COMPARISON OF FEED FORWARD NETWORK AND ELMAN NETWORK FOR FAULT DIAGNOSIS OF CATHODE RAY OSCILLOSCOPE USING ARTIFICIAL NEURAL NETWORK

Ritu Munjal, Deptt. Of ECE, PIET Samalkha, ritumunjal.4@gmail.com

Abstract

Artificial neural networks (ANN) are an information-processing method of a simulation of the structure for biological neurons. This paper makes a research on the approach of the artificial neural network for fault diagnosis of cathode ray oscilloscope. In the last five years, the field of diagnosis has attracted the attention of many researchers, both from the technical area as well as medical area. In this paper, I present my efforts in developing the fast algorithm for the fault diagnosis of CRO using Artificial Neural Network. Eight different fault indications have been considered and numbers of faults for each indication have been taken. Network has eight input nodes, each for every indication and thirteen output nodes, each for every fault. The network ha trained by feed forward network and Elman network.

Introduction

The neural net first learns the different fault situations. After the network has learnt them, it can do the proper fault diagnosis.

Work on artificial neural networks, commonly referred to as 'neural networks' has been motivated right its inception by the recognition that the human brain computes in an entirely different way from the conventional digital computer.A traditional digital computer does many tasks very well. It's quite fast, and it does exactly what you tell it to do. Unfortunately, it can't help you when you yourself don't fully understand the problem you want solved. Even worse, standard algorithms don't deal well with noisy or incomplete data, yet in the real world, that's frequently the only kind available. One answer is to use an artificial neural network (ANN), a computing system that can learn on its own. In this paper, feed forward network and Elman network both applied for fault diagnosis of CRO. The cost function in this case is the Mean Square difference between the desired and actual network outputs. The Gradient search technique is used to minimize this cost function.

The network is trained by initially selecting small random weights and internal thresholds and then presenting all training data reputedly. Weights are adjusted after every trial using side information (desired results) specifying the correct class until weights converge and the cost function is reduced to an acceptable value. An essential component of the algorithm is the iterative method described below that propagates error terms required to adapt weights back from nodes in the output layer to nodes in lower layers.

Inputs from the outside World are fed to the nodes of the input layer. Each input layer node is connected to every node of the first hidden layer through weights. Every node of one hidden layer is connected to every node of the next higher layer (if any) and so on. The nodes of the last hidden layer are connected to the output nodes.

The activation function of the response of the node is a non-linear continuous function. The thresholds of these functions are set considering the requirements. The non-linear function used is the sigmoidal function. The Sigmoidal function is chosen because it functional response is more closely related to biological responses and hence a step towards brain modeling. Since the network training algorithm is supervised, the desired outputs are necessary and serve as a reference to calculate errors.

The weights of the connections between the nodes of one layer and that of the next layer are adjusted accordingly in such a way that the overall Mean Square error is minimized.

Multi-layer feed forward networks are always trained in supervised manner with a highly popular algorithm known as the error back propagation algorithm. Basically, error back propagation learning consists of two passes through the different layers of the network: a forward pass and a backward pass. In the forward pass, an activity pattern (input vector) is applied to the sensory node of the network, and its effect propagates through the network layer by layer. Finally, a set of outputs is produced as the actual response of the network. The synaptic weights of the networks are all fixed during the forward pass. The backward pass starts at the output layer by passing error signals leftward through the network and recursively computing the local gradient for each neuron. This permits the synaptic weights of the network to be all adjusted in accordance with an error-correction rule. A multi-layer perceptron has three distinctive characteristics: -

The model of each neuron in the network includes a non-linear activation function

The network contains one or more layers of hidden neurons that are not part of the input or output of the network;

The network exhibits a high degree of connectivity, determined by the synapses of the network. Multi-layer feed forward structures are characterized by directed layered graphs and are the generalization of those earlier single layer structures.A typical multi-layer feed forward network consists of a set of sensory units that constitute the input layer, one or more hidden layers of computation nodes, and an output layer of computation nodes. The input signal propagates through the network in a forward direction on a layer-by-layer basis. To illustrate this process the three layer neural network with two inputs and one output,which is shown in the picture below 1, is used. Each neuron is composed of two units. First unit adds products of weights coefficients and inputsignals.

