**Introduction to Machine Learning (Spring 2020)**

**Homework #3 (50 Pts, Due Date: May 31st)**

**Student ID**

**Name**

**Instruction:** We provide all codes and datasets in Python. Please write your code to complete activation layers (Sigmoid, ReLU, tanh), Fully Connected Layer (FCLayer), Softmax Layer, and L2 regularization. Submit two files as follows:

* ‘HW3\_STUDENT\_ID\_YourName.zip’: All codes in the directory except ‘data’ directory and your document
* ‘HW3\_STUDENT\_ID\_YourName.pdf’: Your document converted into pdf.

**NOTE 1**: In the next homework, ‘Homework #4’, you will be reusing your code from ‘Homework #3’

1. **[30 pts]** Implement functions in ReLU, Sigmoid, Tanh, FCLayer, SoftmaxLayer, Norm in ‘Answer.py’.
2. **[Activation Layer]** Implement Sigmoid, ReLU, Tanh activation in ‘Answer.py’ (‘Sigmoid’, ‘ReLU’, ‘Tanh’).
3. **[Fully Connected Layer]** Implement Fully Connected layer in ‘Answer.py’ (‘FCLayer’).
4. **[Softmax Layer]** Implement Softmax layer in ‘Answer.py’ (‘SoftmaxLayer’).

Given a mini-batch data the error function for a mini-batch is defined as follows:

**Answer: Fill your code here. You also have to submit your code to i-campus.**

**NOTE 2**: **You should write your codes in ‘EDIT HERE’ signs.** It is not recommended to edit other parts. Once you complete your implementation, run the check code (‘test\_answer.py’) to check if it is done correctly.

**NOTE 3**: **Read the instructions in template codes VERY CAREFULLY.** Functionality and input, output shape of every function must be the same as written.

1. **[20 Pts]** Experiment results
2. **[DNN with different activation layer]** Report test accuracy on MNIST using three different activation function(Sigmoid, ReLU, Tanh) with given DNN architecture and parameters. Explain the differences among three activation functions (Use only one activation function in one experiment among Sigmoid, ReLU, Tanh)

**[DNN Architecture]**

|  |  |
| --- | --- |
| **Layer name** | **Configuration** |
| **FC – 1** | Input dim = 784, Output dim = 500 |
| **[Sigmoid, ReLU, Tanh]** | - |
| **FC – 2** | Input dim = 500, Output dim = 500 |
| **[Sigmoid, ReLU, Tanh]** | - |
| **FC – 3** | Input dim = 500, Output dim = 10 |
| **Softmax Layer** | - |

(num\_epochs = 100, learning\_rate = 0.001, batch\_size=128, reg\_lambda = 1e-8)

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Sigmoid** | **ReLU** | **Tanh** |
| **Test accuracy** |  |  |  |

1. **[Deep Neural Networks]** Adjust the model settings (# of hidden layers, # of hidden nodes, # of epochs, learning rate etc.) to get the best results on FashionMNIST using ‘main.py’. Report your best test accuracy with your fine-tuned hyperparameters. Show the plot of training and validation accuracy every epochs on each case and explain how you determined the model structure or parameters in 4~5 lines.

(batch size = 128)

**Answer: Fill the blank in the table. Show the plot of training & validation accuracy with a brief explanation.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Model structure** | **# of epochs** | **Learning rate** | **L2 lambda** | **Best Validation Acc.** | **Final Test Acc.** |
| **1st Best** | **Ex.) FC-1(784, 10)  Sigmoid  Softmax** |  |  |  |  |  |
| **2nd Best** |  |  |  |  |  |  |
| **3rd Best** |  |  |  |  |  |  |