Project Assignment 2

CZ4042: Neural Networks

Deadline: 13th November 2017

- ✓ The project is to be done in a group of not more than two.
- ✓ Complete both part A and part B. Data files for both parts can be found under Project 2 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_P2_report.pdf and all the source codes should be submitted in a zip file named: your_name_P2_codes.zip
- ✓ Submit both your report and the source codes on line via NTULearn before the deadline. Late submissions will be penalized.
- ✓ The assessment will be based on both the project report and the correctness of the codes 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.
- ✓ For projects, access to Z440 machines Software Lab 3 (N4-B1C-14) is provided on every Monday from 8.30am-5.30pm, Thurday 8.30am-5.30pm and Friday 10.30am-5pm.

The project uses hand-written digit images provided by the MNIST database:

http://yann.lecun.com/exdb/mnist/

The MNIST data and sample codes of the project are given in project2.zip file.

Use only a subset of MNIST data for training and testing. Select first 12,000 images from MNIST train data as training dataset and first 2000 of MNIST test data as testing dataset.

Part A: Deep convolutional neural network

This part of the assignment aims to give you exposure to the use of convolutional neural networks (CNN) for object recognition in images.

1) To recognize MNIST digits, design a convolutional neural network consisting of
    - An Input layer of 28x28 dimensions
    - A convolution layer C_1 of 15 feature maps and filters of window size 9x9. A max pooling layer S_1 with a pooling window of size 2x2.
    - A convolution layer C_2 of 20 feature maps and filters of window size 5x5. A max pooling layer S_2 with a pooling window of size 2x2.
    - A fully connected layer F_3 of size 100.
    - A softmax layer F_4 of size 10.

Train the network using ReLu activation functions for neurons and mini batch gradient descent learning. Set batch size 128, learning rate $\alpha = 0.5$ and decay parameter $\beta = 10^{-6}$.

Plot the training cost and test accuracy with learning epochs.
For two representative test patterns, plot the feature maps at the convolution and pooling layers.

(25 marks)

2) Repeat part 1 by adding the momentum term to mini batch gradient descent learning with momentum parameter $\gamma = 0.5$.

(10 marks)

3) Repeat part 1 by using RMSProp algorithm for learning. Use $\alpha = 0.001$, $\beta = 1e^{-4}$, $\rho = 0.9$, and $\epsilon = 10^{-6}$ for RMSProp.

(10 marks)

4) Write an introduction including the methods used and a discussion on your results in the report.

(5 marks)

Part B: Autoencoders

This part of assignment aims to provide you with some exposure to the use of autoencoders. Use the full MNIST dataset for this problem.

Hints: Use corruption level = 0.1, training epochs = up to about 25, learning rate = 0.1, and batch size = 128 for training of all the layers.

1) Design a stacked denoising autoencoder consisting of three hidden-layers; 900 neurons in the first hidden-layer, 625 neurons in the second hidden-layer, and 400 neurons in the third hidden-layer. To train the network:
    - Use the training dataset of MNIST digits
    - Corrupt the input data using a binomial distribution at 10% corruption level.
    - Use cross-entropy as the cost function

   Plot
   - learning curves for training each layer
   - Plot 100 samples of weights (as images) learned at each layer
   - For 100 representative test images plot
     - reconstructed images by the network.
     - Hidden layer activation

(25 marks)

2) Train a five-layer feedforward neural network recognize MNIST data, initialized by the three hidden layers learned in part (a) and by adding a softmax layer as the output layer. Plot the training error and test error during training.

(10 marks)

3) Repeat part (a) and (b) by introducing the momentum term for gradient descent learning and the sparsity constraint to the cost function. Choose momentum parameter $\gamma = 0.1$, penalty parameter $\beta = 0.5$, and sparsity parameter $\rho = 0.05$. Compare the results with those of part (a) and (b)

(10 marks)

4) Write an introduction including methods used and a discussion on your results in the report.

(5 marks)