HYBRID SYSTEM OF SIMPLE EXPONENTIAL SMOOTHING AND NEURAL NETWORK FOR KSE100 INDEX

SAMREEN FATIMA
FAST - National University,
Karachi Campus
Email: squdduss@yahoo.com

Abstract

This study deals with application of Hybrid forecasting systems to formulate simple exponential smoothing model for KSE 100 index daily share price data. We used 7 days ahead forecasts for daily share price of KSE100 index data. This study is divided into two phases. In the first phase of the study we compared ANN model with SES and found that ANN model gives better forecasting performance then SES model. In the second phase of the study we developed a hybrid system of Artificial Neural Networks (ANN) and Simple Exponential Smoothing (SES) models represented as ANN SES for KSE100 index daily share price data. Empirical results support our proposed hybrid system (ANN SES).

INTRODUCTION:

A very difficult task for the stock market is to forecast the financial data such as exchange rates and daily share price etc. A class of time series model i.e exponential smoothing models has been widely used in business and finance [Gardner (1985) & Leung et. al (2000)]. Simple exponential smoothing method is suitable for the specified time period for financial time series because financial time series contain irregularity, randomness and volatility. However, the exponential smoothing method is only a class of linear model and thus it can only capture linear features of financial time series. We use mathematical models for economical forecasting, but in some cases they fail to forecast economic cycles. Time series models are commonly used for prediction and forecasts, but time series methods such as SES, ARIMA and ARCH/GARCH models are not capable of accurately forecasting the time series as they are based on the theory of stationary stochastic processes and follow normality assumption. Neural networks have received a great deal of importance over the last few years and are being successfully applied across an extra ordinary range of problem domains. They are useful to solve the problems of prediction and classification. Neural Network needs very few assumptions to learn patterns of input variables for prediction purpose. P.H.Franses et.al (1998) used ANN to forecast daily exchange rate returns through analyzing the possible effects of GARCH. Nikola Gradojevic and Jng yang (2000) combined two new approaches ANN and market microstructure to exchange rate determination in order to explain short run exchange rate fluctuations. ANN model gave better forecasting performance as compared to GARCH and ARIMA models for KSE 100 index (daily share price data) by Samreen Fatima and Ghulam Hussain (2001). In this study we compare ANN model with SES model and develop a hybrid system (combination of ANN and SES) for KSE100 index daily share price data.

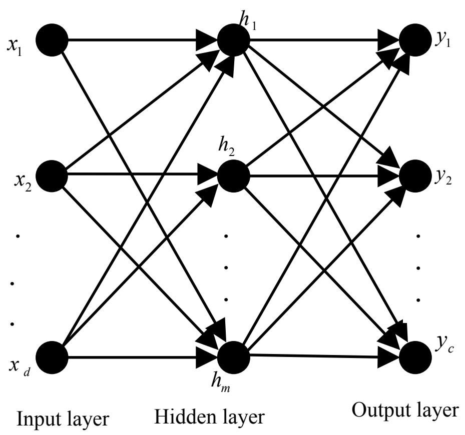
The outline of this paper is as follows. Section 1 presents a brief introduction of ANN and its learning. Section 2 deals with SES model. Section 3 present the results of ANN and SES models based on 1st January 2000 to 11th October 2002 and forecast based on 12th October 2002 to 18th October 2002 of KSE100 index daily share price data. Section 4 deals with conclusion.

1 Introduction to Artificial Neural Network model:

In this section we briefly describe the functional form of feed forward Artificial Neural Network, which is commonly used for analysis of economical and financial time series data (Swanson and White 1995, Kuan and Liu 1995).

In this study we focused on only feed-forward neural network in which ANN allow information to travel from input to output. There are three types of nodes. Input nodes are the processing nodes which receive data from external sources and transmit to the other nodes (hidden nodes).Out put nodes are the results of input data. Hidden nodes are used to connect input and out neurons.

