IC Value 2016 : 61.33| SJIF Impact Factor(2018) : 8.003| ISI Impact Factor (2017):1.365 (Dubai)

Research Paper | Volume - 6, Issue- 7, July 2018 | e-ISSN : 2347 - 9671| p- ISSN : 2349 - 0187

EPRA International Journal of Economic and Business Review

OPTIMAL TECHNICAL TRADING RULE FOR STOCK PRICES USING PAIRED MOVING AVERAGE METHOD PREDICTED BY ARIMA AND ANN MODELS

R. Sivasamy
Professor, Department of Statistics, University of Botswana, Gaborone, Botswana.

Peter O. Peter
Phd Research Scholar, School of Mathematical Sciences, Shanghai Jiao Tong University, Shanghai, China.

ABSTRACT

KEYWORDS:
ARIMA model; ANN model; MA values; predicted prices; OTTR R(t) ratio; Positional profit.

Success of any trade depends on the ability to spot and profit from market swings associated with prices $\{x_t: 1, 2, \ldots, N\}$ of a stock. In this paper an optimal technical trading rule (OTTR) is proposed to identify profitable positions for 'when to buy and when to sell' to help all traders who live and die with minute-by-minute price data. Furthermore a trading rule $G_{SL}(t)$ that assigns selling positions with an upper level price and buying positions with a lower level price is formulated by monitoring the ratio series $R(t)=MA_S(t)/MA_L(t)$ where, $S < L$ with $MA_S(t)$ and $MA_L(t)$ as simple moving averages (MAs) computed from the stock series $\{x_t\}$ under study. We denote the mean and standard deviation measures of the $R_{SL}(t)$ series by 'm' and 's' respectively and the upper level positions (ULPs) are selected above the mean at time 't' if ($R_{SL}(t) > m+ks$, $R_{SL}(t-1) < m+ks$) and lower level positions (LLPs) below the mean are chosen at time 't' if ($R_{SL}(t-1) > m-ks$, $R_{SL}(t) < m-ks$), defining a trading rule $G_{SL}(t)$. A combination (S^*, L^*, h^*) that maximises the total expected profit $P_{SL}(t, h)$ over the positions determined by the OTTR is selected as the 'Optimal technical trading rule (OTTR(S^*,L^*, h^*))' for this investigation. To implement the proposed methodology pertaining to this rule, a training data set and testing data set are simulated and an appropriate model is fitted by hybrid-Auto Regressive Integrated Moving Average (hybrid-ARIMA) and Artificial Neural Network (ANN) methods. Using the estimated values of the parameters by hybrid-ARIMA and NN methods, predictions are made for testing data set. From these predicted values, OTTR(S^*, L^*,h^*) for both hybrid-ARIMA and ANN approaches are obtained and the corresponding maximum profits are compared.

INTRODUCTION

Traders participate in financial markets for buying and selling stocks, futures, forex and other securities through positions, each with an opening and closing out days with the intention of making frequent gains or returns. Technical traders often believe that they have all of the information necessary to make an informed trade by viewing the past trade and price history of a stock. Often they rely on stock charts that are constructed based on trading information such as previous prices and trading volume, plus mathematical indicators.

Several studies on exploring technical analysis have been published in the last five decades. Rodolfo et al. (2017) have well accounted the existing literature on technical analysis by presenting an overview of characteristics of the literature, potential knowledge gaps and focusing on the analysis of stocks and future research in this area. A motivating factor comes from the technical trading method developed by Netfci (1991) showing that most patterns used by technical analysts need to be characterized by appropriate sequences of local minima and/or maxima called signals of market turning points and more often results in nonlinear prediction problems. Similar studies, using technical indices such as moving averages (MA) and relative strength indices (RSI), have been extensively investigated by Asadi et al. (2012) and Chang and Fan (2008). The applications of text mining techniques are discussed in Lo et al. (2000) to find the important information from news articles.

