IUG Journal of Natural Studies
Peer-reviewed Journal of Islamic University-Gaza

ISSN 2409-4587

IUGNES
Vol. 25, No 2, 2017, pp 338-344

Accepted on (14-03-2017)

Comparative Approach of Box-Jenkins Models and Artificial Neural Network Models on Births per Month in Gaza Strip Using R

Suhaila A. Baker[1,*]
Bisher M. Iqelan[1]

[1]Department of Mathematics, Faculty of Science, Islamic University of Gaza, Gaza Strip, Palestine

*** Corresponding author**
e-mail address: biqelan@iugaza.edu.ps

Abstract

Comparative studies of different forecasting techniques can facilitate the selection of the best time series model for forecasting future expectations. In the present study, we address this problem by comparing the forecasting performance of the SARIMA model and four typical artificial neural networks, namely, MLP, ERNN, JRNN, and RBFNN in short-term forecasting for Births in Gaza Strip.

Analyses of Neural network models are done by the package (RSNNS) that is implemented in R program.

We conclude that forecasting with ANNs is accurate and more efficient than the SARIMA. In addition, the most accurate ANNs model among the four examined Neural Networks is RBFNN.

Keywords:
ARIMA,
MLP,
Elman Neural Network,
Jordan Neural Network,
RBF Neural Network,
RSNNS Package.

1. Introduction:

In recent years, there has been an increase in both applied and theoretical research in time series modeling and forecasting. Forecasting future events based on the basis of historical and present data dramatically captures people and specialists' attention. Several techniques have been developed to address this problem and predict the future behavior of a particular series of events.

One of the most important and widely used time series models is the autoregressive integrated moving average (ARIMA) model. The popularity of the ARIMA model exists due to its statistical properties as well as the well-known Box-Jenkins methodology in the model building process. Recently, Artificial Neural Networks (ANNs) have been extensively studied and used in time series forecasting. The major advantage of ANNs models is their flexible nonlinear modeling capability. With ANNs, there is no need to specify a particular model form. Rather, the model is adaptively formed based on the features presented from the data.

We use a data set about the number of births in Gaza strip which is registered by the Palestine health information center (PHIC) in the Ministry of Health (MOH) from January 2000 to December 2015. R-statistical software is used for fitting several Artificial Neural Networks (ANNs) models and (SARIMA) model.

2. Literature Review

Amin-Naseri Rostami Tabar (2008) proposed the use of Recurrent Neural Networks (RNN). The network is composed from four layers: an input layer, a hidden layer, a context layer and an output layer. Their study included real data sets of 30 types of spare parts from Arak petrochemical company in Iran and three performance measures:

Percentage Best (PB), Adjusted Mean Absolute Percentage Error (A-MAPE) and Mean Absolute Scaled Error (MASE). Their results show that NN can be used with promising results in comparison with traditional forecasting methods, like Moving Average (MA) and Single Exponential Smoothing (SES) (Amin- Naseri , Stadi and Tabar , 2007)

R. Samsudin, A. Shabri, and P. Saad (2010) studied the accuracy of the forecasts, in term of MAE has been compared with that of the same number of ANN model which are trained and applied to the same data set used by support vector machine (SVM). The ANN model obtained the best RMSE, MAE (Samsudin, Shabri and Saad ,2010).

R. Ramakrishna (2012) forecasted monthly electricity load in Andhra Pradesh, using Box-Jenkins methodology and feed forward neural networks. He concluded that neural networks model is the best to forecast the future values, because it has minimum measures of forecasting errors such as MAPE, RMSE and MAE (Ramakrishna, 2012).

Dr. Samir Khaled Safi (2013) proposed two efficient approaches of forecasting models. In the first model, Multilayer neural network is trained by minimizing RMSE. The second model consists of using ARIMA model on real data for electricity consumption in Gaza Strip. The results of both models reveal that ANNs outperform and offer consistent prediction performance compared to ARIMA model and hence preferable as a robust prediction model for electricity consumption (Safi, 2013).

3. Artificial Neural Networks (ANNs):

Artificial neural is an abstract computer model of the human brain. The human brain has units called neurons. These neurons are interconnected with links. The three essential features of an ANN are the basic processing elements referred to as neurons or nodes; the network architecture describing the connections between nodes; and the training algorithm used to find values of the network parameters for performing a particular task. An ANN consists of elementary processing elements (neurons), organized in Layers (see Figure 1).

