CSE 474/574: Introduction to Machine Learning (Fall 2019)

Sargur N. Srihari
University at Buffalo, The State University of New York
Buffalo, New York 14260
Contact: 716-645-6162 (O), srihari@buffalo.edu

September 23, 2019

1 Task

This project is to implement neural network and convolutional neural network for the task of classification.

The classification task will be that of recognizing an image and identify it as one of ten classes. You are required to train the classifiers using Fashion-MNIST clothing images. Following are the three tasks to be performed:

1. Build a Neural Network with one hidden layer to be trained and tested on Fashion-MNIST dataset. Code from scratch in Python.
2. Build multi-layer Neural Network with open-source neural-network library, Keras on Fashion-MNIST dataset.
3. Build Convolutional Neural Network (CNN) with open-source neural-network library, Keras on Fashion-MNIST dataset.

Evaluate the results obtained by each of the classifier (Single layer Neural Network, Multi-Layer neural network and CNN) as shown in the evaluation section.

2 Dataset

For training and testing of our classifiers, we will use the Fashion-MNIST dataset. The Fashion-MNIST is a dataset of Zalando's article images, consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image, associated with a label from 10 classes.

Each image is 28 pixels in height and 28 pixels in width, for a total of 784 pixels in total. Each pixel has a single pixel-value associated with it, indicating the lightness or darkness of that pixel, with higher numbers meaning darker. This pixel-value is an integer between 0 and 255. The training

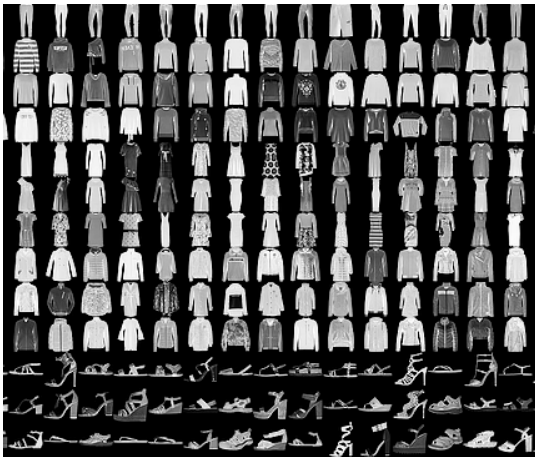

Figure 1: Example of how the data looks like.

and test data sets have 785 columns. The first column consists of the class labels (see above), and represents the article of clothing. The rest of the columns contain the pixel-values of the associated image.

Each training and test example is assigned to one of the labels as shown in table 1.

1	T-shirt/top
2	Trouser
3	Pullover
4	Dress
5	Coat
6	Sandal
7	Shirt
8	Sneaker
9	Bag
10	Ankle Boot

Table 1: Labels for Fashion-MNIST dataset

You can simply load the Fashion MNIST dataset using fashion_mnist_reader notebook present inside the scripts folder.

3 Plan of Work

1. **Extract feature values and labels from the data:** Fashion MNIST dataset is downloaded and processed into a Numpy array that contains the feature vectors and a Numpy array that contains the labels using fashion_mnist_reader notebook present inside the scripts folder.

2. **Data Partitioning:** The Fashion MNIST dataset is originally partitioned into a training set and a testing set. You will use this partition and train your model on the training set.

3. **Train using Neural Network with One Hidden Layer** Use Gradient Descent for neural network to train the model using a group of hyperparameters. (Code from scratch in python)

4. **Train using Multi-Layer Neural Network** with high level Neural Network library, Keras using a group of hyperparameters.

5. **Train using Convolution Neural Network** with high level Neural Network library, Keras using a group of hyperparameters.

6. **Tune hyper-parameters:** For steps 3, 4 and 5: Validate the classfication performance of your model on the validation set. Change your hyper-parameters and repeat the step. Try to find what values those hyperparameters should take so as to give better performance on the testing set.

7. **Test your machine learning scheme on the testing set:** For steps 3, 4 and 5: After tuning the hyper-parameters, fix your hyper-parameters and model parameter and test your models performance on the testing set. This shows the ultimate effectiveness of your models generalization power gained by learning.

4 Evaluation

1. Plot graph of training loss vs number of epochs while training on each classifier (Neural Network with single hidden layer, multi-layer neural network and convolutional neural network).

2. For each classifier evaluate solution on the test set using classification accuracy:

$$Accuracy = \frac{N_{correct}}{N} \tag{1}$$

   Where where $N_{correct}$ is the number of corrected classified data samples, and N is the total number of samples of the validation set.

3. Construct a confusion matrix for each classifier and observe the relative strengths and weaknesses.

5 Deliverables

There are two deliverables: report and code. After finishing the project, you may be asked to demonstrate it to the TAs, particularly if your results and reasoning in your report are not clear enough.

1. Report (30 points)

   The report should describe your results, experimental setup and comparison between the results obtained from different setting of the algorithm. Submit the PDF on a CSE student server with the following script:

   `submit_cse474 proj2.pdf` for undergraduates

   `submit_cse574 proj2.pdf` for graduates

2. Code (70 points)

   The code for your implementation should be in Python only. You can submit multiple files, but the name of the entrance file should be main.ipynb. Please provide necessary comments in the code. Python code and data files should be packed in a ZIP file named `proj2code.zip`. Submit the Python code on a CSE student server with the following script:

   `submit_cse474 proj2code.zip` for undergraduates

   `submit_cse574 proj2code.zip` for graduates