ON THE SUNSPOT TIME SERIES PREDICTION USING JORDON ELMAN ARTIFICIAL NEURAL NETWORK (ANN)

Rohit R. Deshpande and Athar Ravish Khan
Department of Electronics and telecommunication Engg.,
Jawaharlal Darda Institute of Engg. & Tech Yavatmal Maharashtra State, India.

ABSTRACT

In this paper, multi step ahead prediction of monthly sunspot real time series are carried out. This series is highly chaotic in nature [7]. This paper compares performance of proposed Jordan Elman Neural Network with TLRNN (Time lag recurrent neural network), and RNN (Recurrent neural network) for multi-step ahead (1, 6, 12, 18, 24) predictions. It is seen that the proposed neural network model clearly outperforms TLRNN(Time lag recurrent neural network), and RNN(Recurrent neural network) in various performance measures such as MSE (Mean square error), NMSE (Normalized mean square error) and r (correlation coefficient) on testing as well as training data set for 1,6,12,18,and 24 months ahead prediction of sunspot time series. Parameters are calculated by using software, "Neurosolution 5.0". Neurosolution is an object oriented environment for designing, prototyping, simulating, and deploying artificial neural network (ANN) solutions [26].

KEYWORDS: *MSE (Mean square error), NMSE (Normalized mean square error), r (Correlation coefficient), k (Number of step ahead)*

I. INTRODUCTION

The main motivation for analysis and research of Sunspot time series is to predict the future and understand the fundamental feature and effects on nature and human life. Artificial neural network is used for solving the real world problem [6]. This is mainly due to their ability to deal with nonlinearities, non-stationary and non guassianity [24]. The modelling and analysis of chaotic time series has also attracted the attention of many researchers [4]. In this paper, Jordan Elman NN is found an optimal NN as compared to TLRNN & RNN for Sunspot time series prediction. Jordan Elman Neural Network is trained for multi step ahead prediction and the results are compared with reference to the MSE (mean square error), NMSE (normalized mean square error), and r (correlation coefficient), on testing as well as training data sets for short prediction. The number of experiments is carried out by changing various parameters like Error criterion (L1, L2, L3, L4, L5, and L∞), number of iterations, learning rule, percentage of training & testing data sets and transfer functions [1]. Thus the optimum neural network model is proposed for short time prediction of Sunspot time series. Previously, some work has been done on the prediction of sunspot time series. It has been shown that the Fully Recurrent Neural Network (FRNN) model gives the good results for prediction [1].In this paper; we have compared the various parameters for prediction and proposed the Jordon Elman neural network, which gives perhaps the better results. The remainder of the paper is organized as follows. In the next section we present the effective use of Artificial Neural Network for prediction. Section 3 deals with the introduction of proposed Jordon Elman neural network. Section 4 deals with the various performance measures on the basis of which the prediction has been carried out. Section 5 deals with the introduction of Neurosolution, which is used as a tool for the analysis. The graph representing calculation of number of processing elements, parameters obtained for various networks in tabular form, the relation between actual and desired output along with the scatter plots for the variation of number of epochs has been shown in section 6.The paper ends with the comparison tables showing results for various networks and conclusion.

II. NEURAL NETWORKS FOR PREDICTION

The beginning of the prediction of the time series was made in the past century with the introduction of the autoregressive model for the prediction of the annual number of sunspots. Time series are samples of system's behaviour over discrete time values. The neural networks ability to cope up with the nonlinearities, the speed of computation, the learning capacity and the accuracy, made them valuable tools of prediction. To predict the evolution over time of a process implies the prediction of the future values of time series describing the process [1], [4].Time series prediction with classical methods relies on some steps to be followed in order to perform an analysis of the time series, including modelling, the identification of the model and finally parameter estimation. The most difficult systems to predict are a) those with insufficient amount of data in the data series (for instance chaotic signals); b) the systems whose data series are obtained through observation and measurements, in this case data being possibly affected by measurement errors; c) systems whose behaviour varies in time[7].The artificial neuron for prediction receives the delayed signal at its input y(k-i), i=the number of inputs, the inputs are transmitted through i+1 multipliers for scaling, then the scaled inputs are linearly combined to form an output signal that is also passed through an activation function in order to obtain an output signal. The model of the neuron is schematized as shown in figure 1, [2].

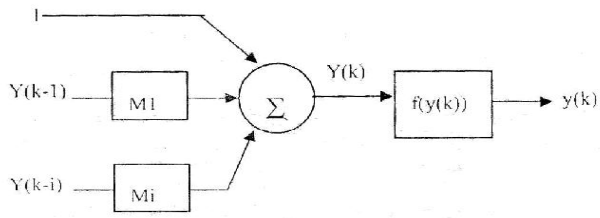


Figure1. The artificial neuron for prediction (M1….i are multipliers)

Consider x (t), the time series data collected in order to build a model. The data in the data series are samples of x(t) with a time step k. the prediction of the time series values can be made single step ahead, in this case one looking for a good estimation X1(t+1) of X(t+1) or multi step ahead prediction, in this case looking for a good estimate X1(t+nk) of X(t+nk), n being the number of steps ahead[9].The first and most common method for the prediction of time series consist in using M past values or M-tuples as inputs and one output.

