Stochastic Gradient Descent

Jue Guo

December 9, 2024

1 Stochastic Gradient Update

In deep learning, the objective function is usually the average of the loss functions for each example in the training dataset. Given a training dataset of n examples, we assume that $f_i(\mathbf{x})$ is the loss function with respect to the training example of index i, where $\mathbf{x}$ is the parameter vector. Then we arrive at the objective function

$$f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x})$$

The gradient of the objective function at $\mathbf{x}$ is computed as

$$\nabla f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(\mathbf{x})$$

If **gradient descent** is used, the computational cost for each independent variable iteration is $\mathcal{O}(n)$, which grows linearly with n. Therefore, when the training dataset is larger, the cost of gradient descent for each iteration will be higher.

Stochastic gradient descent (SGD) reduces computational cost at each iteration. At each iteration of stochastic gradient descent, we uniformly sample an index $i \in \{1, \ldots, n\}$ for data examples at random, and compute the gradient $\nabla f_i(\mathbf{x})$ to update $\mathbf{x}$:

$$\mathbf{x} \leftarrow \mathbf{x} - \eta \nabla f_i(\mathbf{x})$$

where η is the learning rate. We can see that the computational cost for each iteration drops from $\mathcal{O}(n)$ of the gradient descent to the constant $\mathcal{O}(1)$. Moreover, we want to emphasize that the stochastic gradient $\nabla f_i(\mathbf{x})$ is an *unbiased estimate* of the full gradient $\nabla f(\mathbf{x})$ because

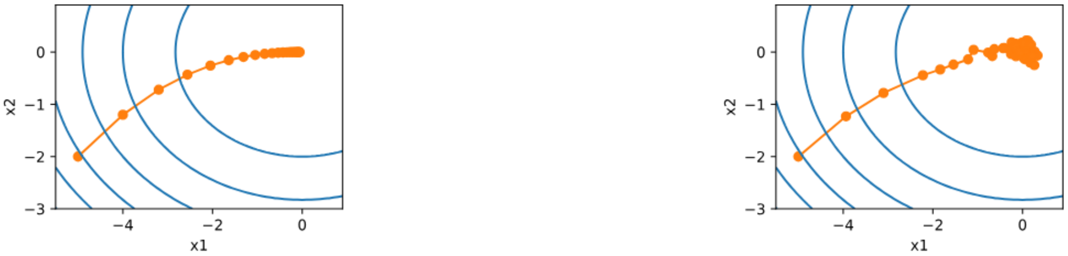
$$\mathbb{E}_i \nabla f_i(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(\mathbf{x}) = \nabla f(\mathbf{x})$$

This means that, on average, the stochastic gradient is a good estimate of the gradient.

Why is Stochastic Gradient Unbiased? The stochastic gradient $\nabla f_i(\mathbf{x})$ that you compute for a single data point i is a good approximation of the full gradient $\nabla f(\mathbf{x})$ that you would compute over the entire dataset.

In other words, while the gradient you compute from a single data point $\nabla f_i(\mathbf{x})$ might vary from the true full gradient $\nabla f(\mathbf{x})$, on average, across many iterations, these stochastic gradients will give you the correct direction for optimization.

As we can see from figure 1, *the trajectory of the variables in the stochastic gradient descent is much more noisy than the one we observed in gradient descent.* This is due to the stochastic nature of the gradient. That is, even when we arrive near the minimum, we are still subject to the uncertainty injected by the instantaneous gradient via $\eta \nabla f_i(\mathbf{x})$.



(a) Gradient Descent

(b) Stochastic Gradient Descent

Figure 1: GD vs SGD

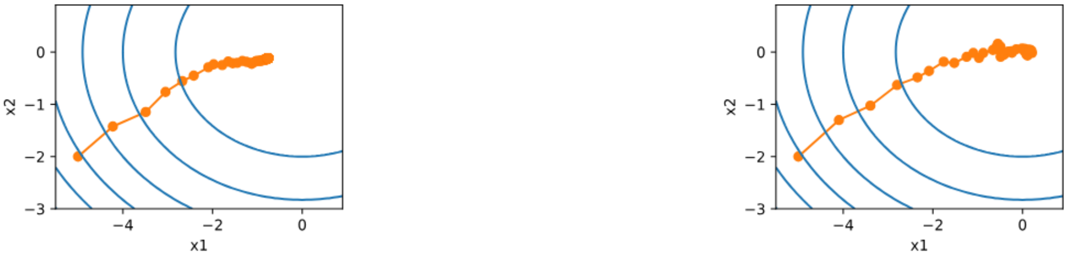
2 Dynamic Learning Rate

Even after 50 steps the quality is still not so good. Even worse, it will not improve after additional steps. This leaves us with the only alternative: change the learning rate η. However, if we pick this too small, we will not make any meaningful progress initially. On the other hand, if we pick it too large, we will not get a good solution, as seen above. The only way to resolve these conflicting goals is to reduce the learning rate dynamically as optimization progresses.