Consider a feed forward network with one hidden layer. The input layer is represented by a vector d = (x1, x2, ...xd), the hidden layer is represented by a vector M=(h1, h2,......hm) and output layer is represented by C= (y1, y2,......yc). See figure 1.
Figure 1. Multilayer Feed Forward Network.



The output of the j^{th} hidden unit is obtained by first forming a weighted linear combination of the 'd' input values and adding a bias(an external input). The activation of hidden unit j can be obtained by transforming the linear sum using a logistic activation function $g(a)$:

$$h_j = g\left(\sum_{i=1}^{d} w_{ji}^{(1)} x_i + \mu_{0j}\right)$$

Where $g(a) = \dfrac{\exp(a)}{1+\exp(a)}$, x_i is the i^{th} input node and μ_{oj} is called bias. For the output layer, the node is define as

$$y_k = f\left(\sum_{j=1}^{m} w_{jk}^{(2)} g\left(\sum_{i=1}^{d} w_{ji}^{(1)} x_i + \mu_{0j}\right) + \mu_{0c}\right) \qquad (1)$$

Where μ_{0c} is bias and $f(.)$ is a non linear activation function if it is linear then equation (1) becomes

$$y_k = \sum_{j=1}^{m} w_{jk}^{(2)} g\left(\sum_{i=1}^{d} w_{ji}^{(1)} xi + \mu_{0j}\right) + \mu_{0c} \qquad (2)$$

If the connection from input layer to the output layer is directed and the activation function of the output layer is linear then the network becomes.

$$y_k = \sum_{i=1 \to c}^{d} \alpha_{ic} x_i + \sum_{j=1}^{m} w_{jk}^{(2)} g\left(\sum_{i=1}^{d} w_{ji}^{(1)} xi + \mu_{0j}\right) + \mu_{0c} \qquad (3)$$

In time series analysis we try to make statistical inferences from a realization to the process that generated it. When we try to predict or forecast the future we need assumptions of stability about the past. These stability assumptions are called stationary conditions. Non stationary time-series processes can be transformed by differencing the series one more time, to make them stationary. Those differences are called lag differences. If the input nodes have lagged values then equation (3) can be written as

$$y_t = \sum_{i=1 \to c}^{d} \alpha_{ic} y_{t-i} + \sum_{j=1}^{m} w_{jk}^{(2)} g\left(\sum_{i=1}^{d} w_{ji}^{(1)} y_{t-i} + \mu_{0j}\right) + \mu_{0c} \qquad (4)$$

Some time we use logarithmic transformation to make the series stationary. In this study we used $y_t = \log \left[\frac{x_t}{x_{t-1}} \right]$ transformation for the purpose of stabilizing variance; details will be discussed in section 3.2. Equation (4) is also known as a semi parametric method.

1.1 Learning Algorithm:

The process of determining the values of the parameters (weights) of ANN model on the basis of the data set is called learning or training. There are two different learning methods, supervised and unsupervised. In supervised learning the network compares its outputs with available training examples (desired output) and receives feed back about any errors. Unsupervised learning method discovers for itself patterns, features regularities, correlation or categories input data and code for them in the output.

Back propagation is the most commonly used learning algorithm, which was invented independently several times by Bryson and Ho(1969), Werbos (1974), Parker(1985) and Rumelhart, Hinton, and Williams (1986a). It is a gradient procedure that first computes the gradient E of the error function with respect to each weight of the network. An error or objective function E is defined as

$$E = \frac{1}{2}\sum_{p=1}^{k}\left(t_p - y_p\right)^2 \qquad (5)$$

It is the sum of overall points p in the data set of the squared difference between the desired target of the output values ‘’ and the model prediction of the values .

$$\frac{\partial E}{\partial w_{ij}} = -\left(t_p - y_p\right)F'\left(\sum x_i w_{ij}\right)x_i$$

$$\Delta w_{ij} = \frac{\partial E}{\partial w_{ij}} = -\left(t_p - y_p\right)\left(1 - y_p\right)y_p x_i$$

Weights update rule is

$$w_{ij}^{t+1} = w_{ij}^{t} + \Delta w_{ij} \qquad (6)$$

Where the current is weight and is the update weights. Equation (6) gives a formula to update the weights from input layer to output units in single layer network.