Zhu and Zhou (2009) discuss the usefulness of technical analysis, specifically the widely employed moving average trading rule from an asset allocation perspective. The authors' further show that when stock returns are predictable, technical analysis adds value to commonly used allocation rules that invest fixed proportions of wealth in stocks. Vanstone and Finnie (2009) have presented a simple methodology for designing, and testing stock market trading systems through a number of models developed using soft computing technologies, specifically artificial neural networks. Ebrahimpour *et al.* (2011) use mixture of Multilayer Perceptron (MLP)-experts for trend forecasting of time series with a case study of the Tehran stock exchange while Sermpinisa *et al.* (2014) employs MLP in trading and hybrid time varying leverage effects.

In this paper, technical data on closing prices of an auto-regressive (AR) stock are simulated and analysed by hybrid-ARIMA and ANN methods. A comparative study has been done in order to high light the advantages of the proposed OTTR. According to the features shown in the performance evaluation, a best profitable stock trading system with price prediction can be achieved using the proposed hybrid-ARIMA-ANN models. This paper introduces a paired moving average method (PMAM) to suggest the best times for daily trading of a single stock market prices.

METHODOLOGY for Model building by ARIMA-ANN approaches and Trading

This paper proposes an effective trading rule based on the concept of market timing that can enhance profitability or reduce losses. Consider an observed set $\{x_t: t=1, 2, \ldots, N, (N+1), \ldots, (N+hh)\}$ of closing price series for a single stock, with (N+ hh) being current day. Divide the data set into two subsets. The subset containing the first N values of $\{x_t: t=1, 2, \ldots, N\}$ is used for fitting a model by both model builders ARIMA and ANN and the estimates of parameters for each model fitting re derived. The second subset that includes 'hh' values $\{x_t: t=N+1, N+2, \ldots, N+hh\}$ is used for future predictions using the appropriate estimates for each method.

Let $MA_k(t)$ denote the value of a moving average of length 'k' as defined in (1) for a given financial series x_t,

$$MA_k(t)=\begin{cases} \frac{1}{k}\sum_{j=0}^{k-1} x_{(t-j)}; & t \ge k \\ \frac{1}{t}\sum_{j=0}^{t-1} x_{(t-j)}; & t < k \end{cases} \quad \ldots (1)$$

Let S and L denote the time span of the short length moving average $MA_S(t)$ and long period moving average $MA_L(t)$ series where $S \geq 2$ and $L > S$. The main focus of this paper is on the ratio series R(t) computed from a pair of moving averages MAS(t) and MAL(t) of lengths S and L respectively i.e.

$$R(t) = R_{SL}(t) = \frac{MA_S(t)}{MA_L(t)}, \quad t=1, 2, \ldots, N \quad \ldots (2)$$

and a rate of h-day ahead returns series Z(t+h) defined for h=1, 2, …, (N-h)

$$Z(t+h)= \frac{x_{t+h} - x_t}{x_t} \quad \ldots (3)$$

where "h' is an optional integer value to be decided by the trader. The main objective here is to determine a best combination (S^*, L^*, h^*) from among the possible triplets (S, L, h) that maximizes an objective profit function $P_{SL}(R_{SL}(t), Z_h(t))$. To achieve this goal, we develop an algorithm to determine a feasible trading rule $G_{SL}(t)$ by monitoring the ratio series R(t) around few extremes obtained by its mean m = mean of R(t) and s = standard deviation of R(t) as appropriate times to sell or keep (do nothing) existing stocks or buy the stock.

Suppose the observed closing stock price series $\{x_t\}$ is traded according to a trading policy $G_{SL}(t)$ for maximizing the profit $P_{SL}(t, h)$ based on the distance of $R_{SL}(t)$ from its mean value 'm' and the rate of return $Z_h(t)$:

$$G(t)=\begin{cases} -1 & \text{signal to sell existing share for a high price} \\ 0 & \text{signal to do nothing} \\ 1 & \text{signal to buy new shares for a low price} \end{cases} \quad \ldots (4)$$

As forecasts from stationary models revert to mean, let us now introduce an ideal value 'k' to define positions for buying and selling days of the stock through the measures 'm and s' and thus the trading rule $G_{SL}(t)$ without using the information about $Z_h(t)$,

$$G_{SL}(t)=\begin{cases} -1 & \text{if } R_{SL}(t) \ge m + k\,s \text{ and } R_{SL}(t-1) < m + k\,s \\ 1 & \text{if } R_{SL}(t) \le m - k\,s \text{ and } R_{SL}(t-1) > m - k\,s \\ 0 & \text{otherwise} \end{cases} \quad \ldots (5)$$