The Layers between the input and the output layers are called "hidden". The neuron computes its output y as a certain function of net, $y = f(net)$. This function is called the activation (or sometimes transfer) function. For more details about activation function see (Ohno-Machado, 1996).

The number of input units is determined by the application. Types of ANNs and their uses are very high. Since the first neural model by McCulloch and Pitts (1943) (Ohno-Machado, 1996), hundreds of different models have been developed and considered as ANNs. The differences in them might be the functions, the accepted values, the topology, the learning algorithms, etc.

The structure of a feed-forward ANN is shown in Figure1.

The relationship between the input observations $(y_{t-1}, y_{t-2}, .., y_{t-p})$, and the output value y_t. Assuming a linear output neuron is given by

$$y_t = g\left(b_0 + \sum_{j=1}^{q} a_j f(w_{0j} + \sum_{i=1}^{p} w_{ij} y_{t-1}) \right)$$

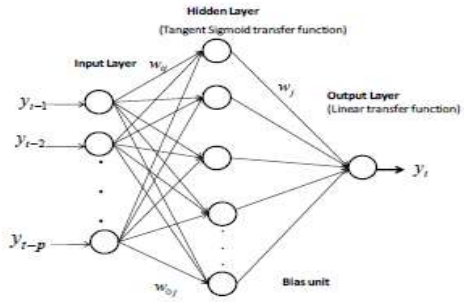

Figure 1 *Architecture of three layers feed-forward back-propagation ANN*

Where b_j $(j = 0,1,2,.....,q)$ is a bias on the j^{th} unit, and $w_{ij}(i = 1,2,...., p; j = 0,1,2,....., q)$ are connection weights, f and g are hidden and output layer activation functions, respectively (Zou, Xia,Yang and Wang , 2007). Several optimization algorithms can be used to train the ANN. Among the several training algorithms available, the back-propagation has been the most popular and most widely used, see (Heaton, 2005; Munakata, 2008). In a back-propagation network, the weights and bias values are initially chosen as random numbers and then fixed by the results of a training process. The goal of training algorithm is to minimize the global error.

A. Feed Forward Neural Network:

In this research we will examine two types of feed forward ANNs, Multilayer Perceptron ANN (MLP) and Radial Basis Function Neural Networks (RBFNN). MLP are constructed of multiple layers of computational units. Each MLP is composed of a minimum of three layers consisting of an input layer, one or more hidden layers and an output layer as shown in Figure 1.

The overall performance of the MLP is measured by the mean square error (MSE) expressed by:

$$MSE = \sum_{p=1}^{Nv} \sum_{i=1}^{M} \frac{(d(i) - y(i))^2}{N}$$

Note: $d(i)$ is the desired output, $y(i)$ is the actual output, N_v is a set of training patterns, where p represents the pattern number. M -dimensional output vector from the trained network for the p^{th} pattern, N -dimensional input vector of the p^{th} training pattern.

Another feed forward ANN is Radial basis function Neural Network (RBFNN). In RBFNN an input vector is placed into each node of the hidden layer, and each node calculates the distance from the input vector to its own center as shown in Figure 2. The resulting distance value is transformed via the Gaussian function, and output from each node. The output value from the hidden layer is multiplied by weighting values. The product is placed into the output layer node, which sums all the products. The output of RBFNN with the Gaussian basis functions is

$$O = \sum w_i \exp(-\frac{1}{2\sigma^2} \|x_p - c_i\|)$$

where x_p is the p^{th} input sample of the network, p is the number of the radial basis functions, j is the connection weight between the j^{th} nodes to the output node. c_i The center of the basis functions, σ the spread of the radial basis functions is the output of the network (Wedding and Cios, 1996).

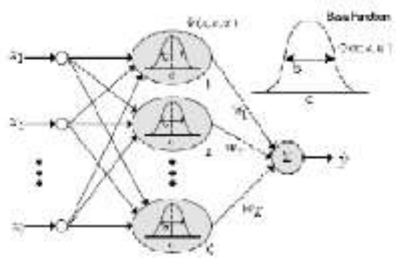

Figure 2 *RBFNN*

B. Recurrent Neural Network:

This neural network contains connected units in which every unit is both input and output unit.

