

# Africa Economic Growth Forecasting Research Based on Artificial Neural Network Model: Case Study of Benin

Songbian Zime

Department of Engineering,  
School of Management and Economic,  
University of Electronic Science and Technology of China

**Abstract-** Economic growth forecasting is important to make the policy on national economic development. Neural Networks are an artificial intelligence method for modeling complex target functions. For certain types of problems, such as learning to interpret complex real world sensor data, Artificial Neural Networks (ANN) are among the most effective learning methods currently know. The objective of this paper is to provide an intelligent algorithm and a practical in the design of a neural network for economic growth forecasting.

Based on the original data from Benin economic database, we extracted the knowledge of economic classification, which includes attribute discretization, attribute importance ranking, attribute reduction and prediction rule. Then input the extracted key components into neural network as the input training sample. This method reduced the structure of neural network, and improved the training speed and the accuracy of prediction. The Benin Case study shows that the neural network can solve nonlinear problem and had been proved that the method is effective and feasible with high accuracy. The model has a good reference value for practical application.

**Keywords:** *Neural networks; economic forecasting; back propagation; NARX networks, data preprocessing; Training; Testing; Validation; Network architecture; algorithm, Learning rate*

## INTRODUCTION

Macroeconomic forecasting is a very difficult task due to lack of an accurate, convincing model of the economy. According to the World Bank data report<sup>1</sup>, GDP growth in Sub-Saharan Africa is projected to strengthen to 4.9 percent in 2013, rising to 5.3 percent in 2014 and 5.5 percent in 2015.

Usually, people care most about the precision and confidence in economic forecasting process mostly. Generally speaking, it may be considered from two aspect<sup>2</sup>, one is to use some historical information which can reflect the intrinsic behavior in economy process; the other is to choose a correct forecast method and model which can describe the developing condition of economy system. However, it is usually very difficult problem to collect enough high quality sample data, as a result of the complexity and dynamic of economic process. The best forecast economy key is to choose the appropriate forecast model which can give high accuracy. It was pointed out<sup>3</sup> that forecast methods commonly used in Africa are: univariate time series model (ARIMA), Phillips curve, interest model, the naïve model, regression analysis,

causal model, quantity economic forecasting and output-input model;- all these models are classified under statistical analysis model.

These techniques usually need to establish model in advance according to experience or economic theory, so is often has much subjectivity, then the forecast error is often very big using these methods.

The modern data technology which appeared at the end of 1980s also has obtained wide application in the economic forecasting field. The method used in data mining are statistics mining, the decision tree mining, the connection mining, the neural network mining...etc<sup>4</sup>. Among these methods, the artificial neural network (ANN) is an auto-adapted dynamics system. ANN are a very powerful tool in modern quantitative economic and have emerged as a powerful statistical modeling technique. ANN provides an attractive alternative tool for both researches and practitioners. They can detect the underlying functional relationships within a set of data and perform tasks such as pattern recognition, classification, evaluation, modeling, prediction and control. Several distinguishing features of ANN make them valuable and attractive in forecasting. First, ANN are nonlinear data-driven. They are capable to perform nonlinear modeling without a prior knowledge about the relationships between input and outputs variables. The non-parametric ANN model may be preferred over traditional parametric statistical models in situations where the input data do not meet the assumptions required by the parametric model, or when large outliers are evident in dataset<sup>5</sup>. Second, ANNs are universal functions approximation. It has been shown that a neural network can approximate any continuous function to any desire accuracy. Third, ANN can generalize. After learning the data presented to them, ANN can often correctly infer the unseen part of a population even if the sample data contain noisy information. Neural Networks are able to capture the underlying pattern or autocorrelation structure within a time series even when the underlying law governing the system is unknown or too complex to describe. In the past few years, artificial neural network has widely applied in economic growth forecasting. The shortcoming including locally optimal solutions and over-fitting limits the practical application of artificial neural network<sup>6</sup>.

This paper is to provide an overview of a step by step methodology to design a neural network for forecasting

economic time series data. First, the architecture of a back propagation (BP) and Nonlinear Autoregressive models with exogenous is briefly discussed. Secondly, design architecture of neural network and establish of algorithms and thirdly, the experimental result and conclusion. However, these methods need profound mathematics, algorithm knowledge and computer program technology to understand for application. For the study case, we applied these methods to provide one kind of easy and high accuracy forecast for Benin national economic growth.