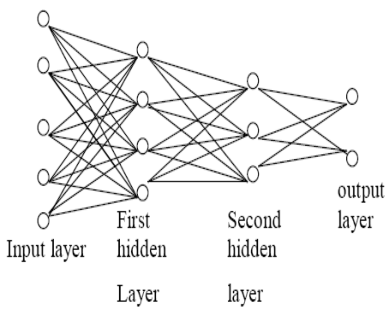


Figure-1

The second unit realise nonlinear function,called neuron activation function. Signal e is adder output signal, and y = f(e) is output signal of nonlinear element. Signal y is also output signal of neuron.

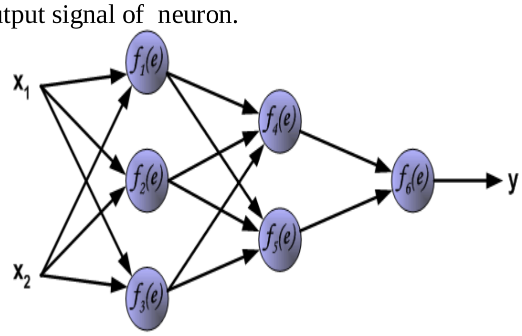


Figure-2

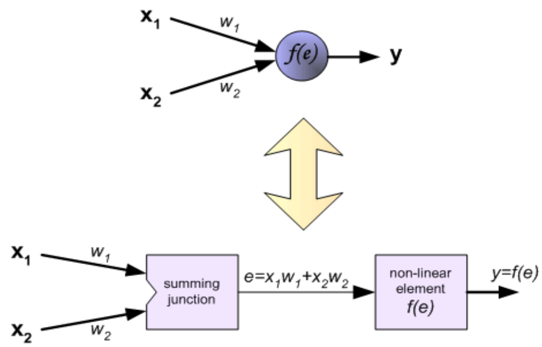


Figure-3

y_n represents output signal of neuron n.

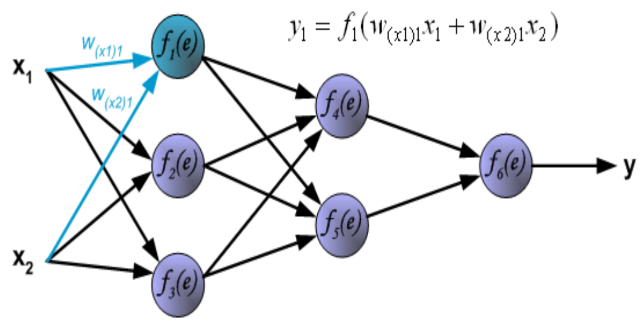


Figure-4

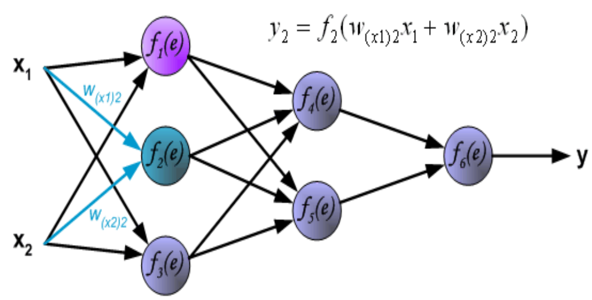


Figure-5

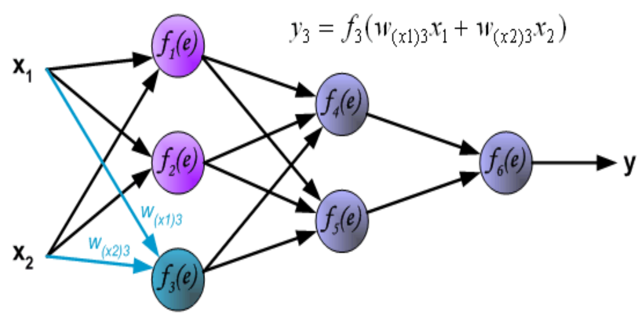


Figure-6

Propagation of signals through the hidden layer. Symbols w_{mn} represents weights of connections between output of neuron m and input of neuron n in the next layer.

