International Journal of Research and Development in Applied Science and Engineering (IJRDASE)
ISSN: 2454-6844

ANN Application in Data Mining for Time Series Forecasting

Ankita Agarwal
Computer Science & Engg. Dept
BBDNITM, Lucknow (U.P.)
agarwalankita.lko@gmail.com

Pooja Khulbe
Computer Science & Engg. Dept
BBDNITM, Lucknow (U.P.)
pooja.khulbe29@gmail.com

Rohit Agarwal
Computer Science & Engg. Dept
B.I.E.T., Lucknow (U.P.)
rohitagarwal202@gmail.com

Abstract-- Time series information analysis for forecasting is one of the most vital factors for the realistic usage, showing the way in which the past impacts the destiny. In this paper, Artificial Neural Network (ANN) approach has been used to broaden one month and two month in advance forecasting fashions using monthly rainfall statistics of Northern India (1871-2014), in which Feed Forward Neural Network (FFNN) the usage of Back Propagation set of rules and Levenberg-Marquardt training characteristic has been used. The work has been completed using Neural Network Toolbox in MATLAB. The overall performance of both the fashions have been assessed primarily based on Regression evaluation, Mean Square Error (MSE) and Magnitude of Relative Error (MRE). Although the ANN model showed advantageous effects for both the fashions for forecasting, however still it become able to perform higher for one month ahead as opposed to months in advance forecasting.

Keywords: ***Time series data analysis, ANN, FFNN, MSE, MRE, Regression, MATLAB***

1. Introduction:

Time series forecasting (TSF),the forecast of a time ordered variable, it turns on into a decisive tool in problem solving, since it allow one to model complex system where the goal is to predict the system's behavior and not how the system works. In the last few decades an increasing focus has been put over this field. Contributions from the arenas of statistics, operational research, and computer science has lead to TSF methods like exponential smoothing or regression that can replace the old fashioned ones, those were primarily based on intuition. In the present paper an attempt has been made to develop forecasting models using Artificial Neural Network (ANN), a soft computing technique.

2. Related Work:

ANNs are used for solving many types of nonlinear problems that are difficult to solve by traditional techniques. Often, time series processes exhibit temporal and spatial variability, and are suffered by issues of nonlinearity of physical processes, conflicting spatial and temporal scale and uncertainty in parameter estimates. ANNs have been able to tackle theses problems effectively. A lot of work has been done in this direction. Hu(1964) was the first to use the implementation of ANN, a very important soft computing technique in weather prediction. French et al. (1992), took a leading work in applying ANN for rainfall prediction, that applied a neural network approach to forecast two-dimensional rainfall. Abraham et al. (2001) used an ANN with scaled conjugate gradient algorithmic rule (ANN-SCGA) and evolving fuzzy neural network (EfuNN) for predicting the rainfall time series. Pucheta Julian A, et. al , (2010), obtained a feed-forward NN based NAR model for forecasting time series. R. Adhikari and R.K.Agarwal, (2012), have demonstrated the effectiveness of artificial neural network in recognizing and forecasting strong seasonal patterns without removing them from the raw data. Seyed Taghi Akhavan Niaki and Saeid Hoseinzade, 2013, demonstrated that the main aim of their research is to predict the daily direction of Standard & Poor's 500 (S&P 500) index using an artificial neural network (ANN).

3. Data Used:

For the present model development monthly time series rainfall data of the whole of India for the period 1871 to 2012 (141 yrs.) has been used. The data has been obtained through the internet from Indian Meteorological Department, Pune site.

The data used as input and output variables for optimum model development are given in the table below. Here in the present work two models shall be developed, model M1 is for one month ahead prediciton and the other model M2 is for two month ahead prediction. In both M1 and M2 models three input variables have been used which include successive months of raifall data. The output in both the models is the predicted rainfall, which is one month and two month ahead values.

