{\tilde{i}_t}\right)^2\right] \\
&= \frac{\alpha^2}{T^2}\sum_{t=0}^{T-2}\mathbf{E}\left[\left(\sum_{i=0}^{T-1-(t+1)}(1-\alpha)^{i}u_{\tilde{i}_t}\right)^2\right] \\
&= \frac{\alpha^2}{T^2}\sum_{t=0}^{T-2}\left(\sum_{i=0}^{T-1-(t+1)}(1-\alpha)^{i}\right)^2\sigma^2 \\
&\leq \frac{\alpha^2}{T^2}\sum_{t=0}^{T-2}\left(\sum_{i=0}^{\infty}(1-\alpha)^{i}\right)^2\sigma^2 \\
&= \frac{\alpha^2}{T^2}\sum_{t=0}^{T-2}\left(\frac{1}{1-(1-\alpha)}\right)^2\sigma^2 \\
&= \frac{1}{T^2}\sum_{t=0}^{T-2}\sigma^2 \\
&\leq \frac{\sigma^2}{T}.
\end{aligned}$$

This is actually decreasing with T, even though our baseline result without averaging wasn't!

Variance reduction. Idea: modify the update step to decrease the variance of SGD. There are many ways to do this. From the original Stochastic Variance Reduced Gradient (SVRG) paper, here's one way to do it.



```
Procedure SVRG
Parameters update frequency $m$ and learning rate $\eta$
Initialize $\tilde{w}_0$
Iterate: for $s = 1, 2, \ldots$
  $\tilde{w} = \tilde{w}_{s-1}$
  $\tilde{\mu} = \frac{1}{n}\sum_{i=1}^{n}\nabla\psi_i(\tilde{w})$
  $w_0 = \tilde{w}$
  Iterate: for $t = 1, 2, \ldots, m$
    Randomly pick $i_t \in \{1, \ldots, n\}$ and update weight
      $w_t = w_{t-1} - \eta(\nabla\psi_{i_t}(w_{t-1}) - \nabla\psi_{i_t}(\tilde{w}) + \tilde{\mu})$
  end
  option I: set $\tilde{w}_s = w_m$
  option II: set $\tilde{w}_s = w_t$ for randomly chosen $t \in \{0, \ldots, m-1\}$
end
```