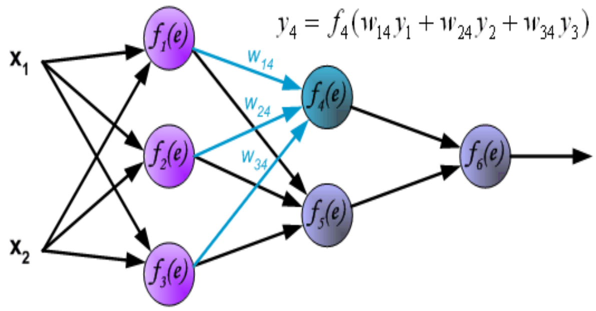


Figure-7

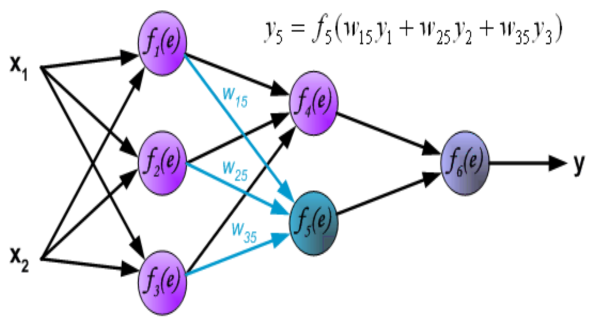


Figure-8

Propagation of signals through the output layer.

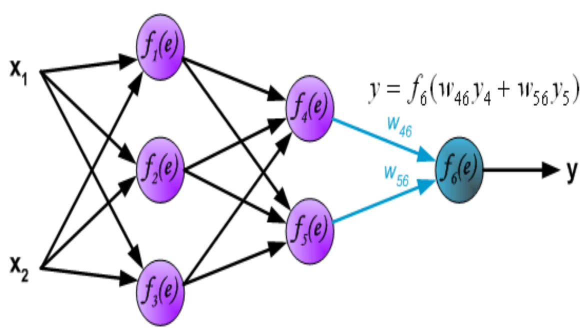


Figure-9

Training

The network is trained by feed forward and Elman network. The Elman network commonly is a two-layer network with feedback from the first-layer output to the first layer input. This recurrent connection allows the Elman network to both detect and generate time-varying patterns. The Elman network differs from conventional two-layer networks in that the first layer has a recurrent connection. The delay in this connection stores values from the previous time step, which can be used in the current time step.

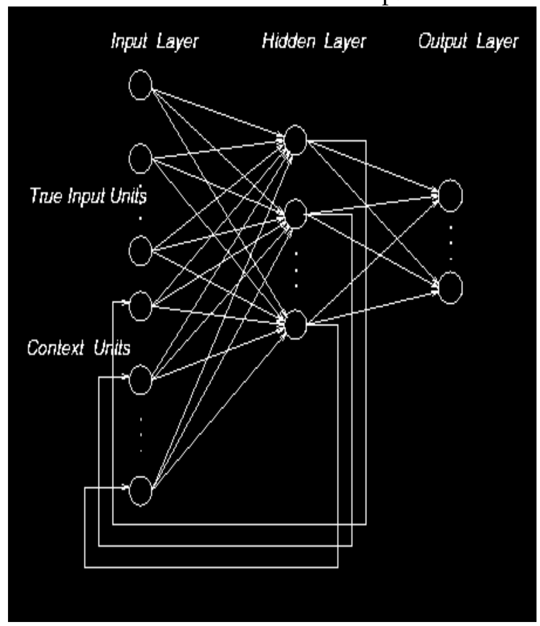


Figure-10

Random Initialization

The choice of initial weights will influence whether the net reaches a global (or only a local) minima of the error and, if so, how quickly it converges. The values for the initial weights must not be too large, or the initial input signals to each hidden or output unit will be likely to fall in the region where the derivative of the sigmoid function has a very small value (the saturation region). On the other hand, if the initial weights are too small, the net input to a hidden or output unit will be close to zero, which also causes extremely slow learning.

