

## ECE/CS/ME 539 Introduction to Artificial Neural Networks and Fuzzy Systems

### Homework #2

This homework covers the following topics: Multi-layer perceptron, pattern classification

1. (25 points) *Error Back-propagation learning*

Download the training data file `hw2train1`, and the testing data file `hw2test1` from the course web page. Each data sample is a 1 by 4 vector (one row in the file). The first two elements in each vector give the coordinates of the data sample, and the third element is the target value (0 or 1).

Use the training data samples to train a multi-layer perceptron network. Then test the trained network using the test data samples. You may use the back-propagation training algorithm demonstrated in class or develop your own BP training algorithms. For each experiment described below, you should repeat the same experiment 10 times and report the mean value and standard deviation of the result. The following default parameters will be used unless stated otherwise: Epoch size  $K = 64$ , maximum epochs used for training = 1000, input samples will be scaled to  $[-5, 5]$ , and output samples will be scaled to  $[0.2, 0.8]$ . The default learning rate  $\eta = 0.1$  and momentum  $\mu = 0.8$ . By default, hidden neurons use hyperbolic tangent activation function ( $T = 1$ ) and output layer neurons use sigmoidal activation function ( $T = 1$ ). Default MLP configuration is 2-2-2 (two outputs, 2 neurons in the hidden layer, and two inputs).

- (a) (5 points, CC) Tabulate the mean values and standard deviations of the testing set classification rate with respect to  $\alpha = 0.01, 0.1, 0.2, 0.4, 0.8$ . For each value of  $\alpha$ , use two different values of momentum:  $\mu = 0$  and  $0.8$ . Remember for each combination of  $\alpha$  and  $\mu$ , 10 repeated trials are to be performed. Discuss the results briefly.
  - (b) (5 points, CC) Tabulate the mean values and standard deviation of the testing set classification rate when the MLP configuration is 2-h-2 where the number of hidden neurons  $h = 2, 5, 20$ . Discuss the results briefly.
  - (c) (5 points, CC) Tabulate the mean values and standard deviation of the testing set classification rate when the MLP configuration may contain more than one hidden layers. Perform experiments specifically on the following configurations: 2-3-2, 2-3-3-2, 2-3-3-3-2. Discuss the results briefly.
  - (d) (5 points, CC) Based on the experimentation results performed in (a)—(c), determine what will be the best choices of  $\alpha$ ,  $\mu$ , and MLP configurations that give satisfactory results. Briefly justify your choice.
  - (e) (5 points) For the network chosen in part (d), illustrate decision region of each class in the 2D  $x_1$ - $x_2$  feature space. One way to do so is to evaluate the output of the MLP over the grid points within the square region bounded by  $(-10, -10)$  and  $(10, 10)$ . In Matlab, the command `meshgrid` will be useful to generate the plot.
2. (15 points) Consider a two-layer MLP with 2 output neurons, two hidden layer neurons and one input neurons. Label the output layer neuron's activation outputs as  $z_1$ , and  $z_2$ , their corresponding net functions as  $v_1$  and  $v_2$ , their corresponding delta error terms  $\delta_1$ ,  $\delta_2$ ; the hidden layer activation outputs as  $y_1$  and  $y_2$ , their corresponding net functions as  $u_1$  and  $u_2$ , their corresponding delta error terms  $\gamma_1$ ,  $\gamma_2$ ; and the input as  $x$ . Their relations are as follows:

$$\begin{aligned}
v_1 &= w_{10} * y_0 + w_{11} * y_1 + w_{12} * y_2, & y_0 &= 1 \\
v_2 &= w_{20} * y_0 + w_{21} * y_1 + w_{22} * y_2, & y_0 &= 1 \\
z_1 &= f(v_1), & z_2 &= f(v_2) \\
u_1 &= c_{10} * x_0 + c_{11} * x, & x_0 &= 1 \\
u_2 &= c_{20} * x_0 + c_{21} * x, & x_0 &= 1 \\
y_1 &= f(u_1), & y_2 &= f(u_2)
\end{aligned}$$

