VOL. 11, NO. 22, NOVEMBER 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

A NEURAL NETWORK AUTOREGRESSION MODEL TO FORECAST PER CAPITA DISPOSABLE INCOME

Debasish Sena[1] and Naresh K. Nagwani[2]
[1]Computer Technology, National Institute of Technology Raipur, Raipur, Chhattisgarh, India
[2]Computer Science and Engineering, National Institute of Technology Raipur, Raipur, Chhattisgarh, India
E-Mail: nknagwani.cs@nitrr.ac.in

ABSTRACT

Time series analysis is an important technique for future forecasting of time dependent variables. Keeping future visualization in mind, time series analysis is applicable to a wide variety of applications. In this work, neural network autoregressive (NNAR) model, a non-linear model is applied for forecasting of per capita disposable income. The average available money per person after the deduction of income taxes is called as the per capita disposable income. It is an indicator of the economic condition of a nation. Forecasting of per capita disposable income is essential in helping the government assessing its economic state with respect to the economy of other developing countries of the world. Financial critical situation like inflation can also be assessed by forecasting of per capita disposable income. Future policies and plans can also be formulated by the planning commission of a country upon observations of the results obtained from this work.

Keywords: time series analysis, NNAR model, forecasting, prediction model, artificial neural network.

INTRODUCTION

Among the latest forecasting techniques, time series analysis is an important one. This focuses on prediction of the future values of a time dependent variable based on its past observed values. Time series analysis is usually carried out by designing suitable mathematical models. The models for time series analysis can be classified into two categories such a linear and non-linear model. A lot of works have already been performed involving time series analysis. In this work, NNAR model is used for forecasting purpose. Neural network models have been applied to forecast electric load (Park, El-Sharkawi, Marks II, Atlas and Dambotg, 1991), airline passenger number (Faraway and Chatfield, 1998) and bankruptcy prediction (Jo and Han, 1997).

In addition techniques like Case-Based Reasoning (CBR), Support Vector Machine (SVM), Autoregressive Integrated Moving Average (ARIMA) model and Markov-Chain model can be used for time series analysis. Min *et al.* (Min and Lee, 2005) has proposed an SVM based prediction model for bankruptcy prediction. Hansen *et al.* (Hansen and Riordan, 2000) has performed weather prediction using a hybrid model of CBR and fuzzy set theory. Singh *et al.* have used an ARIMA model for prediction of software bugs (Singh, Abbas, Ahmed and Ramaswamy, 2010). Generalized discrete GM (1, 1) model based forecasting model is presented by You *et al*. (You, Zhang and Periodical, 2013) to predict the per capita disposable income of urban residents.

In this paper, the focus is on forecasting the quarterly per capita disposable income of West Germany using NNAR model. Per capita disposable income is the total personal income excluding the current taxes. As a key economic indicator to assess the overall state of the economy, per capita disposable income is regularly monitored. Per capita disposable income helps countries gauge their economic growth. The per capita income of a country is severely affected by factors like poverty and inflation. So in order to avoid future financial crisis, forecasting of per capita disposable income is very essential.

NNAR ANALYSIS

Neural networks are a kind of statistical models, popularly used in machine learning. The concept of artificial neural network is inspired from the biological neural network of the central nervous system. These are better used to estimate functions based on large volume of training sets and are network of multiple nodes working in parallel as shown in the Figure-1. NNs are capable of modelling complex non-linear relationship without prior assumption of the underlying relationship. Each node is associated with an activation function which transforms the input to the node to an output which in turn acts as an input for the nodes in subsequent layer. Each node multiplies the input signal with a weight w_{ij}, characteristic of the connection the connection between nodes i and j of adjacent layers and then sums up the weighted input. Each hidden layer node performs a single 'sigmoid' transformation of its input (BuHamra, Smaoui and Gabr, 2003).

