TIME SERIES MODELING AND PREDICTION: A REVIEW

[1*]HOWARD, C.C., [2]HOWARD, I. C. AND [1]ETUK, E. H.
[1]Department of Mathematics/Statistics, Faculty of Science, Rivers State University Port Harcourt, Nigeria. [2] Department of Chemistry/Biochemistry Federal Polytechnic, Nekede, Owerri. Imo State, Nigeria

Abstract

Time series modeling and prediction has fundamental importance to various practical domains. Thus a lot of active research works is going on in this subject in recent times. Many important models have been proposed in literature for improving the accuracy and efficiency of time series modeling and prediction. The aim of this paper is to present a concise description of some popular time series modeling and prediction using stochastic models, artificial neural network model (ANN) and hybrid stochastic and ANN models with their salient features.

[redacted]

KEYWORDS: Time series, Model, Prediction, Accuracy, Hybrid

Introduction

Time series data is an ordered sequence of values of a variable at equally spaced time intervals (Yan and Zou 2013, YanAshwini and Bhavya, 2013, Farhath *et al.*, 2016, Doulah, 2019). In simple terms, a time series is simply a sequence of data (usually numbers/images etc.) collected at regular intervals over a period of time. One of the main characteristic of time series is that it is stochastic in nature. A stochastic process is a sequence of random variables, all defined on the same probability space with discrete time parameter (usually T), which depends on both chance and time.

Time series modeling is a dynamic research area which has attracted attentions of researchers' community over last few decades. The main aim of time series modeling is to carefully collect and rigorously study the past observations of a time series to develop an appropriate model which describes the inherent structure of the series. This model is then used to generate future values for the series, i.e. to

INTERNATIONAL JOURNAL – ASD
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

make forecasts. Time series forecasting thus can be termed as the act of predicting the future by understanding the past (Raicharoen, *et. al*., 2003, Wang *et al*., 2013, Doulah, 2019). Due to the indispensable importance of time series forecasting in numerous practical fields such as business, economics, finance, science and engineering, etc. (Tong, 1983, Zhang, 2003, Zhang, 2007, Wang *et al*., 2013, Adhikari and Agrawal, 2013) proper care should be taken to fit an adequate model to the underlying time series. It is obvious that a successful time series forecasting depends on an appropriate model fitting. A lot of efforts have been done by researchers over many years for the development of efficient models to improve the forecasting accuracy. As a result, various important time series forecasting models have been evolved in literature.

One of the most popular and frequently used stochastic time series models is the *Autoregressive Integrated Moving Average (ARIMA)* (Box and Jenkins, 1970; Hipel and Mcleod, 1994; John 1997; Zhang, 2003;) model. The basic assumption made to implement this model is that the considered time series is linear and follows a particular known statistical distribution, such as the normal distribution. ARIMA model has subclasses of other models, such as the *Autoregressive (AR)* (Box and Jenkins 1970; Hipel and Mcleod,1994; Lee, 2000) *Moving Average (MA)* (Hipel and Mcleod, 1994; Box and Jenkins, 1970) and *Autoregressive Moving Average (ARMA)* (Box and Jenkins, 1970; Hipel and Mcleod, 1994; John, 1997) models. For seasonal time series forecasting, Box and Jenkins, (1970) had proposed a quite successful variation of ARIMA model, viz. the *Seasonal ARIMA (SARIMA)* (Hamzacebi, 2008; Box and Jenkins, 1970; Hipel and Mcleod, 1994). The popularity of the ARIMA model is mainly due to its flexibility to represent several varieties of time series with simplicity as well as the associated Box-Jenkins methodology (Box and Jenkins, 1970; Hipel and Mcloed, 1994; Zhang, 2003; Hamzacebi, 2008), for optimal model building process. But the severe limitation of these models is the pre-assumed linear form of the associated time series which becomes inadequate in many practical situations. To overcome this drawback, various non-linear stochastic models have been proposed in literature (Parrelli, 2001; Zhang, 2003; Zhang, 2007; Wang *et al*., 2013) however from implementation point of view these are not so straight-forward and simple as the ARIMA models.

Recently, artificial neural networks (ANNs) have attracted increasing attentions in the domain of time series forecasting (Hipel and Mcleod, 1994; Kihoro *et.al*., 2000; Zhang, 2007). Although initially biologically inspired, but later on ANNs have been successfully applied in many different areas, especially for forecasting and classification purposes (Kihoro *et. al*., 2000; Joarder *et. al*., 2002). The excellent feature of ANNs, when applied to time series forecasting problems is their inherent capability of non-linear modeling, without any assumption about the statistical distribution followed by the observations. The appropriate model is adaptively formed based on the given data. Due to this reason, ANNs are data-driven and self-adaptive by nature. During the past few years a substantial amount of research works have been carried out towards the application of neural networks for time series modeling and forecasting. A state-of-the-art discussion about the recent works in neural networks for tine series forecasting has been presented by (Zhang *et. al.*, 1998). There are various ANN forecasting models in literature. The most common and popular among them are the multi-layer perceptrons (MLPs), which are characterized by a single hidden layer *Feed Forward Network (FNN)* (Zhang *et. al*., 1998; Zhang, 2003). Another widely used variation of FNN is the *Time Lagged Neural Network (TLNN)* (Faraway and Chartfield, 1998; Kihoro *et. al*., 2000). Hamzacebi, (2008) had presented a new ANN model, viz. the *Seasonal Artificial Neural Network (SANN)* model for seasonal time series forecasting. His proposed model is surprisingly simple and also has been experimentally verified to be quite successful and efficient in forecasting seasonal time series. Of course, there are many other existing neural network structures in literature due to the continuous ongoing research works in this field. However, in the present paper we shall mainly concentrate on the above mentioned ANN forecasting models.