III. JORDAN ELMAN NN

The theory of neural networks with context units can be analyzed mathematically only for the case of linear PEs. In this case the context unit is nothing but a very simple low pass filter. A low pass filter creates an output that is a weighted (average) value of some of its more recent past inputs. In the case of the Jordan context unit, the output is obtained by summing the past values multiplied by the scalar as shown in the figure 2.Notice that an impulse event x (n) (i.e. x (0) =1, x (n) =0 for n>0) that appears at Time n=0, will disappear at n=1. However, the output of the context unit is t1 at n=1, t2 at n=2, etc. This is the reason these context units are called memory units, because they "remember" past events. t should be less than 1, otherwise the context unit response gets progressively larger (unstable).The Jordan network and the Elman network combine past values of the context units with the present inputs to obtain the present net output. The input to the context unit is copied from the network layer, but the outputs of the context unit are incorporated in the net through adaptive weights [9], [24]. NeuroSolutions uses straight back propagation to adapt all the network weights [2].

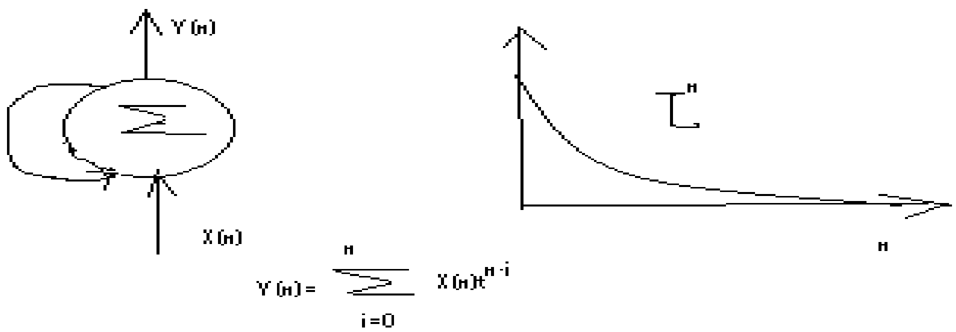


Figure 2. Context unit response

In the Neural Builder, the context unit time constant is pre-selected by the user. One issue in these nets is that the weighting over time is kind of inflexible since we can only control the time constant (i.e. the exponential decay). Moreover, a small change in t is reflected in a large change in the weighting (due to the exponential relationship between time constant and amplitude). In general, we do not know how large the memory depth should be, so this makes the choice of t problematic, without a mechanism to adapt it. See time lagged recurrent nets for alternative neural models that have adaptive memory depth. The Neural Wizard provides four choices for the source of the feedback to the context units (the input, the 1st hidden layer, the 2nd hidden layer, or the output). In linear systems the use of the past of the input signal creates what is called the moving average (MA) models. They represent well signals that have a spectrum with sharp valleys and broad peaks. The use of the past of the output creates what is called the autoregressive (AR) models. These models represent well signals that have broad valleys and sharp spectral peaks. In the case of nonlinear systems, such as neural nets, these two topologies become nonlinear (NMA and NAR respectively). The Jordan net is a restricted case of an NAR model, while the configuration with context units fed by the input layer is a restricted case of NMA. Elman's net does not have a counterpart in linear system theory. As you probably could gather from this simple discussion, the supported topologies have different processing power, but the question of which one performs best for a given problem is left to experimentation[3],[7].

IV. PERFORMANCE MEASURES

The generalization performance of the network is validated on the basis of the following parameter.

4.1. MSE (Mean Square Error)

It is the average of the square of the difference between each output processing element and the desired output. It is used to determine how well the network output fits the desired output, but it doesn't reflect whether two sets of the data move in same direction [1].

Yij = network output for exemplar i at PE j
P = number of output PEs(processing elements)
N = number of exemplar in datasheet
Dij = desired output for exemplar I at PEj

4.2. NMSE (Normalized Mean square error)

$$NMSE = \frac{p.n.MSE}{\sum_{j=0}^{p}\left[\frac{N\sum_{i=0}^{N} dij^2 - \left(\sum_{i=0}^{N} dij\right)^2}{N}\right]} \quad (1)$$

NMSE is given as -

P = number of output PEs(processing elements)
N = number of exemplar in datasheet
dij = desired output for exemplar I at PEj

4.3 Correlation coefficient (r)

It reflects whether the two sets of data moves in same direction .The correlation coefficient are between the network output X and the desired output D is

$$r = \frac{\sum_i (xi - \bar{x})(di - \bar{d})}{\sqrt{\sum_i \frac{(di - \bar{d})}{N}}\sqrt{\sum_i \frac{(xi - \bar{x})^2}{N}}} \tag{2}$$

Where,

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} xi \quad \text{and} \quad \bar{\mathrm{d}} = \frac{1}{N}\sum_{i=1}^{N} di \tag{3}$$

