Gradient Descent Convergence

Due: Sun Apr 24, 2022 11:00pm

Attempt 1 | IN PROGRESS | Next Up: Submit Assignment | Add Comment

Theory

The questions below ask for justification of steps taken in the proofs of Gradient Descent Provably Optimizes Over-parameterized Neural Networks

I Lemma 3.2

The first step in the proof of Lemma 3.2 defines an event A_{ir} over the random initialization of $\mathbf{w}_r(0)$, i.e., the weights of the rth hidden neuron.

1. Assume that event A_{ir} occurs for some realization $\mathbf{w}_r(0)$. Then show that the realization $\mathbf{w}_r(0)$ is such that $|\mathbf{w}_r(0)^T\mathbf{x}_i| < R$.
2. Assume that a realization $\mathbf{w}_r(0)$ satisfies $|\mathbf{w}_r(0)^T\mathbf{x}_i| < R$. Then show that the event A_{ir} is true for that realization $\mathbf{w}_r(0)$.

The above two combined prove that event A_{ir} occurs *"if and only if"* $|\mathbf{w}_r(0)^T\mathbf{x}_i| < R$.

II Lemma 3.3

The following screenshot is from page 7 of the paper, and is a step from the proof of Lemma 3.3 that obtains a bound on the length of the gradient at each time s.

$$\left\|\frac{d}{ds}\mathbf{w}_r(s)\right\|_2 = \left\|\sum_{i=1}^{n}(y_i-u_i)\frac{1}{\sqrt{m}}a_r\mathbf{x}_i\mathbb{I}\left\{\mathbf{w}_r(s)^\top\mathbf{x}_i\geq 0\right\}\right\|_2$$
$$\overset{\text{(i)}}{\leq}\frac{1}{\sqrt{m}}\sum_{i=1}^{n}|y_i-u_i(s)|\overset{\text{(ii)}}{\leq}\frac{\sqrt{n}}{\sqrt{m}}\left\|\mathbf{y}-\mathbf{u}(s)\right\|_2\leq\frac{\sqrt{n}}{\sqrt{m}}\exp(-\lambda_0 s)\left\|\mathbf{y}-\mathbf{u}(0)\right\|_2.$$

Justify why the inequalities marked as (i) and (ii) are correct.

Hint: Equation (4) in https://en.wikipedia.org/wiki/Jensen%27s_inequality . Also note that for a scalar c and vector **v**, *it is true that* $\|c\mathbf{v}\|_2 = |c|\|\mathbf{v}\|_2$.

In the lectures we have only looked at gradient flow on the weights $\mathbf{w}_r(t)$ coming in to the hidden layer and kept the "outgoing" weights $\mathbf{a}$ fixed. But GD is applied on all the weights in a neural network. Theorem 3.3 proves the same convergence guarantees when both $\mathbf{w}_r(t)$ and $\mathbf{a}(t)$ are updated according to the gradient flow equations:

$\frac{d\mathbf{w}_r(t)}{dt} = -\frac{\partial L(\mathbf{W}(t),\mathbf{a}(t))}{\partial \mathbf{w}_r(t)}$

$\frac{d\mathbf{a}(t)}{dt} = -\frac{\partial L(\mathbf{W}(t),\mathbf{a}(t))}{\partial \mathbf{a}(t)}$

1. The following screenshot from page 16 shows a step in the proof of Theorem 3.3. Show all the calculations needed to derive equation (i).

$$\sum_{r=1}^{m}\frac{du_i(t)}{d\mathbf{a}_r}\cdot\frac{d\mathbf{a}_r(t)}{dt}\overset{\text{(i)}}{=}\sum_{r=1}^{m}\left(y_j-u_j(t)\right)\mathbf{G}_{ij}(t)$$

where

$$\mathbf{G}_{ij}(t)=\frac{1}{m}\sigma\left(\mathbf{w}_r^\top\mathbf{x}_i\right)\sigma\left(\mathbf{w}_r^\top\mathbf{x}_j\right).$$

2. The following screenshot from page 16 shows a step in the proof of Lemma A.2. Justify why the inequality labelled as (ii) is true.

$$\frac{d}{dt}\left\|\mathbf{y}-\mathbf{u}(t)\right\|_2^2 = -2\left(\mathbf{y}-\mathbf{u}(t)\right)^\top\left(\mathbf{H}(t)+\mathbf{G}(t)\right)\left(\mathbf{y}-\mathbf{u}(t)\right)$$
$$\overset{\text{(ii)}}{\leq}-2\left(\mathbf{y}-\mathbf{u}(t)\right)^\top\left(\mathbf{H}(t)\right)\left(\mathbf{y}-\mathbf{u}(t)\right)$$

Practical

The MNIST-1D dataset is obtained using the code in https://colab.research.google.com/github/greydanus/mnist1d/blob/master/building_mnist1d.ipynb , it is assigned to a variable named data. In this assignment, you will verify some implications of the theory discussed in class. Recall the matrix entry $\mathbf{H}^{\infty}_{i,j}$ from Assumption 3.1 in Gradient Descent Provably Optimizes Over-parameterized Neural Networks. We can simplify the expression as follows.

$$
\begin{aligned}
&\mathbf{H}^{\infty}_{i,j} \\
&= \mathbb{E}_{\mathbf{w}\sim N(\mathbf{0},I)}[\mathbf{x}_i^T\mathbf{x}_j \mathbb{I}\{\mathbf{w}^T\mathbf{x}_i \geq 0, \mathbf{w}^T\mathbf{x}_j \geq 0\}] \\
&= \mathbf{x}_i^T\mathbf{x}_j \mathbb{E}_{\mathbf{w}\sim N(\mathbf{0},I)}[\mathbb{I}\{\mathbf{w}^T\mathbf{x}_i \geq 0, \mathbf{w}^T\mathbf{x}_j \geq 0\}] \\
&= \mathbf{x}_i^T\mathbf{x}_j \frac{\pi - \text{angle}(\mathbf{x}_i,\mathbf{x}_j)}{2\pi}.
\end{aligned}
$$

In the last equality (which is true by a geometric argument) the term $\text{angle}(x_i, x_j)$ = `np.arccos(np.dot(x_i, x_j))` is the angle between x_i and x_j (x_i and x_j need to have unit length).

1. Call `make_dataset` to obtain a dataset containing 500 samples and normalize each sample to have unit length. Modify the `MLPBase` class and write a new class for a fully connected neural network with a single hidden layer with relu activation and standard normal initialization. Initialize an instance of the fully connected class with 1000 hidden units.
2. Calculate the minimum eigenvalue of $\mathbf{H}^{\infty}$ and $\mathbf{H}(0)$ matrices and confirm if they are positive.
3. Modify the `train_model` function to use SGD optimizer and mean squared error loss function. Apply the modified training function on the instance in part 1 for 10000 steps. Plot the training loss values and compare it to the theoretical upper bound in Theorem 3.2 of Gradient Descent Provably Optimizes Over-parameterized Neural Networks
4. Initialize a new fully connected neural network with 1000 hidden units. Train it only on images of digit 0 for the first 1000 steps, followed by training on images of 0 and 1 for the next 1000 steps, followed by training on images of 0, 1, 2 for the next 1000 steps, and so on such that the last 1000 steps are trained on images from all 10 digits. Compare the training loss values for this new "stagewise" training procedure to the previous plot in part 3.