Name:

STN:

In The Name of Almighty
ECE Dept. Isfahan University of Technology
Statistical Pattern Recognition
Computer Assignment #4

In this assignment, you are to implement a Multi-Layer Perceptron (MLP) ANN and train it using the Back Propagation (BP) algorithm. The Normal database is to be used to train the classifier and then test it. The Normal database samples are two-dimensional and there are two classes. Therefore, the MLP will have three input nodes (x_1, x_2, and a bias node with a constant value of +1) and two output nodes corresponding to class 1 and class 2, respectively. You are to use four hidden nodes plus a bias (+1) node in the hidden layer. Use a Sigmoid function of the form $f(x) = 1/(1+\exp(-x))$ for the transfer function of the hidden and output layer nodes.

The provided Excel spreadsheet contains the initial weights and the normalized training and testing data as discussed in the following.

1. The "Initial Weights" tab contains the randomly assigned initial connection weights. Use these weights to initialize the BP algorithm.
2. The "Training Set" tab contains the training sample pairs with each including the two features along with the desired output node values for this sample. The x_1 and x_2 values do not match with the values you previously had in the Normal database. The reason is that they have been linearly transformed to lie in the range of 0.2 to 0.8 to avoid saturation of the sigmoid functions in the hidden layer. Also note that the order of the samples is changed to alternate between classes 1 and 2. You need to use this ordering sequence for training to avoid biasing the training to a particular class. The desired values for the output nodes are set as (0.95, 0.05) and (0.05, 0.95) for class 1 and class 2, respectively.
3. The "Testing Set" tab contains the normalized testing samples. The order of testing does not matter.

Train this MLP using exactly 500 epochs (iterations over the training set) and a learning rate of 0.2. The weights obtained at the end of the 500th epoch are then taken as the trained weights to be used for classifying the testing set.

To classify a test sample, apply it to the input layer and using the trained weights find the corresponding value for the two output nodes. Then compare these two output values and find which one is higher. If the value of the first output node is higher, the sample gets classified to class 1 and vice versa.

Provide the following statistics:

1- The values of the trained weights in the same format as the initial weights given in the Excel file.

2- The confusion matrix for the classification result.

3- Listing of samples that have been misclassified, e.g., sample no. 35 from class 1, sample no. 234 from class 2, etc.