Volume 3, Issue 4, April 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Application of the Neural Network in System Identification

Sunil Kumar Yadav [1]
Department of Computer Science & Engineering
Amity School of Engineering & Technology
Amity University Uttar Pradesh, Noida, India.

Prof. (Dr.) Sanjeev Bansal [2]
Department of Computer Science & Engineering
Amity School of Engineering & Technology
Amity University Uttar Pradesh, Noida, India.

Abstract: This paper present the new application of ANN in any model identification .For this we adopt an artificial neural network (ANN) architecture which include input layer, hidden layer and output layer ,we apply linear function at the input layer sigmoid activation function at the hidden layers and output layers. The training algorithm is based on back propagation algorithm, the number of neuron in the input layer and output layer completely depend on the mathematical expression of the system.

Keywords-: Artificial neural network (ANN), feed forward architecture.

Introduction

Now a days the use of the ANN in different sectors such as electric power supply, electronic field computer science and engineering, pattern reorganization, artificial intelligence mathematical system etc. By taking a simple mathematical function which includes independent and dependent variables, first we have to prepare the data sheet in the form of pattern (t,s) where "t" represents the target output and "s" is the input sets. In this paper we use the back propagation training algorithms (weight adjustment in the reverse model) one of new ANN technologies is based on operating principal of human brain nerve neural, basically it perform the function of input output mapping or pattern recognition due to its property of associative memory .Here we used the supervised learning based training algorithm. The network has the capability to learn because of the distributed intelligence contributed by the weights.

Artificial Neural Networks (ANN) has emerged as a powerful learning technique to perform complex tasks in highly nonlinear dynamic environments. Some of the prime advantages of using ANN models are their ability to learning based optimization of an appropriate error function and their excellent performance for approximation of nonlinear function

An artificial neural network basically consists a number of computing elements, called neurons that perform the weighted sum of the input signal and the connecting weight. The sum is added with the bias or threshold and the resultant signal is then passed through a non-linear element such as tanh(.) type. Each neuron is associated with three parameters for which learning of neuron can be adjusted, these are the connecting weights, the bias and the slope of the non-linear function. From the structural point of view, a neural network (NN) may be single layer or it may be multi-layer. In multi-layer structure, there may be more than one hidden layers and there is one or many artificial neurons in each layer and for a practical case there may be a number of layers. Each neuron of one layer is connected to each and every neuron of the next layer.

We use ANN because of some given ability-:

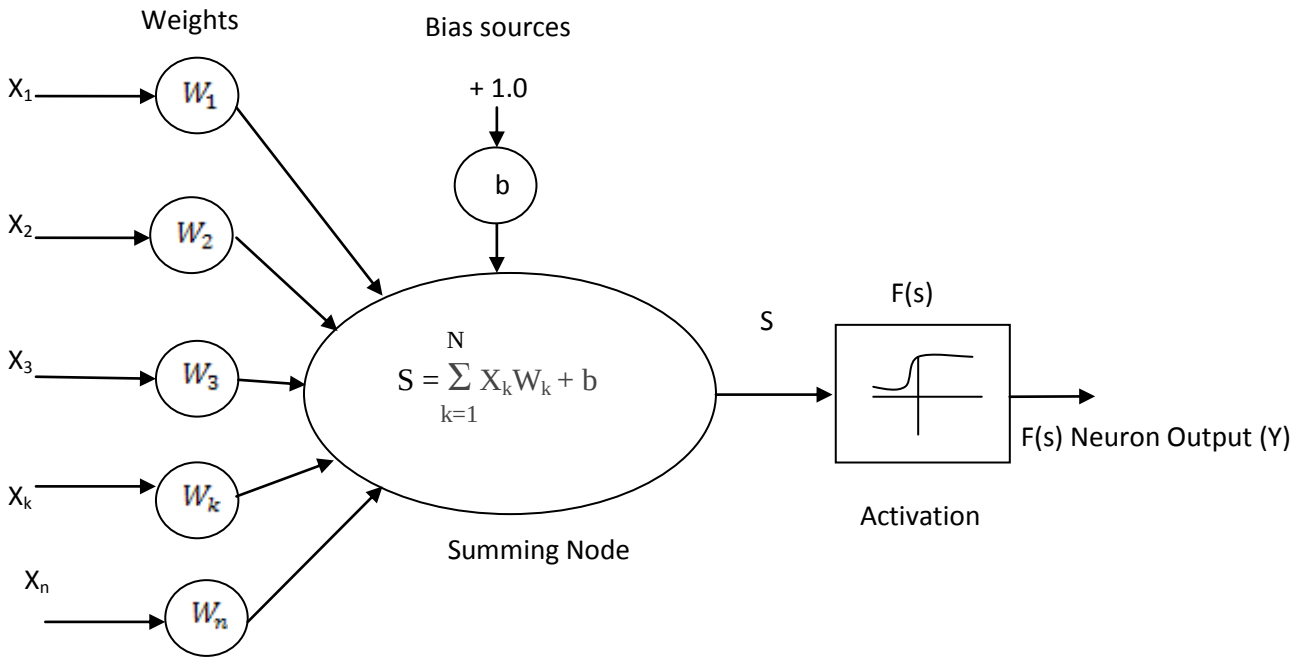
1) All the ANN signals are propagated in unidirectional.
2) The ability of ANN to learn the parameters.
3) The ability to creating the parallel signal in both analog as well as discrete system.
4) The adaptive ability.

