Expert Systems with Applications 36 (2009) 6800–6808

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Development and performance evaluation of FLANN based model for forecasting of stock markets

Ritanjali Majhi [a], G. Panda [b,*], G. Sahoo [c]

[a] Centre of Management Studies, National Institute of Technology, Warangal, India
[b] Department of E&CE, National Institute of Technology, Rourkela, India
[c] Department of CSE, BIT, Mesra, Ranchi, India

ARTICLE INFO

Keywords:
Forecasting
Stock market
Artificial neural network
Functional link artificial neural network(FLANN)

ABSTRACT

A trigonometric functional link artificial neural network (FLANN) model for short (one day) as well as long term (one month, two months) prediction of stock price of leading stock market indices: DJIA and S&P 500 is developed in this paper. The proposed FLANN model employs the least mean square (LMS) as well as the recursive least square (RLS) algorithms in different experiments to train the weights of the model. The historical index data transformed into various technical Indicators as well as macro economic data as fundamental factors are considered as inputs to the proposed models. The mean absolute percentage error (MAPE) with respect to actual stock prices is selected as the performance index to gauge the quality of prediction of the models. Extensive simulation and test results show that the application of FLANN to the stock market prediction problem gives out results which are comparable to other neural network models. In addition the proposed models are structurally simple and requires less computation during training and testing as the model contains only one neuron and one layer. Between the two models proposed the FLANN-RLS requires substantially less experiments to train compared to the LMS based model. This feature makes the RLS-based FLANN model more suitable for online prediction.

1. Introduction

Forecasting the price movements in stock markets has been a major challenge for common investors, businesses, brokers and speculators. As more and more money is being invested the investors get anxious of the future trends of the stock prices in the market. The primary area of concern is to determine the appropriate time to buy, hold or sell. In their quest to forecast, the investors assume that the future trends in the stock market are based at least in part on present and past events and data (Tan, Quek, & Ng, 2005). However financial time-series is one of the most 'noisiest' and 'non-stationary' signals present and hence difficult to forecast (Oh & Kim, 2002 and Wang, 2003).

The Dow Jones Industrial Average (DJIA) index was launched in 1896 with 12 stocks and is now the world's most often quoted stock exchange index, based on a price-weighted average of 30 significant companies traded in the New York Stock Exchange (NYSE) and NASDAQ. This index gives a general indication of the behavior of the market towards different information. Standard & Poor's 500, a basket of 500 stocks was created in 1957. It is weighted by market value, and its performance is thought to be representative of the stock market as a whole. The index selects its companies based upon their market size, liquidity, and sector. Most experts consider the S&P 500 as one of the best benchmarks available to judge overall U.S. market performance. Many researchers in the past have applied various statistical and soft computing techniques such as neural networks to predict the movements in these stock indices. Generally technical indicators like moving averages and relative strength indices derived from the time series of these indices is employed in this regard.

Financial time-series has high volatility and changes with time. In addition, movements of stock markets are affected by many macro-economical factors such as political events, firms' policies, general economic conditions, investors' expectations, institutional investors' choices, movement of other stock market, psychology of investors (Wang, 2002). Nevertheless there has been a lot of research in the field of stock market prediction across the globe on numerous stock exchanges; still it remains to be a big question whether stock markets can really be predicted and the numerous challenges that exist in its everyday application on the stock floor by the institutional investors to maximize returns. Generally there are three schools of thoughts regarding such prediction (Badawy et al., 2005). The first school believes that no investor can achieve above average trading advantages based on historical and present

* Corresponding author. Tel.: +91 661 2462455, mobile: +9437048906; fax: +91 661 2462963.
E-mail address: ganapati.panda@gmail.com (G. Panda).

0957-4174/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2008.08.008

information. The major theories include the random walk hypothesis and the efficient market hypothesis (Taylor, 1986). The second view is that of fundamental analysis. Analysts undertake indepth studies into the various macro-economic factors and look into the financial conditions and results of the industry concerned to discover the extent of correlation that may exist with the changes in the stock prices. Technical analysis presents the third view on market price prediction. Analysts attempt to extract trends in market using past stock prices and volume information. These trends give insight into the direction taken by the stock prices which help in prediction. Technical analysts believe that there are recurring patterns in the market behavior, which can be identified and predicted. In the process they use number of statistical parameters called technical indicators and charting patterns from historical data. As the underlying theory behind all these techniques is totally different they generally give quite contradictory results. More importantly, these analytical tools are heavily dependent on human expertise and justification in areas like, the location of reversal (or continuation) pattern, market pattern, and trend prediction. For such reasons researchers have stressed on developing models for accurate prediction based on various statistical and soft computing techniques.

One such statistical technique employed in this regard is the auto-regressive integrated moving average (ARIMA) based model(Schumann & Lohrbach, 1993). Different time-series in practice have different frequency components. However, there is no systematic approach or a suitable class of models available in the literature to accommodate, analyze and forecast time-series with changing frequency behavior via a direct method. The ARIMA model is obtained by differenciating an assumed non-stationary process to obtain a locally wide sense stationary and locally ergodic process.

The recent advancement in soft computing has given new dimension to the field of financial forecasting. Tools based on ANN have increasingly gained popularity due to their inherent capabilities to approximate any nonlinear function to a high degree of accuracy. Neural networks are less sensitive to error term assumptions and they can tolerate noise and chaotic components (Masters, 1993). Banks and financial institutions are investing heavily in development of neural network models and have started to deploy it in the financial trading arena. Its ability to 'learn' from the past and produce a generalized model to forecast future prices, freedom to incorporate fundamental and technical analysis into a forecasting model and ability to adapt according to the market conditions are some of the main reasons for its popularity.