The correlation coefficient is confined to the range [-1, 1].When r =1, there is perfect positive linear correlation between X & d that is they co vary means they vary by the same amount [T. Edwards, D.S.W Tansley, R.J. Frank, N. Davey .etal], [Lapedes A. and Farber R.etal] .The above parameters are calculated by using software, “Neurosolution 5.0”

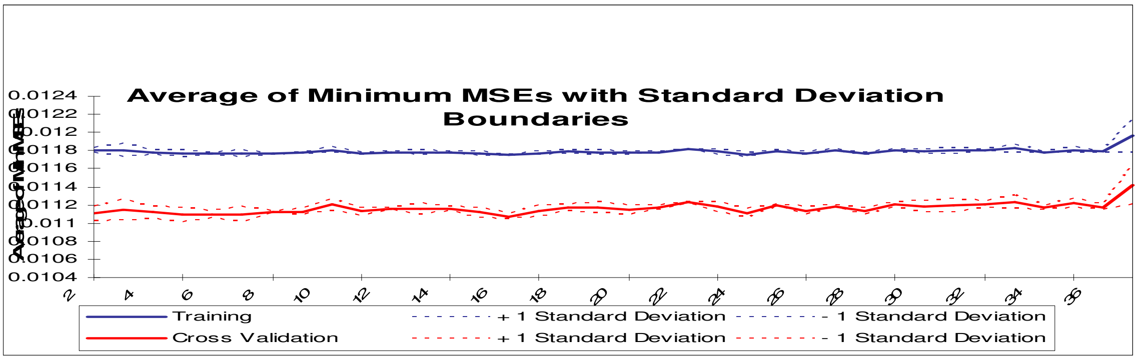
V. NEUROSOLUTION 5.0

Neurosolution is an object oriented environment for designing, prototyping, simulating, and deploying artificial neural network (ANN) solutions. This environment is inherently modular, as a neural network breaks down into a fundamental set of neural components. Individually simple, these components combine to form networks that can solve complex problems. Neurosolution supports a practically infinite number of neural models. It provides the user with an icon-based interface from which the neural components are arbitrarily interconnectable [2].

VI. CASE STUDY

The different neural network models like Jordan Elman , time lag recurrent network and recurrent neural network are trained for multi step ahead [1,6,12,18,24] predictions and the results are compared with reference to MSE (Mean square error), NMSE (Normalized Mean Square Error), r (Correlation coefficient) on testing as well as training data set for short term prediction. The number of experiments is carried out by changing various parameters like number of processing elements, number of hidden layers, number of iterations, transfer function, learning rule. Here we are finding out the number of processing elements for which the network gives minimum MSE.

Figure 3.shows the graph obtained by varying various parameters giving the number of processing elements for which the network gives minimum MSE. The Following Tables gives values of various parameters such as MSE (Mean square error), NMSE (Normalized Mean Square Error), and r (Correlation coefficient) for different look ahead values for different Neural Networks. Table 1 gives values of MSE, NMSE, and r of Jordan Elman NN for multi step ahead (1, 6, 12, 18, 24) prediction for Sunspot time series.



Best Networks	Training	Cross Validation
Hidden 1 PEs	15	5
Run #	3	2
Epoch #	1000	1000
Minimum MSE	0.011739974	0.011022748
Final MSE	0.011739974	0.011022748

Figure 3. Graph and Table of varying a parameter

Similarly table 2 and table 3 shows various values of MSE, NMSE, and r for Time Lagged Recurrent Neural Network and Recurrent Neural Network. Comparing the performance measures like MSE, NMSE and r for 60% samples as training, 25%samples as testing and 15% as cross validation, it is found that the performance of the selected model is optimal for 15 neurons in the hidden layer. Next the proposed model is trained with different error criterion L1, L2, L3, L4, L5 and L∞.

Table1. Type of network-Jordan Elman NN

	For South			For North		
K (step)	**MSE**	**NMSE**	**R**	**MSE**	**NMSE**	**r**
1	**0.00301352**	**0.12175413**	**0.9372644**	**0.00039479**	**0.01198851**	**0.99412255**
6	**0.00315088**	**0.12097912**	**0.9384633**	**0.00131084**	**0.00339036**	**0.98179276**
12	**0.01120112**	**0.23450117**	**0.8901836**	**0.01142852**	**0.16956471**	**0.93439364**
18	**0.00992762**	**0.43738162**	**0.8810052**	**0.01900812**	**0.26451196**	**0.91967542**
24	**0.00969921**	**0.47540716**	**0.8735284**	**0.01007496**	**0.27224255**	**0.90736715**

Table 2. Type of Network Time Lagged Recurrent Neural Network

	For South			For North		
K (step)	**MSE**	**NMSE**	**R**	**MSE**	**NMSE**	**r**
1	**0.0031452**	**0.1338280**	**0.930992**	**0.001002**	**0.030435**	**0.9832722**
6	**0.0038522**	**0.1372878**	**0.927177**	**0.001627**	**0.004266**	**0.9726262**
12	**0.0116022**	**0.2429242**	**0.882282**	**0.011626**	**0.164626**	**0.9302727**
18	**0.0165966**	**0.3459452**	**0.840728**	**0.021829**	**0.322258**	**0.8775592**
24	**0.0997141**	**0.3538383**	**0.813266**	**0.031833**	**0.363333**	**0.8610236**