- (a) (10 points) Derive the delta error back-propagating equations for the four delta error terms:  $\delta_1, \delta_2$  (output layer), and  $\gamma_1, \gamma_2$  (hidden layer). Assume the number of samples used per epoch  $K = 1$ . Hence the index  $k$  can be omitted. Also, the layer index (**I**) will not be needed because neurons in different layers are labeled differently. Assume the sigmoidal activation function is used. Note the expression should not consist of derivatives of a certain function. Rather, the results must be represented with the symbols defined as above.
- (b) (5 points) Derive the weight-update equations for  $w_{21}$ , and  $c_{20}$  using notations defined above.
3. (20 points) *Encoding problem*  
 Consider a special MLP configuration 8-3-8. That is, 8 inputs, 8 outputs, and 3 hidden nodes in the hidden layer. Only 8 training samples as shown in the table below:

Feature								Target							
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Note that each target output vector is identical to the input feature vector.

- (a) (10 points) Develop a MLP using  $\tanh()$  activation function for hidden nodes and sigmoidal activation function for output nodes. Use back-propagation algorithm to train this MLP with above training data. Use the entire training data set as the tuning set. to learn the weights of this MLP. Terminate the training when the tuning error is sufficiently small (you have to try to decide it), AND the position of the maximum output matches the position of 1 in the target vector for each of the eight feature vectors. Turn in (i) the hidden layer weights, (ii) output layer weights, (iii) the output of the hidden nodes and the output nodes for each training sample. Each number should have 4 or fewer fraction digits.
- (b) (5 points, CC) If you are lucky, the output of the 3 hidden nodes with respect to each of the 8 feature vector can be quantized into 8 different binary numbers (not necessarily in this order) 000, 001, 010, 011, 100, 101, 110, 111. Examine the hidden node output reported in part (a) to see if this is the case. You may repeat part (a) several times to observe this to happen. Under this circumstance, the three hidden node can be regarded as a nonlinear *binary encoding* of the eight input features.

Turn in both the hidden node output values, their quantized binary equivalence, and the corresponding output of the output layer nodes in a tabular form

- (c) (5 points, CC) Suppose that both hidden layer and output layer neurons have *linear* activation functions  $f(u) = u$ . Suppose the MLP configuration remains at 8-3-8. Is it possible to derive a set of hidden layer and output layer weights such that the inputs and outputs are identical? If your answer is yes, give an example. If your answer is no, explain why.

4. (20 points) *MLP training using 3way cross validation*

Download three data files **wine1**, **wine2**, and **wine3**. These are 3-way partition of a set of data file for the purpose of wine-recognition. The feature space dimension is 13, and there are 3 classes. The objective of this problem is to find out an optimal configuration of MLP with *one* hidden layer using 3-way cross validation. The activation functions for the output layer will be sigmoidal and for the hidden nodes will be hyperbolic tangent. For each value of  $h$ , perform the following training and testing steps:

Train with wine1, wine2, and test the trained network using wine3.

Train with wine2, wine3, and test the trained network using wine1.

Train with wine3, wine1, and test the trained network using wine2.

The testing results then will be averaged by adding the corresponding confusion matrices, and then calculate the classification rate from the combined confusion matrix. Our goal is to choose a value of  $h$ ,  $1 \leq h \leq 10$  that will yield the highest classification rate as computed according to above method.

- (a) (10 points) Discuss steps you performed to analyze the data and pre-process the data before applying them to the MLP training algorithm.
- (b) (10 points, CC) Submit a table that lists the value of  $h$  versus the corresponding classification rate. Discuss briefly the potential advantages and disadvantages of this approach.

5. (20 points) *Pattern Classifiers Comparison*

Use 3-way cross validation to design and compare the performance of  $k$ -nearest neighbor classifier, and maximum likelihood (Bayesian) classifier with mixture of Gaussian likelihood likelihood function.

- (a) (10 points) Use 3-way cross validation to design a *k-nearest neighbor (kNN)* classifier. Use  $k = 1, 2, 3, 4$ , and  $5$  and decide which value of  $k$  yields best performance.
- (b) (10 points) Design a mixture of Gaussian maximum likelihood (Bayesian) classifier. Assume data samples within each class can be modeled by one or two Gaussian mixtures. Since there are 3 classes, thus, there are 8 possible combinations. Use 3-way cross validation to determine which of the 8 configurations yields highest classification rate.