A lot of research has gone into the development of models based on a range of intelligent soft computing techniques over the last two decades. Early models employed the multi layer perceptron (MLP) architecture using back propagation(BP) algorithm, while a lot of recent work is based on evolutionary optimization techniques such as genetic algorithms (GA). A brief review of work that has gone into the field of application of ANN to stock price prediction is discussed here.

In Japan, technology major Fujitsu and investment company, Nikko Securities joined hands to develop a stock market prediction system for TOPIX, the Tokyo based stock index, using modular neural network architecture (Kimoto et al., 1990). Various economic and technical parameters have been taken as input to the modular neural network consisting of multiple MLP used in parallel. A study has been made to investigate the effect of change of network parameters of MLP model on the stock price prediction problem (Clarence and Wittig, 1993). The paper gives insights into the role of the learning rate, momentum, activation function and the number of hidden neurons to the prediction. In addition to ANN using BP, the probabilistic neural network (PNN) has also been employed to stock prediction (Saad, Prokhorov, & Wunsch, 1998 and Tan, Prokhorov, & Wunsch, 1995). In this work, the model is used to draw up a conservative thirty day stock price prediction of a specific stock: Apple Computers Inc. Due to their bulky nature owing to the large training data, the PNN is not popular among forecasters. In the process lots of newer architectures has been reported in the literature. Ornes & Sklansky (Ornes & Sklansky, 1997) in their paper present a visual neural network (VNN), which combines the ability of multi expert networks to give low prediction error rates with visual explanatory power of nonlinear dimensionality reduction. They conclude that the VNN is a powerful means of interactive neural network design, which provides both better prediction accuracy and good visual explanatory ability. Another architecture introduced to the prediction problem is the multi branch neural network (MBNN) proposed by (Yamashita, Hirasawa, & Hu Jinglu, 2005) and has been applied to the TOPIX (Tokyo Stock Exchange). The simulations show that MBNN, based on the concept of universal learning networks (ULN), have higher accuracy of prediction than conventional NNs. In a recent paper, (Chen Yuehui & Zhao Yaou, 2005) investigate how the seemingly chaotic behavior of stock market could be well represented using local linear wavelet neural network (LLWNN) technique. They considered the NASDAQ-100 index and S&P CNX NIFTY index (India). The LLWNN is optimized by using estimation of distribution algorithm (EDA). Results show that the LLWNN model performs marginally better than conventional NN models. The hybrid architectures are also being deployed in recent times. Lee (2004) has proposed a hybrid radial basis function recurrent network (HRBFN) stock prediction system called the iJADE stock advisor. The stock advisor was applied to major Hong Kong stocks and produced promising results in terms of efficiency, accuracy and mobility. A second hybrid AI approach to the implementation of trading strategies in the S&P 500 index futures market is proposed by (Ray, Yenshan, & Lai Charles, 1998) which integrates the rule-based systems techniques with reasoning neural networks (RN) to highlight the advantages and overcome the limitations of both the techniques. They demonstrate that the integrated futures trading system (IFTS) based on this hybrid model outperforms other conventional NNs.

There are also instances of application of fuzzy logic based models to the stock market prediction as well. Hiemstra proposes a fuzzy logic forecast support system to predict the stock prices using parameters such as inflation, GNP growth, interest rate trends and market valuations (Hiemstra, 1994). According to the paper, the potential benefits of a fuzzy logic forecast support are better decision making due to the model-based approach, knowledge management and knowledge accumulation. Another effort towards the development of fuzzy models for stock markets has been made by (Sheta, 2006) using Takagi-Sugeno (TS) fuzzy models. Sheta uses the model for two non-linear processes, one pertaining to NASA and the other to prediction of next week S&P 500 index levels. The two steps involved in the process are the determination of the membership functions in the rule antecedents using the model input data; and the estimation of the consequence parameters. Parameters are estimated using least square estimation. The application of evolutionary optimization techniques such as genetic algorithm (GA) has given an entirely new dimension to the field of stock market prediction.(Badawy et al., 2005) conducted simulations using GA to find the optimal combination of technical parameters to predict Egyptian stocks accurately. (Tan et al., 2005) introduce a novel technique known as genetic complementary learning (GCL) to stock market prediction and give comparisons to demonstrate the superior performance of the method. Another paper introducing GA approach to instance selection (GAIS) (Kyoung-jae, 2006) for ANN in financial data mining has been reported. Kim introduces this technique to select effective training instances out of a large training data set to ensure efficient and fast training for stock market prediction networks. The GA also evolves the weights that mitigate the well known limitations of the gradient descent algorithm. The study demonstrates enhanced prediction performance at reduced training time. A hybrid model proposed by (Kuo, Chen, & Hwang, 2001) integrates GA based fuzzy logic and ANN. The model involves both quantitative factors (technical parameters) and qualitative factors such as political and psychological factors. Evaluation results indicate that the neural network considering both the quantitative and qualitative factors excels the neural network considering only the quantitative factors both in the clarity of buying-selling points and buying-selling performance. Another hybrid model involving GA proposed by (Rafiul, Baikunth, & Michael, 2007) utilizes the strengths of hidden Markov models (HMM), ANN and GA to forecast financial market behavior. Using ANN, the daily stock prices are transformed to independent sets of values that become input to HMM. The job of the GA is to optimize the initial parameters of HMM. The trained HMM is then used to identify and locate similar patterns in the historical data. A similar study investigates the effectiveness of a hybrid approach based on time delay neural networks (TDNN) and GA (Hyun-jung & Kyung-shik, 2007). The GA is used to optimize the number of time delays in the neural network to obtain the optimum prediction performance. Other studies and research in the field of stock market prediction using soft computing techniques include comparative investigation of both the ANN and the statistical ARIMA model (Schumann & Lohrbach, 1993) for the German stock index (DAX).The ANN method uses the four layer counter propagation network. The paper compares the results provided by both the methods and concludes that the efficient market hypothesis does not hold good. A data compression techniques for stock prediction (Azhar, Badros, Glodjo, Kao, & Reif, 1994) has been reported that uses the vector quantization method as an example of lossy data compression and Lempel-Ziv method as an example of lossless data compression technique to predict most of the well known indices across the globe.