Table 3. Type of Network Recurrent Neural Network

	For South			For North		
K(step)	**MSE**	**NMSE**	**R**	**MSE**	**NMSE**	**r**
1	**0.0023245**	**0.1339322**	**0.951992**	**0.004242**	**0.014249**	**0.9812222**
6	**0.0390393**	**0.1463663**	**0.927135**	**0.001634**	**0.042527**	**0.9788397**
12	**0.0143535**	**0.2967211**	**0.884222**	**0.014892**	**0.214893**	**0.9236422**
18	**0.0178532**	**0.3721266**	**0.819626**	**0.021780**	**0.321859**	**0.8610656**
24	**0.0128535**	**0.4238543**	**0.777512**	**0.013118**	**0.302349**	**0.8423390**

The best combination network is then trained and tested for different transfer functions such as a) Tanh axon b) Sigmoid c) Linear Tanh axon d) Linear Sigmoid. The proposed Jordan Elman NN model is trained for the best combinations resulted for training and testing exemplars and it is experimented for 1000 to 20000 iterations for getting an optimum results for each multi step ahead (k=1, 6, 12, 18, 24) of chaotic sunspot time series **.**The number of epochs are varied from 1000 to 20000 in the step 2000 and the graphs are plotted for all the steps. These Graphs are shown in Figure 4 to Figure 8.

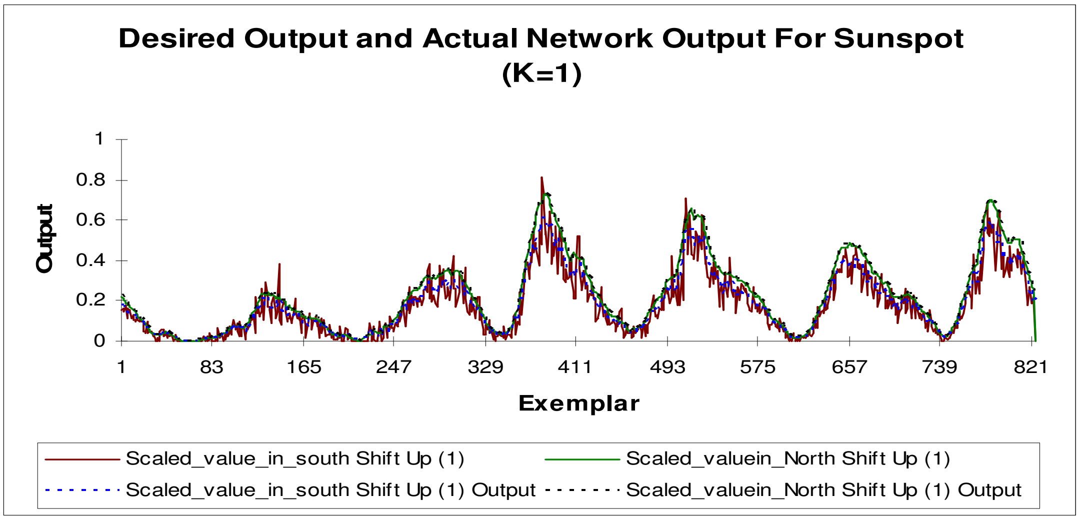


Figure 4. Graph For (k=1)

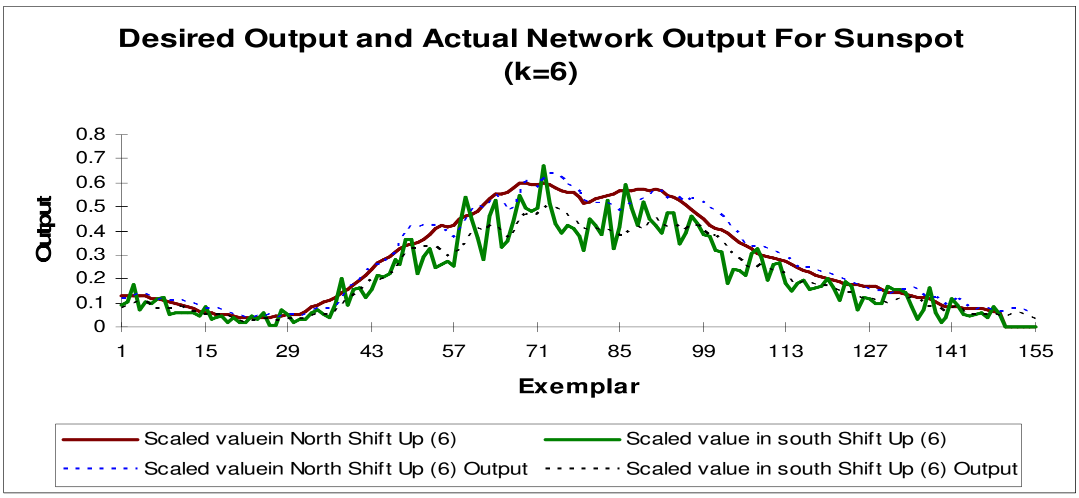


Figure 5. Graph For (k=6)