How rapidly η should decay?

$$\begin{aligned} &\eta(t) = \eta_i \text{ if } t_i \leq t \leq t_{i+1} && \text{piecewise constant} \\ &\eta(t) = \eta_0 \cdot e^{-\lambda t} && \text{exponential decay} \\ &\eta(t) = \eta_0 \cdot (\beta t + 1)^{-\alpha} && \text{polynomial decay} \end{aligned}$$

In the first piecewise constant scenario we decrease the learning rate, e.g., whenever progress in optimization stalls. This is a common strategy for training deep networks. Alternatively we could decrease it much more aggressively by an exponential decay. Unfortunately this often leads to premature stopping before the algorithm has converged. A popular choice is polynomial decay with $\alpha = 0.5$. In the case of convex optimization there are a number of proofs that show that this rate is well behaved. The behavior of the learning rate decay approahes can be seen in figure 2



(a) exponential decay

(b) polynomial decay

Figure 2: exponential decay vs polynomial decay

3 Convergence Analysis for Convex Objectives

Can you perform convergence analysis on stochastic gradient descent for convex objective function? Suppose that the objective function $f(\boldsymbol{\xi}, \mathbf{x})$ is convex in $\mathbf{x}$ for all $\boldsymbol{\xi}$. More concretely, we consider the stochastic gradient decent update:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta_t \partial_{\mathbf{x}} f\left(\boldsymbol{\xi}_t, \mathbf{x}\right)$$

where $f\left(\boldsymbol{\xi}_t, \mathbf{x}\right)$ is the objective function with respect to the training example $\boldsymbol{\xi}_t$ drawn from some distribution at step t and $\mathbf{x}$ is the model parameter. Denote by

$$R(\mathbf{x})=E_{\xi}[f(\boldsymbol{\xi}, \mathbf{x})]$$

the expected risk and by R^* its minimum with regard to $\mathbf{x}$. Last let $\mathbf{x}^*$ be the minimizer (we assume that it exists within the the domain where $\mathbf{x}$ is defined). In this case we can track the distance between the current parameter $\mathbf{x}_t$ at time t and the risk minimizer $\mathbf{x}^*$ and see whether it improves over time:

$$\begin{aligned}
& \left\|\mathbf{x}_{t+1}-\mathbf{x}^*\right\|^2 \\
& =\left\|\mathbf{x}_t-\eta_t \partial_{\mathbf{x}} f\left(\boldsymbol{\xi}_t, \mathbf{x}\right)-\mathbf{x}^*\right\|^2 \\
& =\left\|\mathbf{x}_t-\mathbf{x}^*\right\|^2+\eta_t^2\left\|\partial_{\mathbf{x}} f\left(\boldsymbol{\xi}_t, \mathbf{x}\right)\right\|^2-2 \eta_t\left\langle\mathbf{x}_t-\mathbf{x}^*, \partial_{\mathbf{x}} f\left(\boldsymbol{\xi}_t, \mathbf{x}\right)\right\rangle
\end{aligned} \tag{1}$$

We assume, control the size of the gradient and ensure the update step does not get too large, that the l_2 norm of stochastic gradient $\partial_{\mathbf{x}} f\left(\boldsymbol{\xi}_t, \mathbf{x}\right)$ is bounded by some constant L, hence we have that

$$\eta_t^2\left\|\partial_{\mathbf{x}} f\left(\boldsymbol{\xi}_t, \mathbf{x}\right)\right\|^2 \leq \eta_t^2 L^2$$

We are mostly interested in how the distance between $\mathbf{x}_t$ and $\mathbf{x}^*$ changes in expectation. In fact, for any specific sequence of steps the distance might well increase, depending on whichever $\boldsymbol{\xi}_t$ we encounter. Hence we need to bound the dot product. Since for any convex function f it holds that $f(\mathbf{y}) \geq f(\mathbf{x})+\left\langle f^{\prime}(\mathbf{x}), \mathbf{y}-\mathbf{x}\right\rangle$ for all $\mathbf{x}$ and $\mathbf{y}$, by convexity we have

$$f\left(\boldsymbol{\xi}_t, \mathbf{x}^*\right) \geq f\left(\boldsymbol{\xi}_t, \mathbf{x}_t\right)+\left\langle\mathbf{x}^*-\mathbf{x}_t, \partial_{\mathbf{x}} f\left(\boldsymbol{\xi}_t, \mathbf{x}_t\right)\right\rangle$$

Clarification $f(\mathbf{y}) \geq f(\mathbf{x})+\left\langle f^{\prime}(\mathbf{x}), \mathbf{y}-\mathbf{x}\right\rangle$, the function $f(\mathbf{y})$ is at least as large as the value of the function at $\mathbf{x}$, plus the linear approximation (tangent plane) around $\mathbf{x}$. This inequality holds for all $\mathbf{x}$ and $\mathbf{y}$.