Thus the review of the existing literature reveals that varieties of ANN structures such as MLP, PNN, VNN, MBNN, LLWNN, HRBFN, RN and TDNN have been employed to develop forecasting models to predict different stock indices. In most cases it is noticed that the development of these models and testing involve large computational complexity as well as more prediction and testing time. Therefore in the present paper our interest is to develop a low complexity and accurate prediction model which is better suited for online purpose. The present paper is proposed with two principal objectives. The first and most important objective is to introduce a functional link single layer artificial neural network(FLANN) for developing efficient stock market prediction model. The second objective is to show that the proposed model is computationally efficient, involves less complexity and provides similar or better prediction performance compared to other standard models.

In this study we propose a functional link ANN (FLANN) architecture based model to predict the movements of prices in the DJIA and S&P500 stock indices. The functional link ANN is a novel single neuron based architecture first proposed by Pao (Pao, 1989). It has been shown that this network can be conveniently used for functional approximation and pattern classification with faster convergence rate and lesser computational load than a multi-layer perceptron (MLP) structure. The structure of the FLANN is fairly simple. It is a flat net without any need for a hidden layer. Therefore, the computations as well as learning algorithm used in this network are simple. The functional expansion of the input to the network effectively increases the dimensionality of the input vector and hence the hyper-planes generated by the FLANN provide greater discrimination capability in the input pattern space (Pao, Phillips, & Sobajic, 1992). A number of research papers on system identification and control of nonlinear systems, noise cancellation and channel equalization has been reported in recent times (Patra, Pal, Chatterji, & Panda, 1999). These experiments have proven the ability of FLANN to give out satisfactory results to problems with highly non-linear and dynamic data. In this paper, the ability of the FLANN architecture based model to predict stock index movements, both for short term (next day) and medium term (one month to two months) prediction using statistical parameters such as technical indicators based on historical index data and fundamental economic factors is shown and analyzed. Apart from the basic architecture, two different adaptive algorithms are used to develop the FLANN model for the specified time-series prediction. One of the major criticisms against using neural networks for financial forecasting has been the black box nature of its operation. The network can give a good prediction but it is unable to explain how it reached that decision. Sometimes only the predicted value is not important, but more importantly, how the network reached its decision and which inputs play an important role in deciding the output is more important. Some efforts has been made in the paper to identify such inputs.

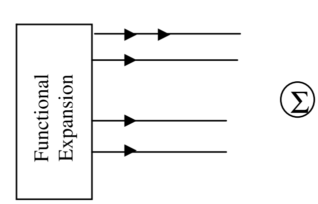
The rest of the paper is organized as follows: Section 2 deals with model development using FLANN structure using the LMS and RLS algorithms. The details of experimental setup including data collection, experimental conditions and testing and training process are discussed in Section 3. The simulation study and the results obtained from various experiments are provided in Section 4. In the final section the conclusion has been made.

2. Model development

2.1. The FLANN Model

The architecture of the FLANN is different from the linear weighting of the input pattern produced by the linear links of the multilayer perceptron (MLP). In a FLANN, each input to the network undergoes functional expansion through a set of basis functions. The functional link acts on an element or the entire pattern itself by generating a set of linearly independent functions. The input, expanded by a set of linearly independent functions in the functional expansion block causes an increase in the input vector dimensionality. This enables the FLANN to solve complex classification problems by generating non-linear decision boundaries.

A simplified FLANN Model is shown in Fig. 1. Let the kth input pattern vector to the model is given by $\underline{X}_k = [x_1(k), x_2(k) \ldots x_j(k) \ldots x_J(k)]^T$. When it is applied to the functional expansion (FE) block each element $x_j(k)$ of the vector is functionally expanded to generate the basis functions $\{\varphi_1(k)\varphi_2(k)\ldots\ldots\ldots\ldots \varphi_{(2N+1)}(k)\}^T$. The conventional nonlinear functional expansions which can be employed are trigonometric, power series or Chebyshev type. Based on our experience it is observed that the



use of trigonometric expansion provides better prediction capability of the model and hence in the present case trigonometric expansion is employed. If trigonometric polynomial basis function is used then

$$\underline{\phi}\{x_j(k)\} = \{x_j(k), \cos \pi x_j(k), \sin \pi x_j(k) \ldots \cos N\pi x_j(k), \sin N\pi x_j(k)\} \tag{1}$$