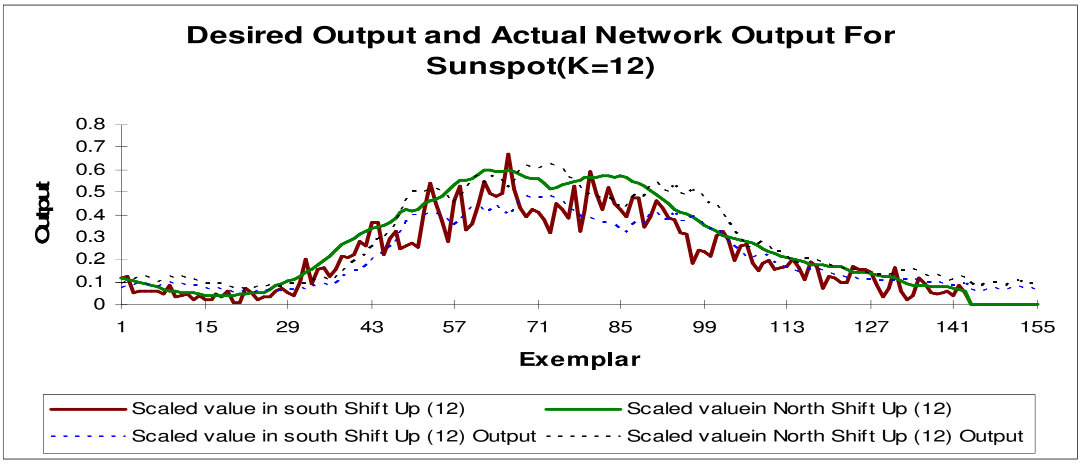


Figure 6. Graph For (K=12)

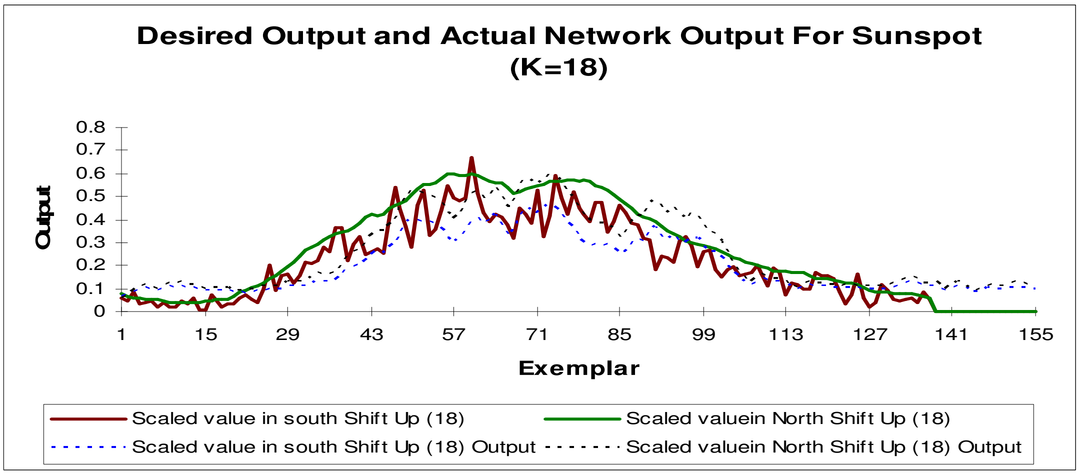


Figure 7. Graph For (K=18)

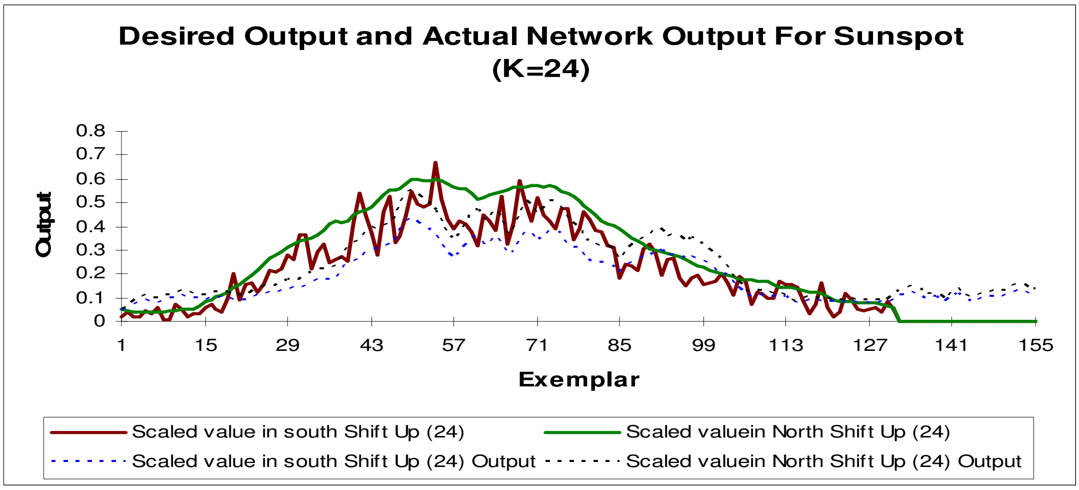


Figure 8. Graph For (K=24)

Then, we have varied number of epochs means number of iterations. Epochs are varied from 1k to 20 k (1000 to 20000).By analyzing its results we have plotted the scatter plots of number of epochs and variation of MSE, NMSE, & r.