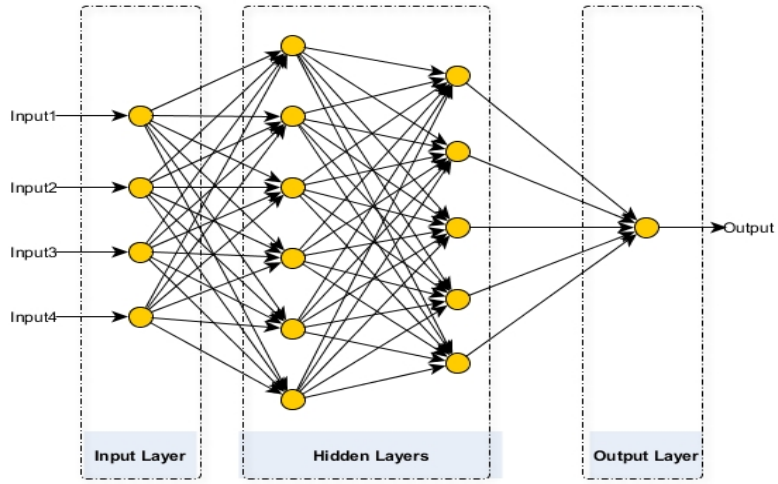

Figure-1. NN architecture consists of four input nodes, two hidden layers and one node output layer.

$$z_j = g(\sum_i y_i w_{ij} - \beta_j), \quad (1)$$

where z_j is the output of the jth node, y_i is the ith input, β_j is the bias of the jth node and $g(\square)$ is the sigmoidal function given b

$$g(x) = \frac{1}{1+e^{-x}} \quad (2)$$

The successful implementation of an NN is dependent on proper training of the network with large volume of training sets. In the process of training the NN learns to adjust the weights associated with each connection with the help of a learning algorithm. The back-propagation algorithm is the most famous and widely used learning algorithm of all neural network paradigms. This algorithm trains the network based on the response obtained for each received input. Then it adjusts the weights by comparing the obtained response with the target output so as to minimize the sum squared error (SSE) of the network which is given by

$$SSE = \frac{1}{2}\sum_p \sum_k \left(\tilde{Z}_k - Z_k\right)_p^2 \quad (3)$$

where Z_k and $\tilde{Z}_k$ are the target output and the response vector of the *k*th output node. The *p* subscript refers to the specific input vector pattern used. The weights obtained after training phase are the best suited for producing best output at the output layer.

The NNAR model is a parametric non-linear model of forecasting. Forecasting in this model is performed in two steps. In the first step, the order of auto regression is determined for the given time series. The order of auto-regression indicates the number of previous values which the current value of the time series is dependent upon. In the second step, the NN is trained with a training set prepared by taking the order of auto-regression into consideration. The number of input nodes is determined from the order of auto-regression and the inputs to the NN are the previous, lagged observations in univariate time series forecasting. The predicted values are the output of the NN model. The number of hidden nodes is often decided by trial-and-error or through experimentation due to lack of any theoretical basis for selection (Zhang, Patuwo and Hu, 2001). The number of iteration should be selected properly in order to avoid the problem of over fitting.

IMPLEMENTATION AND DISCUSSIONS

This section focuses on the implementation details of the NNAR model using a dataset and the result obtained is thoroughly discussed also.

Dataset

The dataset used in this work is taken from Datasets for Applied Time Series Econometrics[1]. The dataset of quarterly, seasonally adjusted West German real per capita disposable income is chosen for the NNAR model implementation. The dataset is as shown in the Figure-2 and shows the quarterly, seasonally unadjusted West German real per capita disposable income from 1960 to 1987.

Building the model

The 'forecast' package of the 'R' statistical tool[1] has been used here for developing the NNAR model for the chosen dataset. Figure-3 represents the time plot of the chosen dataset. A step by step procedure has been followed for building the neural network model. The steps followed are:

Step 1: Exploratory data analysis is carried out with the help of some traditional time series analysis methods to estimate the lag difference in the data.

Step 2: Data has been partitioned into training and validation sets. Data from first quarter of 1960 to fourth quarter of 1984 are used as training set and data from first quarter of 1985 to fourth quarter of 1987 are used as validation set.

Step 3: A rough layout of the neural network is created by considering the number dependent lags for input and number of hidden layers.