This input/output pattern matching is possible if the appropriate weights are selected. We can write output signals as function of inputs

$$Y_1 = \mathbf{f}(X_1, X_2, X_3)$$

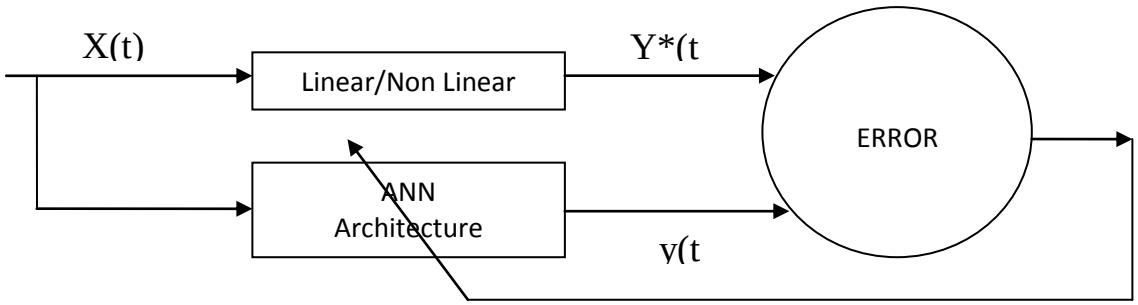
Where Y_1 is output and X_1, X_2, X_3 are inputs

For each input pattern of signals, there will be a corresponding desired output pattern



Structure of Artificial neural Network

PROBLEM STATEMENT



Where
Y*(t) = Target output, Y(t) = Actual output
Error = Y*(t) – y(t), X(t) = Input signal

