Lecture 5: Stochastic Gradient Descent

CS4787 — Principles of Large-Scale Machine Learning Systems

Combining two principles we already discussed into one algorithm.

- Principle: Write your learning task as an optimization problem and solve it with a scalable optimization algorithm.
- Principle: Use subsampling to estimate a sum with something easier to compute.

Recall: we *parameterized* the hypotheses we wanted to evaluate with parameters $w \in \mathbb{R}^d$, and want to solve the problem

$$\text{minimize: } R(h_w) = \frac{1}{n}\sum_{i=1}^{n} L(h_w(x_i), y_i) = f(w) = \frac{1}{n}\sum_{i=1}^{n} f_i(w) \text{ over } w \in \mathbb{R}^d.$$

Stochastic gradient descent (SGD). Basic idea: **in gradient descent, just replace the full gradient (which is a sum) with a single gradient example**. Initialize the parameters at some value $w_0 \in \mathbb{R}^d$, and decrease the value of the empirical risk iteratively by sampling a random index $\tilde{i}_t$ uniformly from $\{1, \ldots, n\}$ and then updating

$$w_{t+1} = w_t - \alpha_t \cdot \nabla f_{\tilde{i}_t}(w_t)$$

where as usual w_t is the value of the parameter vector at time t, α_t is the *learning rate* or *step size*, and ∇f_i denotes the gradient of the loss function of the ith training example. Compared with gradient descent and Newton's method, SGD is simple to implement and runs each iteration faster.

A potential objection: **this is not necessarily going to be decreasing the loss at every step!** So we can't demonstrate convergence by using a proof like the one we used for gradient descent, where we showed that the loss decreases at every iteration of the algorithm. The fact that SGD doesn't always improve the loss at each iteration motivates the question: **does SGD even work? And if so, why does SGD work?**

Demo. Gradient descent versus stochastic gradient descent on linear regression.

Why might it be fine to get an approximate solution to an optimization problem for training?

Takeaway:

Why does SGD work? Unlike GD, SGD does not necessarily decrease the value of the loss at each step. Let's just try to analyze it in the same way that we did with gradient descent and see what happens. But first, we need some new assumption that characterizes *how far* the gradient samples can be from the true gradient. Assume that, for some constant $\sigma^2 > 0$, the mean-squared error of our gradient samples from the true gradient is bounded, for all $w \in \mathbb{R}^d$, by

$$\mathbf{E}\left[\|\nabla f_i(w) - \nabla f(w)\|^2\right] = \mathbf{E}\left[\|\nabla f_i(w)\|^2\right] - \|\nabla f(w)\|^2 \leq \sigma^2.$$

Here the expectation is taken over a uniform random selection of a component loss function f_i. In other words, since $\mathbf{E}\left[\nabla f_i(w)\right] = \nabla f(w)$, this is a global bound on the variance of the gradient samples. As before, we will also assume that for some constant $L > 0$, for all x in the space and for any vector $u \in \mathbb{R}^d$,

$$\left|u^T \nabla^2 f(x) u\right| \leq L \|u\|^2.$$

From here, we can analyze SGD like we did with gradient descent, first *without assuming convexity* and using a constant step size. From Taylor's theorem, using the same argument as for gradient descent, we can get

$$f(w_{t+1}) \leq f(w_t) - \alpha \nabla f_{\tilde{i}_t}(w_t)^T \nabla f(w_t) + \frac{\alpha^2 L}{2} \left\|\nabla f_{\tilde{i}_t}(w_t)\right\|^2.$$

Now we're faced with a problem. The term

$$-\alpha \nabla f_{\tilde{i}_t}(w_t)^T \nabla f(w_t)$$

is not necessarily nonnegative, so we're not necessarily making any progress in the loss. The key insight: we are making progress **in expectation**. If we take the expected value of both sides of this expression (where the expectation is taken over the randomness in the sample selection $\tilde{i}_t$), we get

$$\begin{aligned}
\mathbf{E}\left[f(w_{t+1})\right] &\leq \mathbf{E}\left[f(w_t) - \alpha \nabla f_{\tilde{i}_t}(w_t)^T \nabla f(w_t) + \frac{\alpha^2 L}{2} \left\|\nabla f_{\tilde{i}_t}(w_t)\right\|^2\right] \\
&\leq \mathbf{E}\left[f(w_t)\right] - \alpha \mathbf{E}\left[\|\nabla f(w_t)\|^2\right] + \frac{\alpha^2 L}{2}\left(\sigma^2 + \mathbf{E}\left[\|\nabla f(w_t)\|^2\right]\right) \\
&= \mathbf{E}\left[f(w_t)\right] - \left(\alpha - \frac{\alpha^2 L}{2}\right) \mathbf{E}\left[\|\nabla f(w_t)\|^2\right] + \frac{\alpha^2 \sigma^2 L}{2}.
\end{aligned}$$

Assuming that $\alpha L < 1$, we can simplify this to

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

Rearranging the terms, summing up over T iterations, and telescoping the sum,

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

Rearranging and dividing both sides by $\alpha T/2$, as we did in the analysis of GD, and noticing that $f(w_T) \geq f^*$, where f^* is the global minimum of f,

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

The term on the left is the expected squared-norm of the gradient of a point randomly chosen from the trajectory of SGD.

How should we interpret this?

So SGD with constant step size converges to a noise ball!

Even if we run for a very large number of iterations,

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

For many applications this is fine...but it seems somehow lacking.