II- ARTIFICIAL NEURAL NETWORK PRINCIPLE AND FORECAST

An Artificial Neural Network (ANN) is a mathematical model that tries to simulate the structure and functionalities of biological neural networks. Basic building block of every artificial neural network is artificial neuron, that is, a simple mathematical model or function. Such a model has three simple sets of rules: multiplication, summation and activation. At the entrance of artificial neuron the inputs are weighted which means that every input value is multiplied with individual weight. In the middle section of artificial neuron is sum function that sums up all weighted inputs and bias (Figure1). Artificial neural network is a method of information processing, which is developed by the biological neural systems inspired. Based on the learning sample process, the artificial neural network analyzes the data mode, builds the model and then finds some new knowledge. Neural network can automatically adjust the neurons input and output in accordance with the rules through learning, to change the internal state.

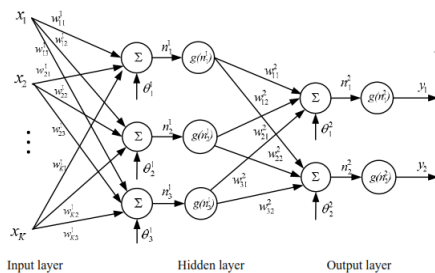


Figure 1: A multilayer perceptron network with one hidden layer.

A- Back Propagation

Back propagation (BP) neural networks consist of a collection of inputs and processing units known as neurons, or nodes (Figure 2). The neurons in each layer are fully interconnected by connection strengths called weights which, along with the network architecture, store the knowledge of a trained network. In addition to the processing neurons, there is a bias neuron connected to each processing unit in the hidden and output layers.

The standard BP network is made up of three kinds of neural unit layers. The lowest layer is called the input layer. The middle one is the implication layer (may be multi-layer). And the top one is the output layer. Every layer of neural unit forms fully-connection, and the neural unit in each layer has

no connection. The study course of BP algorithm is made up of propagation and backpropagation<sup>7</sup>

In the propagation course, the input information is transferred and processed through input layer and implication layer. The state of every neural unit layer only affects the state of next layer. If the expected information cannot be got in the output layer, the course will turn into the back-propagation and return the error signal along the former connection path. Altering the connection weight between each layer, the error signal is transmitted orderly to the input layer, and then passes the propagation course. The repeated application of these two courses makes the error become smaller and smaller, until it meets the error demand.

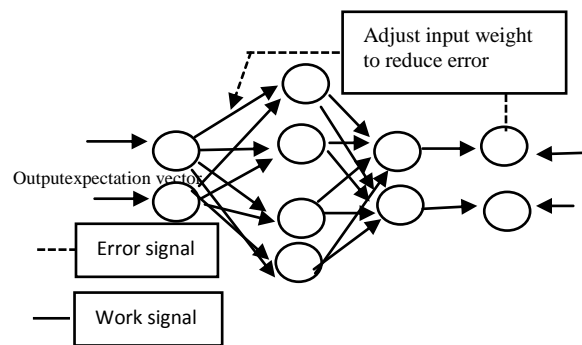


Figure2: Back propagation architecture.

B- NARX networks

An important and useful class of discrete time nonlinear systems is the “Nonlinear Autoregressive models with exogenous input (NARX model)”, are therefore called NARX recurrent neural networks<sup>8, 9</sup>. This is a powerful class of models which has been demonstrated that they are well suited for modeling nonlinear systems and specially time series. One principal application of NARX dynamic neural networks is in control systems. Some important qualities about NARX networks with gradient-descending learning, gradient algorithm have been reported:<sup>10</sup> learning is more effective in NARX networks than in other neural network (the gradient descent is better in NARX) and these networks converge much faster and generalize better than other networks<sup>11, 12</sup>. The simulated results show that NARX networks are often much better at discovering long time – dependences than conventional recurrent neural networks. An explanation why output delays can help long-term dependences can be found by considering how gradients are calculated using the back-propagation-through-time (BPTT) algorithm. A state space representation of recurrent NARX neural networks can be expressed as<sup>11</sup>:

$$z_k(k + 1) = \begin{cases} \phi u(k), z_i(k), & i = 1 \\ z_i(k), & i = 2,3, \dots, N, \end{cases}$$

Where the output  $y(k) = z_i(k)$  and  $z_i, i=1, 2 \dots N$  are the state variable of recurrent network. The recurrent network exhibition forgetting behavior, if

$$\lim_{m \rightarrow \infty} \left( \frac{\partial z_i(k)}{\partial z_i(k - m)} \right) = 0 \quad \forall k, m \in K, i \in O, j \in I$$

Where  $z$  is state variable, "I" denote the set of input neurons, "O" denotes the set of output neurons and  $k$  denotes the time index set.