A common procedure is to initialize the weights to random values between -0.5 and 0.5 (or between -1 and 1 or some other suitable interval). The values may be positive or negative because the final weights, after training, may be of either sign.

Data Representation

In many problems, input vectors and output vectors have components in the same range of values. Because one factor in the weight correction expression is the activation of the lower unit, units whose activations are zero will not learn. This suggests that learning may be improved if the input is represented in bipolar form and the bipolar sigmoid is used for the activation function.

Number Of Hidden Layers

Theoretical results show that one hidden layer is sufficient for a feed forward net to approximate any continuous mapping from the input patterns to the output patterns to an arbitrary degree of accuracy. Although, two hidden layers may make training easier in some situations, yet the complexity of the circuit increases and the training becomes much slower because more layers require more weights. So the number of layers should be judiciously selected taking into consideration the complexity and economy of the circuit and its solution.

Number Of Hidden Units

A relationship for the number of hidden units is given as follows

h may equal or greater than this expression $\log_{10}m$ / $\log s_{10}2$ depends on the problem

where h are the number of hidden units and m are the number of output units.

Fault Diagnosis With Multi Layer Perceptron

The network has eight input nodes, one for every indication in fault and thirteen output nodes, one for every examined faulty section. The number of nodes in the hidden layer is nine. Both the hidden nodes and the output nodes use the bipolar sigmoid as the activation function given as follows:

$$f(z_{in}(j)) = \frac{2}{1+\exp(-z_{in}(j))} - 1 \quad \text{for hidden nodes}$$

$$f(y_{in}(k)) = \frac{2}{1+\exp(-y_{in}(k))} - 1 \quad \text{for output nodes.}$$

Eight different fault indications has been considered[23]. They are as listed below:

Symptom No.1: Main fuse blown out on switching on the instrument.

a) *Fault No.1:* Shorted Power supplies
b) *Fault No.2:* Shorted primary winding of the transformer

***Symptom No. 2* :** No synchronization

a) *Fault No.3.*Synchronization ckt. Defective

***Symptom No. 3*:** No spot or trace

a) *Fault No.4* grid to cathode potential of CRT was being improper
b) *Fault No.5*. Bad CRT
c) *Fault No.6*. Open filament and cathode connection of CRT

Symptom No.4: Pilot lamp does not glow

a) *Fault No.7*. Defective lamp
b) *Fault No.8* Defective mains cable or fuse blown out

Symptom No.5: General loss of intensity

a) *Fault No.5*. Bad CRT
b) *Fault No.9* Defective high voltage power supply
c) *Fault No 10*. Mains voltage being too low

Symptom No.6:.No vertical shift

a) *Fault No.11* Defective vertical amplifier

Symptom No. 7: No horizontal shift

a) *Fault No.12*. Defective horizontal amplifier

Symptom No. 8: No horizontal deflective improper trace length

a) *Fault No.13* Defective time base

The training data consists of 8 different inputs corresponding to the above mentioned 8 different indications and are as given below

X0	X1	X2	X3	X4	X5	X6	X7
1	-1	-1	-1	-1	-1	-1	-1
-1	1	-1	-1	-1	-1	-1	-1
-1	-1	1	-1	-1	-1	-1	-1
-1	-1	-1	1	-1	-1	-1	-1
-1	-1	-1	-1	1	-1	-1	-1
-1	-1	-1	-1	-1	1	-1	-1
-1	-1	-1	-1	-1	-1	1	-1
-1	-1	-1	-1	-1	-1	-1	1

The output patterns corresponding to the different input patterns are given as under

to	t1	t2	t3	t4	t5	t6	t7
-1	1	-1	-1	-1	-1	-1	-1
1	-1	-1	-1	-1	-1	-1	-1
1	-1	-1	-1	-1	-1	-1	-1
-1	-1	1	-1	-1	-1	-1	-1
-1	-1	1	-1	1	-1	-1	-1
-1	-1	1	-1	-1	-1	-1	-1
-1	-1	-1	1	-1	-1	-1	-1
-1	-1	-1	1	-1	-1	-1	-1
-1	-1	-1	-1	1	-1	-1	-1
-1	-1	-1	-1	1	-1	-1	-1
-1	-1	-1	-1	-1	1	-1	-1
-1	-1	-1	-1	-1	-1	1	-1
-1	-1	-1	-1	-1	-1	-1	1

After training, fault number 1 should produce a positive value at the second and third output and a negative value for all other outputs. Similarly, fault no 3 should produce a positive value for the fourth , fifth and sixth output nodes and negative values for all other nodes.