Over the years, several forecasting methods have been proposed with the aim of producing increasingly accurate predictions of (stochastic) time series. In general, they could be split into two great exclusive categories: the individual (or single) predictive methods, such as the well-known *Auto-Regressive Integrated Moving Average* (ARIMA) models, (Hamilton, 1994); and the combination of individual predictive methods, proposed initially by (Bates and Granger, 1969). Indeed, the collection of single predictive methods might yet be regrouped into two exclusive classes: the statistical one (here it lies, for instance, the (linear) ARIMA models), and the machine learning one (here it lies the (non-linear) *Artificial Neural Networks* (ANNs), as in (Haykin, 2001). By hybrid forecasting methods, it means those ones that always carry out the modeling of a given time series, denoted by y_t (t = 1,… .,T), according to the following three steps:

Step 1, a single forecasting method from statistical class/machine learning class is applied to y_t (t =

INTERNATIONAL JOURNAL – ASD
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

1,….,T) for producing its forecasts as well as its residuals;
Step 2, the forecasting errors generated in Step 1 are predicted by using an individual forecaster from machine learning class/statistical class; and,
Step 3, the forecasts provided in Step 1 are "corrected" by the predictions of the residuals produced in Step 2 such that to generate the hybrid forecasts of the underlying time series y_t (t = 1,…..,T). In effect, a hybrid predictive method can be referred to as particular case of combined forecasters.
Hybrid techniques that decompose a time series into its linear and nonlinear form are one of the most popular hybrid models, which have recently been shown to be successful for single models (Khashei and Bijari 2011a, Sharma, 2012, Yan and Zou 2013, Suresh *et al*., 2017). The linear ARIMA and the nonlinear multilayer perceptions are jointly used in these hybrid models in order to capture different forms of relationship in the time series data. The motivation of these hybrid models come from the following perspectives. First, it is often difficult in practice to determine whether a time series under study is generated from a linear or nonlinear underlying process; thus, the problem of model selection can be eased by combining linear ARIMA and nonlinear ANN models (Khashei and Bijari 2011b, Sharma, 2012, Wang *et al*., 2013). Second, real-world time series are rarely pure linear or nonlinear and often contain both linear and nonlinear patterns, which neither ARIMA nor ANN models alone can be adequate for modeling in such cases; hence the problem of modeling the combined linear and nonlinear autocorrelation structures in time series can be solved by combining linear ARIMA and nonlinear ANN models. Third, it is almost universally agreed in the forecasting literature that no single model is the best in every situation, due to the fact that a real-world problem is often complex in nature and any single model may not be able to capture different patterns equally well. Therefore, the chance in order to capture different patterns in the data can be increased by combining different models (Zhang, 2003, Suhartono and Guritno 2005, Yan and Zou 2013).

Time series predictions using stochastic models

The selection of a proper model is extremely important as it reflects the underlying structure of the series and this fitted model in turn is used for future forecasting. A time series model is said to be linear or non-linear depending on whether the current value of the series is a linear or non-linear function of past observations.
In general models for time series data can have many forms and represent different stochastic processes (Doulah 2019). There are two widely used linear

INTERNATIONAL JOURNAL – ASD
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

time series models in literature, viz. *Autoregressive (AR)* (Box and Jenkins, 1970; Hipel and Mcleod, 1994; Lee, 2000) and *Moving Average (MA)* (Box and Jenkins, 1970., Hipel and Mcloed, 1970) models. Combining these two, the *Autoregressive Moving Average (ARMA)* (Box and Jenkins, 1970., Hipel and Mcleod, 1994., Lee, 2000., John, 2000) and *Autoregressive Integrated Moving Average (ARIMA)* (Box and Jenkins, 1970., Hipel and Mcleod, 1994., John, 1997) models have been proposed in literature. The *Autoregressive Fractionally Integrated Moving Average (ARFIMA)* (Park, 1999, Galbraith and Zinde, 2001) model generalizes ARMA and ARIMA models. For seasonal time series forecasting, a variation of ARIMA, viz. the *Seasonal Autoregressive Integrated Moving Average (SARIMA)* (Box and Jenkins, 1970., Hipel and Mcleod, 1994., Hamzacebi, 2008) model is used. ARIMA model and its different variations are based on the famous Box-Jenkins principle (Box and Jenkins, 1970., Hipel and Mcleod, 1994., Lee, 2000., Zhang, 2003) and so these are also broadly known as the Box-Jenkins models. Linear models have drawn much attention due to their relative simplicity in understanding and implementation. However many practical time series show non-linear patterns. For example, as mentioned by (Parrelli, 2001), non-linear models are appropriate for predicting volatility changes in economic and financial time series. Considering these facts, various nonlinear models have been suggested in literature. Some of them are the famous *Autoregressive Conditional Heteroskedasticity (ARCH)* (Park, 1999., Parrelli, 2001), model and its variations like *Generalized ARCH (GARCH)* (Park, 1999., Parrelli, 2001), *Exponential Generalized ARCH (EGARCH)* etc., the *Threshold Autoregressive (TAR)* (Tong, 1983., Zhang, 2003) model, the *Non-linear Autoregressive (NAR)* (Zhang, 2007) model, the *Nonlinear Moving Average (NMA) (Parelli, 2001)* model, etc.

The Autoregressive Moving Average (ARMA) Models

An ARMA(*p, q*) model is a combination of AR(*p*) and MA(*q*) models and is suitable for univariate time series modeling. In an AR(*p*) model the future value of a variable is assumed to be a linear combination of *p* past observations and a random error together with a constant term. Mathematically the AR(*p*) model can be expressed as(Hipel and Mcleod, 1994., Lee, 2000).

$$y_t = c + \sum_{i=1}^{p} \varphi_t y_{t-1} + \varepsilon_t = c + \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + \dots\dots\dots.. + \varphi_p y_{t-p} + \varepsilon_t \qquad (2.1)$$

Here y_t and ε_t are respectively the actual value and random error (or random shock) at time period *t* , φ_i (*i*= 1,2,..., *p*) are model parameters and *c* is a constant.

The integer constant *p* is known as the order of the model. Sometimes the constant term is omitted for simplicity. Usually for estimating parameters of an AR process using the given time series, the Yule-Walker equations (Hipel and Mcleod, 1994) are used.

Just as an AR(*p*) model regress against past values of the series, an MA(*q*) model uses past errors as the explanatory variables. The MA(*q*) model is given by (Hipel and Mcleod, 1994., John, 1997., Lee, 2000):

$$y_t = \mu + \sum_{j=1}^{q} \theta_j y_{t=j} + \varepsilon_t = \mu + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots\dots\dots\dots + \theta_q \varepsilon_{t-q} + \varepsilon_t \qquad (2.2)$$

Here μ is the mean of the series, θ_j (*j*=1,2,...,*q*) are the model parameters and *q* is the order of the model. The random shocks are assumed to be a white noise (John, 1997., Hipel and Mcleod, 1994) process, i.e. a sequence of independent and identically distributed (i.i.d) random variables with zero mean and a constant variance δ^2. Generally, the random shocks are assumed to follow the typical normal distribution. Thus conceptually a moving average model is a linear regression of the current observation of the time series against the random shocks of one or more prior observations. Fitting an MA model to a time series is more complicated than fitting an AR model because in the former one the random error terms are not fore-seeable.