1 -The NARX models

The NARX model for approximation of a function  $G$  can be implemented in many ways, but the simpler seems to be by using a feed forward neural network (FFNEW) with the embedded memory (a first tapped delay line), as is shown in figure 3, plus a delayed connection from the output of the second layer to input (a second tapped delay line). Making the network dependent on  $d_u$  previous sequence elements is identical to using  $d_u$  input units being fed with  $d_u$  adjacent sequence elements. This input is usually referred to as a time window since it provides a limited viewed on part of the series. It can also be viewed as a simple way of transforming the temporal dimension into another spatial dimension.

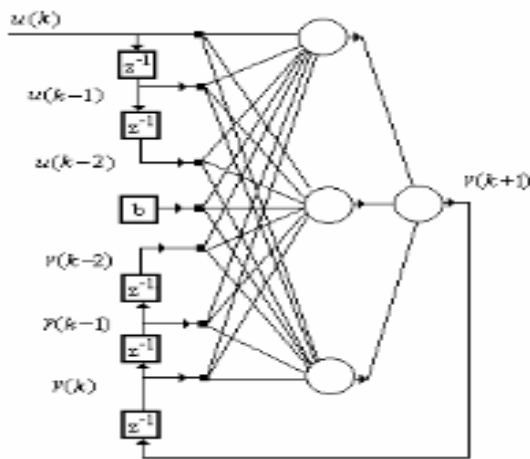


Figure 3: NARX model with tapped delay line at input.

For the architectural model in figure 3 the notation used is  $NN(d_u, d_y; N)$  to denote the NN with  $d_u$  input delay,  $d_y$  output delay and  $N$  neurons in layer 1. Similarly, for the architecture model in figure 2 the notation used is  $NN(d_{u1}, d_{u2}, d_y; N)$ . For the NN models used in this work, with two levels (level 1 surnamed input layer and level 2 or output layer), the general prediction equations for computing the next value of time series  $y(k+1)$  (output) using model in figure 3 for the past observation  $u(k), u(k-1), \dots, u(k-d_u)$  and the past output  $y(k), y(k-1), \dots, y(k-d_y)$  as inputs, may be in the function:

$$y(k+1) = \phi_0 \left\{ w_{ho} + \sum_{h=1}^N w_{ho} \cdot \phi_h \left( w_{ho} + \sum_{i=0}^{d_u} w_{ih} u(k-i) + \sum_{j=0}^{d_y} w_{jh} \cdot y(k-j) \right) \right\}$$

2- NARX neural network training

The training process has some difficulties. One is related to the number of parameters, which refers to how many connections or weights are contained in network. Usually, this number is large and there is a real danger of "overtraining" the data and producing a false fit which does not lead to better forecasts. For NARX neural network model the number is given by  $p = (d_u + d_y + 2) N$ . A solution is penalizing the

parameter increase<sup>12</sup>. This fact motivates the use of an algorithm including the regularization technique, which involves modifying the performance function for reducing parameters value. Practically, the typical performance function used in training, MSE, is replaced by a new one, MSEreg, as follows:

$$MSE = \frac{1}{N} \sum_{i=1}^N (e_i)^2 = \frac{1}{N} \sum_{i=1}^N (t_i - y_i)^2$$

$$MSE = \frac{1}{N} \sum_{j=1}^n w_j^2$$

$$MSE_{reg} = \epsilon MSE + (1 - \epsilon) MSW$$

Where  $t_i$  is the target and  $\epsilon$  is the performance ratio. The new performance function causes the network to have smaller weights and biases, and in this way forces the network response to be smoother and less likely to over fit.

III- DESIGNING ANEURAL NETWORKFORFORECASTINGMODEL

The design of a neural network successfully forecasting economic a time series is a complex task. The individual steps of this process are listed below:

- Step 1: Variable selection
- Step 2: Data collection
- Step 3: Data preprocessing
- Step 4: Training, testing, and validation sets
- Step 5: Neural network design
  - Number of hidden layer
  - Number of hidden neuron
  - Number of output neuron