Network Design And Training

The step by step procedure followed for the deign and training of the neural network is as given below

Design and training of the neural net

A multilayer perceptron network is designed such that it has ni input nodes (each corresponding to some indication),op output nodes (each corresponding to a server) and h hidden units in a single hidden layer.

Each input symptom in the form of a bipolar representations given to the feed forward network

For every iteration, all the input sets are given to the net and weights are modified accordingly.

The training continues till sufficient number of epochs after which the network responds correctly to each of the input symptom.

The weights of the network are then final and the network is said to have completed training.

The trained network is now ready for fault classification.

Fault classification

Any one of the symptom of the faults represented in a bipolar from are given to the network as an input.

Of the outputs, the one which has a positive value or is > 0 is the corresponding class number.

The algorithm has been implemented in MATLAB.

Results and conclusions

I applied the feedforward network and elman network for fault diagnosis of CRO and found that elman neural network is faster than feed forward network.Feed forward net has trained the network in 248 epochs whereas elman net has trained the network in 210 epochs.

TRAINGDX, Epoch 0/500, MSE 1.71973/0.01, Gradient 0.55609/1e-006

TRAINGDX, Epoch 5/500, MSE 1.70404/0.01, Gradient 0.552355/1e-006

TRAINGDX, Epoch 10/500, MSE 1.68679/0.01, Gradient 0.547909/1e-006

TRAINGDX, Epoch 15/500, MSE 1.66661/0.01, Gradient 0.543041/1e-006

TRAINGDX, Epoch 20/500, MSE 1.64217/0.01, Gradient 0.538421/1e-006

TRAINGDX, Epoch 25/500, MSE 1.61201/0.01, Gradient 0.535109/1e-006

TRAINGDX, Epoch 30/500, MSE 1.5743/0.01, Gradient 0.533991/1e-006

TRAINGDX, Epoch 35/500, MSE 1.52666/0.01, Gradient 0.534567/1e-006

TRAINGDX, Epoch 40/500, MSE 1.46616/0.01, Gradient 0.534717/1e-006

TRAINGDX, Epoch 45/500, MSE 1.38942/0.01, Gradient 0.534604/1e-006

TRAINGDX, Epoch 50/500, MSE 1.29203/0.01, Gradient 0.536584/1e-006

TRAINGDX, Epoch 55/500, MSE 1.17015/0.01, Gradient 0.51953/1e-006

TRAINGDX, Epoch 60/500, MSE 1.03104/0.01, Gradient 0.47975/1e-006

TRAINGDX, Epoch 65/500, MSE 0.880328/0.01, Gradient 0.460508/1e-006

TRAINGDX, Epoch 70/500, MSE 0.721281/0.01, Gradient 0.381354/1e-006

TRAINGDX, Epoch 75/500, MSE 0.572699/0.01, Gradient 0.321887/1e-006

TRAINGDX, Epoch 80/500, MSE 0.449076/0.01, Gradient 0.252017/1e-006