Each jth input sample of the kth pattern is expanded to N sine terms, N cosine terms plus the sample itself. Hence for an J element input pattern the total number of elements formed is given by $M = J(2N + 1)$. Let these expanded values for the entire J elements is denoted as

$$\underline{\phi}(k) = [\phi_1(k)\ \phi_2(k) \ldots \phi_{2N+1}(k) \ldots \phi_M]^T \tag{2}$$

Then the estimated output of the model during the application of kth pattern of the lth experiment is given by

$$\hat{y}(k,l) = f(s(k,l)) \tag{3}$$

where

$$s(k,l) = \underline{\phi}(k)^{\mathrm{T}}\underline{W} \tag{4}$$

$$\underline{W} = [w_1(l), w_2(l), \ldots, w_M(L)]^{\mathrm{T}} \tag{5}$$

The term $\underline{W}$ represents the weight vector of the model at lth experiment and $f(\bullet)$ denotes the nonlinear function which is taken to be tanh function. The FLANN model with a set of basis functions attempts to approximate the nonlinear time series using a set of K training patterns. When all the K patterns are applied the input-output relationship is expressed as

$$\underline{S} = \underline{\phi}\underline{W} \tag{6}$$

where $\underline{\phi}$ is $(K \times M)$ dimensional matrix given by

$$\underline{\phi} = [\underline{\phi}^T(1)\ \underline{\phi}^T(2) \ldots \underline{\phi}^T(k) \ldots \underline{\phi}^T(K)]^T \tag{7}$$

and $\underline{S}$ is a K- dimensional vector represented as

$$\underline{S} = [s(1,l)\ s(2,l) \ldots s(K,l)]^T \tag{8}$$

From (6) it is evident that K simultaneous linear equations are to be solved to obtain the weight of the FLANN model. The number of basis functions, M is chosen to be $K \leqslant M$. Depending on the condition three distinct cases arises (Pao, 1989). In the first case when $K = M$ and the determinant of $\underline{\phi}$ is not zero, that is $Det\ \underline{S} \neq 0$, then

$$\underline{W} = \phi^{-1}\underline{S} \tag{9}$$

where the dimensions of $\underline{\phi}$ are $K \times M$. In the second case when $K < M$, $\underline{\phi}$ is partitioned to obtain $\underline{\phi}_F$ of dimension $K \times K$. Under such situation $\underline{W}$ is modified to $\underline{W}_F$ by setting $w_{k+1} = w_{k+2} = w_M = 0$.

$$\text{If } Det\ \underline{\phi}_F \neq 0, \text{then } \underline{W} = \underline{\phi}_F^{-1} * \underline{S} \tag{10}$$

If the portioning is not done then a large of number of solutions can be obtained. The more interesting case is when $K > M$. In this case the functional link can generate an infinitely large number of orthonormal functions. If the columns of $\underline{\phi}$ are enhanced so that N is increased to M to match P that is $M = P$, the solution of the weights is given by

$$\underline{W}_M = \underline{\phi}_M^{-1}\underline{S} \tag{11}$$

where $\underline{\phi}_M$ are of dimensions $K \times M$. Expression (11) is an exact flat net solution. If $M > P$, and rank of $\underline{\phi}_M = K$, then one can proceed as in (10). This analysis indicates that the functional expansion model always yields a flat net solution if sufficient number of additional orthonormal functions are used in the enhancement.

In the proposed FLANN model the gradient descent and the recursive least square algorithms are used to update its associated weights.

2.2. Training of the model by gradient descent method

Let K training patterns are applied to the model in a sequential manner and this process is repeated for L experiments. Application of K patterns of input to the model and update of all weights once after application of all patterns constitutes one experiment. The weights of the model are updated at the end of each experiment (l) by computing the average change mth weight as

$$w_m(l+1) = w_m(l) + \Delta w_m(l) \tag{12}$$

The change of mth weight at lth experiment is given by

$$\Delta w_m(l) = \frac{\sum_{k=1}^{K} 2\mu\phi_m(k)\delta(k,l)}{K} \tag{13}$$

The symbol $\varphi_m(k)$ represents the mth expanded value at the application of kth pattern and the error term is computed as

$$\delta(k,l) = \left[\frac{1-\hat{y}^2(k,l)}{2}\right]e(k,l) \tag{14}$$

where

$$e(k,l) = y(k) - \hat{y}(k,l) \tag{15}$$

and $\hat{y}(k,l)$ stands for the estimated output of the model when the kth pattern of lth experiment is applied.

2.3. Training of the model by recursive least square (RLS) method

In this case the weights of the model is trained by minimizing the cost function iteratively. The cost function $E(l)$ is defined as

$$E(l) = \sum_{k=1}^{K-k} \lambda^{K-k}|e(k,l)|^2 \tag{16}$$

where λ is the forgetting factor which is nearly equal to unity and $e(k,l)$ is defined in (15). The average change in weight in each path of the model which minimizes (16) is computed as

$$\Delta w_m(l) = \frac{\sum_{k=1}^{K} G_m(k)\delta(k,l)}{K} \tag{17}$$

where

$$G_m(K) = \frac{\lambda^{-1}\underline{P}(k-1)\underline{\phi}(k)}{1+\lambda^{-1}\underline{\phi}(k)^{\mathrm{T}}\underline{P}(k-1)\underline{\phi}(k)} \tag{18}$$

$$\text{and} \quad \underline{P}(k) = \lambda\underline{P}(k-1) - \lambda^{-1}G_m(k)\underline{\phi}^{\mathrm{T}}(k)\underline{P}(k-1) \tag{19}$$

The matrix $\underline{P}(k)$ is initialized by setting $\underline{P}(k) = \frac{1}{\varepsilon}\underline{I}_M$, where ε is a small positive number and $\underline{I}_M$ is an $(M \times M)$ identity matrix. Using (17)–(19) the weights of the RLS based FLANN model are updated during training.

