EECE 6036: Intelligent Systems

Homework 3: Given 3/16/16; Due 4/7/16

1. (200) Write a program for implementing multi-layer feed-forward neural networks and training them with back-propagation including momentum. Your program should allow you to choose the number of hidden layers, number of hidden neurons in each layer, number of output nreurons, learning rate, momentum, etc. It should be possible to set the learning rate and momentum se[arately for each layer of weights. You do not need to include adaptation of learning rate unless you want to.

Repeat the classification problem given to you in Homework 2 using this network, plotting all the same graphs as you did for the perceptron. For the decision boundary of the final system, you will need to use the method you used to find the decision boundaty of kNN in Homework, i.e., classify points on a grid and color them by the estimated class.

You should use a network with one hidden layer, but you will need some trial-and-error to figure out the best number of hidden neurons.

Write a report providing the same format as that used for the perceptron part of the Homework 2 problem.

Submit a printout of your program as an appendix with the report. You may use any programming language, but you *cannot* use neural network or other simulators that provide pre-programmed versions of back-propagation or any other algorithms. You must write full programs yourself.

The text part of the report, excluding the figures and program, should be no more than 2 pages, 12 point type, double spaced.

2. (200 points) One of the most widely used data set for evaluating machine learning applications in the image analysis area is the MNIST dataset, which provides images of handwritten digits and letters. In this homework, you will use the numbers subset from this dataset.

Two data files are included:

- **Image Data File:** *MNISTnumImages5000.txt* is a text file that has data for 5,000 digits, each a grayscale image of size 28 × 28 pixels (i.e., 784 pixels each). Each row of the data file has 784 values representing the intensities of the image for one digit between 0 and 9. The first hundred images are shown in the included file *first100.jpg*. The MATLAB script used to produce this image is also included as *showMNISTnum.m*.
- **Label Data File:** *MNISTnumLabels5000.txt* is a text file with one integer in each row, indicating the correct label of the image in the corresponding row in the image data file. Thus, the first entry '7' indicates that the first row of the image data file has data for a handwritten number 7.

You need to do the following:

1. Randomly choose 4,000 data points from the data files to form a training set, and use the remaining 1,000 data points to form a test set.
2. Using the program from Problem 1, train a 1-hidden layer neural network to recognize the digits using the training set. You will probably need a fairly large number of hidden neurons – in the range of 100 to 200, but you can try fewer – and several output neurons. I suggest using 10 output neurons – one for each digit – such that the correct neuron is required to produce a 1 and the rest 0. To evaluate performance during training, however, you can use lower thresholds, such as 0.75 and 0.25, as discussed in class. You will probably need hundreds of epochs for learning, so consider using stochastic gradient descent, where only a random subset of the 4,000 points is shown to the network in each epoch. The

performance of the network in any epoch is measured by the fraction of correctly classified points in that epoch. Save this value at the beginning, and then in every tenth epoch (as in Homework 2).

3. After the network is trained, test it on the test set. To evaluate performance on the test data, you can use a soft-max approach, where you consider the output correct if the correct output neuron produces the largest output among all 10 output neurons – even if that output is not above 0.75 or 0.5.

Write a report providing the following information. Each item required below should be placed in a separate section with the heading given at the beginning of the item.:

- **System Description:** A description of all the choices you made – number of hidden neurons, learning rate, momentum, output thresholds, rule for choosing initial weights, criterion for deciding when to stop training, etc.
- **Results:** Report performance of the final network on the training set and the test set using sensitivity, specificity, PPV and NPV just as you did in Homework 2. However, *you will not do 9 runs in this case, just one* (though you may need to try out many networks before finding one that works well). Also plot the time series of the error during training using the data saved at every tenth epoch.
- **Analysis of Results:** You should describe, discuss and interpret the results you got, and why you think they are as they are.

The text part of the report, excluding the figures and program, should be no more than 2 pages, 12 point type, double spaced.

There is no need to submit a separate program for this problem because you would have used the same program as you did for Problem 1.

Save all your data, results, code, etc., for this homework because the next homework will require you to use the same code and produce results that build on the results from Problem 2 in this homework.

If you cannot access the data, please send me mail at. Ali.Minai@uc.edu.

Points will be awarded for: 1) Correctness ; 2) Clarity of description; 3) Quality of the strategy; and 4) Clarity of arguments and presentation.

As in previous homeworks, the report text should not be mixed in with the program. It should be a stand-alone document with text, tables, figures, etc., with the program as an appendix. *None of the information required in the report should be given as a comment or note in the program. It must all be in the report.*

You may consult your colleagues for ideas, but *please write your own programs and come to your own conclusions.*