- Step 6: Evaluation criteria
- Step 7: Neural network training

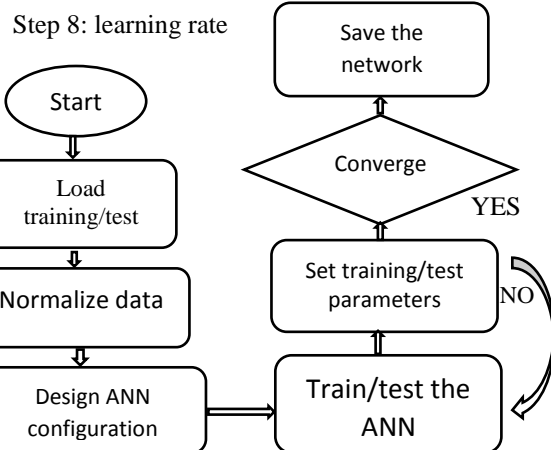


Figure 4: Development of ANN training/testing algorithms

#### a- Variable selection

Success in designing a neural network depends on a clear understanding of the problem. Knowing which input variables are important in the market being forecasted is critical. This is easier said than done because the very reason for relying on a neural is for its powerful ability to detect complex nonlinear relationships among a number of different variables. However, economic theory can help in choosing variables which are likely important predictors. At this point in design process, the concern is about the raw data from which a variety of indicators will be developed. These indicators will from the actual inputs to the neural networks.

The model applied in this paper uses lagged values of the dependent variables as a result of correlation and autocorrelation analysis.

#### b- Data collection

The cost and availability for data must be considered by researcher. Technical data is readily available from many vendors at a reasonable cost whereas fundamental information is more difficult to obtain. Time spend collecting data cannot be used for preprocessing, training and evaluating network performance. The vendor should have a reputation of providing high quality data; however, all data should still be checked for errors by examine day to day changes, ranges, logical consistency and missing observations .

Missing observations which often exist can be handled in a number of ways. All missing observations can be dropped or a second option is to assume that the missing observations remain the same by interpolating or averaging from nearby values.

#### c- Data processing

Some transfer functions need that the inputs and targets are scaled so that they fall within a specified range. In order to meet this requirement we need to pre-process the data. This refers to analyzing and transforming the input and output variables to minimize noise, highlight important relationships, detect trends, and flatten the distribution of the variable to assist the neural network in learning the relevant patterns. The input and output variables for which the data was collected are rarely fed into three networks in raw form. As the very least, the raw data must be scaled between the upper and lower bonds of the transfer functions.

#### d- Training, testing, and validation sets

The network takes a certain percentage (usually 70%) of the input data, and uses it as the training set, another percentage (usually 15%) and uses it as the validation set, and the remaining percentage (usually 15%) as the testing set. Common practice is to divide the time series into three distinct sets called the training, testing and validation (out-of-sample) sets. A final check on the validation set chosen must strike a balance between obtaining a sufficient sample size to evaluate a trained network and having enough remaining observations for both training and testing.

#### e- Neural Network design

A neural network's architecture defines its structure including the number of neurons in each layer and the number and type of interconnections.

The number of input neurons is one of the easiest parameters to select once the independent variables have been preprocessed because each independent variable is represented by its own input neuron. This section will address the selection of the number of hidden layers, hidden layer neurons, output neurons, and choice of weight value

Three layer neural networks can satisfy the majority application, but more hidden layers may be adopted for more complex situation. The node value has important influence on training result. Too little node value leads to poor training effect, or is unable to distinguish new sample as well as poor fault tolerance, while too big node value needs to excessively learning time, low generalization capacity both can drop network performance. Therefore, the best node value must exist. The reference<sup>13</sup> proposed following three empirical formulas to be referred to:  $C_{ni}^i$

- (1)  $\sum_{i=0}^n C_{ni}^i > k$ , in which k is sample number, n1 is node value of hidden layer, and n is node value of input layer
- (2) If  $i > n_1$ , then  $C_{ni}^i = 0$
- (3)  $n_1 = \sqrt{m + n} + a$ , in which m is node value of output layer, and n is node value of input layer, as well as a is a constant between 1 and 10,
- (4)  $n_1 = \log_2 n$ , in which n is node value of input layer.
- (5) The reference<sup>14</sup> provide an empirical formula as follows:

$$n_1 = \begin{cases} m - 0.618 * (m - n), & n < m \\ n + 0.618 * (n - m), & n \geq m \end{cases}$$

n is node value input layer, m is node value of output layer

The choice of initial weight value is very important because the economic system is non-linear, that value for network have to reach partial smallest and convergence. An important request is that initial weight value can make each neuron's condition value approach to zero when input signals are accumulated. The initial weight value cannot take constant; only can take accidental value in certain scope, these values is between 0 and 1

#### f- Evaluation criteria

The most common error function minimized in neural networks is the sum of squared errors. Other error functions offered by software vendors include least absolute deviations, least fourth powers, asymmetric least squares, and percentage differences. These error functions may not be the final evaluation criteria since other common forecasting evaluation methods such as the mean square error (MSE) are typically not minimized in neural networks. To judge the network performance, regression analysis can be use between predict value and actual value.