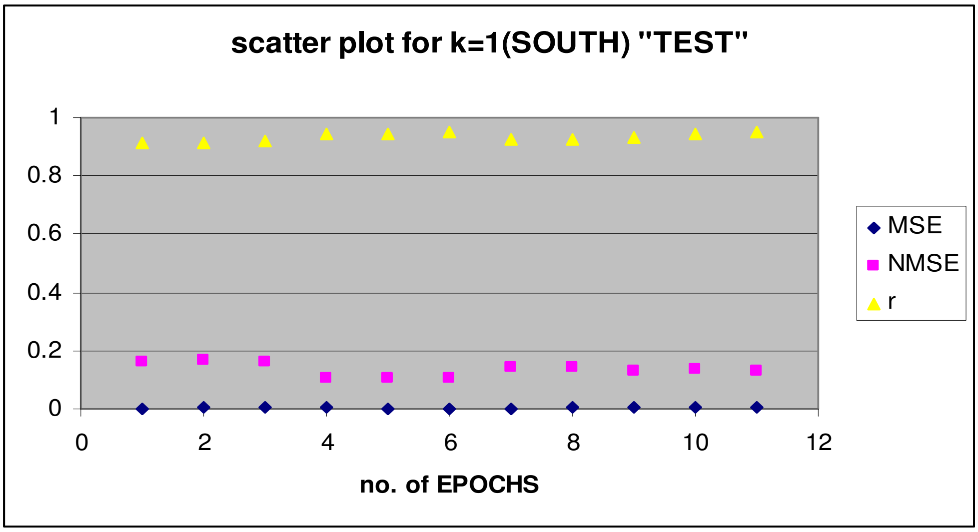


Figure 9. Scatter Plot For K=1

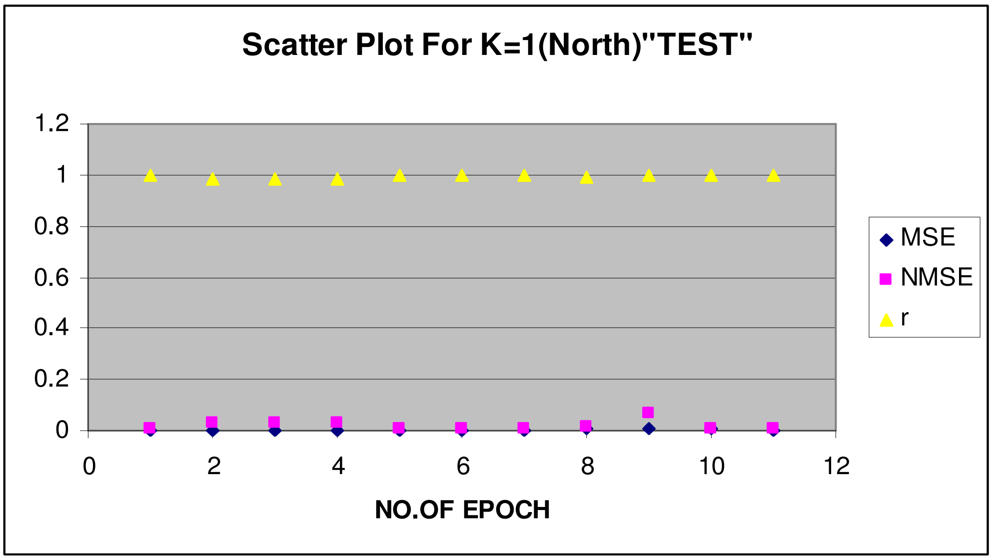


Figure 10. Scatter Plot For K=1

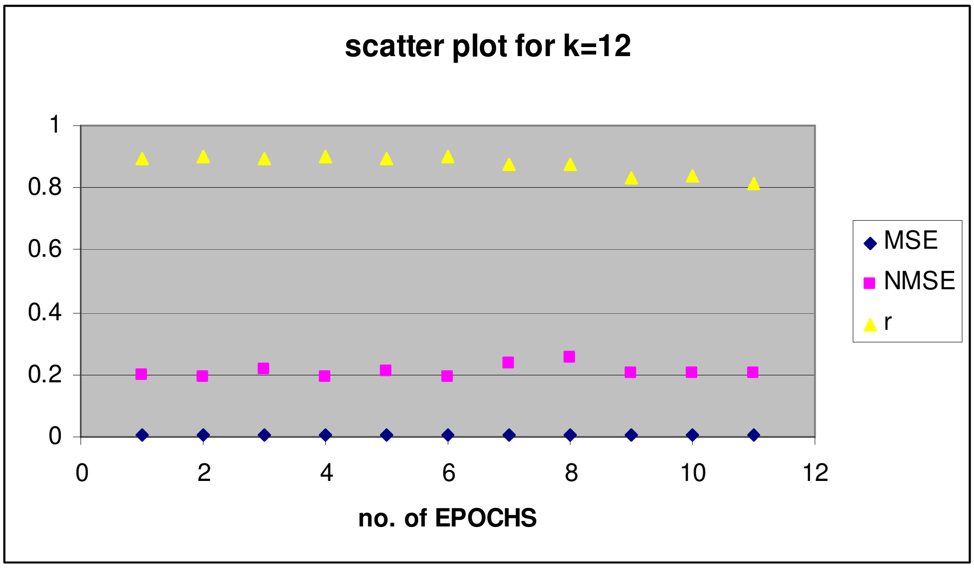


Figure 11. Scatter Plot For K=12

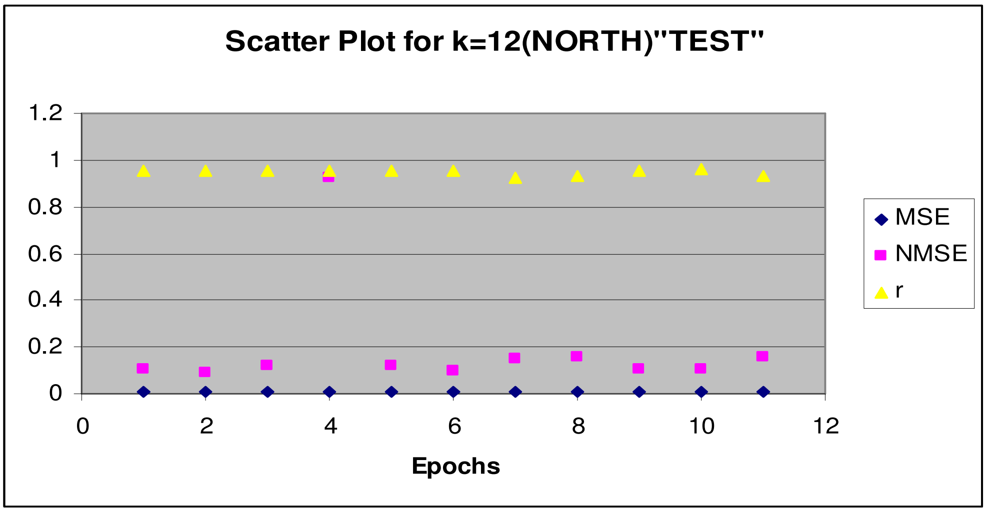


Figure 12. Scatter Plot For K=12

VII. Results

Comparison Tables of JENN, TLRNN, FRNN For chaotic time series (k= 1 & k=6)

Table 4. For k=1

S. No.	NN MODEL	SOUTH			NORTH		
		R	MSE	NMSE	r	MSE	NMSE
1	JORD-ELMN	0.9398	0.0028	0.1170	0.9931	0.0004	0.0138
2	TLRNN	0.9394	0.0029	0.1181	0.9853	0.0010	0.0304
3	FRNN	0.9155	0.0040	0.1618	0.9812	0.0012	0.0372

Table 5. For k=6

S. No.	NN MODEL	SOUTH			NORTH		
		r	MSE	NMSE	r	MSE	NMSE
1	JORD-ELMN	0.9384	0.0031	0.1209	0.9817	0.0013	0.0339
2	TLRNN	0.9292	0.0038	0.1398	0.9809	0.0014	0.0380
3	FRNN	0.9271	0.0390	0.1461	0.9788	0.0016	0.0372

Tables 4 and 5 shows comparison between Jordn-Elman NN, TLRNN, and FRNN for shift(k=1,6).Thus here we have compared various NN on the basis of MSE, NMSE, and r, and an optimal NN for prediction of Sunspots time series is found.

VIII. Conclusions