The architecture of this model ranges from partially to fully interconnected network. There are two types of (RNN): Jordan Recurrent Neural Network (JRNN), and Elman Recurrent Neural Network (ERNN). Compared to MLP, ERNN and JRNN have a context layer that can send the feedback from the output connections to the hidden layer shown in Figure 3 (Jayawardena; Fernando, 1998). Jordan network is similar to Elman network. The only difference is that context units are fed from the output layer instead of the hidden layer. The dynamics of ERNN showing in Figure 3 are described by the following equations

$$y(k) = g(w_3 x(k))$$
$$x(k) = f(w_1 x_c(k) + w_2(u(k-1)))$$
$$x_c = x(k-1)$$

Where: $u(k-1)$ and $y(k)$ are the input and output of the network, respectively, at a discrete time k, $x_c(k)$ and $x(k)$ are the nodes of the context and the hidden layers, respectively, and w_1, w_2, w_3 are the weight matrices for the context-hidden, input-hidden, and the hidden output layers, respectively.

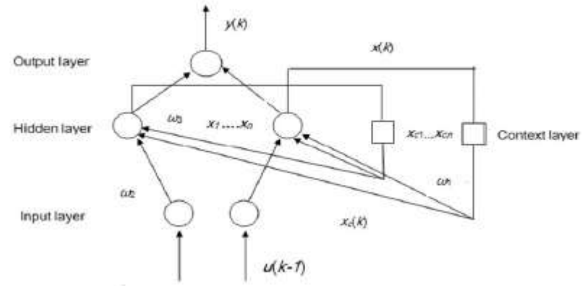

Figure 3 *Elman Recurrent Neural Network*

SARIMA Models

The process $\{x_t\}$ is called an autoregressive moving average (ARMA) process, denoted by ARMA (p, q) is given by

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + ... + \phi_p x_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - ... - \theta_q \varepsilon_{t-q}, (1)$$

or simply

$$\phi_p(B) x_t = \theta_q(B) \varepsilon_t,$$

where

$$\phi_p(B) = 1 - \phi_1 B - \phi_2 B^2 - .... - \phi_p B^p$$

is an AR polynomial of B for order p.

Also,

$$\theta_q(B) = 1 - \theta_1 B - \theta_2 B^2 - .... - \theta_q B^q$$

is a MA polynomial of B for order q.

In the above, B is the backshift operator, used to simplify the representation of lag values, by $Bx_t = x_{t-1}$.

A generalization of ARMA model, to cover a wide class of non-stationary time series, is achieved by proposing “differencing” in the model. A non-stationary time series $\{x_t\}$ is said to follow a non-stationary autoregressive integrated moving average (ARIMA) denoted by ARIMA (p, d, q) if it is expressed as:

$$\phi_p(B)\nabla^d x_t = \mu - \theta_q(B)\varepsilon_t, \qquad (2)$$

where ε_t are identically and independently distributed as $N(0,\sigma^2)$, $t = 1, 2, \ldots, N$ and N is the number of samples, d is the order of non-seasonal differences and ∇ is the non-seasonal differencing operator, $\nabla = 1 - B$, μ is the mean of a series assuming that after differencing is stationary. One can notice that when $d = 0$, the ARIMA (p, d, q) model becomes an ARMA (p, q) model. More details can be found in (Iqelan, 2015; Brockwell and Davis, 1991; Shumway and Stoffer, 2011; Cryer and Chan, 2008).

The ARIMA model is for non-stationary non-seasonal observations. Box and Jenkins in (Cryer, and Chan, 2008) popularized this model to deal with seasonality. For time series possessing a seasonal component that repeats every s observations, their proposed model is known as the seasonal ARIMA model.