II. Simple Exponential Smoothing (SES) model:

Exponential smoothing has become very popular as a forecasting method for a wide variety of time series data. The method of simple exponential forecasting takes the forecast for the previous period and adjusts it using the forecast error. The forecast for the next period is defined as:

$F_{t+1} = F_t + \alpha(Y_t - F_t)$

Or in another way we can write

$$F_{t+1} = \alpha Y_t + (1-\alpha)F_t \qquad (7)$$

Where the forecast of current period is F_{t+1}, F_t is the forecast for pervious. α is the smoothing constant and Y_t is available observation (KSE100 index daily share price). The forecast (F_{t+1}) is based on weighting the most recent observation (Y_t) with a weight value (α) and weighting the most recent forecast (F_t) with a weight of $1-\alpha$.

Where $F_t = \alpha Y_{t-1} + (1-\alpha)F_{t-1}$

After substituting the above result equation 7 becomes.

$$F_{t+1} = \alpha Y_t + \alpha(1-\alpha)Y_{t-1} + (1-\alpha)^2 F_{t-1}$$

Similarly, substituting recursively for $F_{t-1}, F_{t-2}, \ldots, F_1$ we obtain

$$F_{t+1} = \alpha Y_t + \alpha(1-\alpha)Y_{t-1} + \alpha(1-\alpha)^2 Y_{t-2} + \alpha(1-\alpha)^3 Y_{t-3} + \alpha(1-\alpha)^4 Y_{t-4} + \ldots . .$$
$$+ \alpha(1-\alpha)^{t-1} Y_1 + (1-\alpha)^t F_1$$

Where F_1 is the initial forecast and the weight attached to F_1 is usually small.

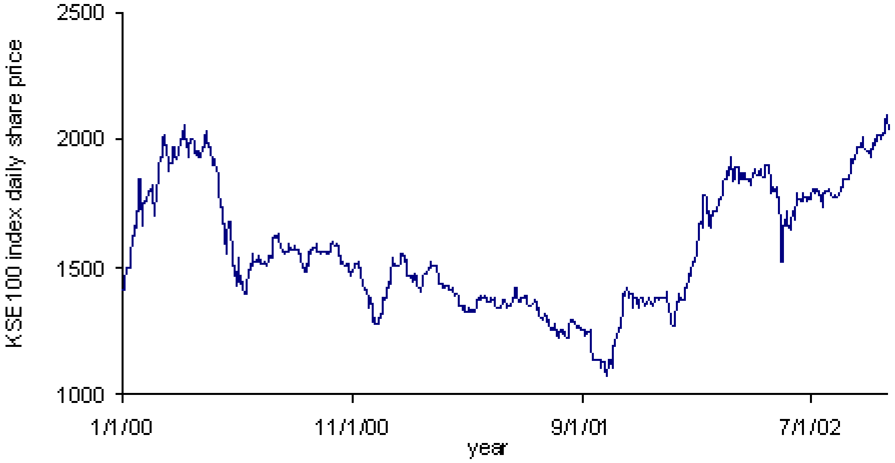
Exponential Smoothing assigns exponentially decreasing weights as the observation gets older. In other words, recent observations are given relatively more weight in forecasting than the older observations Spyros Makridakis et.al (1998). Value of between 0 to1

Estimating the best value from the data:

In practice the smoothing parameter is often chosen by a grid search of the parameter space that is, different solutions for are tried starting, for example, with = 0.1 to = 0.9, with increments of 0.1.

III Data analysis:

Figure2. KSE Daily Share Index Data 2000 – 2002



In this study we used data for the period 1st January 2000 to 18th October 2002. The data were available on daily basis at the world wide web of Karachi stock exchange, www.s_exchange .com.