3. Experimental setup

3.1. Data collection and feature extraction

The data for the experiment on stock market prediction has been collected for two stock indices namely DJIA and S&P 500. The data are collected from January 1994 to October 2006, totaling 3200 data patterns for both DJIA and S&P 500 indices. The data obtained for the stock indices consisted of the closing price, opening price, and lowest value in the day, highest value in the day and the total volume of stocks traded in each day. The forecasting model of Fig. 1 is simulated to predict the closing price of the index on each

Table 1
Selected technical indicators and their formulae

Technical indicators	Formula
Simple moving average (SMA)	$\frac{1}{N}\sum_{i=1}^{N} x_i$, N = No. of days, x_i = today's price
Exponential moving average (EMA)	$(P \times A) + (\text{Previous EMA} \times (1 - A))$; A = 2/($N$ + 1), P – current price, A – smoothing factor, N – time period
Accumulation/distribution oscillator (ADO)	$\frac{(C.P-L.P)-(H.P-C.P)}{(H.P-L.P)\times(\text{Period's Volume})}$ C.P – closing price, H.P – highest price, L.P – lowest price
Stochastic oscillator (STO)	$\%K = \frac{(\text{Today's Close-Lowest Low in } K \text{ period})}{(\text{Highest High in } K \text{ period-Lowest Low in } K \text{ period})} \times 100$
	%D = SMA of % K for the period
On balance volume (OBV)	If Today's Close > Yesterday's Close OBV = Yesterday's OBV + Today's Volume If Today's Close < Yesterday's Close OBV = Yesterday's OBV − Today's volume
WILLIAM's %R	$\%R = \frac{(\text{Highest High in } n \text{ period}-\text{Today's Close})}{(\text{Highest High in } n \text{ period}-\text{Lowest Low in } n \text{ period})} \times 100$
Relative strength index (RSI)	$\text{RSI} = 100 - \frac{100}{1+(U/D)}$ U = total gain/n, D = total loss/n, n = number of RSI period
Price rate of change (PROC)	$\frac{(\text{Today's Close}-\text{Close } X-\text{period ago})}{(\text{Close } X-\text{period ago})} \times 100$
Closing price acceleration (CPACC)	$\frac{(\text{Close Price}-\text{Close Price } N-\text{period ago})}{(\text{Close Price } N-\text{period ago})} \times 100$
High price acceleration (HPACC)	$\frac{(\text{High Price}-\text{High Price } N-\text{period ago})}{(\text{High Price } N-\text{period ago})} \times 100$

day of the forecasting period. Ten technical and fundamental indicators defined in Table 1 are used as inputs to the network. Technical indicators are any class of metrics whose value is derived from generic price activity in a stock or asset. Technical indicators look to predict the future price levels, or simply the general price direction, of a security by looking at past patterns. A brief explanation of each indicator is provided here.

(i) Simple moving average (SMA)
It is the simple average of the values by taking a window of the specified period.

(ii) Exponential moving average (EMA)
It is also an average of the values in the specified period but it gives more weightage to recent values and thus it is more close to the actual values. EMAs of 10, 20 and 30 days window have been considered in the experiments.

(iii) Accumulation/distribution oscillator (ADO)
It measures money flow in the security. It attempts to measure the ratio of buying to selling by comparing price movements of a period to the volume of that period. The ADO also has been calculated for each day/pattern in the experiment.

(iv) Stochastic oscillator (STO)
Stochastic oscillator is a momentum indicator that shows the location of the current close relative to the high/low range over a set of number of periods. Closing levels which are consistently near the top of the range indicates accumulation (buying pressure) and those near the bottom of the range indicate distribution (selling pressure). Two oscillator indices %K and %D as defined in Table 1 are used.

(v) On balance volume (OBV)
It is a momentum indicator that relates volume to price change.

(vi) William's %R (WILLIAMS)
It is a momentum indicator that measures overbought/oversold levels.

(vii) Relative strength index (RSI)
It calculates the internal strength of the security. In the present case the periods have been taken as 9 days (RSI9) and 14 days (RSI14).

(viii) Price rate of change (PROC)
The PROC indicator displays the difference between the current price and closing price x-time periods ago. Through experimental results it has been found that value of x taken as 12 and 27 are considered best for technical analysis.

(ix) Closing price acceleration (CPACC)
It is the acceleration of the closing prices in the given period.

(x) High price acceleration (HPACC)
It is the acceleration of the high prices in the given period.Fundamental analysis is the study of economic, industry, and company conditions in an effort to determine the value of a company's stock. Apart from these technical parameters which depend on the past value of the data for forecasting there are fundamental analysis factors. These are generally macro economic parameters known to affect the stock market. Five fundamental factors used in the study are crude oil prices, United States GDP growth rate, corporate dividend rates, federal interest rates and commodity price index (CPI). The technical indicators used in this paper and their formulae are listed in Table 1.