Table 1 :- Model Input / Output Variables

Model	Input Variables Rainfall (mm)	Output Variables Rainfall (mm)
M1	R(t-2)	R(t+1)
	R(t-1)	
	R(t)	
M2	R(t-2)	R(t+2)
	R(t-1)	
	R(t)	

4. ANN Model Development:

NN Model is developed using Matlab graphical user interface (GUI). Trials are first conducted by randomly selecting

number of processing elements. Judgment of the accuracy of prediction is done on the basis of the mean square error at the end of the training. The range of selection of the processing elements was narrowed down carefully, on the basis of result of performance for prediction. Among the range of 5 to 40 neurons, NN model was observed to perform with very good accuracy for prediction. NN models are created with one hidden layer and varying number of processing elements or neurons.

Optimal network geometry was found out, using trial and error approach, in an attempt to create more optimum model. The training parameters used for model training are given in table below.

Performance Function	Performance Function
Transfer Function	Transfer Function
No. Of neurons used for hidden layer	No. Of neurons used for hidden layer

Table 2: Model parameter values for Back Propagation Algorithm for all the models

Parameters used for Network Training	
Network Type	Network Type
Training Functions Used	Training Functions Used
Adaption Learning Function	Adaption Learning Function

5. Results and Discussions

Network performance of both the models M1 and M2 are given in tables below

Table. 3. Network Performance for Model M1 for one Month Ahead Forecasting

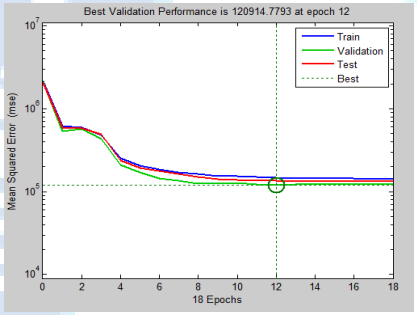
No. of neurons	**Validation Performance**	**Epochs**	**Reasons for stopping**	**Network Configuration**
5	120914.77	12	Val. Stopped/Max. iterations	3-5-1
10	120332.90	14	Val. Stopped/Max. iterations	3-10.1
15	129833.71	22	Val. Stopped/Max. iterations	3-15-1
20	136442.98	8	Val. Stopped/Max. iterations	3-20-1
25 (best)	**112461.95**	**4**	**Val. Stopped/Max. iterations**	**3-25-1**
30	152499.50	23	Val. Stopped/Max. iterations	3-30-1
35	103683.66	24	Val. Stopped/Max. iterations	3-35-1
40	122817.65	13	Val. Stopped/Max. iterations	3-40-1
45 (best)	132373.42	8	Val. Stopped/Max. iterations	3-45-1
50	154452.31	6	Val. Stopped/Max. iterations	3-50-1

Table. 4. Network Performance for Model M2 for two Month Ahead Forecasting

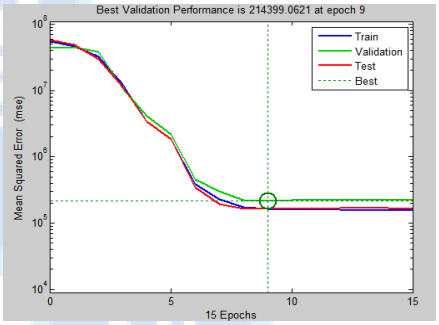
No. of neurons	**Validation Performance**	**Epochs**	**Reasons for stopping**	**Network Configuration**
5	196653.42	34	Val. Stopped/Max. iterations	3-5-1
10	169111.91	44	Val. Stopped/Max. iterations	3-10.1
15	142875.66	29	Val. Stopped/Max. iterations	3-15-1
20	146643.47	12	Val. Stopped/Max. iterations	3-20-1
25	198612.44	11	Val. Stopped/Max. iterations	3-25-1
30	220510.21	8	Val. Stopped/Max. iterations	3-30-1
35	208755.18	5	Val. Stopped/Max. iterations	3-35-1

40	233804.28	10	Val. Stopped/Max. iterations	3-40-1
45	159821.84	13	Val. Stopped/Max. iterations	3-45-1
50 (best)	**138643.47**	**12**	**Val. Stopped/Max. iterations**	**3-50-1**

From the table 3 and 4 given above it is seen that the arrangement of the table is done in ascending order of the number of neurons in the hidden layer and the validation performance is measured accordingly. Further it is seen that the best network strucutres are 3-25-1 and 3-50-1 for M1 and M2 models respectively. It was also noticed that the performance did not necessarily improve even when the network error was low. In the following figures 3.3 (a) and (b) given below one can notice that roughly after 12 and 9 epochs, the performance of training, testing and validation errors were somewhat stagnant. This shows that after epochs 12 and 9 there is no further improvement in the performance of the network and the network seems to have saturated.