#### g- Training Neural Network

The objective of training is to find the set of weights between the neurons that determine the global minimum of the error function. Unless the model is over fitted, this set of weights should provide good generalization. The back propagation network, applied in this work, uses the gradient descent

training algorithm which adjusts the weights to move down the steepest slope of the error surface

For training, the MATLAB neural network toolbox provide many training algorithms, the following are commonly used:

Gradient drop algorithm<sup>15</sup>, it's the most basic training algorithm, which revises each weight value in the gradient's dropping direction, and its characteristic is that convergence rate is very slow.

- 1- Gradient drop algorithm attached with momentum: based on the above algorithm, an additional value change in the last training process is added to each weight value change.
- 2- Gradient drop algorithm with auto-adapted learning rate: this algorithm can adjust learning rate according to error performance function, to solve the problem of improper learning rate choice basic BP algorithm, and it is a kind of gradient drop algorithm with dynamic learning rate.
- 3- Conjugate gradient algorithm: the MATLAB nerve network toolbox provide four kinds of conjugate gradient algorithms: *traincgf*, *traincgp*, *traincgb* and *traincsg*, the convergence rates of these algorithms are faster than that of basic gradient drop algorithms.
- 4- Levenberg-Marquardt optimization algorithm: This algorithm is the union of gradient drop algorithm and Gauss-Newton algorithm. Its learning rate is much big and its robustness is the best, but need much more memory.

#### h- Learning rate

A BP network is trained using a gradient descent algorithm which follows the contours of the error surface by always moving down the steepest slope. During training, a learning rate that is too high is revealed when the error function is changing wildly without showing a continued improvement. A very small learning rate also requires more training time. In either case, the research must adjust the learning rate during training or "brainwash" the network by randomizing all weights and changing the learning rate for the new run through the training set. Two empirical formulas are:  $\beta = \frac{1}{\sqrt{n_1}}$  and  $\beta = \frac{2}{n_1+1}$  in which  $n_1$  is the node value of hidden layer. It can be also use dynamic or auto-adapted learning rate.

#### Normalization of sample data

Before apply neural network model, the economic sample data need pre-treatment. The purpose is to make input and output data maintain at steep sector of the sigmoidal transfer function, strengthen data's effective recognition, and increase forecasting precision.

For input sample data  $X: x_1, x_2, \dots, x_i$ . The normalize  $X'$  is 
$$X'_i = \frac{X_i - \min(X)}{\max(X) - \min(X)}$$

After transformation, the data fall in [0,1].

## IV-EXPERIMENTAL RESULTS AND ANALYSIS

### A-Forecasting GDP and comparison between the predicted and the real value

This section shows how to use ANN to forecast GDP (Growth Domestic Product) of Benin National economic with input and output from 1992 to 2012. The original data was extracted from report (February 20014) of Benin Minister of Finance and National Institute of Statistic and Economy. The data is also available at ([www.insae.bj.org](http://www.insae.bj.org)). This paper use the variable from  $x_1$  to  $x_{16}$  as follow: the table 1 and GDP (Y) are output data of Benin national economy approved by Minister of Finance and International Monetary Fund.. It showed that these variables have serious co linearity and sequence relevance through regression analysis. So we cannot forecast with traditional method, but ANN model.

$X_i$	Variables (1.10 <sup>9</sup> CFA. [\$1 ≈ 478 CFA])
$X_1$	Agriculture
$X_2$	Rearing, animals
$X_3$	Fishing, forest
$X_4$	Industry attractive
$X_5$	Factory industry
$X_6$	Energy
$X_7$	BTP (Public building and worker sector)
$X_8$	Trade
$X_9$	Transport and Telecommunication
$X_{10}$	Bank, insurance
$X_{11}$	Another services
$X_{12}$	No commercial services
$X_{13}$	PISB
$X_{14}$	DTI and VAT
$X_{15}$	Export
$X_{16}$	Import
Y	GDP(output)

Table 1: Variables for economic forecasting

In this paper, we selected 23 groups of data. The first 21 groups data from 1992 to 2010 as training sample, the last 2 groups 2011 and 2012 as test sample. But first we normalized all data to centralize their value between [0, 1]. We designed an ANN model with three-layer, 16 input nodes and one output node. According to the formulas<sup>16</sup> of node value of hidden layer above, we can work out that the node value of hidden layer is 26. After training, we should simulate the data in order to examine the network's simulation performance. All the different value is showed on the following figures.