Profit Function: Let us now make a precise decision at time 't' with regard to profit through the function $P_{SL}(t, h)$:

- if $G_{SL}(t)=1$ (i.e. buy signal) and $Z(t+h) < 0$ $\Rightarrow x_t > x_{t+h}$ then it does not help the trader to buy a low price which gives a loss of - Z(t+h) (> 0) or profit= Z(t+h)
- if $G_{SL}(t)=1$ (i.e. buy signal) and $Z(t+h) > 0$ indicates that $x_t < x_{t+h}$ then it helps the trader to buy a low price which gives a positive profit Z(t+h)
- if $G_{SL}(t)= -1$ (i.e. sell signal) and $Z(t+h) < 0$ then it helps the trader to sell a high price stock to make a positive amount –Z(t+h) of profit
- if $G_{SL}(t)= -1$ and $Z(t+h) > 0$ then the trader has to sell a low price stock to face loss amount of Z(t+h) or profit= - Z(t+h)

The profit function $P_{SL}(t,h)$ is defined as follows:

$$P_{SL}(t,h) = \begin{cases} Z(t+h) & \text{if } G_{SL}(t)=1,\ Z(t+h)<0 \\ Z(t+h) & \text{if } G_{SL}(t)=1,\ Z(t+h)>0 \\ -Z(t+h) & \text{if } G_{SL}(t)=-1,\ Z(t+h)<0 \\ -Z(t+h) & \text{if } G_{SL}(t)=-1,\ Z(t+h)>0 \\ 0 & \text{if } G_{SL}(t)=0 \end{cases} \quad \ldots (6)$$

Let β denote the total number of times the ratio R(t) series takes the extreme values falling outside the boundary limits $m \pm ks$ for t=1,2, ,…, (N-h). Then the total expected rate of return (TERR) is

$$TERR(S, L, h) = \frac{1}{\beta}\sum_{t=1}^{N-h} P_{SL}(t, h) \quad \ldots (7)$$

Thus (7) is a typical type of optimization problem, a solution of (7) that gives a maximum from among various combinations assigned by a rule to the vector (S, L, h) is called an optimal technical trading rule OTTR(S*, L*, h*).

ANN Architecture versus Hybrid-ARIMA Modelling(s)

Many NN models are similar or identical to popular statistical techniques as each perceptron computes a linear combination of the inputs with a *bias* term called the *netinput* (j) to the j^{th} neuron of the hidden layer. A possibly nonlinear *activation* function 'f' maps any real value received from the input layer into an image f(*netinput*(j)) as a bounded value, often between -1 to 1. In an MLP, each net input to the hidden layer is a linear combination of the inputs as specified by the weights w_{ij} called connecting weights from a neuron labelled as 'i' in the input layer to the j^{th} neuron of the hidden layer where the bias term is 'w_{0j}'. The final outputs that fall close to our targeted values are then computed as linear combinations of the hidden images f(.) with an identity linear function. Diagrammatic representations of NN(5,3,1) and NN(3,0,2) neural networks with their connecting weights (assigned randomly) are displayed in Figure 1 and Figure 2 respectively.