TRAINGDX, Epoch 85/500, MSE 0.360604/0.01, Gradient 0.192213/1e-006

TRAINGDX, Epoch 90/500, MSE 0.294751/0.01, Gradient 0.156627/1e-006

TRAINGDX, Epoch 95/500, MSE 0.249597/0.01, Gradient 0.124825/1e-006

TRAINGDX, Epoch 100/500, MSE 0.214911/0.01, Gradient 0.0869454/1e-006

TRAINGDX, Epoch 105/500, MSE 0.188086/0.01, Gradient 0.0751302/1e-006

TRAINGDX, Epoch 110/500, MSE 0.161704/0.01, Gradient 0.0648984/1e-006

TRAINGDX, Epoch 115/500, MSE 0.134777/0.01, Gradient 0.0555856/1e-006

TRAINGDX, Epoch 120/500, MSE 0.113518/0.01, Gradient 0.0559803/1e-006

TRAINGDX, Epoch 125/500, MSE 0.0981795/0.01,Gradient 0.0312805/1e-006

TRAINGDX, Epoch 130/500, MSE 0.0902003/0.01,Gradient 0.0177374/1e-006

TRAINGDX, Epoch 135/500, MSE 0.0857136/0.01,Gradient 0.0121819/1e-006

TRAINGDX, Epoch 140/500, MSE 0.0827705/0.01,Gradient 0.00764228/1e-006

TRAINGDX, Epoch 145/500, MSE 0.0808744/0.01,Gradient 0.00521507/1e-006

TRAINGDX, Epoch 150/500, MSE 0.0796412/0.01,Gradient 0.0035515/1e-006

TRAINGDX, Epoch 155/500, MSE 0.0788306/0.01,Gradient 0.00269549/1e-006

TRAINGDX, Epoch 160/500, MSE 0.0782529/0.01,Gradient 0.00200196/1e-006

TRAINGDX, Epoch 165/500, MSE 0.0778168/0.01,Gradient 0.00177295/1e-006

TRAINGDX, Epoch 170/500, MSE 0.0773679/0.01,Gradient 0.00245677/1e-006

TRAINGDX, Epoch 175/500, MSE 0.0748283/0.01,Gradient 0.0170256/ 1e-006

TRAINGDX, Epoch 180/500, MSE 0.0434709/0.01,Gradient 0.0390613/ 1e-006

TRAINGDX, Epoch 185/500, MSE 0.03953/0.01, Gradient 0.00477501/ 1e-006

TRAINGDX, Epoch 190/500, MSE 0.0389715/0.01,Gradient 0.00241079/ 1e-006

TRAINGDX, Epoch 195/500, MSE 0.0387296/0.01,Gradient 0.00129727/ 1e-006

TRAINGDX, Epoch 200/500, MSE 0.0386164/0.01,Gradient 0.000600437/ 1e-006

TRAINGDX, Epoch 205/500, MSE 0.0385496/0.01,Gradient 0.000268482/ 1e-006

TRAINGDX, Epoch 210/500, MSE 0.0385223/0.01,Gradient 0.000217285/ 1e-006

TRAINGDX, Epoch 215/500, MSE 0.0385049/0.01,Gradient 0.000107306/ 1e-006

TRAINGDX, Epoch 220/500, MSE 0.0384943/0.01,Gradient 9.73086e-005/ 1e-006

TRAINGDX, Epoch 225/500, MSE 0.0384836/0.01,Gradient 7.81641e-005/1e-006

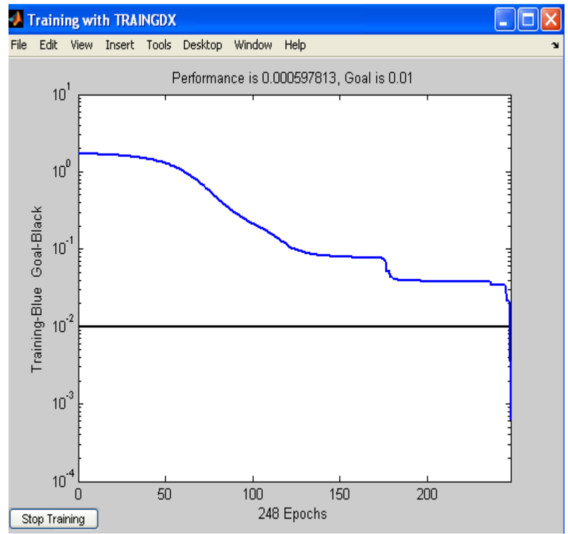
TRAINGDX, Epoch 230/500, MSE 0.0384707/0.01,Gradient 0.000118986/ 1e-006

TRAINGDX, Epoch 235/500, MSE 0.0383714/0.01,Gradient 0.000857009/ 1e-006

TRAINGDX, Epoch 240/500, MSE 0.0349635/0.01,Gradient 0.0216903/ 1e-006

TRAINGDX, Epoch 245/500, MSE 0.0349635/0.01,Gradient 0.0216903/ 1e-006

TRAINGDX, Epoch 248/500, MSE 0.000597813/0.01, Gradient 0.00396039/ 1e-006



TRAINGDX, Performance goal met.