Autoregressive (AR) and moving average (MA) models can be effectively combined together to form a general and useful class of time series models, known as the ARMA models. Mathematically an ARMA(*p, q*) model is represented as (Hipel and Mcleod, 1994., John., 1997., Lee, 2000) :

$$y_t = c + \varepsilon_t + \sum_{i=1}^{p} \varphi_i y_{(t-i)} + \sum_{i=1}^{p} \varphi_j y_{(t-j)} \qquad (2.3)$$

Here the model orders *p*, *q* refer to *p* autoregressive and *q* moving average terms.

Usually ARMA models are manipulated using the lag operator notation. The lag or backshift operator is defined as $Ly_t = y_{t-1}$. Polynomials of lag operator or lag polynomials are used to represent ARMA models as follows(John, 1997):

AR(*p*) model: $\varepsilon_t = \varphi(L) y_t$.

MA(q) model: $y_t = \theta(L)\varepsilon_t.$

ARMA(*p, q*) model: $\varphi(L)y_t = \theta(L)\varepsilon_t.$

Here $$\varphi(L)=1-\sum_{i=1}^{p}\varphi_i(L^i) \quad and \quad \theta(L) = 1+\sum_{j=1}^{p}\varphi_j(L_j)$$

It is shown in (Hipel and Mcleod, 1994) that an important property of AR(*p*) process is invertibility, i.e. an AR(*p*) process can always be written in terms of an MA(∞) process. Whereas for an MA(*q*) process to be invertible, all the roots of the equation θ(*L*) = 0 must lie outside the unit circle. This condition is known as the *Invertibility Condition* for an MA process.

Stationarity Analysis

When an AR(*p*) process is represented as $\varepsilon_t = \varphi(L)y_t$, then φ(*L*) = 0 is known as the characteristic equation for the process. It is proved by (Box and Jenkins, 1970) that a necessary and sufficient condition for the AR(*p*) process to be stationary is that all the roots of the characteristic equation must fall outside the unit circle. (Hipel and McLeod, 1994) mentioned another simple algorithm for determining stationarity of an AR process. For example as shown in (Lee, 2000) the AR(1) model $y_t = c + \varphi_1 y_{t-1} + \varepsilon_t$ is stationary when $|\varphi_1| < 1,$ with a constant mean μ = $\frac{c}{1-\varphi_1}$ and constant variance $\gamma_0 = \frac{\sigma^2}{1-\varphi_1^2}$

An MA(*q*) process is always stationary, irrespective of the values the MA parameters (Hipel and Mcleod, 1994). The conditions regarding stationarity and invertibility of AR and MA processes also hold for an ARMA process. An ARMA(*p, q*) process is stationary if all the roots of the characteristic equation φ(*L*) = 0 lie outside the unit circle. Similarly, if all the roots of the lag equation θ(*L*) = 0 lie outside the unit circle, then the ARMA(*p, q*) process is invertible and can be expressed as a pure AR process.

Autocorrelation and Partial Autocorrelation Functions (ACF and PACF)

To determine a proper model for a given time series data, it is necessary to carry out the ACF and PACF analysis. These statistical measures reflect how the observations in a time series are related to each other. For modeling and

INTERNATIONAL JOURNAL – ASD
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

forecasting purpose it is often useful to plot the ACF and PACF against consecutive time lags. These plots help in determining the order of AR and MA terms. Below we give their mathematical definitions:

For a time series$\{x(t), t = 0,1, 2,...\}$the *Autocovariance* (John, 1997, Hipel and Mcleod, 1994) at lag *k* is defined as:

$$\gamma_o = Cov(x_t, x_{t+k}) = E[(x_t - \mu)(x_{t+k} - \mu)] \qquad (2.4)$$

The *Autocorrelation Coefficient* (John, 1997., Hipel and Mcleod, 1994) at lag *k* is defined as:

$$\rho_k = \frac{\gamma_k}{\gamma_0}$$

Here μ is the mean of the time series, i.e. $\mu = E[x_t]$. The autocovariance at lag zero i.e. γ_0 is the variance of the time series. From the definition it is clear that the autocorrelation coefficient γ_k is dimensionless and so is independent of the scale of measurement. Also, clearly $-1 \leq \rho_k \leq 1$. Statisticians (Box and Jenkins, 1970) termed ρ_k as the theoretical *Autocovariance* Function (ACVF).

Another measure, known as the *Partial Autucorrelation Function (PACF)* is used to measure the correlation between an observation *k* period ago and the current observation, after controlling for observations at intermediate lags (i.e. at lags < *k*) (Lee, 2000). At lag 1, PACF (1) is same as ACF(1). The detailed formulae for calculating PACF are given in (Box and Jenkins, 1970, Hipel and Mcleod, 1974).

Normally, the stochastic process governing a time series is unknown and so it is not possible to determine the actual or theoretical ACF and PACF values. Rather these values are to be estimated from the training data, i.e. the known time series at hand. The estimated ACF and PACF values from the training data are respectively termed as sample ACF and PACF (Box and Jenkins, 1970., Hipel and Mcleod, 1974). As given in (Hipel and Mcleod, 1974), the most appropriate sample estimate for the ACF at lag *k* is

$$c_k = \frac{1}{n}\sum_{t=1}^{n-k}(x_t - \mu)(x_{t+k} - \mu) \qquad (2.6)$$

Then the estimate for the sample ACF at lag *k* is given by

$$r_k = \frac{c_k}{c_0} \qquad (2.7)$$

Here $\{x(t), t = 0,1,2,.......\}$ is the training series of size *n* with mean μ .

As explained by (Box and Jenkins, 1970), the sample ACF plot is useful in determining the type of model to fit to a time series of length *N*. Since ACF is symmetrical about lag zero, it is only required to plot the sample ACF for positive lags, from lag one onwards to a maximum lag of about *N/4*. The sample PACF plot helps in identifying the maximum order of an AR process. The methods for calculating ACF and PACF for ARMA models are described in (Hipel and Mcleod, 1994).

Autoregressive Integrated Moving Average (ARIMA) Models

The ARMA models, described above can only be used for stationary time series data. However in practice many time series such as those related to socio-economic (Hipel and Mcleod, 1994) an business show non-stationary behavior. Time series, which contain trend and seasonal patterns, are also non-stationary in nature (Faraway and Chatfield, 1998., Hamzacebi, 2008). Thus from application view point ARMA models are inadequate to properly describe non-stationary time series, which are frequently encountered in practice. For this reason the ARIMA model (Box and Jenkins, 1970., Hipel and Mcleod, 1994., Lonbardo and Flaherty, 2000) is proposed, which is a generalization of an ARMA model to include the case of non- stationarity as well.