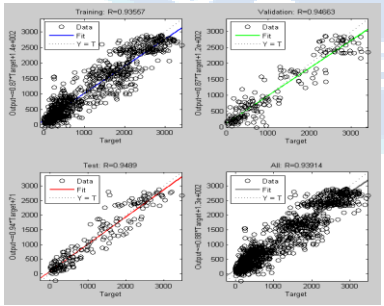


(a)

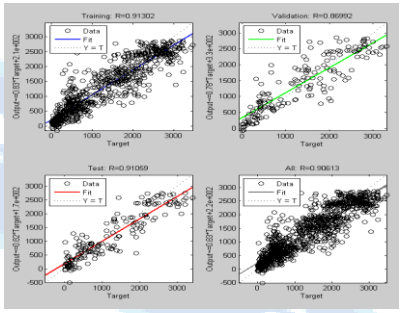


(b)

Fig. 1: Training of NN model gauged by MSE for M1 and M2 Models respectively



(a) for N=45



(B) for N=50

Fig. 2. Scatter Plot of R Values for Training, Testing and Validating datasets for best developed network for M1 and M2 Models Respectively

Further figure 1 given above represents a linear regression analysis between the network response and the network output. It can be inferred that NN model does good mapping. 15% of the data which was used for validation was not used for training at all. Hence performance of these machining conditions is something that neural network model has never experienced before. Therefore, one can consider this mapping to be true and representing functional relationship.

Following figures 3 given below depicts the comparison between actual and simulated data for surface roughness. It is noticed that except for rare occasions, simulated surface roughness values for the designated parameters are in acceptable proximity with actual values. This representation therefore agrees with the conclusion that, high accuracy of prediction is attained by Neural Network Model after successful completion of training criteria i.e. with the value of MSE being within acceptable range as well as agreeable performance measure. Hence, from the results it is inferred that the performance of thew NN model is acceptable. Thus can further be confirmed from the scatter plots shown in 4.

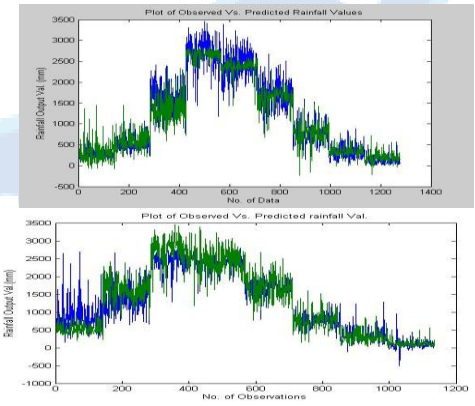


Fig 3. Comparison of observed and predicted values by best M1 & M2 models

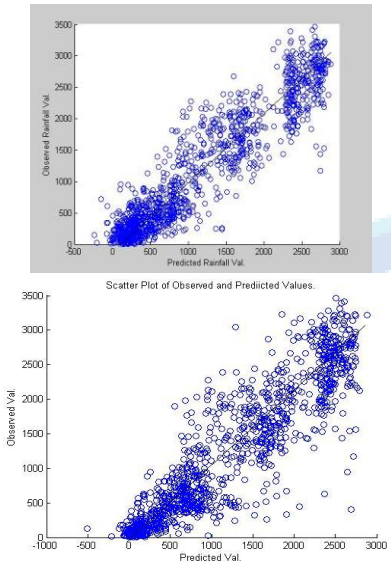


Fig. 4. Scatter Plot of Observed Vs. Predicted Rainfall values for M1 & M2 Models

Further analysis of the observed and predicted values for both M1 nad M2 models on the basis of MRE valuers is shown in figures 5 below.

