Assignment 3

MSBD 5002 Data Mining, Spring 2020

Due: 11:59 pm, Apr 26th

1 Submission Guidelines

- **Please note that assignment 3 should be submitted via canvas**.
- You need to zip the following two files together:
  - assignment3_studentid_report.pdf/.docx: Please put your report in this file. (Attachments should be original .pdf or .docx, NOT compressed)
  - assignment3_studentid_code.zip: The zip file contains all your source codes for this project.
- All attachments, including report and code, should be named in the format of: student_studentid.zip.
- Submissions not following the rules above are NOT accepted.
- 20 marks will be deducted for every one hour after the deadline.
- Your grade will be based on the correctness, efficiency and clarity.
- The email for Q&A: sdiaa@cse.ust.hk.
- **Plagiarism will lead to zero mark.**

2 Objective

The objective of this project is twofold:

- To acquire a better understanding of neural networks by using a public-domain software package called PyTorch.

3 Major Tasks

This assignment consists of the following tasks:

- To install and learn to use PyTorch.
- To train multilayer perception (MLP) neural network models using all data sets provided, including five binary data sets and one multi-class data set.
- To write up a report.

These tasks will be illustrated in the following sections.

3.1 PyTorch Installation

Detailed guides on installing PyTorch for different computing platforms can be found at the website https://pytorch.org/get-started/locally/. Installing PyTorch using Anaconda or PIP is the recommended way since it is simple and straightforward.

For this assignment, it suffices to use the CPU only version (higher than 1.0) of TensorFlow with Python 3.x. You may also use PyTorch with GPU support if you have a powerful GPU.

You may also need to install matplotlib for plotting figures. However, other high-level machine learning frameworks such as Keras should not be used for this assignment.

4 Data Sets

4.1 Five Binary Classification Data Sets

You will use five binary classification data sets which are available in the ZIP file (datasets.zip). The table 1 shows the number of features, number of training examples, and number of test examples for each data set.

When you load each .npz data file, you will find four NumPy arrays train X, train Y, test X and test Y. Each row of X stores the features of one example and the corresponding row of Y stores its class label (0 or 1). As is always the case, the class label files for the test sets should not be used for classifier training but only for measuring the classification accuracy on the test data.

Table 1: Summary of Binary Classfication Data Sets

Data Set	**#features**	**# train**	**# test**
Breast cancer	10	547	136
Diabetes	8	615	153
Digit	64	800	200
Iris	4	120	30
Wine	13	142	36

4.2 Multi-class Data Sets

The dataset consists of a training set of 10,000 examples and a test set of 1,000 examples. Each example is a 28x28 grayscale image, associated with a label from 10 classes. Please load the data set with *scipy* such as *from scipy.io import loadmat*.

5 Train and Test the Nerual Networks Models

As discussed in class, neural network classifiers generalize logistic regression by introducing one or more hidden layers. Learning of both models may use a (batch or stochastic) **gradient descent** algorithm by minimizing the **cross-entropy loss**. It requires that the step size parameter be specified. Try out a few values and choose one that leads to stable convergence. You may also decrease gradually during the learning process to enhance convergence. A common criterion used for early stopping is when the improvement between iterations does not exceed a small threshold or when the number of iterations has reached a prespecified maximum. Since the solution found may depend on the initial weight values chosen randomly, you may repeat each setting multiple times and report the average classification accuracy.

5.1 Neural Networks Models for Binary Classifcation Data Sets

For 5 binary classifcation data sets $\mathcal{D} = \{D_1, D_2, \cdots, D_5\}$, you are required to construct a set of **single hidden layer** neural network models. The number of hidden units H should be determined using cross validation. The generalization performance of the model is estimated for each candidate value of $H \in \{1, 2, \cdots, 10\}$. This is done by randomly sampling 80% of the training instances to train a classifier and then testing it on the remaining 20%. Hence, given 5 datasets, you are required to perform 5 such random data splits in order to find the most suitable $\mathcal{H}^* = \{H_1^*, H_2^*, \cdots, H_5^*\}$ for 5 data sets, respectively.

Subsequently, given any binary data set D_i, you are requied to train a neural network classifier with H_i^* hidden units in a single layer, which is trained from scratch using all the training instances available.

5.2 Neural Networks Models for Multi-class Data Sets

For the multi-class data set, you are required to construct a set of **two hidden layers** neural network models. The number of hidden units for first layer L_1 is choosen from $\{50, 75, 100\}$, while the number of hidden units for second layer L_2 is choosen from $\{10, 15, 20\}$. Like the single layer NNMs above, the best combination of hidden numbers for each layer is decided by cross validation (randomly sampling 80% of the training instances to train a classifier and then testing it on the remaining 20%).

5.3 Report Writting

In your report, you are expected to present the parameter settings and the experiment results. Besides reporting the classification accuracy (for both training and test data) in numbers, graphical aids should also be used to compare the performance of different settings visually. Some utilities in *scikit-learn* such as *auc* and *confusion matrix* are recommended for analyzing and reporting the experiment results. For the CPU time information, you may just report it in numbers.

6 Tips

6.1 Programming Tips

As is always the case, good programming practices should be applied when coding your program. Below are some common ones but they are by no means complete:

- Using functions to structure program clearly.
- Using meaningful variable and function names to improve readability.
- Using indentation
- Using consistent styles
- Including concise but informative comments

7 Grading Schema

The assignment grading schema has been listed as:

- **Coding:** (60 marks) Essential comment will be helpful for your grading.
  - Build neural network models and adopt the gradient descent optimization algorithm (20 marks).
  - Tune the parameters using cross validation techniques (20 marks).

- Compute the cross entropy loss and the accuracy of the neural network model on both the training and test sets (20 marks).

- **Project Report:** (40 marks)
  - Present the experiment settings of the neural network models such as optimizer and learning rate (10 marks).
  - Report the parameter tuning result of the neural network model using cross validation (20 marks).
  - Report the accuracy of the selected neural network models on the test set (10 marks).

8 Academic Integrity

While you may discuss with your classmates on general ideas about the assignment, your submission should be based on your own independent effort. In case you seek help from any person or reference source, you should state it clearly in your report. Failure to do so is considered plagiarism which will lead to appropriate disciplinary actions.