The seasonal autoregressive integrated moving average model is given by

$$\phi_P(B^s)\phi_p(B)\nabla_s^{\,d}\nabla^d x_t = \mu + \theta_Q(B^s)\theta_q(B)\varepsilon_t \qquad (3)$$

and is denoted by SARIMA(p, d, q) (P,D,Q)$_s$, where D is the order of seasonal differences and ∇s is the seasonal differencing operator, $\nabla s = 1 - B^s$ .For monthly time series s=12 and for quarterly time series s=4 .It is possible that several SARIMA models may be identified, and the selection of an optimum model is necessary. Such selection of models is usually based on the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC), also known as Schwarzs Information Criterion. It is defined respectively as follows:

$$AIC = -2\log(\max imumlikelihood) + 2k$$

$$BIC = -2\log(\max imumlikelihood) + k\log(n)$$

The corrected form is defined as

$$AICc = AIC + 2\frac{(k+1)(k+2)}{(n\text{-}k\text{-}1)}$$

where k is the number of free parameters ($k = p + q + P + Q$) and n is the number of residuals that can be computed for the time series. For more details about the maximum likelihood see (Cryer, and Chan, 2008). The choice of each parameter calls for a minimization of the AIC, AICc and BIC.

Performance tests:

The forecast models are predestined in terms of their capability to predict the future values. In order to compare forecasting performance of different models, many statistical measures can be used for this purpose. The most widely reported error measure is RMSE. The other frequently used measures are MAE and MAPE. Smaller values indicate better model performance. Both of MAE and RMSE together can be used to diagnose the variation in the errors of predicted values. Whenever value of MAE is less than RMSE, there is a variation in the errors. The three prediction error estimators are defined as follows (see (Iqelan, 2015; Shumway and Stoffer, 2011):

• Mean Absolute Error (MAE):

The MAE is used to measure how close forecasts or predictions are to the actual data and is given by:

$$MAE = \frac{1}{n}\sum_{t=1}^{n}\left|X_t - \bar{X}_t\right|$$

• Root Mean Square Error (RMSE):

The RMSE is a quadratic formula which measures the differences between values predicted by hypothetical model and observed values and is given by:

$$RMSE = \sqrt{\frac{1}{n}(X_t - \bar{X}_t)^2}$$

• Mean absolute percentage error (MAPE):

The MAPE measures the relative dispersion of forecast errors and is defined by the following formula:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{X_t - \bar{X}_t}{X_t}\right| * 100$$

4. Computational Results:

The observations in this research are about number of births in Gaza Strip, Table 1 represents some descriptive statistics of the time series. The plot of the collected data from January 2000 to December 2014 is shown in Figure 4. Observations of January 2015 to December 2015 are considered for predicted Births.

Table 1 *Descriptive statistics*

Statistics	Obs.	Min.	Median	Mean	Max.
Births	180	1799	3936	3966	5338

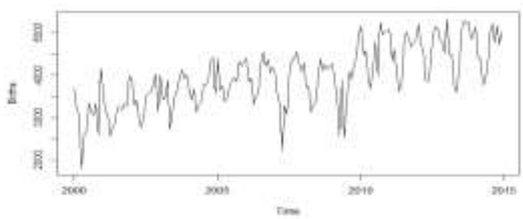

Figure 4 *Time series plot of Births in Gaza Strip*

The data are clearly non-stationary as the series wanders up and down for long periods. Consequently, we will take a first difference of the data. The differenced data are shown in Figure 5, these look stationary.

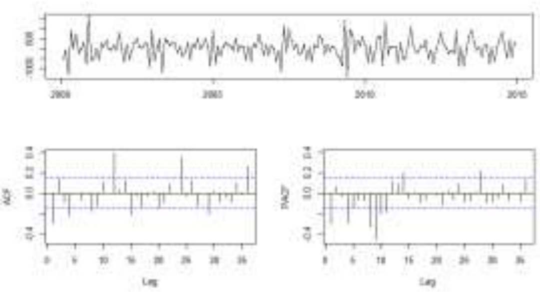

Figure 5 *ACF and PACF of first difference of Births time series*

Fitting of ARIMA Model:

Our aim now is to find an appropriate SARIMA Model based in the ACF and PACF shown in Figure 5. In the non-seasonal lags, there are three significant spikes at lag 1, 4, 8 in the PACF suggesting a possible AR (3) term. The significant spikes at lag 1 and lag 4 in the ACF suggest a non-seasonal MA (2) component, for the seasonal behavior, we check the situation on around lags 12, 24 and so on. In the ACF, there is a collection of positive spikes around lag 12 and 24. In the PACF there is no clear spike at lag 12.