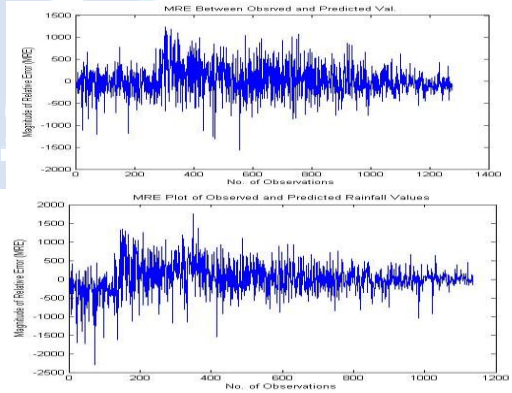


Fig 5. Deviation of Predicted value from Observed value for M1& M2 Models

Further, table 3 and 4 below shows the computed values of Regression and MSE values for the various M1 and M2 models considering different network structures. For the network identification used in the second column of table, the first number indicates the number of neurons in the input layer, the last number indicates the number of neurons in the output layer, and the numbers in between represent neurons in the hidden layer.

For M1 Model:- From table 5 it is clear that for M1 model 3-25-1 is the best model developed with the least MSE value of 112461.95 and the best regresion values of 0.946 and 0.948 both the validating and testing data sets respectively, although the training regression values is slightly inferior in respect to other models. The same has been depicted graphically in figure 6 given below.

Table 5: Statistical Parameters for different netwok structures for M1 Model

Structure No.	No. of Neurons	Regression Values			MSE
		Training Val.	Validating Val.	Testing Val.	All
1	3-5-1	0.9211	0.927	0.93	0.923
2	3-10-1	0.936	0.937	0.941	0.937
3	3-15-1	0.943	0.931	0.92	0.937
4	3-20-1	0.946	0.921	0.916	0.937
5	**3-25-1**	**0.935**	**0.946**	**0.948**	**0.939**
6	3-30-1	0.948	0.912	0.929	0.939
7	3-35-1	0.95	0.941	0.922	0.945
8	3-40-1	0.951	0.931	0.93	0.945
9	3-45-1	0.942	0.936	0.943	0.942
10	3-50-1	0.949	0.916	0.937	0.941

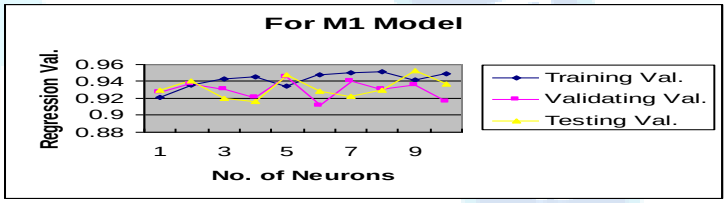


Fig. 6. Graphical representation of Regression values for M1 Model

Further from table 5 and figure 6 one can see that in general the R values for training are in general better than testing and validating datasets, which shows that the network seems to have performed well for training datasets. Further from the detailed analysis of figure 3.5 one can see that as the size of the network structure increases the Rvalue for training dataset becomes better but there is a slight fall in the values of R for testing and validating datasets.

For M2 Model:- From table 6 it is clear that for M2 model 3-50-1 is the best model developed with the least MSE value of 138643.47 and the best regresion values of 0.913 and 0.910 both the training and testing data sets respectively, with overall R value of 0.906, although the validating regression values is slightly inferior in respect to other models. The same has been depicted graphically in figure 6 given below.

Table 6 :- Statistical Parameters for different netwok structures for M2 Model

Structure No.	No. of Neurons	Regression Values			MSE
		Training Val.	Validating Val.	Testing Val.	All
1	3-5-1	0.897	0.886	0.903	0.896
2	3-10-1	0.901	0.904	0.884	0.900
3	3-15-1	0.901	0.904	0.905	0.902

4	3-20-1	0.907	0.924	0.849	0.901
5	3-25-1	0.912	0.879	0.892	0.904
6	3-30-1	0.908	0.876	0.875	0.899
7	3-35-1	0.905	0.889	0.88	0.899
8	3-40-1	0.905	0.856	0.909	0.899
9	3-45-1	0.909	0.894	0.901	0.905
10	**3-50-1**	**0.913**	**0.869**	**0.910**	**0.906**