We used 1st January 2000 to 18th October 2002 data for model building and from 12th October 2002 to 18th October 2002 for testing the estimated model. The monthly share price data display irregular, randomcity, non seasonal and non-linear behavior that shows the process is non stationary. Therefore using suitable transformation data is converted into stationary process which we will describe in the next section.

III.1 Neural Network Modeling:

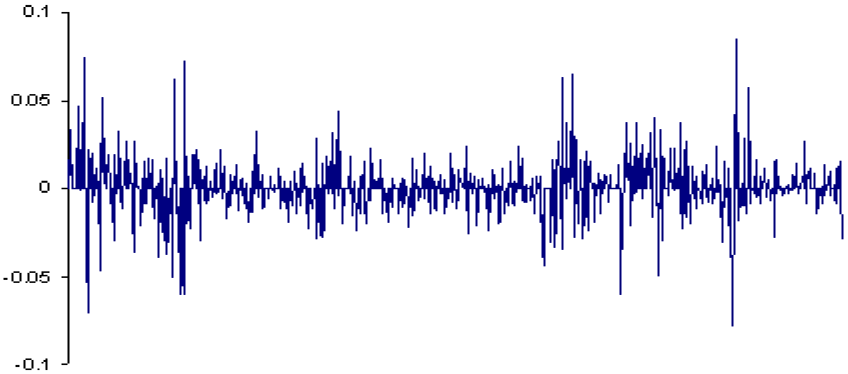
The neural network development followed several steps.

1. Selection of the input and output variables, number of hidden layers, and number of hidden neurons, training algorithms, activation functions and initial weights. The selection of hidden layer and hidden neuron is difficult as large numbers of hidden neurons make the network complicated while with a few hidden neurons the network is unable to learn the relationship among the data and the error will not reach an acceptable level.
2. Division of the data set into three sets, network training, validation and testing sets.
3. For evaluation of the forecast performance MSE was used.
4. Step 1 to 3were repeated until the error reaches the acceptable level.

In ANN modeling, we used transformation which stabilized the variance. where

where $\{x_t\}$ is the series of KSE100 index daily share prices data.

Fig.3. Transformed Data of KSE 100 Daily Index 2002 – 2002



We built many ANN models by taking different input nodes and different hidden nodes keeping one output node and found that 7-1-1(seven input nodes, one hidden node and one output node) was suitable for KSE100 index data. As MSE was least in 7-1-1among all ANN models. Therefore the ANN model, which we used in our experiment, is single layer with seven input nodes, one hidden node and one output node using non sigmoid logistic function. The data from 1st January 2000 to 18th October 2002 were divided into three subsets: a training set 1st Jan 2000 to 11th October 2002 and validation 7th Jan 2000 to 11th October 2002 and test set 12th October 2002 to 18th October 2002. The training subset was used to estimate the parameters. The second subset, validation set was used to monitor the estimation process. The third data set was not used for estimation, but was used for comparing different models. Network training was started with small random weights and propagated the error backward in the network for adjustment of weights by using back propagation learning algorithm which we have discussed in section 1.1 The network training was started with seven input lagged values $y_{t-1}, y_{t-2}, y_{t-3}, y_{t-4}, y_{t-5}, y_{t-6}, y_{t-7}$ that gave the model. Using that model we predicted $\widehat{y}_{t-8}$.Computed validation error of the y_{t-8} (lagged value) and $\widehat{y}_{t-8}$ (predicted value) using error sum of squares. Similarly second set $y_{t-2}, y_{t-3}, y_{t-4}, y_{t-5}, y_{t-6}, y_{t-7}, y_{t-8}$ inputs using model of this training data predicted $\widehat{y}_{t-9}$ and then computed validation error of $\widehat{y}_{t-9}$ and y_{t-9}. This process was continued until all data were used. We selected the minimum of all the validation errors and used it for estimating parameters of the selected model. We used MSE for comparing different models. Before computing MSE, we transformed the neural net output into original data scale by using the transformation $\hat{x}_t = x_{t-1}e^{y_t}$.