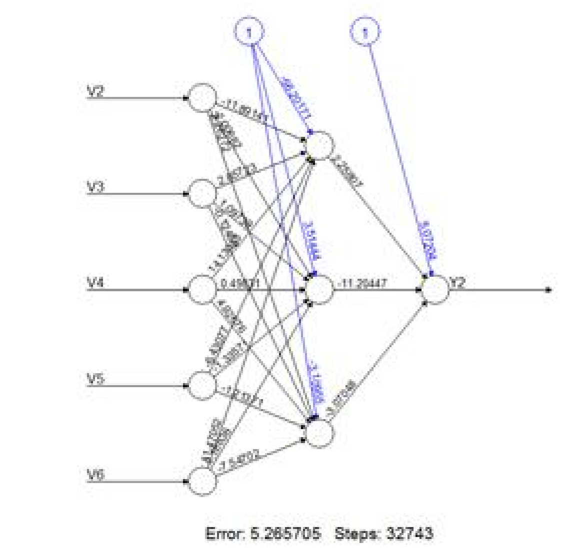

Figure 1: Representation of NN(5,3,1) neural network

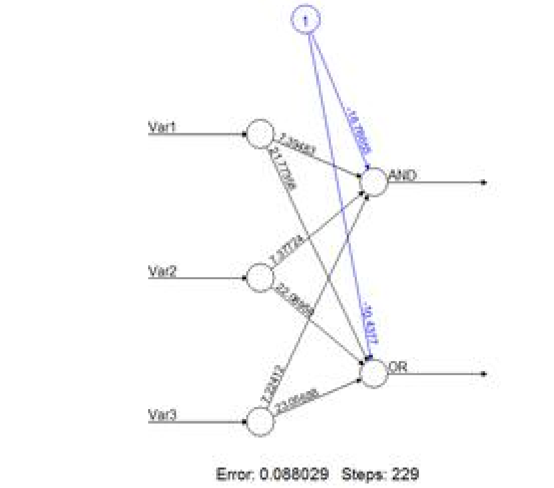

Figure 2: Representation of NN(3,0,2) neural network

As the series $\{x_t\}$ is expected to have a linear autocorrelation part L_t and non-linear component NL_t, we assume an additive structure as

$$x_t = L_t + NL_t \qquad \text{.... (8)}$$

The linear component L_t is estimated as $\hat{L}_t$ by ARIMA model. Passing the command ar = auto.arima(X), X = ts(x[1:N1]), the fitted model is detected as AR(1): $x_t = \varphi_0 + \varphi_1 x_{t-1} + a_t$ resulting in the estimated parameters as mean = $\hat{\mu} = \hat{\varphi}_0 / (1-\hat{\varphi}_1)$ =7.065938, $\hat{\varphi}_1$ =0.892105 for which the details and the predictions for the testing data set are available in the Appendix A. Let e_t denote the residual of (8) then

$$NL_t = x_t - \hat{L}_t \qquad \text{... (9)}$$

It has been advised by several authors see Zhang (2001), that one can capture non-linear features, say $\hat{NL}_t$, by employing an ANN modelling architecture to the residuals NL_t of (9) effectively than any other alternative method. The model so constructed is called the hybrid ARIMA-ANN model which gives the final forecast of x_t as

$$\hat{x}_t = \hat{L}_t + \hat{NL}_t \qquad \text{... (10)}$$

Under the neural network NN (p, q, 1) method, number of input patterns used is n = N-p during the training period for fitting the following model to the given time series of stock prices $\{x_t : t=1, 2, \dots, N\}$,

$$x_{t+p} = f(x_t, x_{t+1}, \dots, x_{t+p-1}, W), \quad t = 1, 2, \dots, (N-p)$$

$$= w_0 + \sum_{j=1}^{q} w_j \left\{ f\left(W_{0j} + \sum_{i=1}^{p} W_{ij}\, x_{t+i-1} \right) \right\} \qquad \text{... (11)}$$

Once the network weights and biases have been initialized, the network is ready for training. The training process requires, initial network weights and biases, a set of examples/patterns (called target outputs (N-p)) each with network inputs p from $\{x_t\}$ series. During training the weights and biases of the network are iteratively adjusted to minimize the network performance function. For each forecast $\hat{x}_{t+p}$ of x_t, an algorithm of fitting the NN(p, q, 1) of (11), estimates $\hat{w}/\hat{W}$ of all the connection weights w/W is derived by minimising the mean square error $S^2 = SS/(n-\tau)$, where SS is the sum of squares of the residuals

$$SS = \sum_{t=1}^{N-p} \left(x_{t+p} - \hat{x}_{t+p} \right)^2 \qquad \text{... (12)}$$

In order to avoid overfitting / under fitting, one has to use an optimum number of neurons in the hidden layer of the NN(p,q,1) architecture. For this purpose 'Akaike

Information Criterion (AIC)' and the 'Bayesian Information Criterion (BIC)' are used as minimization tool for isolating the best model NN(p, q^*,1) from among candidate NN(p,q,1) models.

