CIS 520, Machine Learning, Fall 2021
Homework 4
Due: Sunday, October 10th, 11:59pm
Submit to Gradescope

Your name here

1 Neural Networks: Backpropagation

Your task is to now compute the derivatives of the loss function given in 1 with respect to W_1, W_2, b_1 and b_2 by hand, i.e., $\frac{\partial Loss}{\partial W_1}$, $\frac{\partial Loss}{\partial b_1}$, and $\frac{\partial Loss}{\partial b_2}$.

$$Loss = y * ln(\sigma(z_2)) + (1 - y) * ln(1 - \sigma(z_2)) \tag{1}$$

Show all the intermediate derivative computation steps. You might benefit from making a rough schematic of the backpropagation process. Also recall the derivatives of the softmax function and the tanh function:

$$\frac{d\sigma(z_2)}{dz} = \sigma(z_2) * (1 - \sigma(z_2)) \tag{2}$$

$$\frac{\partial tanh(z_1)}{\partial z_1} = 1 - tanh^2(z_1) \tag{3}$$

$$Loss = y * \ln(\sigma(z_2)) + (1-y) * \ln(1-\sigma(z_2))$$

where $z_2 = \tanh(xW_1 + b_1)W_2 + b_2$

$$\frac{\partial Loss}{\partial z_2} = y \frac{\partial \ln(\sigma(z_2))}{\partial z_2} + (1-y) \frac{\partial \ln(1-\sigma(z_2))}{\partial z_2}$$

$$= y \cdot \frac{1}{\sigma(z_2)} \cdot \sigma(z_2)(1-\sigma(z_2)) + (1-y) \cdot \frac{1}{1-\sigma(z_2)} (-\sigma(z_2))(1-\sigma(z_2))$$

$$= y(1-\sigma(z_2)) - (1-y)\sigma(z_2) = y - y\sigma(z_2) - \sigma(z_2) + y\sigma(z_2)$$

$$= y - \sigma(z_2)$$

$$\frac{\partial Loss}{\partial W_1} = \frac{\partial Loss}{\partial z_2} \frac{\partial z_2}{\partial W_1} = \frac{\partial}{\partial W_1}(-\sigma(z_2)) = \frac{\partial}{\partial W_1}(\tanh(xW_1+b_1)W_2)\ (y-\sigma(z_2))$$

$$= x^T(1-\tanh^2(xW_1+b_1))(y-\sigma(z_2))\ W_2^T$$

$$\frac{\partial Loss}{\partial b_1} = \frac{\partial Loss}{\partial z_2} \frac{\partial z_2}{\partial b_1} = \frac{\partial}{\partial b_1}(-\sigma(z_2))(y-\sigma(z_2)) = (1-\tanh^2(xW_1+b_1))(y-\sigma(z_2))W_2^T$$

$$\frac{\partial Loss}{\partial W_2} = \frac{\partial Loss}{\partial z_2} \frac{\partial z_2}{\partial W_2} = \frac{\partial}{\partial W_2}(-\sigma(z_2))(y-\sigma(z_2)) = \tanh(xW_1+b_1)\ (y-\sigma(z_2)) = a_1^T(y-\sigma(z_2))$$

$$\frac{\partial Loss}{\partial b_2} = \frac{\partial Loss}{\partial z_2} \frac{\partial z_2}{\partial b_2} = \frac{\partial}{\partial b_2}(-\sigma(z_2))(y-\sigma(z_2)) = y-\sigma(z_2)$$

Figure 1: Question 1

2 Convolutional Neural Networks

2.1 Theory

1. The output volume is:

$$((98 + 2 * 2) - 6) + 1 = 25$$
$$\implies 25 * 25 * 64$$

2. Including bias parameters, this hidden layer has the following number of parameters:

$$(99 * 99 * 3 + 1) * 50 * 50 * 6$$
$$= 441060000$$

3. Including bias parameters, this hidden layer has the following number of parameters:

$$(3 * 3 * 3 + 1) * 6 = 168$$

4. The output from the simple convolutional layer for the given filter and input is:

Row	**Column**	**Filter**	**Value**	**Value**(without flipmirror)
1	1	1	-13	9
1	1	2	44	21
1	2	1	18	-10
2	1	1	0	-10

Table 1: Output from convolution

5. State two advantages of convolutional neural networks over fully connected networks.

   1. Much fewer parameters needed for the network. 2. Convolution involves more relationship between pixels, thus making some tasks like edge detection easier.

2.2 Programming

For this question, refer to the Jupyter Notebook. You will be using PyTorch to implement a convolutional neural network – the notebook will have detailed instructions. We will be using the fashion MNIST dataset for a classification task.