The neural network model is given below.

$$
\begin{aligned}
y_t =& 0.00028456 + 0.0121466\ y_{t-1} + 0.0549527\ y_{t-2} + 0.0374384\ y_{t-3} - 0.024525\ y_{t-4} \\
& - 0.0139442\ y_{t-5} - 0.0035911\ y_{t-6} + 0.0145222\, y_{t-7} - 1.80486\ \mathrm{g}\big(-12.28 + 14.0093\ y_{t-1} \\
& - 0.00288558\ y_{t-2} + 0.0354713\ y_{t-3} + 0.142565\ y_{t-4} \\
& + 7.01083\ y_{t-5} + 0.0985209\ y_{t-6} \\
& + 4.12412 y_{t-7}\big).
\end{aligned}
$$

Where $\mathrm{g}\,(x) = \dfrac{1}{1+e^{-x}}$ is logistic function.

The network has 17 parameters.

We checked the accuracy of the model by MSE = $\dfrac{\sum_{t=1}^{m} (y_t - \hat{y}_t)^2}{7}$ = 3.89914.

For building above model we used Mathematica4® packages.

3.2 Simple Exponential smoothing modeling:

For simple exponential smoothing we used Eviews3® package, on the basis of trail and error method best value of α is 0.999 for KSE100 index data.

Simple Exponential Smoothing model of KSE100 index is $F_{t+1} = 0.999\ \ Y_t + (1-0.999)\ \ F_t$.

Where Y_t is the most recent actual value, F_t is the latest forecast, F_{t+1} is the forecast for the next period, and α is the smoothing constant.
Table 2 shows the corresponding Sum of Squared Residuals and Root Mean Squared error after fitting the model on KSE100 index daily share price.

Table 1: Sum of Squared Residuals and

Root Mean Squared Error of the fitted mod

Sum of Squared Residuals	522309.6
Root Mean Squared Error	22.68459

Forecast mean square error is= $\frac{\sum_{t=1016}^{1022}\left(F_t - \hat{F}_t\right)^2}{7}$ = 1956.4549

3.3 The Hybrid system forecasting Methodology

Exponential smoothing model capture linear characteristics of the financial data (KSE100 index daily share price data), while ANN capture nonlinear patterns of the financial data. To get more satisfactory forecasting results we developed hybrid system, in which ANN combined with SES.

Study of our proposed hybrid system is divided into two phases (See figure3). The first phase of the study describes the modeling of KSE100 index from 1st January 2000 to 18th October 2002 using SES model, which we have discussed in section 3.2. While the second phase of study is Hybrid financial system, which is obtained by feeding the predicted data of SES into the ANN model for forecasting of KSE100 daily share price outputs.

Figure.3 shows how our proposed hybrid system works to predict or forecast a time series daily share price data.

Figure 4. Hybrid System for Forecasting KSE 100 Index

For ANN_{SES} input values were the predicted values of SES model but the building process of the hybrid system of ANN_{SES} was the same as ANN process, which we have discussed in section 3.1.

Table 2. FMSE of SES and ANN

Model	F MSE
SES	1956.4549
ANN	3.89914

Table 3. FMSE of ANN $_{SES}$ and ANN

Model	F MSE
ANN$_{SES}$	3.8928
ANN	3.89914

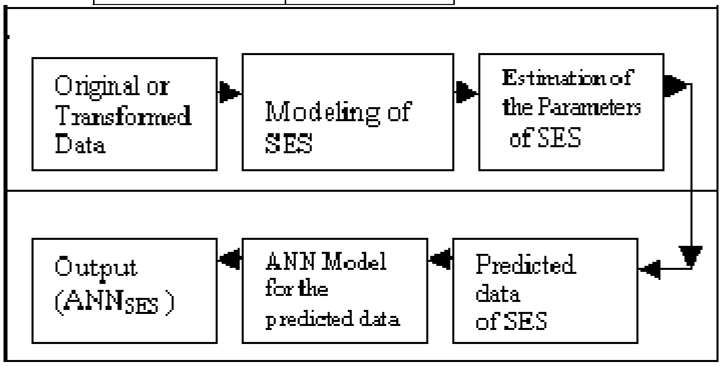

Original or Transformed Data
Modeling of SES
Estimation of the Parameters of SES
Predicted data of SES
ANN Model for the predicted data
Output (ANN$_{SES}$)