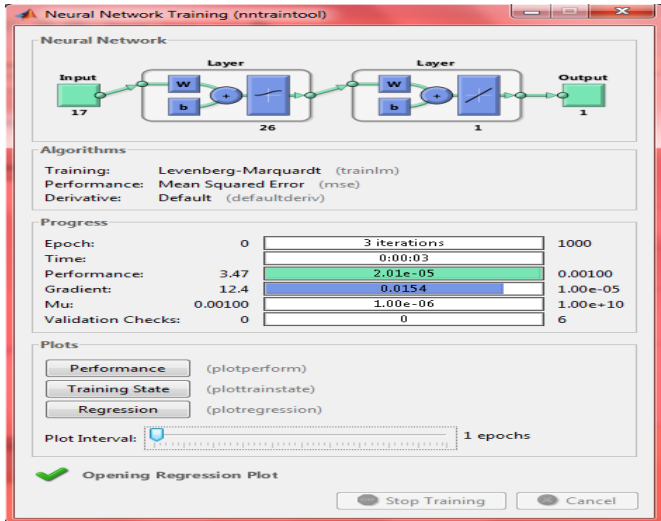


Figure 5: Training data on Matlab software

The results of the performance evaluation and the predicted value are showed in the different tables and figures.

The best performance of network is obtain at epoch= 3, the gradient error is 0,060431.

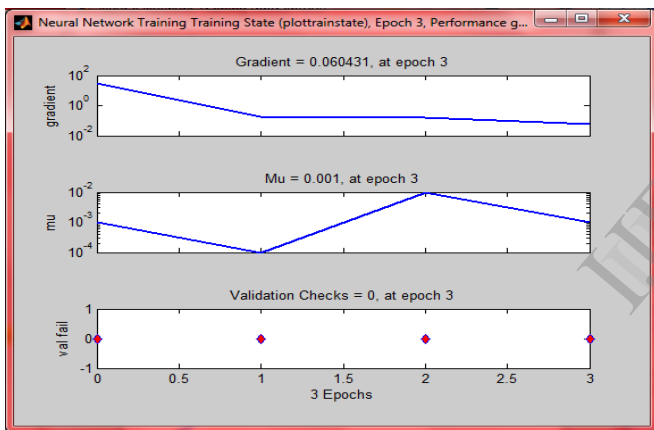


Figure 6: network training state.

The training is  $R=0,9999$ , there is high correlation between output and target.

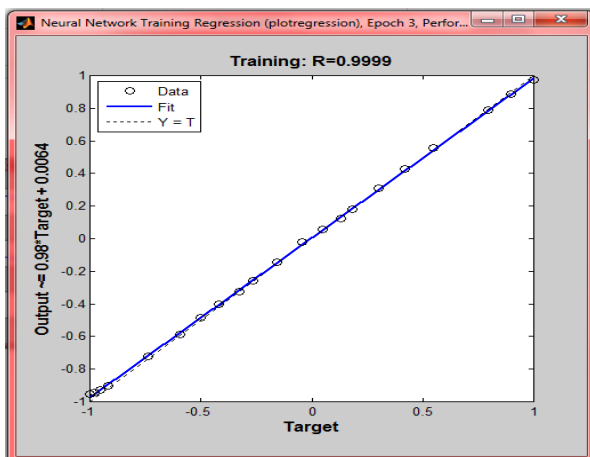


Figure 7: network training regression

The training performance is 0,00032433 at epoch=3 the Mean Square Error MSE= MSE = 597.3723 (figure 8)

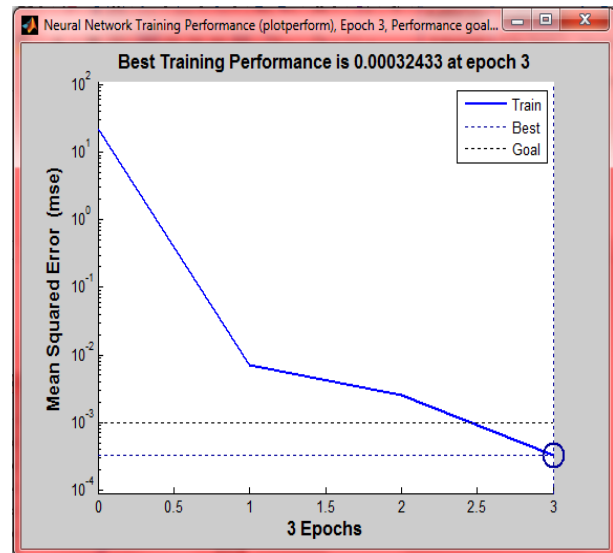


Figure 8: Training performance

The regression comparison between the target and prediction value is showed on figure 9.