In ARIMA models a non-stationary time series is made stationary by applying finite differencing of the data points. The mathematical formulation of the ARIMA(*p*,*d*,*q*) model using lag polynomials is given below (Hipel and Mcleod, 1994., Lonbardo and Flaherty, 2000):

$$\varphi(L)(1-L)^d y_t = \theta(L)e_t, i.e$$

$$\left(1-\sum_{t=1}^{p}\varphi_t L^t\right)(1-L)^d y_t = \left(1+\sum_{j=1}^{q}\theta_j L^j\right)\varepsilon_t \quad (2.8)$$

- Here, *p*, *d* and *q* are integers greater than or equal to zero and refer to the order of the autoregressive, integrated, and moving average parts of the model respectively.
- The integer *d* controls the level of differencing. Generally *d* =1 is enough in most cases.
  When *d*=0, then it reduces to an ARMA(*p,q*) model.
- An ARIMA(*p*,0,0) is nothing but the AR(*p*) model and ARIMA(0,0,*q)* is the MA(*q*) model
- ARIMA(0,1,0), i.e.$y_t = y_{t-1} + \varepsilon_t$ is a special one and known as the *Random Walk* model

(John, 1997., Lee, 2000., Zhang, 2003). It is widely used for non-stationary data, like economic and stock price series.

A useful generalization of ARIMA models is the Autoregressive Fractionally Integrated Moving Average (ARFIMA) model, which allows non-integer values of the differencing parameter *d*. ARFIMA has useful application in modeling time series with long memory (Galbraith and Zinde-Walsh, 2001). In this model the expansion of the term $(1-L)^d$ is to be done by using the general binomial theorem. Various contributions have been made by researchers towards the estimation of the general ARFIMA parameters.

Seasonal Autoregressive Integrated Moving Average (SARIMA) Models

The ARIMA model (2.8) is for non-seasonal non-stationary data. Box and Jenkins, (1970) have generalized this model to deal with seasonality. Their proposed model is known as the Seasonal ARIMA (SARIMA) model. In this model seasonal differencing of appropriate order is used to remove non-stationarity from the series. A first order seasonal difference is the difference between an observation and the corresponding observation from the previous year and is calculated as $z_t = y_t - y_{t-s}$. For monthly time series s = 12 and for quarterly time series s = 4. This model is generally termed as the SARIMA(*p,d,q*)× $(P,D,Q)^s$ model.

The mathematical formulation of a SARIMA(*p,d,q*)×$(P,D,Q)^s$ model in terms of lag polynomials is given below(Kihoro *et. al.*, 1994):

$$\Phi_p(L^s)\phi_p(L)(1-L)^d(1-L^s)^D y_t = \Theta_q(L^s)\theta_q(L)\varepsilon_t, \tag{2.9}$$

$$i.e.\ \Phi_p(L^s)\phi_p(L)z = \Theta_Q(L^s)\theta_q(L)\varepsilon_t,$$

Here z_t is the seasonally differenced series.

Some Non Linear Time Series Models

Having discussed linear time series models, we need to now concern ourselves with non-linear models as not all models are linear. For a better time series analysis and prediction, nonlinear models must also be considered. Campbell *et al.*, (1997) made important contributions towards this direction. According to them almost all non-linear time series can be divided into two branches: one includes models non-linear in mean and other includes models non-linear in variance (heteroskedastic). As an illustrative example, Parrelli (2001), gave two nonlinear time series models

- *Nonlinear Moving Average (NMA) Model*:

$\mathrm{y}_t = \varepsilon_t + \sigma\varepsilon_{t-1}^2$ This model is non-linear in mean but not in variance.

- The simplest ARCH model (*Engle's 1982*), ARCH(1), is

$$y_t = \sigma_t \varepsilon_t \qquad \sigma_t^2 = \sigma_0 + \sigma_1 y_{t-1}^2$$

with $\varepsilon_t \sim N(0,1)$ and the sequence of ϱt variables being independent. Here $\alpha_1 > 0$ has to be satisfied to avoid negative variances. Note that the conditional distribution of Y_t given $Y_{t-1} = y_{t-1}$ is

$$N(0, \sigma_0 + \sigma_1 y_{t-1}^2)$$

This model is heteroskedastic, i.e. non-linear in variance, but linear in mean. This model has several other variations, like GARCH, EGARCH etc.

GARCH and other models

The ARCH model can be thought of as an autoregressive model in y_t^2. An obvious extension of this idea is to consider adding moving average terms as well. This generalization of ARCH is called GARCH. The simplest GARCH model is GARCH(1,1):

$$y_t = \sigma_t \varepsilon_t, \qquad \sigma_t^2 = \alpha_1 y_{t-1}^2 + \beta_1 \sigma_{t-1}^2$$

The sequence is second-order stationary if $\alpha 1 + \beta 1 < 1$.

The simplest estimation scheme for the GARCH(1,1) model uses some initial sample of observations to obtain a crude estimate of σ_t^2, and then use maximum likelihood estimation based on the prediction error decomposition.

A further extension (EGARCH, where E is for exponential) is to model the log of σ_t^2 as a function of the magnitude, and of the sign, if $\alpha_1 + \beta_1 < 1$.

Box-Jenkins Methodology

Having discussed various time series models, there is need to select an appropriate model that can produce accurate forecast based on a description of historical pattern in the data and how to determine the optimal model orders. Box and Jenkins (1970), developed a practical approach to build ARIMA model, which best fit to a given time series and also satisfy the Parsimony principle. Their concept has fundamental importance on the area of time series analysis and for easting (Lombardo and Flaherty, 2000., Zhang, 2003). The Box-Jenkins methodology does not assume any particular pattern in the historical data of the series to be forecasted. Rather, it uses a three step iterative approach of *model identification*, *parameter estimation* and *diagnostic checking* to determine the best parsimonious model from a general class of ARIMA models (Box and Jenkins, 1970., Lee, 1978., Lombardo and Flaherty, 2000., Zhang, 2003). This three-step process is repeated

INTERNATIONAL JOURNAL – ASD
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

several times until a satisfactory model is finally selected. Then this model can be used for forecasting future values of the time series. The Box-Jenkins forecast method is schematically shown in Figure 2.1

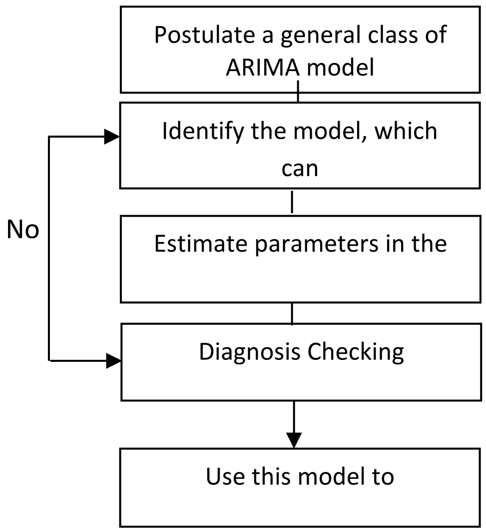

Figure 2.1: The Box-Jenkins methodology for optimal model selection

(Adapted from Adhikari and Agrawal, 2013)