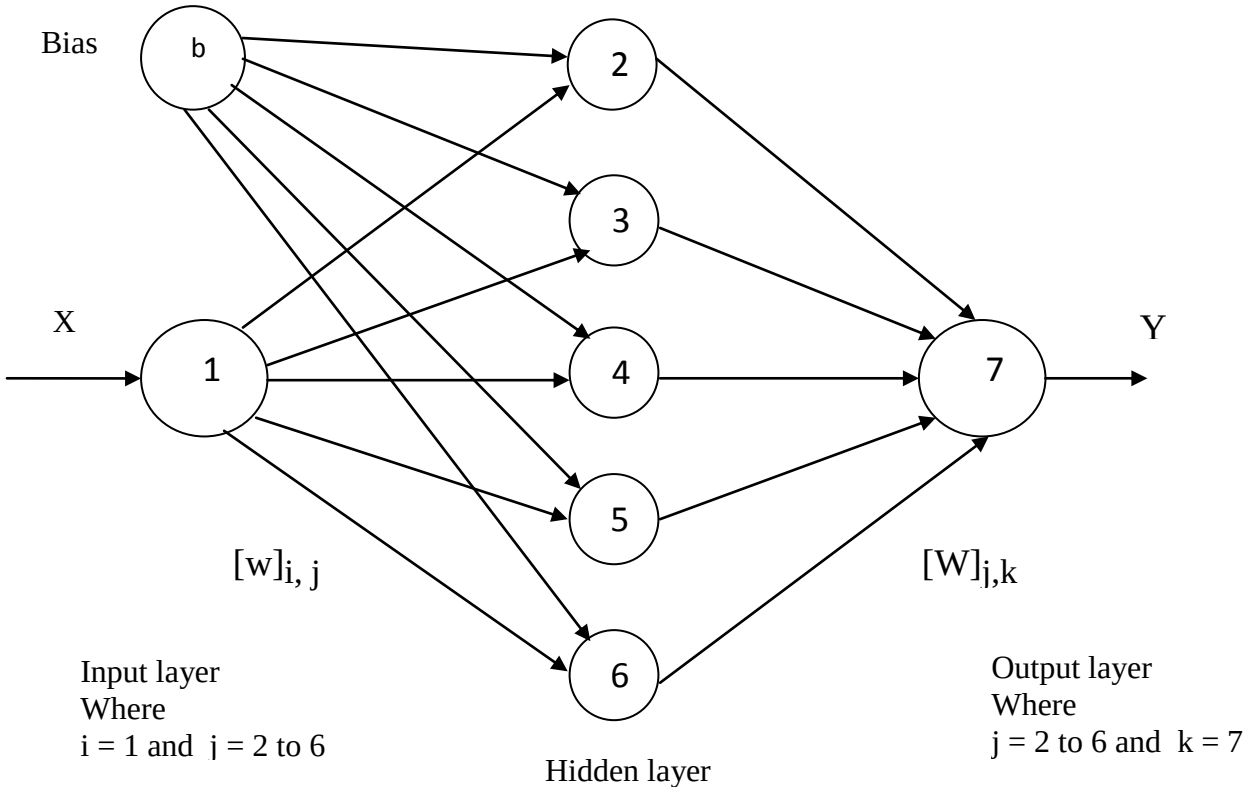
Data sheet: y = AsinX (A=2) ($-\pi \leq X \leq \pi$)

X	A sinX	Actual Values
$\frac{\pi}{6}$	0.5*2	1
$\frac{\pi}{4}$	0.7071*2	1.4142
$\frac{\pi}{3}$	0.866*2	1.732
$\frac{\pi}{2}$	1*2	2

π	0*2	2
$\frac{-\pi}{2}$	-1*2	-2

$\frac{-\pi}{4}$	-0.7071*2	-1.414
$\frac{-\pi}{6}$	-0.5*2	-1
$\frac{\pi}{10}$	0.3090*2	0.6180
$\frac{\pi}{8}$	0.3826*2	0.7653
$\frac{\pi}{7}$	0.4335*2	0.86776
$\frac{\pi}{12}$	0.2588*2	0.5176

MATHEMATICAL FUNCTION AND FORMULATION



Step1: Now calculating the input and output values at the input layer

By figure. $I_{\text{input layer i}} = x$ 1

$O_{\text{input layer i}} = x = I_{\text{input layer i}}$ 2

Step 2: By using sigmoid function calculating the input and output values at the hidden layer