Consequently, this primary analysis proposes that the advisable model for these data is an SARIMA (3,1,2)(0,0,1). We fit this model, along with some variations on it, and compute their AICc values which are shown in Table 2.

Table 2 *AICc Values of Suggested ARIMA Models*

	AIC	AICc	BIC
SARIMA(0,1,1)(0,0,1)	2681.85	2681.99	2691.41
SARIMA(1,1,1)(1,0,1)	2608.84	2609.19	2624.18
SARIMA(2,1,2)(1,0,1)	2607.32	2607.98	2629.63
SARIMA(3,1,2)(0,0,1)	2679.55	2680.21	2701.86
SARIMA(3,1,2)(1,0,1)	2603.82	2604.67	2629.32
SARIMA(3,1,2)(2,0,1)	2600.32	2601.39	2629.01
SARIMA(3,1,2)(0,0,2)	2641.35	2642.2	2666.85

Suggestion-SARIMA (3,1,2)(2,0,1) based on the smallest values of AIC, AICc, and BIC among the other SARIMA choices. The p value of the Liunge-Box test is 0.4508, which is greater than 0.05, then the residuals are independent which we want for the model to be correct.

All information criteria prefer the SARIMA (3, 1, 2)(2, 0, 1) model, which is the model displayed in Equation 3. The fitted model in this case as in Equation 3 is

$$(1+0.5668B-0.3838B^2-0.3064B^3)(1-1.2242B^{12}$$
$$+0.2245B^{24})=c+(1+0.1632B+0.7731B^2)(1+0.9753B^{12})\varepsilon_t$$

The Predicted values of the best model in Box-Jenkins modeling will later be compared with the four Artificial Neural network models.

Fitting of Artificial Neural Network Models:

An efficient preprocessing of the data is necessary to input it into the net. The simulation processing is realized by R-program.

Preparing data:

Due to the nature of activation functions, scaling the network inputs and outputs to an appropriate range usually between [0,1] by applying normalization. Then we split it into training set (a sit of samples used to adjust or train the weights in the neural network to produce the desired outcome) and to testing set (a set used to determine the generalization error of the final chosen model) (Priddy and Kellerem, 2005). The Caret Package in R does the splitting of the data. Partitioning the data into several partitions have been done with several ratios as seen in Table 4.

One of the most important characteristics of a perceptron network is the number of neurons in the hidden layer(s). If an inadequate number of neurons are used, the network will be unable to model complex data, and the resulting fit will be poor, for more details see (Heaton, 2005). Analysis of Neural network models by the package RSNNS (Bergmeir and Benitez, 2012) that implements an R (R Development Core Team 2011) interface to the Stuttgart Neural Network Simulator (Zell, 1998).

The performance of two feed forward (multilayer perceptron and radial basis function), and two recurrent (Elman and Jordan) ANNs is analyzed. We use the data of births from 2000 to 2014 for training and testing and the data of births collected in year 2015 for comparison. All the models are with learning rate=0.1 and iteration=100, the back-propagation algorithm is used as the learning process for all ANN studied models (Bergmeir and Benitez, 2012). We try several numbers of neurons (5-10-20) in the hidden layer and estimate the MSE, RMSE, MAE, MAPE for each model. Also we try the different partitions for the training and testing set, Table 3 shows the effect of partitioning on the value of RMSE and MAPE for each Neural Network model considered in this research.

Table 3 *Pertationing and Performance test for ANN Models*

Model	Neurons	Tests	Training – Testing Ratio		
			70-30%	80-20%	90-10%
MLP	5	RMSE MAPE	0.32 55.17	0.33 52.12	0.32 54.02

	10	RMSE	0.32	0.33	0.33
		MAPE	55.07	52.57	54.38
	20	RMSE	0.32	0.33	0.32
		MAPE	55.29	52.48	53.71
RBF	5	RMSE	0.30	1.23	0.14
		MAPE	62.31	626.86	42.72
	10	RMSE	5.7	0.87	0.37
		MAPE	91.30	230.95	68.26
ERNN	5	RMSE	0.33	0.35	0.24
		MAPE	39.90	34.82	47.83
	10	RMSE	0.16	0.18	0.16
		MAPE	29.14	34.53	30.90
JRNN	5	RMSE	0.28	0.29	0.25
		MAPE	43.47	42.67	44.24
	10	RMSE	0.25	0.27	0.21
		MAPE	39.80	38.45	39.62
	20	RMSE	0.38	0.39	**0.19**
		MAPE	38.24	34.30	**37.17**