The red curve is for prediction and the blue curve in the target.

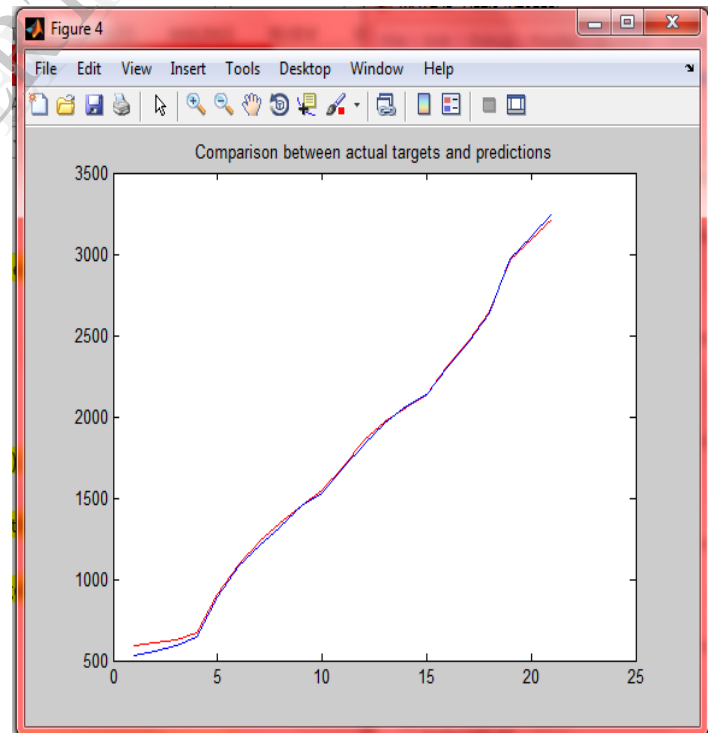


Figure 9: regression comparison between actual target and prediction.

The table below show the comparison between the real value and predicted value after network simulation for training data and sample data.

Year	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000
Simulate value	596.4	608.4	629.0	667.5	912.1	1091.6	1230.5	1343.1	1453.0	1544.1	1692.0
<b>Real value</b>	<b>533.6</b>	<b>560.4</b>	<b>594.4</b>	<b>644.1</b>	<b>887.3</b>	<b>1083.0</b>	<b>1207.8</b>	<b>1323.9</b>	<b>1448.4</b>	<b>1532.4</b>	<b>1679.6</b>

Year	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010
Simulate value	1859.6	1965.1	2056.5	2133.0	2309.4	2468.0	2646.0	2961.3	3090.8	3216.0
<b>Real value</b>	<b>1832.1</b>	<b>1956.9</b>	<b>2067.5</b>	<b>2140.0</b>	<b>2298.7</b>	<b>2460.2</b>	<b>2639.0</b>	<b>2970.5</b>	<b>3109.1</b>	<b>3247.9</b>

Table 2: comparison between the real GDP and Predicted GDP

From the table 2 we can see the simulation result and the real value are too close and very consistent. So, we can get the conclusion that the model based on neural network can be used in economic forecasting, and that is more effective.

#### B-Forecasting for the next 5 years GDP value

We used the NARX models specified above and the presented datasets to predict the next 5 years GDP of Benin economic. The NARX are implemented over the sets mentioned earlier. The 5-step-ahead forecasting is considered. The evaluation of different performance are employed as the forecasting accuracy measures .Figure 4 gives the result of model performance evaluation, we obtained the graphs listed in Figure 5 for the value of 5 step ahead forecasting. The result are summarize in the table4.

Parameters	Value
Performance	1.4814e+004
Train Performance	1.7674e+003
Val Performance	5.4519e+004
Test Performance	4.0342e+004
Closed Loop Performance	1.2359e+004
Early Predict Performance	1.4814e+004