$$I_{Hj} = X_1W_1 + X_2W_2 + X_3W_3 + X_4W_4 + X_{n-1}W_{n-1} + X_nW_n$$

$$= O_{\text{input i}} W_{i,j}$$

$I_{Hj} = I_{\text{input layer i}} W_{i,j}$ by the help of equation 2

Output of hidden layer neurons

$O_{Hj} = 1/1+e^{-\lambda I_{hj}}$

Assume that λ=1

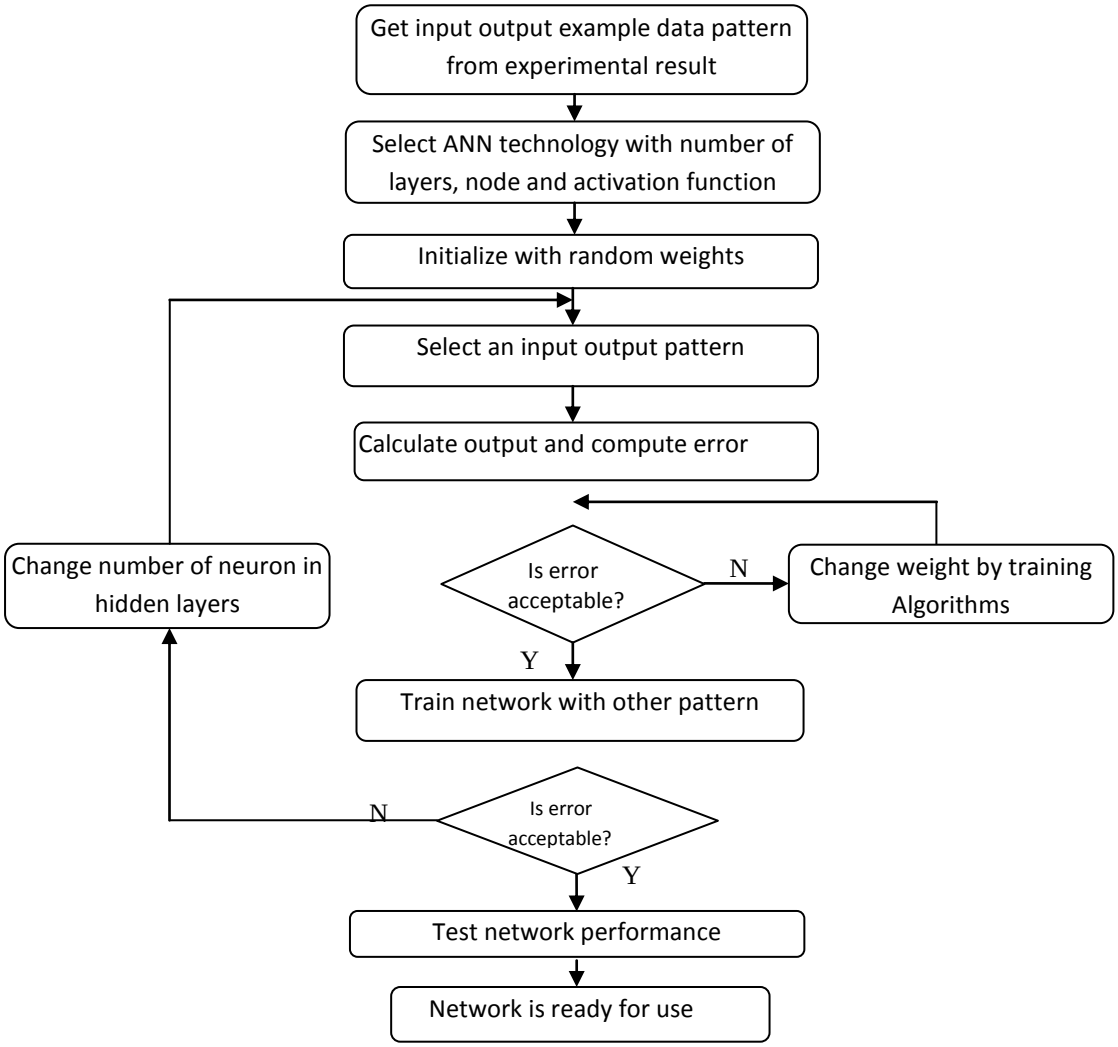
Step 3: Now calculating the input and output values at the output layer

