CSCE 636: Neural Networks (Fall 2018)
Assignment #2
Due 10/5/2018

1. Please read and follow submission instructions. No exception will be made to accommodate incorrectly submitted files/reports.

2. You need to submit a report in hard-copy before lecture. The hard-copy report should contain everything except for the code files. For example, all answers to problems and summary of experimental results must be included in the hard-copy report.

3. The code files need to be emailed to `csce636hw@gmail.com`. Before submission, compress all your code files into a zip file named 'firstname_lastname_HW#.zip'. Make sure the email title is 'firstname_lastname_HW#'. Note that your email submission should contain your code files ONLY.

4. Hard-copy is due in class before lecture and code part is due 11:30AM on the due date.

5. Only one submission is allowed.

6. LFD refers to the textbook "Learning from Data".

7. Write your code between the following lines. Do not modify other parts. Only the code between the following lines will be graded. Don NOT include experimental results in your code submission. Do NOT change file names.

   ### YOUR CODE HERE

   ### END YOUR CODE

---

1. (50 points)(Coding Task) **Neural Networks for MNIST Handwritten Digits Classification**: In this assignment, you will implement a feed-forward neural network on MNIST using Tensorflow. In this classification task, the model will take a 16 × 16 image of a handwritten digit as inputs and classfy the image into a digit in [0,9] . Note that unlike HW1, we have 10 classes here. The "code" folder provides the starting code. You must implement the model using the starting code.

   Requirements: Python 3.6, Tensorflow 1.10, tqdm

   This assignment is designed to help you get familiar with Tensorflow. To help you understand all necessary parts, only basic Tensorflow APIs in tf.layers and tf.nn are allowed to use. Most parts of the code are given. Please make sure that you understand how the code works for future assignments.

   In "MNIST_tutorial.py", an implementation using advanced APIs is provided. You can use it to check your results. A link to Tensorflow turorials is also provided.

   In this assignment, you may not need a GPU. The training process takes about 5s/epoch on a GeForce GTX 1080Ti GPU and 100s/epoch on MacBook Pro's CPU. The next assignment will require a GPU.

   (a) (5 points) Read "DataReader.py". Explain why we need to do "flatten" in "load_data()".

(b) (15 points) Complete “Network.py”. Only basic Tensorflow APIs in tf.layers and tf.nn are allowed to use.

(c) (15 points) Complete the validation part in “Model.py”. Use the given testing part as an example.

(d) (10 points) You will run “python main.py” for three times. Follow the instructions in “main.py” and complete it.

(e) (5 points) Report the hyperparameters of your best model and the testing accuracy.

2. (15 points) Exercise 7.7 (e-Chap:7-11) in LFD.

3. (20 points) Exercise 7.8 (e-Chap:7-15) in LFD.

4. (15 points) Exercise 7.14 (e-Chap:7-29) in LFD.