A crucial step in an appropriate model selection is the determination of optimal model parameters. One criterion is that the sample ACF and PACF, calculated from the training data should match with the corresponding theoretical or actual values (Hipel and Mcleod, 1994., Faraway and Chatfield., Kihoro, *et. al.*, 2000.,). Other widely used measures for model identification are *Akaike Information Criterion (AIC) (Faraway and Chatfield*, 1998., Kihoro *et. al.*, 2000), *Bayesian Information Criterion (BIC)* (Faraway and Chatfield, 1998., Kihoro *et. al.*, 2000) which are defined below Faraway and Chatfield, 1998):

$$\text{AIC(p)} = \text{n In}\left(\hat{\sigma}_e^2 / n \right) + 2p$$

$$\text{BIC(p)} = \text{n In}\left(\hat{\sigma}_e^2 / n \right) + p + p\, In(n)$$

Here n is the number of effective observations, used to fit the model, *p* is the number of parameters in the model and $\hat{\sigma}_e^2$ is the sum of sample squared residuals. The optimal model order is chosen by the number of model parameters, which minimizes either AIC or BIC. Other similar criteria have also been proposed in literature for optimal model identification

Time series predictions using Artificial Neural Network

Artificial neural networks (ANNs) approach has been suggested as an alternative technique to time series forecasting and it gained immense popularity in last few years (Tealab 2018, Doulah, 2019). The basic objective of ANNs was to construct a model for mimicking the intelligence of human brain into machine (Kihoro *et. al*., 2000., Sousa *et al*., 2006). Similar to the work of a human brain, ANNs try to recognize regularities and patterns in the input data, learn from experience and then provide generalized results based on their known previous knowledge. Although the development of ANNs was mainly biologically motivated, but afterwards they have been applied in many different areas, especially for forecasting and classification purposes (Kihoro *et. al*., 2000., Khashei and Bijari 2010, Sharma, 2012, Tealab 2018). The important features of ANNs, which make them quite favorite for time series analysis and forecasting are stated below

- ANNs are data-driven and self-adaptive in nature (Zhang *et. al*., 1998., Joader *et. al*., 2000). There is no need to specify a particular model form or to make any *a priori* assumption about the statistical distribution of the data; the desired model is adaptively formed based on the features presented from the data. This approach is quite useful for many practical situations, where no theoretical guidance is available for an appropriate data generation process.
- ANNs are inherently non-linear, which makes them more practical and accurate in modeling complex data patterns, as opposed to various traditional linear approaches, such as ARIMA methods (Zhang *et. al*., 1998., Jaoder, *et. al*., 2000., Zhang 2003). There are many instances, which suggest that ANNs made quite better analysis and forecasting than various linear models (Sharma, 2012).
- According to Hornik *et al*., (1989), ANNs are universal functional approximators. They have shown that a network can approximate any continuous function to any desired accuracy (Zhang, 1998). ANNs use parallel processing of the information from the data to approximate a large class of functions with a high degree of accuracy. Further, they can deal with situation, where the input data are erroneous, incomplete or fuzzy (Joader *et. al*., 2000).

The ANN Architecture

The most widely used ANNs in forecasting problems are multi-layer perceptrons (MLPs), which use a single hidden layer feed forward network (FNN) (Zhang *et. al*., 1998., Zhang, 2003). The model is characterized by a network of three layers, viz. input, hidden and output layer, connected by acyclic links. There may be more than one hidden layer. The nodes in various layers are also known as processing elements. The three-layer feed forward architecture of ANN models can be diagrammatically depicted as shown in Figure 3.1 below:

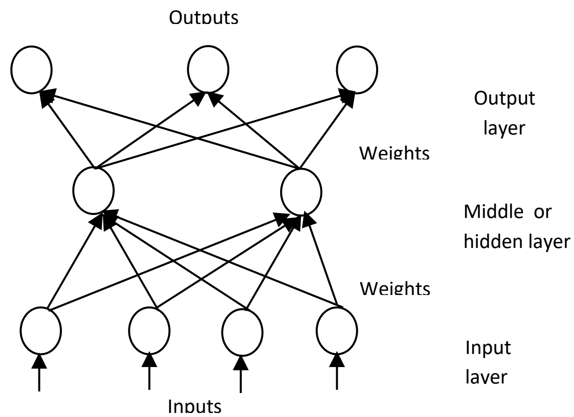

Figure 3.1: The three-layer feed forward ANN architecture

The output of the model is computed using the following mathematical expression (Zhang, 2007)

$$y_t = \alpha_0 + \sum_{j-1}^{q} \alpha_j \, g\left[\beta_{0j} + \sum_{i-1}^{p} \beta_y y_{t-1}\right] + \varepsilon_t, \forall t \ldots \ldots \ldots \ldots \ldots \ldots \ldots . (3.1)$$

Here $y y_{t-1} (i = 1,2, \ldots \ldots, p)$ are the *p* inputs and y_t is the output. The integers *p*, *q* are the number of input and hidden nodes respectively. $\alpha_j (j = 0,1,2, \ldots \ldots, q) and\ \beta_y (i = 0,1,2, \ldots \ldots, p, j = 0,1,2, \ldots . q)$ 0,1,2,…,*q*) are the connection weights and ε_t is the random shock; α_0 and β_{0j} are the bias terms. Usually, the Ψ logistic sigmoid function $g(x) = \frac{1}{1 + e^{-x}}$ is applied as the nonlinear activation function. Other activation functions, such as linear, hyperbolic tangent, Gaussian, etc. can also be used.

The feed forward ANN model (3.1) in fact performs a non-linear functional mapping from the past observations of the time series to the future value, i.e $y_t = f(y_{t-1}, y_{t-2}, \ldots \ldots . . y_{t-p}, w) = \varepsilon_t,$ where **w** is a vector of all parameters and *f* is a function determined by the network structure and connection weights (Zhang *et. al*., 1998., Zhang, 2003)

To estimate the connection weights, non-linear least square procedures are used, which are based on the minimization of the error function (Kihoro *et. al*., 2000):

INTERNATIONAL JOURNAL – ASD
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

$$F(\psi) = \sum_t e_t^2 = \sum_t [(y_t - y_t)]^2 \qquad \text{……………………………. (3.2)}$$

Here Ψ is the space of all connection weights.