Table 3: performance of 5-step ahead GDP prediction

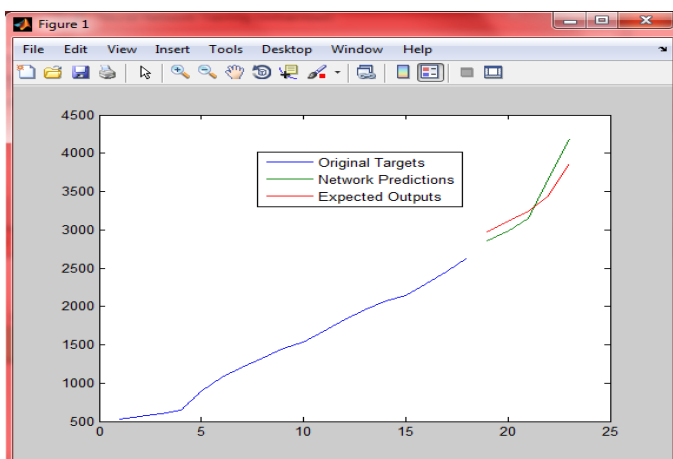


Figure 10: Prediction of 5 years ahead GDP value

Predicted period	Predicted value
Y+1(2008)	2.8557e+003
Y+2(2009)	2.9905e+003
Y+3(2010)	3.1531e+003
Y+4(2011)	3.6685e+003
Y+5(2012)	4.1895e+003

Table 4: predicted GDP value of next 5 years.

This last figure, as well as other resource measures done during training, validation and testing, showed that artificial neural network has high accuracy. In terms of performance, experiments showed that the difference between real value and the predicted is rather small, then excellent performance.

## CONCLUSION

In this paper, we combined Back propagation (BP) and Nonlinear Autoregressive models with exogenous input (NARX) to compare respectively the predicted and the next 5 years GDP value for economic growth forecasting. Through the establishment of a set of economic variable, we could evaluate the performance and accuracy of the model. After simulation on Matlab software, it had been proved that the method is effective and feasible with high accuracy. It is obvious that the neural network can solve nonlinear problem, the value predicted is relatively accurate and the error is small, therefore, the method is practicable and good for economic growth forecasting problem.

This paper also successfully implemented the real intelligent algorithm to reduce the influence of subjective factors and find important economic forecasting model. This algorithm can be applied to other field such investment risk prediction, business failure prediction, stock market forecasting and help financial, economic institute and government for decision making.

## REFERENCE

- [1] Office of the Chief Economist for the Africa region Arbache, Jorge, Del-n S. Go, and John Page. 2008. "Is Africa's Economy at a Turning Point?" In Africa at a Turning Point, edited by D. Go and J. Page. Washington DC: World Bank, pp. 13-85.
- [2] C.Z.He, Self -organisation data mining and economic forecasting, Scientific publishing House press, Beijing, China, 6-7, 2005

- [3] Gupta R, Kabundi A. 2010. Forecasting macroeconomic variables in a small open economy: A comparison between small- and large-scale models. *Journal of Forecasting* 29 168-185.
- [4] D.J Hand, H. Mannila, P.Smyth, Principle of data mining, Springer press, USA, 2002
- [5] Anderson and Rosenfeld, 1988; Hecht-Nielsen, 1990; Hertz et al., 1991; Hiemstra and Jones, 1994
- [6] Lawrence, 1991; Rumelhart and McClelland, 1986; Waite and Hardenbergh, 1998; Wasserman, 1993.
- [7] Y.S. Chen, G.P. Wang, and S.H. Dong, "Learning with progressive transductive support vector machine", *Pattern Recognition Letters*, 2003, Vol.24, No.12, pp.1845-1855.
- [8] Zhou qming. "The research of Application of Constructional Engineering Cost Estimation by Neural Network Ensemble". *Journal of Chongqing Jiaotong University*, vol. 24, 2005, pp.129-132
- [9] Yan Ping-fan, Zhang Chang-shui. Artificial neural network and Simulation calculation. Beijing: Tsinghua university press, 2000
- [10] Simon Haykin, *Neural Networks*, Second Edition, Pearson Education, 1999
- [11] Tsungnan Lin, Bill G. Horne, Peter Tino, C. Lee Giles, Learning long-term dependencies in NARX recurrent neural networks, *IEEE Transactions on Neural Networks*, Vol. 7, No. 6, 1996, pp. 1329-1351
- [12] Yang Gao, Meng Joo Er, NARMAX time series model prediction: feedforward and recurrent fuzzy neural network approaches, *Fuzzy Sets and Systems*, Vol. 150, No. 2, 2005, pp.331-350
- [13] D.P. Mandic, J.A. Chambers, *Recurrent Neural Networks for Prediction*, John Wiley & Sons, 200
- [14] H.B. Demuth, M. Beale, *Users' Guide for the Neural Network Toolbox for Matlab*, The Mathworks, Natick, MA, 1998
- [15] G.L.Su, F.P.Deng, BP neural network improvement algorithm based on MATLAB language, *Technical Notification*, vol 19, No 3, 132-134, Mar, 2003
- [16] Fecit technical research and development centre, neural network theory and MATLAB 7 realization, Electronics Industry Publishing House press, Beijing, China 2005