Input of the output layer

$I_{ok} = \sum O_{Hj} W_{j,k}$ where j = 2 to 6 and k = 7

$O_{OK} = 1/1+e^{-\lambda I_{ok}}$ for K=1 to n (no. of neuron in the output layer**)**

FLOW CHART FOR BACK PROPAGATION TRAINING OF FEED FORWARD NEURAL NETWORK



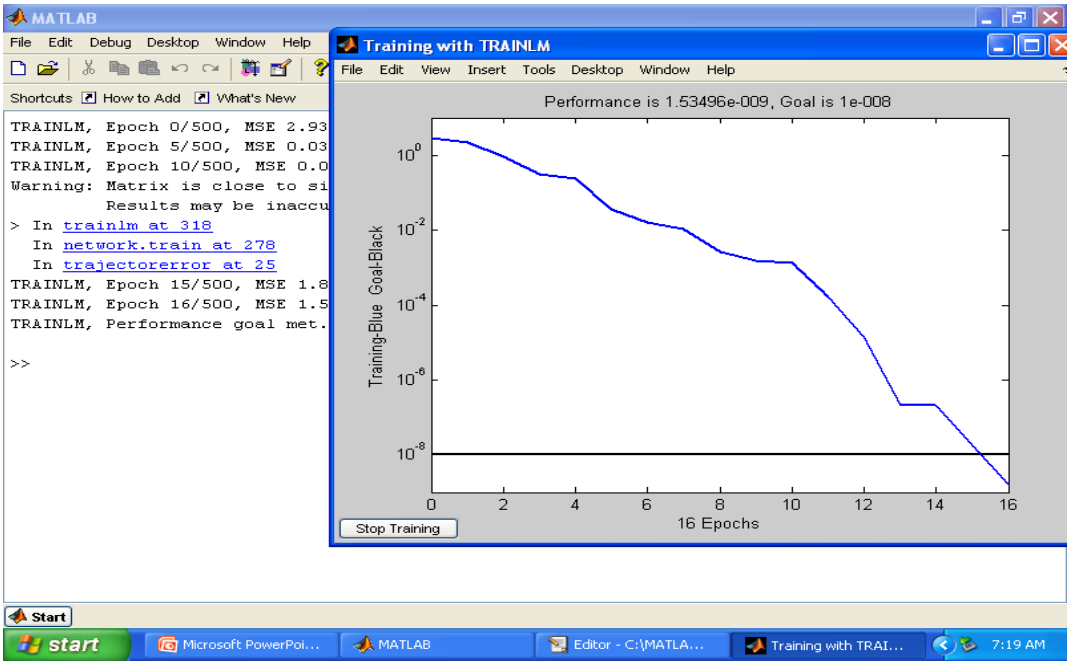
TESTING

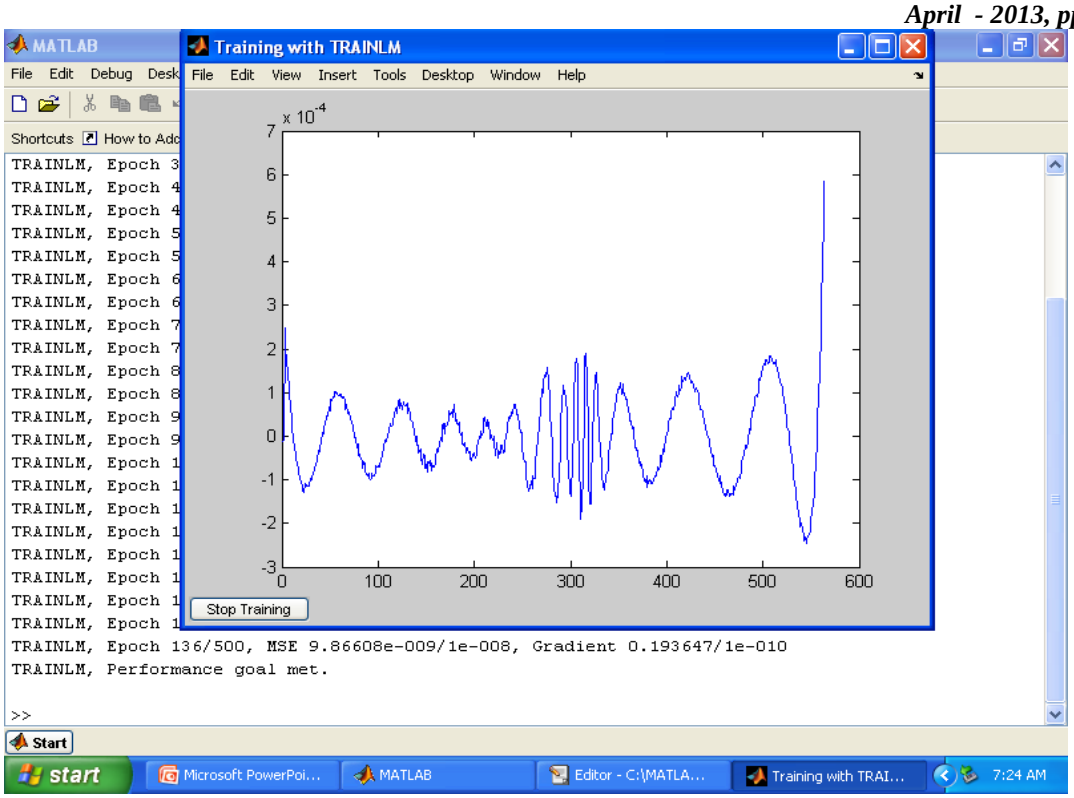
After the successful training of the system we start the testing of the system by the help of input function.

Number of input neuron	1
Number of output neuron	1

Number of hidden layers	1
Number of hidden neuron	4
Number of pattern	12
Max epoch	500
Learning step	2
Error	10^{-8}

RESULT





Conclusion

The connectionist models presented in the paper have been designed in order to obtain an efficient tool for the identification of complex systems, where the dynamic process may be partially or completely unknown by the help of this paper we clearly show that application of neural network for model identification. In this paper we find that the given model is sine curve with two times at the range part.. The maximum error (10^{-7}) will accept.

References

[1] J. Sjöberg, H. Hjalmerson, L. Ljung (1994):"Neural Networks in System Identification", Preprints10th IFAC symposium on SYSID, Copenhagen, Denmark. Vol.2, pp. 49-71, 1994.