The optimization techniques used for minimizing the error function (3.2) are referred as *Learning Rules*. The best-known learning rule in literature is the *Back propagation* or *Generalized Delta Rule* (Kihoro *et. al*., 2000)

Time Lagged Neural Networks (TLNN)

In the FNN formulation, described above, the input nodes are the successive observations of the time series, i.e. the target x_t is a function of the values y_{t-i} , (i =1,2,..., p) where p is the number of input nodes. Another variation of FNN, viz. the TLNN architecture (Faraway and Chatfield, 1998., Kihoro *et. al*., 2000) is also widely used. In TLNN, the input nodes are the time series values at some particular lags. For example, a typical TLNN for a time series, with seasonal period s =12 can contain the input nodes as the lagged values at time t –1, t –2 and t -12.

The value at time t is to be forecasted using the values at lags 1, 2 and 12.

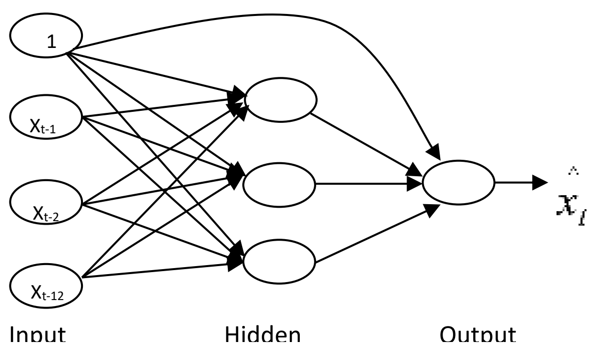

Figure 3.2: A typical TLNN architecture for monthly data

In addition, there is a constant input term, which may be conveniently taken as 1 and this is connected to every neuron in the hidden and output layer. The introduction of this constant input unit avoids the necessity of separately introducing a bias term.

For a TLNN with one hidden level, the general prediction equation for computing a forecast may be written as (Faraway and Chatfield, 1998):

$$\hat{x} = \phi_0 + \left\{ w_{c0} + \sum_b w_{h0}\phi_h \left(w_{ch} + \sum_i w_{ih} x_{t-jh} \right) \right\} \qquad (3.3)$$

Here, the selected past observations $x_{t-j1}, x_{t-j2}, \dots\dots, x_{jk}$ are the input terms, $\{w_{ch}\}$ are the weights for the connections between the constant input and hidden neurons and w_{co} is the weight of the direct connection between the constant input and the output. Also $\{w_{ih}\}$ and $\{w_{ho}\}$ denote the weights for other connections between the input and hidden neurons and between the hidden and output neurons respectively. $\emptyset_o$ and $\emptyset_h$ are the hidden and output layer activation functions respectively. (Faraway and Chatfield, 1998) used the notation NN($j_{\downarrow}1, j_{\downarrow}2, \dots\dots j_{\downarrow}k; h)$ to denote the TLNN with inputs at lags $j_1, j_2, \dots\dots j_k$ and *h* hidden neurons. We shall also adopt this notation in our upcoming experiments. Thus Figure 3.2 represents an NN (1, 2, 12; 3) model.

Seasonal Artificial Neural Networks (SANN)

The SANN structure was proposed by (Hamzacebi, 2008) to improve the forecasting performance of ANNs for seasonal time series data. The proposed SANN model does not require any preprocessing of raw data. Also SANN can learn the seasonal pattern in the series, without removing them, contrary to some other traditional approaches, such as SARIMA, above the author has empirically verified the good forecasting ability of SANN on four practical time data sets. A brief overview of SANN model as proposed by Hamzacebi (2008) is presented below.

In this model, the seasonal parameter *s* is used to determine the number of input and output neurons. This consideration makes the model surprisingly simple for understanding and implementation. The i^{th} and $(i+1)^{th}$ seasonal period observations are respectively used as the values of input and output neurons in this network structure. Each seasonal period is composed of a number of observations. Diagrammatically an SANN structure can be shown as in Figure 3.3 below (Hamzacebi, 2008):

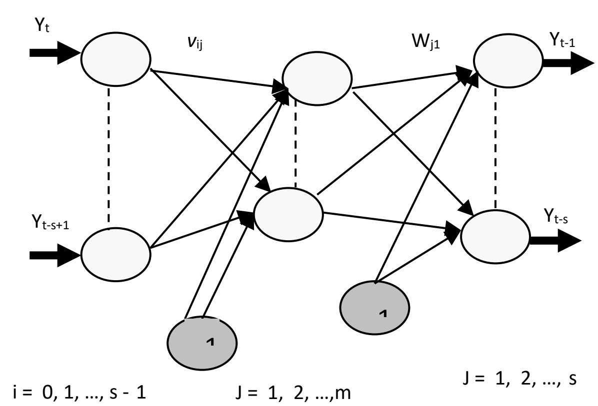

Figure 3.3: SANN architecture for seasonal time series

INTERNATIONAL JOURNAL – ASD
VOL. 10 NO.2 DEC-2019 ISSN: 2760-4106

Mathematical expression for the output of the model is

$$Y_{t+i} = \alpha_i + \sum_{j=1}^{m} w_{jt} f\left(\theta_j + \sum_{i=0}^{s-1} v_y Y_{t-1}\right) \forall; i = 1,2,3, \dots \dots, s \dots \dots \dots \dots \dots . (3.4)$$

Here $Y_{t+i}(l = 1,2,3, \dots \dots, s)$ are the predictions for the future *s* periods and $Y_{t-i}(i = 0,1,2, \dots \dots, s-1)$ are the observations of the previous *s* periods; $v_y(i = 0,1,2,3, \dots \dots, s-1; j = 1,2,3, \dots \dots, m)$ are weights of connections from input nodes to hidden nodes and $w_{jl}\ (j = 1,2,3, \dots, m; l = 1,2,3, \dots \dots .1,2,3, \dots \dots, s)$ are weights of connections from hidden nodes to output nodes. Also $\alpha_l(l = 1,2,3, \dots \dots, s)$ and $\theta_j(j = 1,2,3, \dots \dots, m)$ are weights of bias connection and *f* is the activation function (Hamzacebi 2008).

.

Thus while forecasting with SANN, the number of input and output neurons should be taken as 12 for monthly and 4 for quarterly time series. The appropriate number of hidden nodes can be determined by performing suitable experiments on the training data.