Now we have to rearrange the equation to satisfy the objective, $\left\langle\mathbf{x}^*-\mathbf{x}_t, \partial_{\mathbf{x}} f\left(\xi_t, \mathbf{x}_t\right)\right\rangle \leq f\left(\xi_t, \mathbf{x}^*\right)-f\left(\xi_t, \mathbf{x}_t\right)$. Substitute into the third term of e.q.1:

$$-2 \eta_t\left\langle\mathbf{x}_t-\mathbf{x}^*, \partial_{\mathbf{x}} f\left(\xi_t, \mathbf{x}_t\right)\right\rangle \leq-2 \eta_t\left(f\left(\xi_t, \mathbf{x}_t\right)-f\left(\xi_t, \mathbf{x}^*\right)\right)$$

we have:

$$\left\|\mathbf{x}_{t+1}-\mathbf{x}^*\right\|^2 \leq\left\|\mathbf{x}_t-\mathbf{x}^*\right\|^2-2 \eta_t\left(f\left(\xi_t, \mathbf{x}_t\right)-f\left(\xi_t, \mathbf{x}^*\right)\right)+\eta_t^2 L^2$$

rearrange,

$$\left\|\mathbf{x}_t-\mathbf{x}^*\right\|^2-\left\|\mathbf{x}_{t+1}-\mathbf{x}^*\right\|^2 \geq 2 \eta_t\left(f\left(\xi_t, \mathbf{x}_t\right)-f\left(\xi_t, \mathbf{x}^*\right)\right)-\eta_t^2 L^2 \tag{2}$$

This means that we make progress as long as the difference between current loss and the optimal loss outweighs $\eta_t L^2 / 2$. Since this difference is bound to converge to zero it follows that the learning rate η_t also needs to vanish.

Now we want to take the expectation over e.q.2

$$E\left[\left\|\mathbf{x}_t-\mathbf{x}^*\right\|^2\right]-E\left[\left\|\mathbf{x}_{t+1}-\mathbf{x}^*\right\|^2\right] \geq 2 \eta_t\left[E\left[R\left(\mathbf{x}_t\right)\right]-R^*\right]-\eta_t^2 L^2$$

doing so, we eliminate the randomness due to the stochastic sampling of the gradient. We now sum the inequality for all t from 1 to T. The left-hand side becomes a telescoping sum:

$$\sum_{t=1}^T\left(\mathbb{E}\left[\left\|\mathbf{x}_t-\mathbf{x}^*\right\|^2\right]-\mathbb{E}\left[\left\|\mathbf{x}_{t+1}-\mathbf{x}^*\right\|^2\right]\right)$$

Notice that in this sum, most of the terms cancel out. Here's how it works:

1. When you expand the sum, you get: $\left(\mathbb{E}\left[\left\|\mathbf{x}_1-\mathbf{x}^*\right\|^2\right]-\mathbb{E}\left[\left\|\mathbf{x}_2-\mathbf{x}^*\right\|^2\right]\right)+\left(\mathbb{E}\left[\left\|\mathbf{x}_2-\mathbf{x}^*\right\|^2\right]-\mathbb{E}\left[\left\|\mathbf{x}_3-\mathbf{x}^*\right\|^2\right]\right)+\cdots+\left(\mathbb{E}\left[\left\|\mathbf{x}_T-\mathbf{x}^*\right\|^2\right]-\mathbb{E}\left[\left\|\mathbf{x}_{T+1}-\mathbf{x}^*\right\|^2\right]\right)$
2. Most of the terms cancels out, after all the cancellation, only two terms remain: $\mathbb{E}\left[\left\|\mathbf{x}_1-\mathbf{x}^*\right\|^2\right]-\mathbb{E}\left[\left\|\mathbf{x}_{T+1}-\mathbf{x}^*\right\|^2\right]$

Now, since $\mathbf{x}_{T+1}$ is close to the optimal solution as T increases, we can ignore this term (or assume it converges to a small value). Thus, we have:

$$\left\|\mathbf{x}_1-\mathbf{x}^*\right\|^2 \geq 2\left(\sum_{t=1}^T \eta_t\right)\left[E\left[R\left(\mathbf{x}_t\right)\right]-R^*\right]-L^2 \sum_{t=1}^T \eta_t^2 \tag{3}$$

Note that we exploited that $\mathbf{x}_1$ is given and thus the expectation can be dropped. Last define (*averaging the iteration tends to stabilize the solution and improve convergence behavior*)

$$\overline{\mathbf{x}} \stackrel{\text{def}}{=} \frac{\sum_{t=1}^T \eta_t \mathbf{x}_t}{\sum_{t=1}^T \eta_t}$$