2.2.1 Convolutional Neural Network

Add the accuracy and the loss curve from tensorboard in this report:

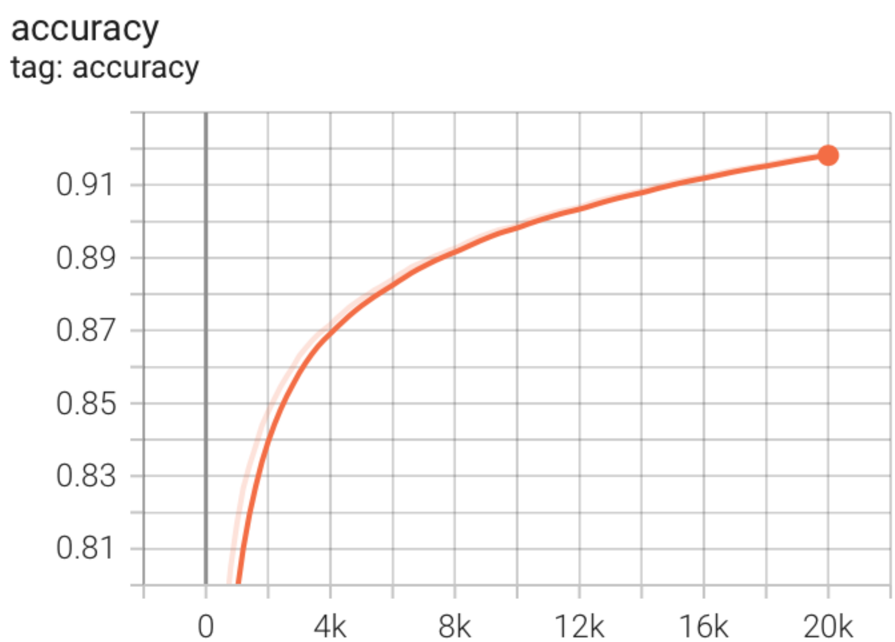

Figure 2: Accuracy curve

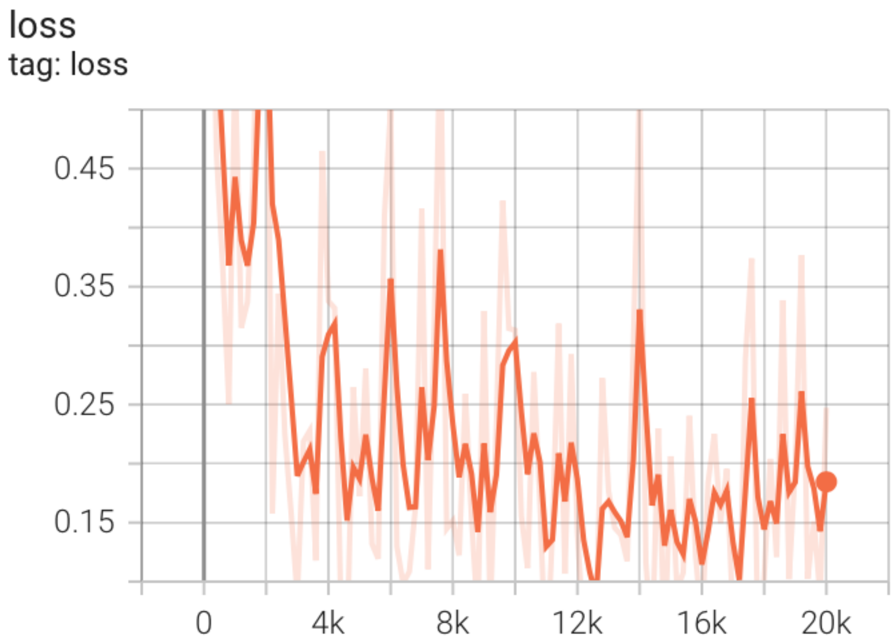

Figure 3: Loss curve

2.2.2 Network Architecture and Implementation

Model Architecture: Used all the layers mentioned in the table with the following hyperparameters

1. Batch Normalization: False
2. Dropout: 0.1
3. (Kernel, Stride, Padding) = (2, 2, 0)

2.2.3 Accuracy

Report the overall accuracy and the per-class accuracy:

Overall Accuracy	90%

Table 2: Overall Accuracy for Convolutional Neural Network

Class	Accuracy
T-shirt/top	82%
Trouser	98%
Pullover	90%
Dress	93%
Coat	78%
Sandal	96%
Shirt	73%
Sneaker	99%
Bag	100%
Ankle boot	93%

Table 3: Per Class Accuracy for Convolutional Neural Network

Identify the problematic classes and list the possible reasons as to why these classes may have significantly lower accuracy compared to other classes.

The most problematic classes are T-shirt/top, coats, and shirts. Because they are just so similar and very hard to distinguish even for humans. Another reason might be the edges are not so obvious compared with other classes, making CNN edge detection hard to work.