Step 4: Training patterns are prepared for the training of neural network. In this experiment training pattern consists of current data, data at lag 1 and data at lag 4.

Step 5: Neural network is trained using the training patterns.

Step 6: Netwok is tested on validation sets by calculating the mean relative error of forecasting.

To determine the order of auto-regression for the dataset, PACF plot is drawn for the dataset and is shown in the Figure-4. The abscissa represents the lags and the ordinate represents the partial autocorrelation values for the corresponding lags. As the dataset is a quarterly dataset, the PACF plot shows a significant partial autocorrelation value at lag 1 and a seasonal partial autocorrelation value at lag 4. This indicates the current time series value is regressed upon its previous value and its previous seasonal value. Hence an NNAR (1, 1) model is proposed for the forecasting of the time series.

	Qtr1	Qtr2	Qtr3	Qtr4
1960	1684.608	1757.207	1842.721	2027.318
1961	1859.056	1832.643	1893.536	2098.923
1962	1873.268	1908.643	1989.580	2203.581
1963	1931.262	1980.327	2065.510	2256.867
1964	2040.467	2098.867	2145.864	2427.700
1965	2178.308	2268.469	2313.577	2548.202
1966	2239.871	2286.016	2379.546	2535.525
1967	2247.802	2311.833	2332.953	2579.139
1968	2348.289	2404.779	2478.411	2808.609
1969	2535.429	2605.453	2688.117	2998.783
1970	2695.444	2764.954	2879.483	3255.635
1971	2825.476	2877.223	2955.720	3378.428
1972	3014.108	3030.190	3092.076	3491.399
1973	3098.311	3091.690	3144.909	3535.088
1974	3096.384	3110.859	3222.077	3613.994
1975	3233.532	3319.459	3323.182	3723.165
1976	3312.514	3324.622	3418.389	3824.463
1977	3414.656	3438.236	3487.294	3973.899
1978	3546.968	3553.151	3655.235	4102.794
1979	3680.968	3783.815	3755.960	4261.406
1980	3797.327	3786.522	3790.926	4282.953
1981	3848.164	3764.649	3759.083	4294.212
1982	3811.583	3709.892	3641.536	4180.713
1983	3734.133	3698.315	3664.942	4237.391
1984	3847.087	3766.339	3783.385	4327.609
1985	3887.862	3839.148	3859.015	4431.022
1986	4049.019	4053.196	4077.438	4653.943
1987	4191.428	4163.433	4177.777	4811.136

Figure-2. Quarterly, seasonally unadjusted West German real per capita disposable income.

By using the 'forecast' package of the 'R' statistical tool, the neural network model is trained with the data from the first quarter of 1960 to fourth quarter of 1984 with 1 hidden layer of 8 nodes and 50 iterations. The neural network is as shown in the Figure-5. The neural network accepts lag 1 value and lag 4 values of the time series as the input and produces the current value of the time series as output. Then forecasting is performed for the next three years using the developed NNAR model. The forecast plot is as shown in the Figure-6. The blue portion of the plot shows the forecasted quarterly per capita disposable income. The forecast also shows a seasonal variation.

DISCUSSIONS

After forecasting for the next three years, the mean relative error (MRE) of forecasting is calculated by comparing the forecasted values with the actual values of the dataset.

ARPN Journal of Engineering and Applied Sciences

Table-1. Forecast details of quarterly per capita disposable income.

Year	Quarter	Actual value	Forecasted value	Relative error
1985	Q1	3887.862	3817.769	0.018028
1985	Q2	3839.148	3836.403	0.000715
1985	Q3	3859.015	4332.733	0.122756
1985	Q4	4431.022	3843.586	0.132573
1986	Q1	4049.019	3882.153	0.041211
1986	Q2	4053.196	4332.654	0.068947
1986	Q3	4077.438	3865.482	0.051982
1986	Q4	4653.943	3922.238	0.157222
1987	Q1	4191.428	4328.445	0.032689
1987	Q2	4163.433	3884.251	0.067055
1987	Q3	4177.777	3957.276	0.052779
1987	Q4	4811.136	4321.003	0.101874