Since, now consider the risk over the iterations,

$$E\left(\frac{\sum_{t=1}^T \eta_t R\left(\mathbf{x}_t\right)}{\sum_{t=1}^T \eta_t}\right)=\frac{\sum_{t=1}^T \eta_t E\left[R\left(\mathbf{x}_t\right)\right]}{\sum_{t=1}^T \eta_t}=E\left[R\left(\mathbf{x}_t\right)\right]$$

From Jensen's inequality, $E\left[R\left(\mathbf{x}_t\right)\right] \geq E[R(\overline{\mathbf{x}})]$, meaning the expected of each risk added and averaged is greater than the risk of average value, thus

$$R(\overline{\mathbf{x}}) \leq \frac{\sum_{t=1}^T \eta_t R\left(\mathbf{x}_t\right)}{\sum_{t=1}^T \eta_t}$$

apply expectation on both side

$$\mathbb{E}[R(\overline{\mathbf{x}})] \leq \mathbb{E}\left[\frac{\sum_{t=1}^T \eta_t R\left(\mathbf{x}_t\right)}{\sum_{t=1}^T \eta_t}\right]$$

then,

$$\mathbb{E}[R(\overline{\mathbf{x}})] \leq \frac{\sum_{t=1}^T \eta_t \mathbb{E}\left[R\left(\mathbf{x}_t\right)\right]}{\sum_{t=1}^T \eta_t}$$

Finally,

$$\sum_{t=1}^T \eta_t E\left[R\left(\mathbf{x}_t\right)\right] \geq \sum_{t=1}^T \eta_t E[R(\overline{\mathbf{x}})]$$

plugging it into e.q.3, we have

$$\left\|\mathbf{x}_1-\mathbf{x}^*\right\|^2 \geq 2 \sum_{t=1}^T \eta_t\left(\mathbb{E}[R(\overline{\mathbf{x}})]-R^*\right)-L^2 \sum_{t=1}^T \eta_t^2$$

we want to isolate $\mathbb{E}[R(\overline{\mathbf{x}})]-R^*$,

$$2 \sum_{t=1}^T \eta_t\left(\mathbb{E}[R(\overline{\mathbf{x}})]-R^*\right) \leq\left\|\mathbf{x}_1-\mathbf{x}^*\right\|^2+L^2 \sum_{t=1}^T \eta_t^2$$

Divide both sides by $2 \sum_{t=1}^T \eta_t$ to isolate $\mathbb{E}[R(\overline{\mathbf{x}})]-R^*$:

$$\mathbb{E}[R(\overline{\mathbf{x}})]-R^* \leq \frac{\left\|\mathbf{x}_1-\mathbf{x}^*\right\|^2+L^2 \sum_{t=1}^T \eta_t^2}{2 \sum_{t=1}^T \eta_t}$$

In the next step, we define $r^2=\left\|\mathbf{x}_1-\mathbf{x}^*\right\|^2$, which is a constant representing the initial distance between the first iterate $\mathbf{x}_1$ and the optimal solution $\mathbf{x}^*$. So the bound becomes:

$$\mathbb{E}[R(\overline{\mathbf{x}})]-R^* \leq \frac{r^2+L^2 \sum_{t=1}^T \eta_t^2}{2 \sum_{t=1}^T \eta_t}$$

- r^2 is the initial squared distance between $\mathbf{x}_1$ and the optimal solution $\mathbf{x}^*$.
- $L^2 \sum_{t=1}^T \eta_t^2$ represents the effect of the stochastic gradient noise over time. The bound depends on how the learning rates η_t scale.

- The denominator $2\sum_{t=1}^{T}\eta_t$ reflects how much total progress the algorithm makes as a function of the learning rates.

We want to 1). maximize $\sum_{t=1}^{T}\eta_t$: This represents the cumulative progress made by the algorithm. Larger values of η_t mean faster progress, and 2). minimize $\sum_{t=1}^{T}\eta_t^2$: This term grows if the learning rate is too large, leading to instability in the updates (overshooting the optimal solution).

The learning rate $\eta = \frac{r}{L\sqrt{T}}$ is intuitive because it starts large (when we are far from the solution) and shrinks as we get closer (as T grows). This mirrors how we would like an optimization process to behave: aggressive updates early on, and fine-tuning near the end. Now we have

$$\mathbb{E}[R(\overline{\mathbf{x}})] - R^* \leq \frac{r^2 + L^2 \cdot \frac{r^2}{L^2}}{2 \cdot \frac{r\sqrt{T}}{L}} = \frac{2r^2}{2 \cdot \frac{r\sqrt{T}}{L}} = \frac{rL}{\sqrt{T}}$$

That is, we converge with rate $\mathcal{O}(1/\sqrt{T})$ to the optimal solution.