It is observed that Jordan Elman Network is able to predict Sunspots time series quite well in comparison with other networks. It is seen that MSE,NMSE & r of the proposed dynamic model for testing data set as well as training data set are significant than other neural models. The network is analyzed for different step ahead (k=1, 6, 12, 18, 24).The better results can be obtained for maximum

24 step ahead by using this network. The following parameters are obtained for 24 step ahead; MSE=0.009457611, NMSE=0.309630953, r=0.87684609 (for South), MSE=0.010051769, NMSE=0.281693745, r=0.908057335 (for North) For Epoch=16000.Here, we can see that the value of correlation coefficient is much closer to unity which makes the proposed network optimal one for prediction. In this paper, we have also studied the variation of error and correlation coefficient with the number of epochs. Also, the variation of actual and desired output can also be studied for different step ahead values (1, 6, 12, 18, and 24) by plotting the graphs. It can be seen that the proposed neural network closely follows the actual output.

REFERENCES

[1] Sanjay L. Badjate, Sanjay M.Gulhane *"Multi-Step Ahead Prediction Of Chaotic Time Series Using Recurrent Neural Network Models "*. International Journal Of Computer Science And Applications Vol. 4, No. 1, April / May 2011 ISSN: 0974-1003

[2] Alluada Neurosolution 5.0.

