Project 1

CZ4042: Neural Networks

Deadline: 16th October 2017

- ✓ The project is to be done in a group of not more than two.
- ✓ Complete both part A and B. Data files for both parts can be found under Project 1 under Assignments on NTULearn.
- ✓ One member of the group needs to submit the project report and codes to NTU Learn. The cover page of the report should contain the names of both members. The report should be in pdf format with filename your_name_P1_report.pdf and all the source codes should be submitted in a zip file named: your_name_P1_codes.zip
- ✓ Submit both your report and the source codes on line via NTULearn before the deadline. Late submissions will be penalised.
- ✓ The assessment will be based on both the project report and the correctness of the code submitted.
- ✓ TA Mr. Sukrit Gupta (SUKRIT001@ntu.edu.sg) is in charge of the course projects. Please see him at the Biomedical Informatics Lab (NS4-04-33) during his office hours: Friday 3:30 P.M. – 5:00 P.M., in case you face issues.

Part A: Classification Problem

This project aims at building neural network to classify Landsat satellite dataset:
https://sites.google.com/site/sukritsite/teaching
https://archive.ics.uci.edu/ml/datasets/Statlog+(Landsat+Satellite)

The dataset contains multispectral values of pixels in a 3x3 neighbourhoods in a satellite images and class labels of the centre pixels in each neighbourhood. The aim is to predict class labels in the test dataset after training the neural networks on the training data.

Training data: sat_train.txt, 4435 samples
Test data: sat_test.txt, 2000 samples

Read the data from the two files and use it as train and test set. Do not use the data in the test dataset during training. It is reserved for the final performance measures. Think of it as unseen data during all of your work.

Each data sample is a row of 37 values: 36 input attributes (4 spectral bands x 9 pixels in the neighbourhood), and the class label. There are 6 class-labels: 1, 2, 3, 4, 5, 7.

1. Design a 3-layer feedforward neural network consisting of a hidden-layer of 10 neurons having logistic activation function and an output softmax layer. Assume a learning rate $\alpha = 0.01$, decay parameter $\beta = 10^{-6}$, and batch size = 32. Use appropriate scaling of input features.

(9 marks)

2. Find the optimal batch size for mini-batch gradient descent while training the neural network by evaluating the performances for different batch sizes. Set this as the batch size for the rest of the experiments.
    a) Plot the training errors and test accuracies against the number of epochs for the 3-layer network for different batch sizes. Limit search space to:$\{4,8,16,32,64\}$.
    b) Plot the time taken to update parameters of the network against different batch sizes.
    c) State the rationale for selecting the optimal batch size.

(9 marks)

3. Find the optimal number of hidden neurons for the 3-layer network designed in part (2). Set this number of neurons in the hidden layer for the rest of the experiments.
    a) Plot the training errors and test accuracies against the number of epochs for 3-layer network at hidden-layer neurons. Limit the search space to the set: $\{5,10,15,20,25\}$.
    b) Plot the time to update parameters of the network for different number of hidden-layer neurons
    c) State the rationale for selecting the optimal number of hidden neurons

(9 marks)

4. Find the optimal decay parameter for the 3-layer network designed in part (3).
    a) Plot the training errors against the number of epochs for the 3-layer network for different values of decay parameters in search space$\{0, 10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}\}$.
    b) Plot the test accuracies against the different values of decay parameter.
    c) State the rationale for selecting the optimal decay parameter.

(9 marks)

5. After you are done with the 3-layer network, design a 4-layer network with two hidden-layers, each consisting of 10 neurons with logistic activation functions, a batch size of 32 and decay parameter 10^{-6}.
    a) Plot the train and test accuracy of the 4-layer network.
    b) Compare and comment on the performances on 3-layer and 4-layer networks.

(9 marks)

6. Additionally, the project report should contain:
    a) An *introduction* to the problem of classification of Landsat satellite dataset and the use of multilayer feedforward networks for solving classification problem.
    b) Described the *methods* used in the experiments

(5 marks)

Hint: Sample code is given in file 'begin_project_1a.py' to help you get started with this problem.

Part B: Approximation Problem

This assignment aims to provide you with some exposure to the use of neural networks for regression/approximation problems. Download the California Housing database: https://sites.google.com/site/sukritsite/teaching
http://www.dcc.fc.up.pt/~ltorgo/Regression/cal_housing.html

This database contains attributes of housing complexes in California such as location, dimensions, etc., together with their corresponding prices. The aim is to predict the housing prices in the test dataset after training the neural networks on other attributes in the training data. Read the data from the file 'california_housing.data'.

This is a model selection task, i.e. you have to try out different parameters for the neural network, forming different models and select the one which gives you highest accuracy.

Divide the data into test and train set at a ratio of 0.3: 0.7. You need to scale and normalise only the input features in the data to zero mean and unit standard deviation**.** You need to perform 5-fold cross-validation on the train data in order to select the best models. For this, you will further divide the train data into five folds.

Each data sample is a row of 9 values: 8 input attributes and median housing price.

1. Design a 3-layer feedforward neural network consisting of a hidden-layer of 30 neurons. Use mini-batch gradient descent (with batch size of 32 and learning rate $\alpha = 10^{-4}$) to train the network. Use up to about 1000 epochs for this problem.
   a) Plot the training error against number of epochs for the 3-layer network.
   b) Plot the final test errors of prediction by the network.

   (9 marks)

2. Find the optimal learning rate for the 3-layer network designed. Set this as the learning rate in first hidden layer for the rest of the experiments.
   a) Plot the training errors and validation errors against number of epochs for the 3-layer network for different learning rates. Limit the search space to: $\{10^{-3}, 0.5 \times 10^{-3}, 10^{-4}, 0.5 \times 10^{-4}, 10^{-5}\}$
   b) Plot the test errors against number of epochs for the optimum learning rate.
   c) State the rationale behind selecting the optimal learning rate.

   (12 marks)

3. Find the optimal number of hidden neurons for the 3-layer network designed.
   a) Plot the training errors against number of epochs for the 3-layer network for different hidden-layer neurons. Limit search space to:$\{20,30,40,50,60\}$.
   b) Plot the test errors against number of epochs for the optimum number of hidden-layer neurons.
   c) State the rationale behind selecting the optimal number of hidden neurons

   (12 marks)

4. Design a four-layer neural network and a five-layer neural network, with the first hidden layer having number of neurons found in step (3) and other hidden layers having 20 neurons each. Use a learning rate of $\alpha = 10^{-4}$. Plot the test errors of the 4-layer network and 5-layer network, and compare them with that of the 3-layer network.

(12 marks)

5. Additionally, the project report should contain:
   a) An *introduction* to the problem of approximation of housing prices in the California Housing dataset and the use of multilayer feedforward networks for solving the prediction problem.
   b) The *methods* used in the experiments.

(5 marks)

Hint: Sample code is given in file 'begin_project_1b.py' to help you get started with this problem.