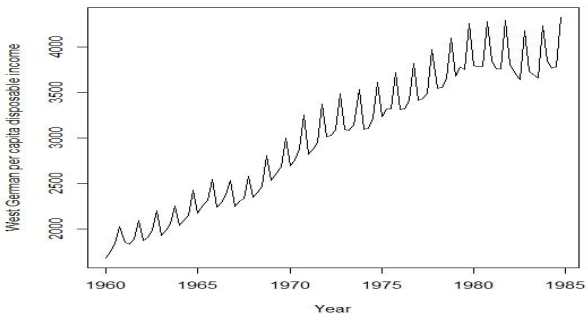

Figure-3. Time plot of quarterly per capita disposable income.

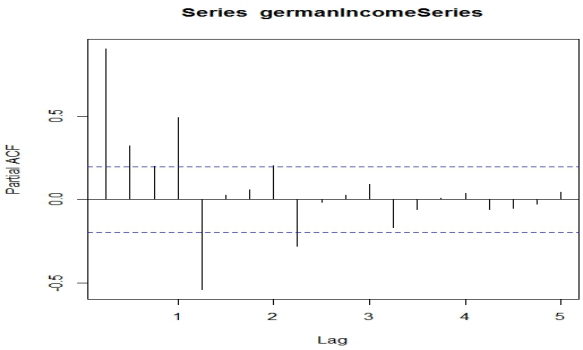

Figure-4. PACF plot of quarterly per capita disposable income.

The MRE is calculated as 0.070652 which confirms the model as a good model for forecasting. The forecasting details are given in the Table-1.

ARIMA based prediction model for forecasting the per capita income is also designed by Sena and Nagwani (Sena and Nagwani, 2015) which gives a relative error of 0.0225 on the same dataset. It demonstrates that the ARIMA based time series model is most appropriate as compared to the model presented in this paper. It also shows that a non linear model does not perform better as compared to ARIMA based model for non stationery time series analysis. ARIMA can suitably be used for non stationery time series by converting it to stationery time series and gives better results in forecasting as compared to the non linear NNAR models.

CONCLUSIONS

In this work, a non-linear time series analysis technique, the neural network model is developed to forecast the quarterly seasonally unadjusted West German per capita disposable income. A satisfactory result of forecasting is obtained from the experiments with a mean relative error of 0.070652. The neural network model fits the non-linearity of the time series well with successive iterations. The comparative study of the presented work is also performed with the ARIMA model for forecasting of the per capita disposable income. The limitation of the NNAR model is to carefully select the number of iteration and the number of neurons in the hidden layer in order to avoid the problem of overfitting. The combinations of the NNAR model with other linear and non-linear model can also be explored in the future scope of this work.

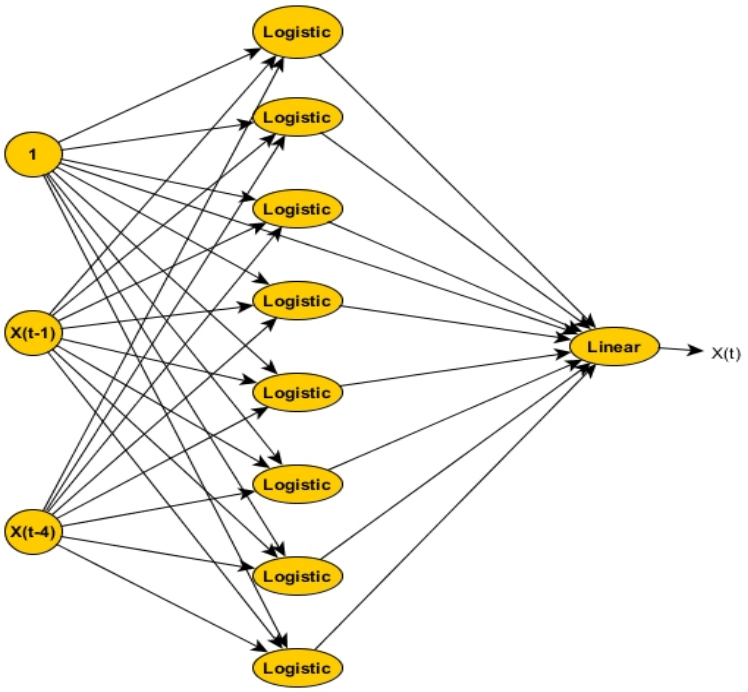

Figure-5. NNAR(1,1) model architecture.