TRAINGDX, Epoch 0/500, MSE 1.34768/0.01, Gradient 0.696783/1e-006

TRAINGDX, Epoch 5/500, MSE 1.32302/0.01, Gradient 0.692354/1e-006

TRAINGDX, Epoch 10/500, MSE 1.29599/0.01, Gradient 0.683516/1e-006

TRAINGDX, Epoch 15/500, MSE 1.26472/0.01, Gradient 0.668984/1e-006

TRAINGDX, Epoch 20/500, MSE 1.22788/0.01, Gradient 0.647374/1e-006

TRAINGDX, Epoch 25/500, MSE 1.18469/0.01, Gradient 0.618259/1e-006

TRAINGDX, Epoch 30/500, MSE 1.13494/0.01, Gradient 0.583299/1e-006

TRAINGDX, Epoch 35/500, MSE 1.07875/0.01, Gradient 0.546534/1e-006

TRAINGDX, Epoch 40/500, MSE 1.01599/0.01, Gradient 0.514086/1e-006

TRAINGDX, Epoch 45/500, MSE 0.945083/0.01, Gradient 0.494702/ 1e-006

TRAINGDX, Epoch 50/500, MSE 0.861842/0.01, Gradient 0.483891/ 1e-006

TRAINGDX, Epoch 55/500, MSE 0.766919/0.01, Gradient 0.413141/ 1e-006

TRAINGDX, Epoch 60/500, MSE 0.685709/0.01, Gradient 0.301192/ 1e-006

TRAINGDX, Epoch 65/500, MSE 0.622299/0.01, Gradient 0.310998/ 1e-006

TRAINGDX, Epoch 70/500, MSE 0.536347/0.01, Gradient 0.368007/ 1e-006

TRAINGDX, Epoch 75/500, MSE 0.431175/0.01, Gradient 0.216346/ 1e-006

TRAINGDX, Epoch 80/500, MSE 0.384465/0.01, Gradient 0.147001/ 1e-006

TRAINGDX, Epoch 85/500, MSE 0.35655/0.01, Gradient 0.111598/1e-006

TRAINGDX, Epoch 90/500, MSE 0.334909/0.01, Gradient 0.100158/ 1e-006

TRAINGDX, Epoch 95/500, MSE 0.315148/0.01, Gradient 0.0748729/

1e-006

TRAINGDX, Epoch 100/500, MSE 0.297444/0.01, Gradient 0.0654476/ 1e-006

TRAINGDX, Epoch 105/500, MSE 0.277474/0.01, Gradient 0.0745771/ 1e-006

TRAINGDX, Epoch 110/500, MSE 0.245961/0.01, Gradient 0.0917746/ 1e-006

TRAINGDX, Epoch 115/500, MSE 0.202905/0.01, Gradient 0.105107/ 1e-006

TRAINGDX, Epoch 120/500, MSE 0.17372/0.01, Gradient 0.0603217/ 1e-006

TRAINGDX, Epoch 125/500, MSE 0.152556/0.01, Gradient 0.0559121/ 1e-006

TRAINGDX, Epoch 130/500, MSE 0.131136/0.01, Gradient 0.0545722/ 1e-006

TRAINGDX, Epoch 135/500, MSE 0.120899/0.01, Gradient 0.0143719/ 1e-006

TRAINGDX, Epoch 140/500, MSE 0.118084/0.01, Gradient 0.00801381/ 1e-006

TRAINGDX, Epoch 145/500, MSE 0.11641/0.01, Gradient 0.0067289/ 1e-006

TRAINGDX, Epoch 150/500, MSE 0.113835/0.01, Gradient 0.0157034/ 1e-006

TRAINGDX, Epoch 155/500, MSE 0.0907214/0.01, Gradient 0.0991985/ 1e-006

TRAINGDX, Epoch 160/500, MSE 0.0510374/0.01, Gradient 0.0508183/1e-006

TRAINGDX, Epoch 165/500, MSE 0.0438112/0.01, Gradient 0.0228428/ 1e-006