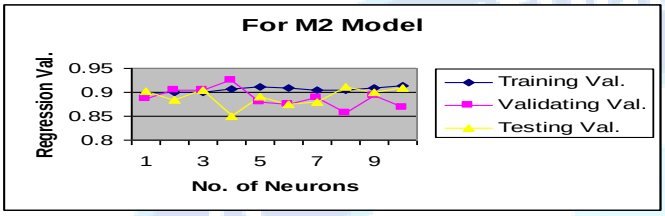


Fig. 7. Graphical representation of Regression values for M2 Model

Further a comparative analysis of M1 and M2 model based on tables 3.5 and 3.6 shows that M1 model with one month ahead prediction of rainfall values has better R values for traing , testing and validating datasets than M2 model with 2 months ahead prediction.

6. Conclusions:

In this study, applicability and capability of ANN technique for long term forecasting has been investigated. For this rainfall data of North India has been procured from IMD,Pune internet site. Two models M1 and M2, for one month and two month ahead prediction of rainfall, having different input variables were trained and tested using ANN technique. In both the cases for M1 and M2 models, 3-25-1 and 3-50-1 network structure model was found to be the best forecasting model. Further from the above analysis done in Results and Discussions, it is seen that ANN has performed better for M1 model than M2 model, showing that ANN technique has been able perform better for one month than two months ahead. Also the regression values for M1 model has shown better results than M2 model.

References:

[1] Asce task committee on application of artificial neural networks in hydrology, (2000a), "artificial neural networks in hydrology. I: preliminary concepts", journal of hydrologic engineering , 5(2), pp 115–123. 6

[2] Faraway, j. And chatfield, c., (1995), "time series forecasting with neural networks : a case study. Research report 95-06 of the statistics group", university of bath, uk. 16

[3] Huang, y.,2(009.), "advances in artificial neural networks – methodological development and application", algorithms, 2, pp 973-1007. 21

[4] Lee, y. And tong, l. (2011), "forecasting time series using a methodology based on autoregressive integrated moving average and genetic programming", knowledge-based systems, vol. 24, pp. 66-72

[5] R. C. Eberhart and y. Shi, computational intelligence: concepts to implementations: morgan kaufmann, 200 54

[6] Xiao hu, peng xu, shaozhi wu, shadnaz asgari and marvin bergsneider. 2010. "a datamining framework for time series estimation. Journal of biomedical informatics" 43(2010) 190–199. Elsevier.

[7] Zhang, g. P., (2003), "time series forecasting using a hybrid arima and neural network model", neurocomputing, 50, pp 159–175. 37

[8] Box, g.e.p., jenkins, g.m. And reinsel, g.c. (1994). Time series analysis : forecasting and control, pearson education, delhi.

[9] Kalogirou, s. A., neocleous, c., constantinos, c. N., michaelides, s. C.& schizas, c. N.,"a time series construction of precipitation records using artificial neural networks. In: proceedings of eufit '97 conference, 8–11 september, aachen, germany. Pp 2409–2413 1997.

[10] Anurag, R. Sharma, " Load Forecasting by using ANFIS", International Journal of Research and Development in Applied Science and Engineering, Volume 20, Issue 1, 2020.

[11] R. Sharma, Anurag, " Load Forecasting using ANFIS A Review", International Journal of Research and Development in Applied Science and Engineering, Volume 20, Issue 1, 2020.

[12] R. Sharma, Anurag, " Detect Skin Defects by Modern Image Segmentation Approach, Volume 20, Issue 1, 2020.

[13] Anurag, R. Sharma, " Modern Trends on Image Segmentation for Data Analysis- A Review", International Journal of Research and Development in Applied Science and Engineering, Volume 20, Issue 1, 2020.

[14] Pucheta julian a, et. Al , "a feed-forward neural networks-based nonlinear autoregressive model for forecasting time series", computación y sistemas vol. 14 no. 4, pp 423-435, 2010.

[15] Adhikari, r., et. Al., "forecasting strong seasonal time series with artificial neural network", journal of scientific and industrial research, vol. 71, pp. 657-666, 2012.