CS 596 Machine Learning

**Homework Assignment 5**

Due: 11:59pm, Mar 17, 2018

The purpose of this homework assignment is to apply the feedforward Neural Network (FNN) to solve a practical classification problem.

**Overview**

In this homework project, you will need to apply the feedforward neural network (FNN) on the MNIST dataset for hand written digit recognition.

**Dataset**

The MNIST database of handwritten digits has a training set of 60,000 digit images, and a test set of 10,000 images. It is a subset of a larger set available from NIST. The digits have been size-normalized and centered in a fixed-size image.

A starter script “main\_ha5.py” is provided for downloading MNIST dataset, splitting datasets, and feature normalization. You don’t need to changes these coding lines. You might need to install the following packages:

* keras
* numpy
* scipy
* matplotlib
* sklearn

**Method**

For every image, we will use its pixel-wise intensities as features. To define a FNN model, you will need to specify the following hyper-parameters:

1. number of hidden layers and number of neural units for each layer;
2. learning rate
3. activation function (sigm, tanh\_opt) and
4. output function (‘sigm’,’linear’, ‘softmax’);

There is no need for you to code the neural network algorithm from scratch. You can use either third-party libraries or the Tensorflow implementations provided along with this homework (folder named ‘code\_FNN\_TF’).

**Question 1**. Please further split the 60,000 training images (and labels) into two subsets: 50,000 images, and 10,000 images. Use these two subsets for training models and validation purposes. In particular, you will train your FNN model using the 50,000 images and labels, and apply the trained model over the rest 10,000 images for evaluation purposes.

Please specify at least three sets of hyper-parameters (see the above). For each set, call the third-party functions or tensorflow to train a FNN model on the training samples (50,000 images in this case), and apply the learned model over the validation set (10,000 images in this case). For each model and its results, please compute its confusion matrix, average accuracy, per-class Precision and Recall. Report the model that achieves the top accuracy.

A sample function for calculating confusion matrix is provided in ‘util.py’

**Question 2**. Apply the top ranked model over the testing samples (10,000 images). Call the above function to compute the confusion matrix, average accuracy, per-class precision/recall rate. In addition, pickup and visualize TEN testing images for which your mode made errors. Try to analyze the reasons of these failure cases.

Note that the validation set are different from the testing set. The former is used to select model hyperparameters and the latter is used to verify the selected model.

**How to submit**

* Submit your source codes and report through the SDSU Blackboard. The codes should be self-contained, and run without any error. Otherwise, severe penalty will be applied.
* Two re-submissions are allowed.
* ***No hard copy required.***