Results of training the Neural Network models using RSNNS package are as follows:

- **MLP Model:**

The model consists of 3 layers, the first layer is the input layer which has 15 nodes as the input for the entire network, the second layer is the hidden layer which consists of many neurons, and the third layer is the output which is only one output. From Table 3, MLP 15-20-1 model with the mini-mum error is the most accurate MLP model.

- **RBFNN Model:**

RBFNN model contains 15 inputs and 1 output that has only one hidden layer with many neurons. For a higher number of neurons, such as 20 neurons in the hidden layer, the response of the model is not stable.In addition, the result of the error is high. Table 3 shows the best model architecture which is RBFNN 15-5-1.

- **ERNN Model:**

Our Elman model has three layers, the input layer with 15 inputs and one hidden layer with many neurons, number of context units equal to the number of neurons in the hidden layer, the activation function is Logistic function and one output in the output layer. From Table 3 the best Elman architecture model with the lowest forecast performance error is ERNN 15-10-1 with 10 context units. For neurons up to 20 and 25, the performance of the model does not increase.

- **JRNN Model:**

Jordan model is the same as Elman; the only difference is that context units are fed from the output layer instead of the hidden layer. Jordan architecture is explained as follows, there are 15 inputs and only one output. The number of the context units is the same as the numbers of the output units which make us have only one context unit. JRNN 15-20-1 is the best.

Comparison between ARIMA and ANNs models:

Firstly, before we made the comparison, de-normalization of the data is required to find out the forecast error measure for each Neural Network we analyzed, and compare it to our proposed SARIMA model as listed in Table 4.

Table 4 *Comparison the performance of ANN models and SARIMA*

Model	MAE	RMSE	MAPE
SARIMA(3,1,2)(2,0,1)	508.94	595.07	10.65
MLP(13-20-)	357.09	378.67	7.82
ERNN(13-10-1)	137.56	182.15	3.04
JRNN(13-20-1)	202.37	231.01	4.45
RBFNN(13-5-1)	132.32	162.05	3.01

From the above table it seems that all ANN models outperform SARIMA performance, and the most accurate model is RBFNN (13-5-1), also, ERNN is more accurate than JRNN, and JRNN outperforms MLP neural network model.

Finally, the predicted value for Births per month using Box-Jenkins SARIMA Model, and Neural network model is compared with actual value observed in year 2015 as illustrated in Table 5.

Table 5 *SARIMA and RBFNN predicted values versus actual*

Months	**Actual**	**SARIMA (3,1,2)(2,0,1)**	**RBF (15-5-1)**
Jan 2015	4955	5225.357	4834.35
Feb2015	4195	4551.144	4396.94
Mar2015	4324	4677.511	4548.09
Apr2015	4676	3994.279	4647.84
May2015	3901	4188.241	3777.72
Jun2015	4900	4325.282	4832.54
July2015	5078	4979.170	5003.26
Aug2015	4262	5115.300	4317.27
Sep2015	4998	5002.300	4751.22
Oct2015	4283	5214.722	4404.55
Nov2015	4089	4993.258	4071.32
Dec2015	4372	5171.558	4044.67

Conclusion:

The forecasting of monthly Births in Gaza-Strip, with Artificial Neural Network is more efficient than the Box-Jenkins methods.

The most accurate ANNs among the four Neural Networks we examined in this research was Radial Basis Function Neural Network (RBFNN) which used the radial function (Gaussian) as its activation function with the lowest error. R-programming is well appropriate for the modeling and predicting of temperature data in this case.

References:

Amin-Naseri M.R, Tabar B.Rostami and Stadi B.O, "Generalized regression neural network in modeling lumpy demand." Presented at the 2007, 8th international conference on operations and quantitative management, Bangkok, Thailand

Bergmeir C, Benıtez JM (2012). Neural Networks in R Using the Stuttgart Neural Network Simulator: RSNNS. R package version 0.4-3, URL http://CRAN.R-Project.org/ package=RSNNS.