Selection of a Proper Network Architecture

So far we have discussed about three important network architectures, viz. the FNN, TLNN and SANN, which are extensively used in forecasting problems under artificial neural network. Some other types of neural models are also proposed in literature, such as the *Probabilistic Neural Network (PNN)* for classification problem and *Generalized Regression Neural Network (GRNN)* (Jorarder *et al*., 2000) for regression problem. After specifying a particular network structure, the next most important issue is the determination of the optimal network parameters. The number of network parameters is equal to the total number of connections between the neurons and the bias terms (Faraway and Chartfield, 1998., Hamzacebi, 2008). A desired network model should produce reasonably small error not only on within sample (training) data but also on out of sample (test) data (Jorarder *et. al*., 2000). Due to this reason immense care is required while choosing the number of input and hidden neurons. However, it is a difficult task as there is no theoretical guidance available for the selection of these parameters and often experiments, such as cross-validation are conducted for this purpose (Zhang, 2003., Hamzacebi, 2008). Another major problem is that an inadequate or large number of network parameters may lead to the overtraining of data

(Chartfield,1996; Zhang, 2003). Overtraining produces spuriously good within-sample fit, which does not generate better forecasts. To penalize the addition of extra parameters some model comparison criteria, such as AIC and BIC can be used (Faraway and Chartfield, 1998). *Network Pruning* (Kihoro *et al.,* 2000) and MacKay's *Bayesian Regularization Algorithm (Faraway and Chartfield, 1998*; Joarder, 2000) are also quite popular in this regard (Adhikari and Agrawal, 2013).

In summary we can say that NNs are amazingly simple though powerful techniques for time series forecasting. The selection of appropriate network parameters is crucial, while using NNs for forecasting purpose. Also a suitable transformation or rescaling of the training data is often necessary to obtain best results.

Hybrid Approach for Time Series Modelings

Artificial neural network models with hidden layers can capture nonlinear patterns in time series because they can be characterized as a class of approximate general functions, which are capable of modeling nonlinearity (Tang and Fishwick, 1993, Khashei and Bijari 2011a, Khashei and Bijari 2011b, Yi *et al.,* 2012,). Thus, it is prudent to combine ANNs and ARIMA in time series forecasting to deal with all of the heterogeneous components of the underlying patterns. Also, it may be judicious to consider a time series, which is composed of a linear autocorrelation structure and a nonlinear component. Figure 4.2 shows a schematic diagram of integrating ANNs with partially known nonlinear relationships. Again, we can consider two models to analyse such time series, additive model (L + N) and multiplicative model (L-N). Figure 4.3 illustrates the combined models of ANNs integrating the two cases. The mathematical expressions for these two cases are given by Equations (4.1) and (4.2) below:

Additive Model: $y_t = L_t + N_t$ (4.1)

Multiplicative Model : $y_t = L_t * N_t$ (4.2)

Where L_t represents the linear component and N_t the nonlinear component. These two components have to be derived from the data using these equations. In contrast, Zhang (2003) proposed a hybrid model of ARIMA and ANN for the additive model.

During the first phase of the proposed hybrid approach, an Arima model is applied to the linear component of time series, which is assumed to be $\{y_t, t=1, 2, \dots\}$, and a series of forecasts are generated, namely $\left\{\hat{L}_t\right\}$. By comparing the actual values

y_t with the forecast value of $\{\hat{L}_t\}$ of the linear component, we can obtain a series of nonlinear components, which are defined to be $\{e_t\}$.

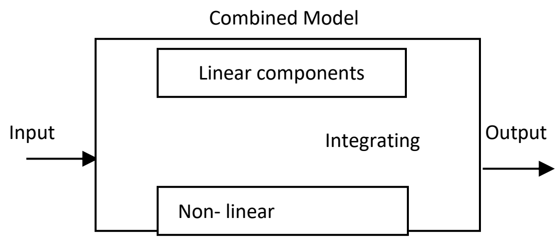

Figure 4.1: Schematic diagram of combined models

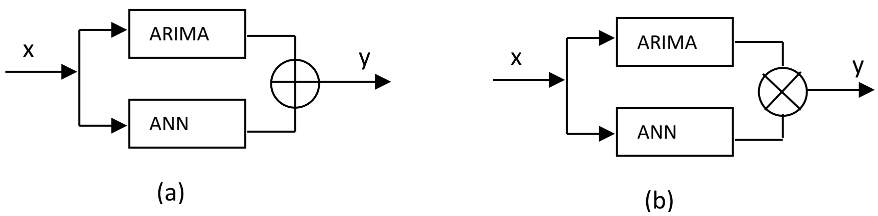

Figure 4.2: Combined models of artificial neural networks with two cases of integrating" (a) additive model, (b) multiplicative model

According to 'multiplicative model', we have

$$e_t = y_t / \hat{L}_t \qquad (4.3)$$

By contrast, according to 'additive model' we have

$$e_t = y_t - \hat{L}_t \qquad (4.4)$$

Thus, a nonlinear time series is obtained. The second phase is concerned ith modeling the nonlinear component of the specified time series in an ANN model. The trained ANNs model is responsible for making a series of forecasts of nonlinear components, denoted by $\{\tilde{N}_t\}$, which are based on the previously deduced nonlinear time series $\{e_t\}$ vales as the inputs. That is, the ANNs time series forecasting model is a nonlinear mapping function, as shown below :

$$e_t = f(e_{t-1}, e_{t-2}, \Lambda, e_{t-n}) + e_t \qquad (4.5)$$

The second phase can be seen as a process of error correction of time series prediction in the ANNs model on the basis of ARIMA model. Thus, for the multiplicative model, the combined forecast can be obtained from equation (8) below

$$y_t = \hat{L}_t * \hat{N}_t \qquad (4.6)$$

However, for the ‘additive model, the combined forecast is given by equation (4.7) below:

$$y_t = \hat{L}_t + \hat{N}_t \qquad (4.7)$$

Conclusion

Time series modeling and prediction is an active research area over the last few decades. The accuracy of time series prediction is fundamental to many decision processes and hence the research for improving the effectiveness of prediction models has never stopped. One of them is the *Combining Approach*, i.e. to combine a number of different and dissimilar methods to improve forecast accuracy. A lot of works have been done towards this direction and various combining methods have been proposed in literature. Together with other analysis in time series forecasting, we have thought to find an efficient combining model in future if possible with the aim of further studies in time series modeling and prediction.

References

Adhikari, R. and Agrawal A An Introductory Study on Time Series Modeling and Forecasting. LAP Lambert Academic Publishing, Germany. 2013, arXiv:1302.6613

Ashwini, N. and Bhavya, G. Using Mining and Intelligent Approach for Time Series Forecasting Problems: *International Journal of Advanced Research in Computer Science and Software Engineering*, 2013, 3(11); 1567-1571.

Bates, J. M. and Granger, C. W. J. The Combination of Forecast. *Journal of the Operational Research Society.* 1969.

Box, G.E.P. and Jenkins, G.. Time Series Analysis, Forecasting and Control: Holden-Day,San Francisco, CA. 1970.