[2] K.S. Narendra and K.parthasarathy, ―Back propagation and dynamical systems. Part II: identification,‖ Centre Syst. Sci., CT, tch, rep. 8902, Feb 2006.

[3] Soft Computing Techniques and applications, D K Chaturvedi, Dayalbagh University, Agra.Springer

[4] H. Nguyen and B.Widrow, ―Neural networks for self-learning control systems,‖ Int. J. Contr., vol. 54, no. 6, pp. 1439–1451, 2001.

[5] Practical Knowledge of Fuzzy Techniques and Neural Network Tecniques from Vimal Kumar Bose,University of Tnnesses Knoxville Mordern Power Electronics and AC Drives,PHI

[6] H. Demuth & M. Beale: "Neural Network Toolbox", The MathWorks Inc., 2003.

[7] Werbos P.J. Backpropagation Through Time: What it Does and How to Do it. Proceedings of the IEEE. Vol. 78 N. 10 pp. 1550-1560 October 2005.

[8] Yazdizadeh, A. and Khorasani, K.: Adaptive time delay neural network structures for nonlinear system identification, Neurocomputing, Vol. 47,No. 1-4, pp. 207-240, 2002.

[9] Rumelhart, D.E., Hinton, G.E. and Williams, R.J.Learning representations by back-propagating errors,Nature, Vol. 323, No. 6088, pp. 533-536, 1986.