AIC=n*ln(SS/n)+2*τ and BIC=n*ln(SS/n)+τ + τ*ln(n) … (13)

Forecasts of Hybrid-ARIMA and NN: To explore the issues discussed above, an illustration is now provided by simulating a stationary series x_t of size (36+20=) 56 through an 'R code'. This is then divided into a training set that includes the first 36 values X={x_t: t=1, 2, … , 36} and a testing set of 20 values {x_t: t=37, 38, …, 56}. An empirical study, for the same 36 values X={x_t: t=1, 2, … , 36}, is performed for testing the predictive effects of hybrid-ARIMA and ANN models. The corresponding predictions for the targeted series {x_t: t=37, 38, …,56} of 20 items are then obtained using the model fitted by both methods and the same values are exhibited in Appendix A. As the minimum AIC corresponds to the N(1,1,1) system, from this system predictions are made for the testing set of 20 values. To compare the forecasting abilities of the two approaches hybrid-ARIMA and NN, those 20 values predicted for the test data {xt: t=37, 38 …, 56) are displayed the in Figure 3.

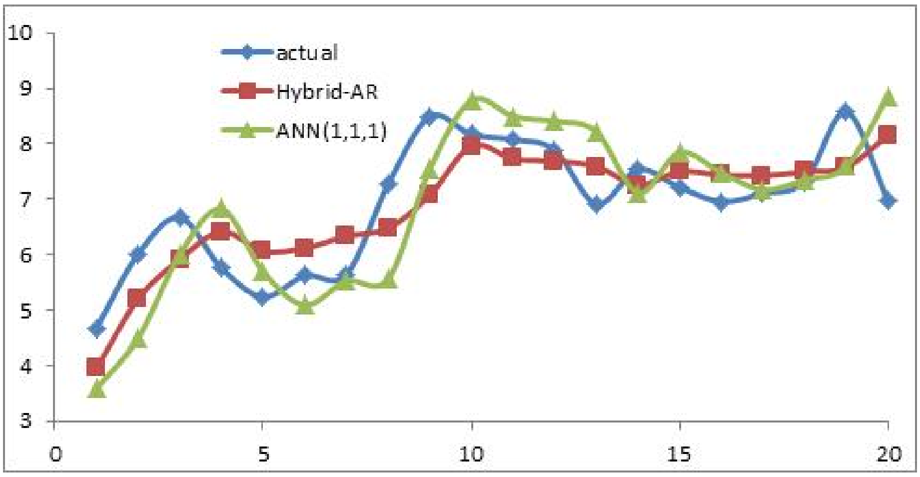

Figure 3: Showing the graphs of actual series against the predicted series by Hybrid-ARIMA and NN Models

From the figure above ('Figure 3'), it can be noted that all predicted values of NN and hybrid-ARIMA move closely with the original series. This happens since both NN hybrid-ARIMA methods have captured linear and non-linear characteristics of the given series with same amount of accuracy and prediction.

Selection of optimal trading rule using technical data

This section deals with the construction of OTTR(S=2, L, h) for deciding the optimal values L^* of L and h^* of h giving the maximum profit with the same trading set X={x_t: t=1, 2, … , 36} considered for model fitting. For the choices of S = 2, L= 4, values of TEER(S, L, h) for different values of h from 1 to 10 are computed and compared as reported under Table 1:

Table 1: Trading Days and Profits for either selling or Buying for the period 1 to 34(=N-h) days with k=0.2, h=2 and N=36					
Day	Signal	Rate of Return	Day	Signal	Rate of Return
3	sell	-0.284174	17	sell	-0.066084
8	buy	0.37722	24	buy	0.15382
10	sell	0.2285	26	sell	0.25135
12	buy	0.04153	28	buy	-0.303536
14	sell	0.01098	34	sell	0.37378
16	buy	0.13622			
				Total	**0.919606**
Mean for 11 trading days=TERR(2,4,2)=0.083601					

