Project 1

CZ4042: Neural Networks

Deadline: 19th October 2018

- ✓ The project is to be done in a group of not more than two students.
- ✓ Need to complete both parts A and B of the project and submit the project report and source codes online via NTULearn before the deadline.
- ✓ Data files for both parts are found in Project 1 folder under Assignments on NTULearn.
- ✓ Both members of the group should submit the project report and codes to NTU Learn, individually using their individual accounts. The cover page of the report should contain the names of both members.
- ✓ The report should contain sections such as (i) introduction, (ii) methods, (iii) experiments and results, and (iv) conclusions. The conclusion should contain answers to all the queries and your experience and conclusions on findings.
- ✓ The assessment will be based on both the project report and the correctness of the code submitted. Late submissions will be penalized: 5% for each day up to three days.
- ✓ The report should be submitted in the following format: Project 1:
  - lastname_firstname_P1_report.pdf (report in pdf format); and
  - lastname_firstname_P1_codes.zip (all the source codes)

- ✓ 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. – 5:30 P.M., in case you face issues.

Part A: Classification Problem

This project aims at building neural networks to classify the Landsat satellite dataset: https://archive.ics.uci.edu/ml/datasets/Statlog+(Landsat+Satellite)

The dataset contains multispectral values of pixels in a 3x3 neighbourhoods in 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 network on training dataset.

Training data: 4435 samples
Test data: 2000 samples

Read the data from the two files: training data from sat_train.txt and testing data from sat_test.txt. Do not use the data in the test dataset during training. It is reserved for computation of 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 feedforward neural network which consists of: an input layer, one hidden perceptron layer of 10 neurons and an output softmax layer. Assume a learning rate $\alpha = 0.01$, L2 regularization with weight decay parameter $\beta = 10^{-6}$, and batch size = 32. Use appropriate scaling of input features.

   (10 marks)

2. Find the optimal batch size by training the neural network by evaluating the performances for different batch sizes.
   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 $S = \{4, 8, 16, 32, 64\}$.
   b) Plot the time taken to train the network for one epoch against different batch sizes.
   c) State the rationale for selecting the optimal batch size.

   Use the batch size for the rest of the experiments.

   (10 marks)

3. Find the optimal number of hidden neurons for the 3-layer network designed in part (2).
   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 number of hidden neurons to $S = \{5,10,15,20,25\}$.
   b) Plot the time to train the network for one epoch for different number of hidden-layer neurons.
   c) State the rationale for selecting the optimal number of hidden neurons.

   (10 marks)

4. Find the optimal decay parameter for the 3-layer network designed with optimal hidden neurons 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 $S = \{0, 10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}\}$.
    b) Plot the test accuracies against the different values of decay parameter.
    c) State the rationale for selecting the optimal decay parameter.

(10 marks)

5. After you are done with the 3-layer network, design a 4-layer network with two hidden-layers, each consisting of 10 perceptrons, trained with 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.

(10 marks)

Hint: Sample code is given in file 'start_project_1a.py' to help you get started with this problem.

Part B: Regression Problem

This assignment uses the data from the California Housing database that contains attributes of housing complexes in California such as location, dimensions, etc., and their corresponding prices. The aim is to predict the housing prices from the other attributes in the data. This task also involves model selection and you will have to select the parameters giving the lowest validation error for prediction.

The California Housing data for this project can be downloaded from
http://www.dcc.fc.up.pt/~ltorgo/Regression/cal_housing.html

Read the data from the file 'california_housing.data'. Each data sample is a row of 9 values: 8 input attributes and the median housing price as targets. Divide the dataset at 70:30 ratio for validation and testing datasets. For selection of the best models, use 5-fold cross-validation on the validation data. The performances should be evaluated on the test data.

1. Design a 3-layer feedforward neural network containing: An input layer, a hidden-layer of 30 neurons having ReLu activation function, and a linear output layer. Use mini-batch gradient descent with a batch size of 32, L_2regularization at weight decay parameter $\beta = 10^{-3}$ and a learning rate $\alpha = 10^{-7}$ to train the network.
   a) Use the validation dataset to train the model and plot validation errors against epochs.
   b) Plot the predicted values and target values for any 50 test samples.

   (10 marks)

2. Find the optimal learning rate for the 3-layer network designed using 5-fold cross-validation on validation data. Let the search space be: $\{0.5 \times 10^{-6}, 10^{-7}, 0.5 \times 10^{-8}, 10^{-9}, 10^{-10}\}$.
   a) Plot cross-validation errors achieved different learning rates.
   b) For the optimal learning rate, plot the test errors against training epochs.

   (13 marks)

3. Find the optimal number of hidden neurons for the 3-layer network designed. Limit search space to: $\{20,40,60,80,100\}$. Use the learning rate from part (2).
   a) Plot the cross-validation errors against the number of hidden-layer neurons.
   b) Plot the test errors against number of epochs for the network consisting of the optimal number of hidden neurons.
   c) State the rationale behind selecting the optimal number of hidden neurons

   (13 marks)

4. Design a four-layer neural network and a five-layer neural network, with the first hidden layer having the number of neurons found in step (3) and other hidden layers having 20 neurons each. Use a learning rate of $\alpha = 10^{-9}$ for all layers. Train four-layer and five-layer networks on validation data and compare their test errors on test data with those on the three-layer networks.

   Introduce dropouts (with a keep probability of 0.9) to the layers and report test errors with and without dropouts.

   (14 marks)

Hint: Sample code is given in file 'start_project_1b.py' to help you get started with this problem.