3.2. Experimental conditions

Each input feature is expanded to five values out of which four are trigonometric expansions and the fifth one is the input itself. The trigonometric functions being used are cos πx, sin πx, cos $3\pi x$, sin $3\pi x$ where x is an input feature. In addition to the expanded inputs a bias is provided to neuron. The linearly weighted sum at the output of the neuron is then applied to sigmoid type (tanh) non-linear activation function for all the experiments. Two adaptive algorithms are used to update the weights during training: least mean square (LMS) algorithm and recursive least square (RLS) algorithm. The convergence coefficient for LMS, μ is set to a constant value of 0.1 for all simulations. Similarly the initialization constant, η for the RLS algorithm is taken as 1000. The inputs are normalized between + 1 and −1 for the proper functioning of the network. This may be achieved by a number of normalization techniques. One of the standard techniques is used here which expresses the actual value in terms of the maximum and minimum of the data set. All the values are normalized by using (20).

$$y = \frac{(2^{*}x - (Max. + Min.)}{(Max. + Min.)} \tag{20}$$

where, y and x represent the normalized and the actual value respectively. The experiments are carried out to test the performance of the network for predicting the close price of the two different indices one day, one month and two months in advance.

3.3. Training and testing process

During the training process the weights are updated once after application of all input patterns. The initial weights of the network are initialized to some random values between −1 to + 1. Each input feature is also normalized prior to the network training. The weights remain unchanged till all of the training data set are applied sequentially into the network, compared with the desired output, their respective error stored and the change of weight in each path is computed. The cost function for the training process considered in the paper is the mean square error (MSE). The training of the network is terminated when the minimum level of the cost function is obtained. At the end of the training process of the network, the weights are frozen for testing the network on inputs that are set apart from the training set. The test set patterns are the input to the network and the output, the predicted index close price is compared with desired output or actual close price. The percentage of errors is recorded for each data set. The criteria for judging the quality of prediction shown by the model is the mean of all the percentage error of the test data set. The mean absolute percentage error (MAPE) is used to gauge the performance of the trained prediction model for the test data. The effort is to minimize the MAPE for testing patterns in the quest for finding a better model for forecasting stock index price movements The MAPE is computed as

$$MAPE = \frac{1}{N}\sum_{j=1}^{N}\left|\frac{y_j - \hat{y}_j}{y_j}\right| \times 100 \tag{21}$$

where y_j is the actual stock market index at jth test pattern, $\hat{y}_j$ is the estimated stock market index at jth test pattern, N is the number of test patterns available for validation.

4. Simulation studies

In the simulation study five different experiments are carried to assess the validity of model.

Experiments and results

Experiment 1: One day in advance prediction using FLANN–LMS model using technical indicators as input. The results are shown in Table 2 and Figs. 2 and 3.

Experiment 2: One month in advance prediction using FLANN–LMS model using technical indicators. The results are shown in Table 3 and Figs. 4 and 5.

Table 2
Results for one day advance prediction of FLANN-LMS model and using different combinations of technical indicators as input

Stock index	Input variables to FLANN–LMS model	Testing period (days)	MAPE (%)
DJIA	EMA10, EMA30, ADO, CPACC, HPACC, STO, RSI9, PROC 12, PROC 27	390	0.64
DJIA	EMA10, EMA20, EMA30 ADO, CPACC, HPACC, RSI9, RSI14, PROC12, PROC27, WILLIAMS	658	0.74
S&P 500	EMA10, EMA30, ADO, CPACC, HPACC, STO, RSI9, PROC 12, PROC 27	390	0.61
S&P 500	EMA10, EMA30 ADO, CPACC, HPACC, STO, RSI9, PROC12, PROC27	658	0.65

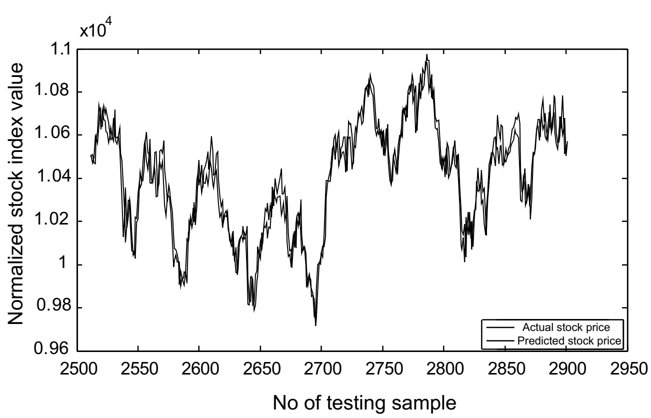


Fig. 2. Comparison of test results with actual stock prices of DJIA (one day in advance) using FLANN-LMS model.

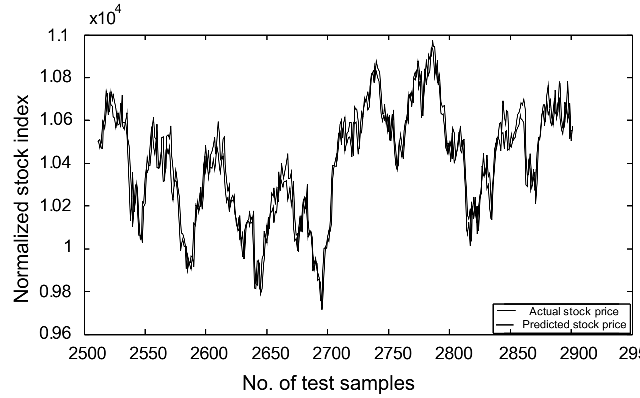


Fig. 3. Comparison of test results with actual stock prices of S&P500 (one day in advance) using FLANN-LMS model.