TRAINGDX, Epoch 170/500, MSE 0.0402424/0.01, Gradient 0.00593147/ 1e-006

TRAINGDX, Epoch 175/500, MSE 0.039275/0.01, Gradient 0.00212098/ 1e-006

TRAINGDX, Epoch 180/500, MSE 0.0389066/0.01, Gradient 0.00154331/ 1e-006

TRAINGDX, Epoch 185/500, MSE 0.0386998/0.01, Gradient 0.000945699/ 1e-006

TRAINGDX, Epoch 190/500, MSE 0.0385461/0.01, Gradient 0.000822602/ 1e-006

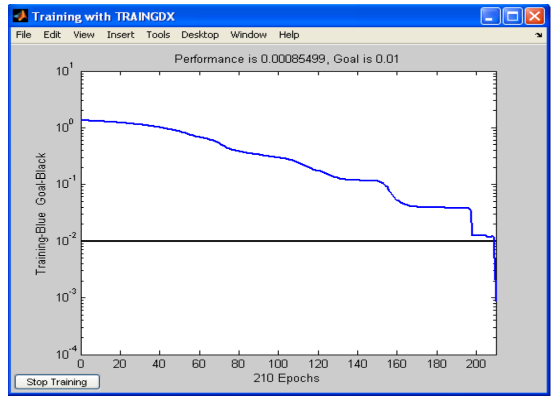
TRAINGDX, Epoch 195/500, MSE 0.0380666/0.01, Gradient 0.00343027/ 1e-006

TRAINGDX, Epoch 200/500, MSE 0.0126426/0.01, Gradient 0.0431767/ 1e-006

TRAINGDX, Epoch 205/500, MSE 0.0126426/0.01, Gradient 0.0431767/ 1e-006

TRAINGDX, Epoch 210/500, MSE 0.00085499/0.01, Gradient 0.00274268/ 1e-006

TRAINGDX, Performance goal met.



References

[1] X.Z Gao,et al.,"*Neural network based Fault Detection with application in Ink jet Printers",*Helsinki University of Technology,Finland

[2] Nikitas J.Dimopoulus,et al.,"*Neural network in Fault Identification in large Communication networks,"*Proceedings of 2nd IEEE Mediterranean Symposium on new directionsin control and automation, June19-22, 1994,Maleme-chania,Crete, Greece

[3] J.M.Zurada*ArtificialNeuralSystems*.West Publishing Co.N.York:Also Jaico Pub. House,Bombay.

[4] Simon Haykin, *Neural Networks- A comprehensive Foundation*.Pentrice Hall, Upper Saddle River, New Jersy07458

[5] B.Yegnanarayana, *Artificial Neural* .PHI

[6] Faisel J Uppal,et al., "*Fault Diagnosis of an Electro-pneumatic Valve Actuator Using Neural Networks with fuzzy capabilities",*Proceedings of the European Symposium on Bruges (Belgium), 24-26 April2002,pp 501-506.

[7] Takashi Torigoe,et.al,"*Application of Neural network to fault diagnosis of Electro-mechanical system",*Memoirs of the Faculty of Engineering, Okayama University, Vol 39, pp 21-27,january,2005

[8] Gurney K., *An Introduction to Neural Networks*, UCL Press, 1997

[9] T. Kohonen, An Introduction to neural computing. Neural Networks Vol. 1 1990, 1-6.

[10] N.K. Bose, P. Liang, Neural network fundamentals with graphs, algorithms and applications. McGraw Hill International Editions, 1996.

[11] Heyi Wang, Yi Gao, "*Elman's Recurrent neural network Applied to Forecasting the quality of water Diversion in the Water Source Of Lake Taihu"* 2010 International Conference on Biology, Environment and Chemistry *IPCBEE vol.1 (2011) © (2011) IACSIT Press, Singapore*.