[3] J.Hertz, A. Krogh, R.G. Palmer. *"Introduction to theory of neural computation"*, vol-1-Addison-Hesley 1991.

[4] William Ditto, Toshinori Munakata *"Principles and applications of chaotic system"*.

[5] Sorin Vlad *"On the prediction methods using neural networks"*.

[6] Tuevo kohonen *"self organizing feature map"*, third edition , Springer 2001,p.28

[7] K.T.Alligood, T.D.Sauer,J.A Yorke, *"CHAOS-An introduction to neural network"*

[8] D.Hammerstram, *"Working with neural network*", IEEE spectrum, 1993, pp46-53.

[9] J.M. Zurada *"Introduction to Neural Systems"*, Jaico Publication, 2004.

[10] Yachanan shachmarove: *"Applying neural network to business, econmics*

[11] Haykin S. Adaptive Filter Theory. Prentice-Hall, Englewood Cliffs, 1991.

[12] Haykin S. Neural Networks, a Comprehensive Foundation. MacMillan, 1994.

[13] Hebb D. The Organization of Behavior. Wiley, New York, 1949.

[14] Hecht-Nielsen R. NeuroComputing. Addison Wesley, 1990.

[15] Hertz J., Krogh A. and Palmer R. Introduction to the Theory of Neural Networks. Addison Wesley, 1991.

[16] Hopfield J. "Neural networks and physical systems with emergent collective computational abilities." Proc. Natl. Acad. Sci. (USA) 79, 2554-2558, 1982.

[17] Jacobs R. "Increased rates of convergence through learning rate adaptation." Neural Networks 1, 295-307, 1988.

[18] Jordan M. "Attractor dynamics and parallelism in a connectionist sequential machine." Proc.8th Annual Conf. on Cognition Science Society, pp 531-546, 1986.

[19] Kohonen T. Self Organization and Associative Memory. Springer Verlag, 1988.

[20] Kohonen T. "The self-organizing map." Proc. IEEE 78, 1464-1480, 1990.

[21] Kosko B. Neural Networks and Fuzzy Systems. Prentice Hall, 1992.

[22] Kung S.Y. Digital Neural Networks. Prentice Hall, 1993.

[23] Lang K., Waibel A. and Hinton G. "A time delay neural network architecture for isolated word recognition." Neural Networks 3(1), 23-44, 1990.

[24] Lapedes A. and Farber R. "Nonlinear signal processing using neural networks: prediction, and system modelling." LA-VR-87-2662, Los Alamos, 1987.

[25] LeCun Y., Denker J., Henderson D., Howard R., Hubbard W. and Jackel L. "Handwritten digit recognition with a back propagation network." In Advances in Neural Information Processing Systems 2, (ed. Touretsky), pp 396-404, 1990.

[26] Lefebvre C. and Principe J. "Object-oriented artificial neural network implementations." World Cong. Neural Networks 4, 436-439, 1993.

[27] Lefebvre C. An Object-Oriented Methodology for the Analysis of Artificial Neural Networks. Master's Thesis, University of Florida, 1992.

[28] Lippman R. "An introduction to computing with neural nets." IEEE Trans. ASSP Magazine 4,4-2, 1987.

[29] Little W. and Shaw G. "Analytical study of the memory storage capacity of a neural network."Mathematical Biosciences 39, 281-290, 1978.

[30] Makhoul J. "Pattern recognition properties of neural networks." Proc. 1991 IEEE Workshop on Neural Networks for Signal Processing, pp 173-187, 1992.

[31] Marr D. "A theory of cerebellar cortex." Journal of Physiology 202, 437-470, 1969.

[32] McCulloch W. and Pitts W. "A logical calculus of the ideas imminent in the nervous activity."Bulletin of Mathematical Biophysics 5, 115-133, 1943.

[33] Minsky M. and Papert S. Perceptrons. MIT Press, 1969.

[34] NeXTStep Operating System, NeXT Computer Documentation, 1991.

Authors

Rohit Ravindra Deshpande was born in Maharashtra, INDIA, in 1986.He received the B.E. degree in electronics and telecommunication engineering, from SGBA University Amravati Maharashtra INDIA, in 2008 and Pursuing M.Tech degree in Electronics Engg from RTMN University Nagpur. In 2008, he joined J.D.I.E.T. Yavatmal Maharashtra INDIA and worked as a lecturer. In June 2011 he became an Assistant Professor there.His current research interests include Artificial Neural Networks and time series prediction.

Athar Ravish Khan was born in Maharashtra, INDIA, in 1979.He received the B.E. degree in electronics and telecommunication engineering, M.E. degree in digital electronics from SGBA University Amravati Maharashtra INDIA, in 2000 and 2011 respectively. In 2000, he joined B.N College of Engineering Pusad and worked as lecturer. In 2006 he joined as lecturer in J.D Institute of Engineering and Technology Yavatmal, Maharashtra INDIA and in March 2011 he became an honorary Assistant Professor there. He is pursuing Ph.D. degree under the supervision of Prof.Dr.Sanjay.M.Gulhane. His current research interests include digital signal processing, neural networks and wireless communications, with specific emphasis on UWB in underground Mines-Tunnels