Table 3
Results for one month advance prediction using different combination of technical indicators

Stock index	Input variables to FLANN–LMS model	Testing period (days)	MAPE (%)
DJIA	ADO, CPACC, HPACC, RSI9, RSI14, PROC12, PROC27, WILLIAMS, OBV,STO	650	16.6
DJIA	EMA10, EMA30 ADO, CPACC, HPACC, RSI9, RSI14, PROC12, PROC27, OBV, STO	650	6.3
DJIA	EMA10, EMA30 ADO, CPACC, HPACC, RSI9, PROC12, PROC27,OBV, STO, WILLIAMS	650	5.9
DJIA	EMA10, EMA30 ADO, CPACC, HPACC, RSI9, PROC12, PROC27,OBV, STO	650	3.61
DJIA	EMA10, EMA20, EMA30, PROC12, PROC27, RSI9, RSI14, STO	650	3.03
DJIA	EMA10, EMA20, EMA30, ADO, RSI9	650	2.92
DJIA	EMA10, EMA20, EMA30, ADO, CPACC, HPACC, RSI9, RSI14	650	2.91
DJIA	EMA10, EMA20, EMA30	650	2.88
DJIA	EMA10, EMA20, EMA30 ADO, CPACC, HPACC, RSI9, RSI14, PROC12, PROC27, WILLIAMS, OBV, STO	650	2.75
DJIA	EMA 10, EMA 20, EMA 30, ADO, CPACC, HPACC, RSI 9, WILLIAMS	60	1.39
S&P 500	EMA10, EMA20, EMA30 ADO, CPACC, HPACC, RSI9, RSI14, PROC12, PROC27, WILLIAMS, STO	658	2.95
S&P 500	EMA10, EMA20, EMA30 ADO, CPACC, HPACC, RSI9, RSI14, PROC27, WILLIAMS	658	2.66
S&P 500	EMA10, EMA20, EMA30 ADO, CPACC, HPACC, RSI9, RSI14, PROC27, WILLIAMS	60	2.22
S&P 500	EMA10, EMA20, EMA30 ADO, CPACC, HPACC, PROC12, PROC27, RSI9, RSI14	60	2.09

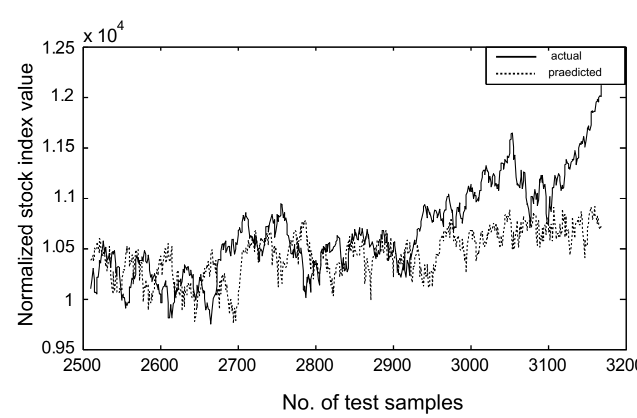


Fig. 4. Comparison of test results with actual stock prices of DJIA (30 days in advance) using FLANN-LMS model.

Experiment 3: Two months advance prediction using FLANN–LMS model using technical indicators only. The results are shown in Table 4 and Fig. 6.

Experiment 4: One month advance prediction with FLANN–RLS model using (a) fixed set of technical indicators and different combinations of fundamental factors and (b) fixed fundamental factors and different combinations of technical indicators.

The results of experiment 4(a) and 4(b) are listed in Table 5 and Table 6, respectively.

Experiment 5: Comparison of performance between FLANN–LMS and FLANN–RLS models. The results of this experiment is shown in Table 7.

Table 8 lists the comparison of number of experiments that are required to train the RLS and LMS based FLANN models. It is observed that the RLS based model require substantially less experiments to train the model and thus can be used for online purpose.

Results and discussions

The purpose of the of Experiment 1 is to obtain the performance of the FLANN-LMS model for one day in advance prediction by considering only technical indicators as inputs. The model provides good prediction performance with a best MAPE of 0.61% for DJIA and 0.65% for S&P 500 index. The results listed in Table 2 shows that the model performs consistently for both small as well as large testing set. Figs. 2 and 3 demonstrate good match between the actual and predicted stock prices of (DJIA and S&P500) respectively when the test data is used as input to the model.