Plotting the results from Table 1 above is provided in Figure 4 below to show the ideal days for selling high price stock or buying the low price stock of X={x_t : t=1, 2, … , 36} when h = 2, S = 2 and L= 4

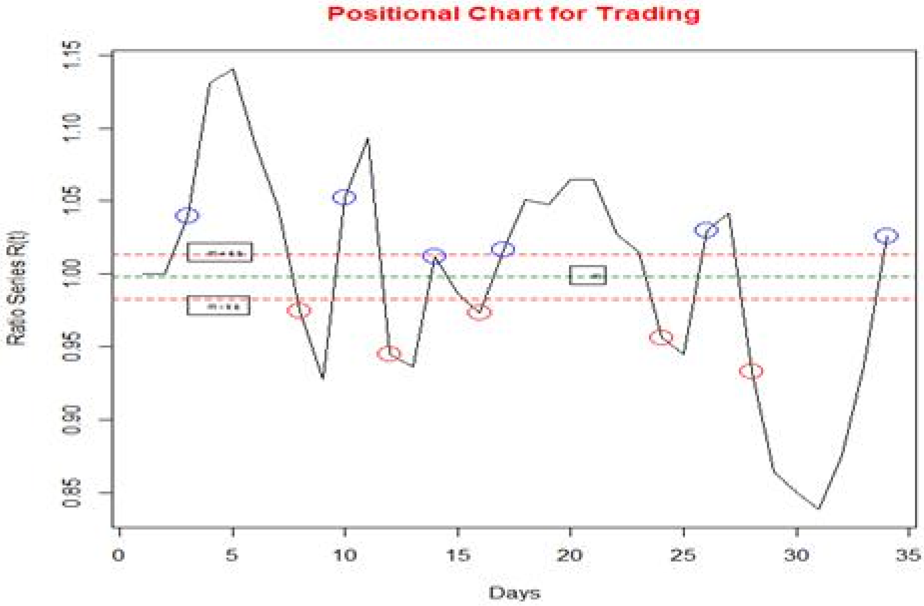

Figure 4: Ratio plot called 'Positional Chart' for 34(=N-h) values fitted by Hybrid-ARIMA Model to the observed data { x_t : t=1, 2, …, N(=36)} with S=2, L=4 and h=2. This graph shows the best days for trading. Blue coloured circles that appear above the upper line indicate signals for selling and red coloured circles that appear below the lower line trading days for buying.

Expected profits for the Predicted values: We now use the predicted values $\hat{x}_t$ of {x_t: 36, 37, …,56} through models fitted by Hybrid-ARIMA and NN(1,1,1). All these predictions and the h-day rate of returns (h = 2) are given in the Appendix A. For the combination (S=2, L=4, h=2), and the corresponding maximum profits are computed and reported in Table 2.

Table 1: Optimal Trading Days for the 20 predicted values

Day	AR:Zh(t)	Signal	Profit	Day	NN:Zh(t)	Profit
1	0.4974	Buy	0.4974	1	0.6749	0.6749
3	0.0245	Sell	-0.0245	3	-0.0528	0.0528
6	0.0587	Buy	0.0587	6	0.0916	0.0916
9	0.0927	Sell	-0.0927	9	0.1266	-0.1266
12	-0.0554	Buy	-0.0554	13	-0.0455	-0.0455
	TERR(ARIMA)		0.3835		TERR(ANN)	0.6471

Profit values are calculated according to the proposed trading rule for the set of 20 predicted values which are displayed under 'Profit' column in Table 2. Both methods produce a common set of 4-days (1, 3, 6, 9) but the fifth day for the Hybrid-ARIMA method is the 12th day and that for the NN method is the 13th day. In these type of comparisons, the sign of the profit is also important other than the magnitude of the total expected profit value. All these factors jointly decide that the NN method performs better that the hybrid-ARIMA method in obtaining optimal trading rule investigations based on technical data.

