EECS 182 Deep Neural Networks
Spring 2023 Anant Sahai

Discussion 3

1. Why learning rates cannot be too big

To understand the role of the learning rate, it is useful to understand it in the context of the simplest possible problem first.

Suppose that we want to solve the scalar equation

$$\sigma w = y \tag{1}$$

where we know that $\sigma > 0$. We proceed with an initial condition $w_0 = 0$ by using gradient descent to minimize the squared loss

$$L(w) = (y - \sigma w)^2 \tag{2}$$

which has a derivative with respect to the parameter w of $-2\sigma(y - \sigma w)$.

Gradient descent with a learning rate of η follows the recurrence-relation or discrete-time state evolution of:

$$\begin{aligned} w_{t+1} &= w_t + 2\eta\sigma(y - \sigma w_t) \\ &= (1 - 2\eta\sigma^2)w_t + 2\eta\sigma y. \end{aligned} \tag{3}$$

(a) **For what values of learning rate $\eta > 0$ is the recurrence (3) stable?**

(HINT: Remember the role of the unit circle in determining the stability or instability of such recurrences. If you keep taking higher and higher positive integer powers of a number, what does that number have to be like for this to converge?)

Solution: We can rewrite the update rule as

$$\begin{aligned} w_{t+1} - \frac{y}{\sigma} &= (1 - 2\eta\sigma^2)(w_t - \frac{y}{\sigma}) \\ w_{t+1} &= \frac{y}{\sigma} + (1 - 2\eta\sigma^2)^{t+1}(w_0 - \frac{y}{\sigma}) \end{aligned} \tag{4}$$

To make the recurrence (3) stable with $\eta > 0$, we need $|1 - 2\eta\sigma^2| < 1$. This gives $\eta < \frac{1}{\sigma^2}$.

(b) The previous part gives you an upper bound for the learning rate η that depends on σ beyond which we cannot safely go. **If η is below that upper bound, how fast does w_t converge to its final solution $w^* = \frac{y}{\sigma}$? i.e. If we wanted to get within a factor $(1 - \epsilon)$ of w^*, how many iterations t would we need?**

Solution:

$$|w_T - w^*| < \epsilon|w^*| \tag{5}$$

Use the derived update rule in (4). We have

$$\begin{aligned}
|w_T - \frac{y}{\sigma}| &< \epsilon|\frac{y}{\sigma}| \\
|(1-2\eta\sigma^2)^T| &< \epsilon \\
T &> \frac{log(\epsilon)}{log(|1-2\eta\sigma^2|)},
\end{aligned} \tag{6}$$

(c) Suppose that we now have a vector problem where we have two parameters $w[1], w[2]$. One with a large σ_ℓ and the other with a tiny σ_s. i.e. $\sigma_\ell \gg \sigma_s$ and we have the vector equation we want to solve:

$$\begin{bmatrix} \sigma_\ell & 0 \\ 0 & \sigma_s \end{bmatrix} \begin{bmatrix} w[1] \\ w[2] \end{bmatrix} = \begin{bmatrix} y[1] \\ y[2] \end{bmatrix}. \tag{7}$$

We use gradient descent with a single learning rate η to solve this problem starting from an initial condition of $\mathbf{w} = \mathbf{0}$.

For what learning rates $\eta > 0$ will we converge? Which of the two σ_i is limiting our learning rate?

Solution: Similarly, we can rewrite the loss function and update rule w.r.t the vector form.

$$\begin{aligned}
L(\mathbf{w}) &= ||\mathbf{y} - \Sigma\mathbf{w}||^2, \Sigma = \begin{bmatrix} \sigma_\ell & 0 \\ 0 & \sigma_s \end{bmatrix} \\
\nabla_{\mathbf{w}} L(\mathbf{w}) &= 2(\Sigma^2\mathbf{w} - \Sigma\mathbf{y}) \\
\mathbf{w}_{t+1} &= (I - 2\eta\Sigma^2)\mathbf{w}_t + 2\eta\Sigma\mathbf{y}
\end{aligned} \tag{8}$$

To ensure the convergence, we need

$$\begin{cases} |1-2\eta\sigma_l^2| < 1 \\ |1-2\eta\sigma_s^2| < 1 \end{cases} \tag{9}$$

$$\eta < \min(\frac{1}{\sigma_l^2}, \frac{1}{\sigma_s^2}) = \frac{1}{\sigma_l^2} \tag{10}$$

(d) **For the previous problem, depending on $\eta, \sigma_\ell, \sigma_s$, which of the two dimensions is converging faster and which is converging slower?**

Solution: We can rewrite the update rule w.r.t each dimension, this gives

$$\begin{aligned}
w[1]_t &= \frac{y[1]}{\sigma_l} + (1-2\eta\sigma_l^2)^t(-\frac{y[1]}{\sigma_l}) \\
w[2]_t &= \frac{y[2]}{\sigma_s} + (1-2\eta\sigma_s^2)^t(-\frac{y[2]}{\sigma_s})
\end{aligned}$$

This faster convergence dimension is $\min(|1-2\eta\sigma_l^2|, |1-2\eta\sigma_s^2|)$

(e) The speed of convergence overall will be dominated by the slower of the two. **For what value of η will we get the fastest overall convergence to the solution?**

Solution: $\eta = \frac{1}{\sigma_l^2+\sigma_s^2}$

2. ReLU with different Optimizers

Work through the notebook to explore how a simple network with ReLU non-linearities adapts to model a function using different optimizers. training the networks takes 5-10 minutes depending on whether you run locally or the server, so you should start the training process (run through the train all layers cell) then return to the theory part of the discussion while training occurs.

Solution: Answers to notebook questions.

- How does the hidden layer width and different optimizers impact the learned function and test error? **Ans:** The larger hidden layer width is, the lower test error we get. Different optimizers result in different performance. Both Adam and SGD with momentum are consistently better than SGD. Generally Adam converges faster than both versions of SGD. In terms of variance (error bar), SGD has the largest variance among different random seeds, and Adam is consistently the most stable one.
- What happens to the elbow locations using different optimizers during training? **Ans:** Converging approximately to the non-linear kinks (for detailed reason, check out discussion 1 notebook).
- Are the circle dots on the graphs above mean or median? Why not plot the other one? **Ans:** Median. It does not make sense to plot the mean since we are plotting at log scale, and averaging points may yield values that are not well represented on the plot.
- How are error bars computed? Are the upper and lower marks maximum/minimum, standard deviation or something else? Does it make sense to plot standard deviation here? **Ans:** The Upper and lower bound are 25% and 75% percentile respectively. Plotting S.D. does not make sense here again due to the log scale plotting.

Contributors:

- Anant Sahai.
- Sheng Shen.
- Kevin Li.
- Kumar Krishna Agrawal.