What if we want an algorithm that actually converges to the optimum? Intuition: for the constant step size approach, we converge down to a gradient magnitude that is proportional to the step size. So if we use a *decreasing step size scheme*, can we get arbitrarily small gradients? That is, we can run the update

$$w_{t+1} = w_t - \alpha_t \nabla f_{\tilde{i}_t}(w_t).$$

Using the same analysis as before, but with α_t in place of α, and assuming that $\alpha_t L < 1$, we can get

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

Rearranging the terms, summing up over T iterations, and telescoping the sum,

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

If we define τ as being the index of a random output model that is selected at random from a weighted distribution over the iterates of SGD, such that for $t \in \{0,\ldots,T-1\}$

$$\mathbf{P}\left(\tau = t\right) = \frac{\alpha_t}{\sum_{t=0}^{T-1}\alpha_t},$$

then

$$\begin{aligned}\mathbf{E}\left[\|\nabla f(w_\tau)\|^2\right] &= \sum_{t=0}^{T-1}\frac{\alpha_t}{\sum_{s=0}^{T-1}\alpha_s}\cdot\mathbf{E}\left[\|\nabla f(w_t)\|^2\right] = 2\left(\sum_{s=0}^{T-1}\alpha_s\right)^{-1}\cdot\sum_{t=0}^{T-1}\frac{\alpha_t}{2}\mathbf{E}\left[\|\nabla f(w_t)\|^2\right] \\ &\leq 2\left(\sum_{s=0}^{T-1}\alpha_s\right)^{-1}\cdot\left(\mathbf{E}\left[f(w_0)\right] - \mathbf{E}\left[f(w_T)\right] + \sum_{t=0}^{T-1}\frac{\alpha_t^2\sigma^2 L}{2}\right).\end{aligned}$$

The norm of the gradient of the output w_τ will be guaranteed to go to zero if

$$\sum_{t=0}^{T-1}\alpha_t \text{ grows much faster than } \sum_{t=0}^{T-1}\alpha_t^2.$$

One example of such a step size rule is $\alpha_t = \frac{1}{L\cdot\sqrt{t+1}}$. Then we have

$$\sum_{t=0}^{T-1}\alpha_t = \sum_{t=0}^{T-1}\frac{1}{L\sqrt{t+1}} \geq \int_1^{T+1}\frac{1}{L\sqrt{x}}\,dx = \frac{2\left(\sqrt{T+1}-1\right)}{L} \quad \text{and} \quad \sum_{t=0}^{T-1}\alpha_t^2 = \sum_{t=0}^{T-1}\frac{1}{L^2(t+1)} \leq 1 + \int_1^T \frac{1}{L^2 x}\,dx = \frac{\log(T)+1}{L^2}.$$

With this, we get

$$\mathbf{E}\left[\|\nabla f(w_\tau)\|^2\right] \leq 2\left(\frac{2\left(\sqrt{T+1}-1\right)}{L}\right)^{-1}\cdot\left(\mathbf{E}\left[f(w_0)\right] - \mathbf{E}\left[f(w_T)\right] + \frac{\log(T)+1}{L^2}\right) = \mathcal{O}\left(\frac{\log(T)}{L\sqrt{T}}\right).$$

This is indeed going to go to zero as $T \to \infty$.

How does this compare to the expression that we got for gradient descent?

Gradient descent for strongly convex objectives. This was without assuming strong convexity. But how does SGD perform on strongly convex problems? As before, we start from this sort of expression

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

and apply the Polyak–Lojasiewicz condition,

$$\left\|\nabla f(x)\right\|^2 \geq 2\mu\left(f(x) - f^*\right);$$

this gives us

$$\mathbf{E}\left[f(w_{t+1})\right] \leq \mathbf{E}\left[f(w_t)\right] - \mu\alpha\mathbf{E}\left[f(w_t) - f^*\right] + \frac{\alpha^2\sigma^2 L}{2}.$$

Subtracting f^* from both sides, we get

$$\mathbf{E}\left[f(w_{t+1}) - f^*\right] \leq (1-\mu\alpha)\mathbf{E}\left[f(w_t) - f^*\right] + \frac{\alpha^2\sigma^2 L}{2}.$$

Now subtracting the fixed point from both sides gives us

$$\begin{aligned}\mathbf{E}\left[f(w_{t+1}) - f^*\right] - \frac{\alpha^2\sigma^2 L}{2\mu\alpha} &\leq (1-\mu\alpha)\mathbf{E}\left[f(w_t) - f^*\right] + \frac{\alpha^2\sigma^2 L}{2} - \frac{\alpha^2\sigma^2 L}{2\mu\alpha} \\ &= (1-\mu\alpha)\left(\mathbf{E}\left[f(w_t) - f^*\right] - \frac{\alpha^2\sigma^2 L}{2\mu\alpha}\right).\end{aligned}$$

Now applying this recursively,

$$\mathbf{E}\left[f(w_T) - f^*\right] - \frac{\alpha^2\sigma^2 L}{4\mu\alpha} \leq (1-\mu\alpha)^K\left(f(w_0) - f^* - \frac{\alpha^2\sigma^2 L}{2\mu\alpha}\right),$$

and so since $(1-\mu\alpha) \leq \exp(-\mu\alpha)$,

$$\mathbf{E}\left[f(w_T) - f^*\right] \leq \exp(-\mu\alpha K)\cdot\left(f(w_0) - f^*\right) + \frac{\alpha\sigma^2 L}{2\mu}.$$

What can we learn from this expression?