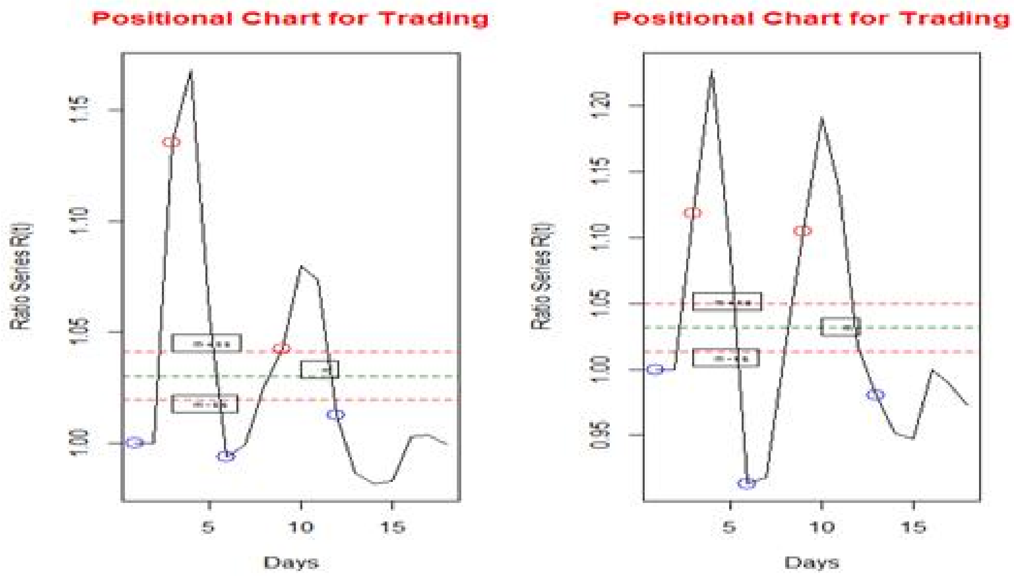

Figure 5: Optimal trading days for 20 predicted values by Hybrid_ARIMA and NN methods

A graph is also drawn in Figure 5 to explore the advantages of employing the Hybrid-ARIMA and NN(1,1,1) methods in linear and non-linear predictions and to trace the optimal days for future trading of a single stock.

CONCLUSION

The aim of this article was to formulate an optimal technical trading rule (OTTR) using the predicted series by hybrid-ARIMA) and ANN models. This approach creates a valid model for stock market prediction. Each method develops an optimal prediction algorithm to produce best future values for closing prices of a stock based on past data sets (technical data) only. Maximization of the theoretical profit of OTTR is determined by a mean profit function that maximizes the total profit. The loss or gain is measured jointly by h-day-ahead rate of returns and the signals detected through a chart representing the ratio series $R(t)=MA_S(t)/MA_L(t)$. For illustrative purposes, technical data on closing prices of stock are simulated and used. A comparative study has been done in order to high light the advantages of the proposed OTTR through a positional chart reflecting the changes of the ratio series R(t) . It is shown that the proposed ANN architecture generalizes better than the hybrid-ARIMA method. According to the features shown in the performance evaluation, a best profitable stock trading system with price prediction can be achieved using the proposed hybrid-ARIMA-ANN models.

APPENDIX - A

Simulation of $x_t = \varphi_0 + \varphi_1 x_{t-1} + a_t$ of size (N_1+hh, N_1=36, hh=20) : $a_t \sim N(0,1)$

```
set.seed(41)
phi=0.92 # AR(1) coefficient
x=0.5 # value of x_0
X=c() # generated AR(1) process #x_0+ phi x(t-1)+e_t#
for (i in 1:(N1+hh)){  # length of the process is 47(N+h)
x= phi*x + rnorm(1) # Gaussian innovations
X=c(X,x)
}
ra=range(X)
X=X+ ra[2]+0.1
N2=N1+hh
sts=ts(X [1:N1])
# Forecast using ARIMA method
sts=ts(X[1:N1])
ar = auto.arima(sts)
ar
testAR=tail(X12,hh)
testAR=data.frame(testAR)
fc.arima = forecast(ar,hh)
fc.arima
```