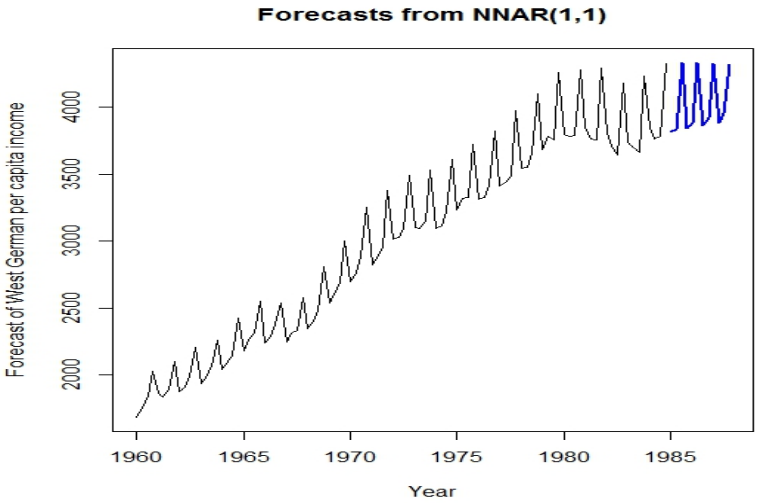

Figure-6. Forecast plot for quarterly per capita disposable income.

ACKNOWLEDGEMENT

The authors thank National Institute of Technology Raipur, India for providing necessary infrastructure to carry out this research work and grant to attend the conference.

REFERENCES

BuHamra, S., Smaoui, N. and Gabr, M. 2003. The Box-Jenkins Analysis and Neural Networks: Prediction and Time Series Modelling. Applied Mathematical Modelling, 28, pp. 805-815.

Faraway, J. and Chatfield, C. 1998. Time Series Forecasting with Neural Networks: A Comparative Study Using the Airline Data. Applied Statistics, 47(2), pp. 231-250.

Hansen, B. K. and Riordan, D. 2000. Weather Prediction Using Case-Based Reasoning and Fuzzy Set Theory. Master of Computer Science Thesis, Technical University of Nova Scotia, Halifax, Nova Scotia, Canada.

Jo, H. and Han, I. 1997. Bankruptcy Prediction Using Case-Based Reasoning, Neural Networks and

Discriminant Analysis. Expert Systems with Applications, 13(2), pp. 97-108.

Min, J. H. and Lee, Y. C. 2005. Bankruptcy Prediction Using Support Vecot Machine with Optimal Choice of Kernel Function Parameters. Expert Systems with Applications, 28, pp. 603-614.

Park, D. C., El-Sharkawi M. A., Marks II, R. J., Atlas, L. E. and Damborg, M. J. 1991. Electric Load Forecasting Using an Artificial Neural Network. IEEE Transaction on Power Systems, 6(2), pp. 442-449.

Sena D., Nagwani N. K. 2015. Application of time series based prediction model to forecast per capita disposable income, 2015 IEEE International Conference on Advance Computing Conference (IACC), Banglore, India, pp. 454-457.

Singh, L. L., Abbas, A. M., Ahmed, F. and Ramaswamy, S. 2010. Predicting Software Bugs Using ARIMA model. 48th Annual Southeast Regional Conference, New York, USA.

You, Z. S., Zhang, X., & Periodical, P. 2013. Per Capita Disposable Income Prediction of Urban Residents Based on GM (1, 1) Model: Take Urban Residents Families in Chongqing as Examples. Journal of Chongqing Normal University (Natural Science), 5, 014.

Zhang, G. P., Patuwo, B. E. and Hu, M. Y. 2001. A simulation study of artificial neural networks for non-linear time series forecasting. Computers & Operations Research, vol. 28, pp. 381-396.