# Softmax for MNIST Classification

## 1 Introduction

MNIST[1] digits dataset is a widely used dataset for image classification in machine learning field. It contains 60,000 training examples and 10,000 testing examples. It is a subset of a larger set available from NIST. The digits have been size-normalized and centered in a fixed-size image. Each sample is a $784 \times 1$ matrix, which is transformed from an original $28 \times 28$ pixels grayscale image. Digits in MNIST range from 0 to 9. Some typical digits images are shown below.

In this homework, you need to use **Softmax Classifier** to perform digits classification.

**Note**: During your training process, information about testing samples in any form should never be introduced.

## 2 Files Description

There are several files included in the source package. Each file's description is listed below.

1) **homework_1.ipynb** is an IPython Notebook file which describes the main content of this homework. You should start the IPython notebook from the *homework-1* directory, with *jupyter notebook* command. Data loading, hyerparameters, training and testing are included in this file. **Please read this file carefully**.
2) **mnist_data_loader.py** is used to load MNIST dataset. Not required reading.
3) **softmax_classifier.py** describes the softmax classifier. **You are required to complete** the function *softmax_classifier(W, input, label, lamda)*. Part of this function is provided and you need to write down your code at '# TODO' to calculate the loss, gradient and prediction. For more details please refer to the file.

1. Here is the mnist page on Prof. LeCun's website.

If you implement the function correctly, just by running homework_1 IPython notebook step by step, you can obtain lines of logging and reach a relatively high test accuracy. Please check your implementation if you encounter any error.

**Note**: Any extra modifications of these files or adding extra Python files should be explained and documented.

## 3 Report

In the experiment report, you are required to complete the following requirements:

1) Record the training and test accuracy, plot the training loss curve and training accuracy curve in the report.
2) The given hyerparameters maybe performed not very well. You can modify the hyerparameters by your own, and observe how does these hyerparameters affect the classification performance. Write down your observation and record these new results in the report.

**Note**: Please convert your report to pdf format.

**Note**: Only some essential lines of codes are permitted to be included for explaining complicated thoughts.

## 4 Attention

1) You need to submit the experiment report and three files mentioned in section 2. Please package and submit it at WebLearning.
2) **Deadline: 2018.10.21 23:30:00**.
3) Any open source neural network toolkits, such as TensorFlow, Caffe, PyTorch, are **NOT** permitted in finishing this homework.
4) **Plagiarism is not permitted**.