Day	Actual	AR(1)	ANN(1,1,1)	Z_h(AR)	Z_h(ANN)
1	4.6798	3.9526	3.5890	0.4974	0.6749
2	6.0176	5.2101	4.5004	0.2320	0.5172
3	6.6753	5.9186	6.0111	0.0245	-0.0528
4	5.7581	6.4190	6.8282	-0.0464	-0.2542
5	5.2422	6.0633	5.6940	0.0449	-0.0287
6	5.6215	6.1214	5.0922	0.0587	0.0916
7	5.6452	6.3353	5.5304	0.1170	0.3639
8	7.2696	6.4806	5.5585	0.2290	0.5806
9	8.4933	7.0764	7.5428	0.0927	0.1266
10	8.1725	7.9644	8.7860	-0.0349	-0.0430
11	8.0781	7.7324	8.4980	-0.0178	-0.0326
12	7.8899	7.6863	8.4077	-0.0554	-0.1535
13	6.9113	7.5945	8.2206	-0.0126	-0.0455
14	7.5379	7.2603	7.1173	0.0247	0.0511
15	7.2167	7.4991	7.8467	-0.0095	-0.0855
16	6.9594	7.4394	7.4813	0.0079	-0.0177
17	7.1040	7.4278	7.1755	0.0208	0.0591
18	7.3185	7.4982	7.3486	0.0876	0.2055
19	8.5797	7.5820	7.5993		
20	6.9827	8.1547	8.8586		

ANN model: NN (1, 1, 1), n = 35 called number of input patterns during training the ANN, 4 number of nodes/neurons in the hidden layer, one output layer, number of parameters estimated=(1+2)4+1=13=τ,say. Sigma =sqrt(SS/(n-tow))= 1.076886612, AIC=n*ln(SS/n)+2*τ=8.937557827 BIC=n*ln(SS/n)+τ + τ*ln(n)= 19.15895007 where SS=residual sum of squares.

ACKNOWLEDGEMENTS

The authors would like to thank their colleagues for their valuable comments and suggestions.

REFERENCES

1. *Asadi, S., E. Hadavandi, F. Mehmanpazir, M.M. Nakhostin,(2012): Hybridization of evolutionary Levenberg–Marquardt neural networks and data pre-processing for stock market prediction, Knowl.-Based Syst. 35 (11) (2012) 245–258.*
2. *Chang, P. C., and C.Y. Fan (2008): A hybrid system integrating a wavelet and TSK fuzzy rules for stock price forecasting, IEEE Trans. Systems, Man Cybern. C: Appl. Rev. 38(6) (2008) 802–815.*
3. *Ebrahimpour, R., Nikoo, H., Masoudnia, S., Yousefi, M. R. , and M. S. Ghaemi, (2011): "Mixture of mlp-experts for trend forecasting of time series: a case study of the tehran stock exchange," International Journal of Forecasting, vol. 27, no. 3, pp. 804–816.*
4. *Netfci, S. N. (1991). Na1ve trading rules in financial markets and Wiener–Kolmogorov prediction theory: A study of technical analysis. Journal of Finance, 64, 549–571.*
5. *Lo, W., Mamaysky, H., and J. Wang, (2000): Foundations of technical analysis: computational algorithms, statistical inference, and empirical implementation, J. Finance 55(4) (2000) 1705–1765.*
6. *Rodolfo Toríbio Farias Nazárioa, Jéssica Lima e Silva b, Vinicius Amorim Sobreiroa, Herbert Kimuraa, A literature review of technical analysis on stock markets, The Quarterly Review of Economics and Finance 66 (2017) 115–126.*
7. *Sermpinisa, G., Stasinakisa, C., and C. Dunisb,(2014) : "Stochastic and genetic neural network combinations in trading and hybrid time-varying leverage effects," Journal of International Financial Markets, Institutions & Money, vol. 30, pp. 21–54.*
8. *Vanstone , B. and Finnie, G (2009) :An empirical methodology for developing stockmarket trading systems using artificial neural networks, Expert Systems with Applications 36 , 6668–6680*
9. *Zhang, G. P., Patuwo, B. E., and Hu, M. Y. (2001). A simulation study of artificial neural networks for nonlinear time-series forecasting. Computers & Operations Research, 28(4), 381-396.*
10. *Zhu, Y. and G. Zhou (2009): Technical analysis : An asset allocation perspective on the use of moving averages , Journal of Financial Economics 92 (2009) 519–544*