Introduction to Neural Networks

Homework #2

Due Date: 10/19/2016 by 11:59:59 PM.

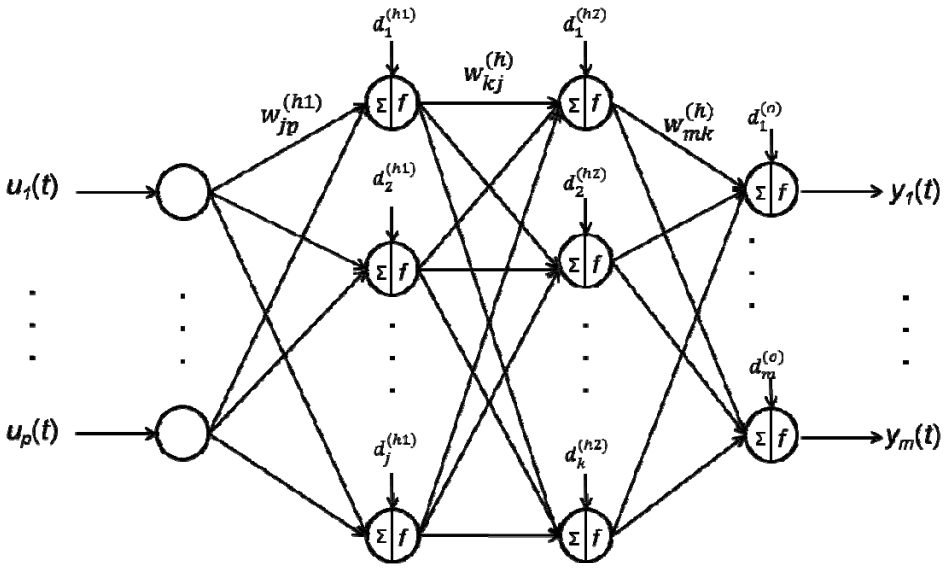

Fig. 1. Structure of the feedforward neural network.

1. Please use the above feedforward neural network to solve the following problems.

Part 1: ***Classification Problems***

(1) Iris (Available at: http://archive.ics.uci.edu/ml/datasets/Iris)

(2) Wine (Available at: http://archive.ics.uci.edu/ml/datasets/Wine)

(3) Breast Cancer Wisconsin (Diagnostic) (Available at: http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29)

(4) Yeast (Available at: http://archive.ics.uci.edu/ml/datasets/Yeast)

The detailed information, including the numbers of features (input variables) and classes, can be obtained from the above websites. It is necessary to change the symbolic features into numerical numbers. For example, in Fig. 1, x_1, x_2, …, x_p are the input variables (features) and y_1, …, y_m are the outputs (types of classes). In the hidden and output neurons, you need to utilize:

a) hyperbolic tangent functions for all the hidden layers

b) sigmoid functions for all the hidden layers

c) hyperbolic tangent functions for the first hidden layer and sigmoid functions for the second layer

as the activation function $f(\bullet)$ to evaluate the network performance.

Part 2: ***Function Approximation***

The target is to use the above network to approximate the following function:

$$\begin{cases} y_1(k) = x_1^2(k) + x_2^2(k) \\ y_2(k) = \dfrac{x_1^2(k) + x_2^2(k)}{3} \end{cases}$$

where $x_1(k)$ and $x_2(k)$ are the inputs, $y_1(k)$ and $y_2(k)$ are the outputs of the FNNBP. A total of

1000 time steps, including 500 time steps of an *i.i.d* uniform sequences within the limits [0, 1] and a sinusoid signals given by $\sin(\pi k/45)$ for the remaining training time, are generated to train the proposed network. In addition, a total of 100 time steps, including 50 time steps of an *i.i.d* uniform sequences within the limits [0, 1] and a sinusoid signals given by $1.05 \times \sin(\pi k/5)$ for the remaining testing time, are generated to test the proposed network.

In your report, you need to finish the following tasks:

1) A working Matlab program
2) Describe the topologies (structures) of the networks that are used to perform the above tasks, including the number of nodes in each layer and the associated weight linkage.
3) The derivations/development of the learning algorithms.
4) Report the best three results out of 10 trials using different initializations. The performance of the networks is based on:
   Part 1: the correct classification rate for each class: correct classifications / total no. of data patterns.
   Part 2: the mean square errors for each output.
5) For the random weight assignment, use the same initial weights that you obtain the best classification result to compare the learning curves with different learning rates (for example: $\alpha = 0.03, 0.1, 0.5$).
6) Discuss how to obtain a better result according to your experience in this homework.