Brockwell, P. J. and Davis, R. A (1991). Time Series: Theory and Methods. 2nd Edition. New York: Springer Series in Statistics.

Cryer, J. D. and Chan, K (2008). Time Series Analysis With Applications in R. 2nd Edition.New York: Springer.

Heaton (2005), Introduction to Neural Networks for Java, Second Edition

Iqelan Bisher (2015) , Time Series Modeling of Monthly Temperature Dataof Jerusalem/Palestine, MATEMATIKA, UTM Centre for Industrial and Applied Mathematics Volume 31, Number 2, 159-176.Issue 1,Jan-Feb 2012, pp.1108-1115.

Jayawardena, A. W. & Fernando, D. A. K. (1998), Use of radial basis function type artificial neural networks for runoff simulation, Computer-Aided Civil and Infrastructure Engineering, 13 (2), 91–9

Munakata Toshinori (2008), Fundamentals of the New Artificial Intelligence,2th Edition.

Ohno-Machado Lucila (1996) , Medical Applications Of Artificial Neural Networks Connectionist Models Of Survival , March.

Priddy Kevin L. , Kellerem Paul E. (2005),Artificial Neural Networks: An Introduction.

Ramakrishna R.,Neural Networks Forecasting Model For Monthly Electricity Load In Andhra Pradesh , International Journal of Engineering Research and Applications(IJERA) ISSN: 2248-9622, Vol. 2,

Safi Samir ,Artificial Neural Networks Approach to Time Series Forecastingfor Electricity Consumption in Gaza Strip, IUG Journal of Natural and Engineering Studies Vol.21, No.2, pp 1-22 2013, ISSN 17266807 .

Samsudin R., Shabri A., Saad P. (2010),A Comparison of Time Series Forecasting Using Support Vector Machine and Artificial Neural Network,Journal of Applie Sciences10(10):950-958.

Shumway, R. H. and Stoffer D. S (2011). Time Series Analysis and Its Applications. NewYork: Springer..

Wedding Ii DK , Cios KJ (1996) , Time series forecasting by combining RBF networks, certainty factors, and the Box-Jenkins model. Neurocomputing 10: 149168.

Zell A, et al. SNNS Stuttgart Neural Network Simulator User Manual, Version 4.2, IPVR, University of StuttgartandWSI, University of Tubingen,1998.

Zou H.F., Xia G.P.,Yang F.T., Wang H.Y.(2007), An investigation and comparison of artificial neural network and time series models for chinese food grain price forecasting,Neurocomputing, 70 , 2913-29

دراسةُ مقارنة لنماذج بوكس –جنكنز ونماذج الشبكة العصبية الاصطناعية على مواليد قطاع غزة بإستخدام برنامج R

الدراسات المقارنة التي تقارن تقنيات التنبؤ المختلفة من الممكن أن تساهم في اختيار أفضل نماذج للسلاسل الزمنية التي يمكن أن تستخدم في التنبؤ بالتوقعات المستقبلية. هذه الدراسة تستحضر هذه المشكلة من خلال دراسة التوقعات المستقبلية لأعداد المواليد في قطاع غزة على المدى القريب باستخدام نموذج الانحدار الذاتي و المتوسطات المتحركة الموسمية (SARIMA) ومقارنة أداء النموذج بالنسبة لأداء أربع نماذج للشبكات العصبية الصناعية (ANN) وهي MLP, ERNN, JRNN RBFNN , تم تحليل نماذج الشبكات العصبية الصناعية بواسطة حقبة RSNNS وهو تطبيق في برنامج R الاحصائي. توصلت الدراسة إلى أن التنبؤ بواسطة نماذج الشبكات العصبية الصناعية أكثر دقة وأكثر كفاءة من برنامج SARIMA. وخلصت الدراسة كذلك إلى أن أكثر نماذج الشبكات العصبية دقة هو نموذج RBFNN.

كلمات مفتاحية:

نموذج الانحدار الذاتي و المتوسطات المتحركةARIMA
الشبكة العصبية متعددة الطبقات MLP
الشبكة العصبية التكرارية إليمانERNN,
الشبكة العصبية التكرارية جوردن ERNN
شبكة دالة القاعدة النصف قطريةRBFNN ,
RSNNS.