Campbell, J. Y. , Lo, A.W. and Mackinlay A.C. The Econometrics of Financial Markets. Princeton University Press. Princeton, New Jersey. 1997.

Chatfield, C. Model uncertainty and forecast accuracy. *J. Forecasting.* 1996, 15, pages: 495–508

Doualh, M.S. Time Series Forecasting: A comparativeStudy of VAR ANN and SVM Models. *Journal of statical and Econometric methods.* 2019, *8(3), 21-34*

Engle, R. F. “Autoregressive Conditional Heteroscedasticity with estimates of the variance of United Kingdom inflation” Econometrics. 1982, 50 (4) 987-1007. DOI: 10.2307/1912773.

Faraway, J. and Chatfield, C.. Time series forecasting with neural networks: a comparative study using the airline data: *Applied Statistics* 1998, 47, pages: 231–250.

Farhath, Z.A., Arputhamary, B. and Arockiam, L. A Survey On Arima Forecastingusing Time Series Model International Journal of Computer Science and Mobile Computing, 2016, 5 (8), 104-109

Firmino, P. R. A.; Mattos Neto, P. S. G. and Ferriera, T. A. E. Error modeling approach to improve time series forecasters. *Neurocomputing*. 2015, 153, 242–254.

Galbraith, J. W and Zinde-Walsh, V. Autoregression-based estimators for ARFIMA Models: CIRANO Working Papers, No: 2011s-11. 2001

Hamilton, J. D. Time Series Analysis,1 ed. New Jersey : Princeton University Press. 1994.

Hamzacebi, C. Improving artificial neural networks performance in seasonal time series forecasting. *Information Science.* 2008, 178, 4550-4559.

Haykin, S. S. Redes Neurais, 2ed. Porto Alegre: Bookman. 2001.

Hipel, K. W. and McLeod, A. I. Time Series Modelling of Water Resources and Environmental Systems, Amsterdam, Elsevier. 1994.

Hornik, K., Stinchcombe, M. and White, H. Multilayer feed-forward networks are universal approximators: *Neural Networks.* 1989, 2 ,: 359–366.

John H. C. Time Series for Macroeconomics and Finance: Graduate School of Business. University of Chicago, Spring . 1997.

Joarder, K., Rezaul, B. and Ruhul, S., Artificial Neural Networks in Finance and Manufacturing: Idea Group Publishing, USA. 2002.

Jun. W., Jian. L. and Jiaquan. W.. Application of chaos and fractal models to water quality time series prediction. *Environmental Modelling & Software,* 2009. 24,(5), 632-636.

Khashei, M & Bijari, M. An artificial neural network (p, d, q) model for time series forecasting. *Expert Systems with Applications*, 2010. 37, (1), 479-489.

Khashei, M.& Bijari, M. Which Methodology is Better for Combining Linear and Nonlinear Models for Time Series Forecasting? *Journal of Industrial and Systems Engineering.* . 2011a, 4,(4), 265-285.

Khashei M.& Bijari M. A novel hybridization of artificial neural networks and ARIMA models for time series forecasting; *Applied Soft Computing.* 2011b, 11; 2664–2675.

Kihoro, J.M., Otieno, R.O. and Wafula, C. Seasonal Time Series Forecasting: A Comparative

Study of ARIMA and ANN Models: *African Journal of Science and Technology (AJST) Science and Engineering Series* 2000, 5,(2), 41-49.

Lee, J., Univariate time series modeling and forecasting (Box-Jenkins Method): Econ 413, lecture 4. 2000.

Lombardo, R and Flaherty, J. Modeling Private New Housing Starts In Australia, Pacific-Rim Real Estate Society Conference, University of Technology Sydney (UTS), 2000. January 24-27.

Park, H.. Forecasting Three-Month Treasury Bills Using ARIMA and GARCH Models. Econ 930, Department of Economics, Kansas State University. 1999.

Parrelli, R.. Introduction to ARCH & GARCH models, Optional TA Handouts, Econ 472 Department of Economics, University of Illinois. 2001.

Raicharoen. T., Lursinsap, C. and Sanguanbhoki, P. Application of critical support vector machine to time series prediction, Circuits and Systems, 2003. ISCAS '03.Proceedings of the 2003 International Symposium. 2003, (5), 25-28 , pages: V-741-V-744.

Sharma, M. A. Comparative study of forecasting models based on weather parameters. A Ph.D thesis in Statistics from the Shobhit Institute of Engineering and Technology (A deemed-to-b- University) Modipuram, Meerut-250110 India. 2012.

Sousa, S. I. V., Martins, F. G., Pereira, M. C., and Alvim-Ferraz, M.C. M. Prediction of ozone concentrations in Oporto city with statistical approaches. *Chemosphere*, 2006, 64(7), 1141– 1149.

Suhartono, S. and Guritno, S. A comparative study of forecasting models for trends and seasonal time series: Does complex model always yield better forecast than simple models? *Journal Teknik Industri*. ,2005, 7(1), pp. 22-30

Suresh, N.N., Saravanane, R., and Sundararajan., T., Application of ANN and MLR Models on Groundwater Quality Using CWQI at Lawspet, Puducherry in India. *Journal of Geoscience and Environment Protection.* 2017, 5, 99-124.

Tang Z, Fishwick P. Feed forward neural nets as models for time series forecasting. *ORSA Journal on Computing.* 1993, 5: 374–385.

Tealab A. (2018). Time series forecasting using artificial neural network methodologies: A systematic review. Future Computing and Informatics Journal 3. 334-340.

Tong, H. Threshold Models in Non-Linear Time Series Analysis: Springer-Verlag, New York. 1983.

Wang, l.; Zou, H.; Su, J.; Li, L.; and Chaudhry, S.; (2013) An ARIMA-ANN Hybrid Model for Time series forecasting. System Research and Behavioural Science System Research 30, 244-259. DOI:10.1002/series.2179

Xiao, Y., Xiao, J. and Wang, S. A hybrid model for times series forecasting. *Human Systems Management* . 2012, 31, 1-11. DOI 10.3233/HSM-2012-0763

Yan, H. and Zou, Z. Application of a Hybrid ARIMA and Neural Network Model to Water Quality Time Series Forecasting. *Journal of Convergence Information Technology* (JCIT). 2013, 8,(4), doi:10.4156/jcit. .vol8.issue4.8

Zhang, G; Patuwo, B. E. and Hu, M. Y. Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting.* 1998, 14 , 35-62.

Zhang, G.P. Time series forecasting using a hybrid ARIMA and neural network Model: *Neuro computing.* 2003, 50, 159–175.

Zhang, G.P. A neural network ensemble method with jittered training data for time series forecasting: *Information Sciences.* ,2007, 177, 5329–5346.