Table 2. FMSE of SES and ANN

Model	F MSE
SES	1956.4549
ANN	3.89914

Table 3. FMSE of ANN $_{SES}$ and ANN

Model	F MSE
ANN$_{SES}$	3.8928
ANN	3.89914

4. Conclusion:

In this study we compared ANN with SES model. We also proposed a hybrid system that combined simple exponential smoothing and Artificial Neural Network for KSE100 index daily share price. Table 2 shows that FMSE of ANN is less as compared to SES therefore, ANN model out played SES model. We compared our proposed hybrid system with standard ANN in table 3, the experimental result supports our proposed methodology. However, the difference between standard ANN and hybrid system of ANN$_{SES}$ are not highly significant. Hybrid system of ANN$_{ARCH/GARCH}$ was superior to ANN and ANN$_{ARIMA}$ in forecasting for the same period of KSE100 index daily share price data by Samreen et. al (2006).

Future work:

For further research we suggest.

i) Other learning should be used in place of supervised learning.

ii) GARCH variants should be used for such data.

iii) Behavior of relationship among Islamabad, Lahore and Karachi stock exchanges can be studied. .

References:

Bryson, A.E and Y. -C .Ho (1969). Applied Optimal Control. New York: Blaisdell

Franses, P.H. and Homelan, V.P. (1998) On Forecasting exchange using Neural Networks, Journal of Applied Financial Economics, 8,589-596.

Gardner, E.S (1985) Exponential Smoothing: The State of the Art. Journal of Forecasting 4 1-28.

Kuan, C.M. and T. Liu (1995). Forecasting Exchange Rates using Feed forward and

Recurrent Neural Networks. Journal of Applied Econometrics 10 (4): 347–64.

Leung, M.T., Daouk, H., Chen, A.S (2000). Forecasting Stock Indices: A Comparison of Classification and Level Estimation Models. International Journal of Forecasting 16 ,173- 190.

Nikola Gradojevic and Jang Yang (2000). The Application of Artificial Neural Networks to Exchange rate Forecasting: The Role of Market Microstructure variable. Bank Canada working paper.

Parker, D.B (1985).Learning Logic. TR-47, Center for Computational Research in Economics and Management Science, Massachusetts Institute of Technology, Cambridge, MA.

Rumelhart, D.E., G.E. Hinton, and R.J. Williams (1986a). Learning Representations by Back-Propagation Errors. Nature 323, 533-536. Reprinted in Anderson and Rosenfeld (1988).

Samreen Fatima and Ghulam Hussain(2001). Use of Artificial Neural Network for predicting KSE100 index: A comparative study .proceedings: Eight Statistical seminar 2001.ISBN-969-8397-07-8.

Samreen Fatima and Ghulam Hussain (2006). Statistical models of KSE100 index using Hybrid Financial Systems. Proceedings: IEEE International Conference on Engineering of Intelligent Systems (ICEIS 2006).

Spyros G. Makridakis , Steven C.Wheelwright, Rob J.Hyndman (1998). “Forecasting methods and applications (3rd edition)”.John Wiley & Sons Inc. ISBN: 0471532339.

Swanson, N.R and White, H. (1995).A model Selection approach to assessing the information in term structure using linear and Artificial Neural Networks. Journal of Business and Economics statistics, 13, 265-75.

Werbos, P. (1974). Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences. Ph.D. Thesis, Harvard University.