Experiment 2 investigates on the performance for one month in advance prediction using different combination of technical indica-

Table 6
Results of prediction performance of FLANN–RLS model using fixed fundamental factors and different combinations of technical indicators

Stock index	Input variables to FLANN model (technical indicators)	Input variables to FLANN model (fundamental factors)	Testing period (days)	MAPE using RLS (%)	RLS initialization constant
DJIA	EMA10	Dividend, interest rate, oil price, GDP rate	60	2.36	1000
DJIA	EMA10, EMA20, EMA30	Dividend, interest rate, oil price, GDP rate	60	2.23	1000
DJIA	EMA10, EMA20, EMA30, ADO	Dividend, interest rate, oil price, GDP rate	60	2.42	1000
DJIA	EMA10, EMA20, EMA30, HPACC	Dividend, interest rate, oil price, GDP rate	60	2.21	1000
DJIA	EMA10, EMA20, EMA30, CPACC	Dividend, interest rate, oil price, GDP rate	60	2.24	1000
DJIA	EMA10, EMA20, EMA30, RSI9	Dividend, interest rate, oil price, GDP rate	60	2.33	1000
DJIA	EMA10, EMA20, EMA30, PROC12	Dividend, interest rate, oil price, GDP rate	60	2.38	1000
DJIA	EMA10, EMA20, EMA30, STO	Dividend, interest rate, oil price, GDP rate	60	2.52	1000
DJIA	EMA10, EMA20, EMA30, WILLIAMS	Dividend, interest rate, oil price, GDP rate	60	2.51	1000
DJIA	EMA10, EMA20, EMA30, OBV	Dividend, interest rate, oil price, GDP rate	60	2.07	1000

Table 7
Comparison of prediction performance between FLANN–LMS and FLANN–RLS models

Stock index	Input variables to FLANN	Days in advance prediction	Testing period (days)	MAPE (LMS) (%)	MAPE (RLS) (%)
DJIA	EMA20, EMA30, ADO, CPACC, RSI9, RSI14, OBV, PROC 27, WILLIAMS	60 days	60	2.25	2.45
DJIA	EMA10, EMA20, EMA30, ADO, CPACC, HPACC, RSI9, WILLIAMS	30 days	60	2.33	2.54
DJIA	EMA10, EMA30, ADO, CPACC, HPACC, STO, RSI9, PROC 12, PROC 27	1 day	390	0.64	0.58
DJIA	EMA10, EMA20, EMA30 ADO, CPACC, HPACC, RSI9, RSI14, PROC12, PROC27, WILLIAMS	1 day	658	0.74	0.61

Table 8
Comparison of number of experiments required to train the FLANN model

Experiment Details	No. of experiments in FLANN-LMS model	No. of experiments in FLANN-RLS model
One day ahead with technical parameters	$2510^*4000 = 10040000$	$2510^*5 = 12550$
One month ahead with technical parameters	$2510^*3500 = 8785000$	$2510^*15 = 37650$
Two months ahead with technical parameters	$2510^*2000 = 5020000$	$2510^*80 = 200800$

considerable deterioration in the prediction performance. In an effort to reduce the number of input to the network and at the same time enhance the performance of the prediction model, the input parameters are varied on a trial-error basis to gain insight into the extent of usefulness of the technical indicators to the prediction model. The stochastic oscillators, SMA, RSI 14 and Williams technical indicators are not found to affect the prediction performance significantly. On the other hand, technical indicators such as exponential moving averages (EMA), on balance volume (OBV), accumulation distribution oscillator (ADO) and price rate of change (PROC) are found to be essential to the model. The best performance obtained for this experiment is 1.39% MAPE for DJIA and 2.09% for S&P 500 index both using a short testing set of 60 days. It is also observed that the model performs much better on shorter testing sets than on larger testing sets for such an experiment.

Experiment 3 is carried out to check the performance of FLANN-LMS model for even longer term prediction i.e. for two months. The best MAPE obtained in this case for DJIA is 2.25% as displayed in Table 4 and the corresponding comparison with test data is shown in Fig. 6.

Experiment 4 is carried out to investigate the effect of including key fundamental factors in improving the prediction performance of the FLANN-RLS model. Five fundamental factors namely – crude oil price, federal interest rates, GDP growth rate (US), Commodity price index and corporate dividend rates are taken as input. In this case the experiments are carried out by taking different combinations fundamental factors, while keeping the technical indicators constant as shown in Table 5. The results show that the inclusion of fundamental factors to the FLANN model has little effect on the prediction performance.

Simulations are also carried out by taking different combinations of technical indicators keeping the fundamental factors constant as shown in Table 6. The exponential moving average (EMA 10, 20 and 30) are mainly considered as technical parameters in the simulations. The best prediction with a MAPE of 2.07% is found with 8 variables including EMAs and on balance volume (OBV). It can be inferred from Experiment 4 that the prediction performance depends largely on the choice of technical indicators.

Finally Experiment 5 compares the prediction performance of LMS and RLS based FLANN models. The results in Table 7 show that the FLANN-LMS model gives comparable results to those obtained from FLANN-RLS model. However, RLS has the distinct advantage of having faster convergence and hence lesser computational load for training of the network. Table 7 reaffirms this advantage of the RLS update algorithm over LMS in terms of computational efficiency which is valid for all the three cases: one day, one month and two months in advance. It is observed from Table 8 that the LMS based model takes roughly 800 times more experiments in case of one day in advance prediction and 25 times more experiments for two months in advance prediction. Comparison of the results obtained by various ANN methods and presented in Table 1 of [1] reveals that the proposed FLANN based method provide better prediction results.

5. Conclusion

In this paper, the FLANN based stock market prediction model employing the LMS and the RLS based weight update mechanism is introduced. The FLANN, model is structurally simple and involves lesser computations compared to models. Experiments also show that the FLANN based model with both LMS as well as RLS update algorithm gives enhanced prediction performance for one day, one month and two month ahead stock market prediction. The use of all the technical indicators as inputs to the model unnecessarily loads the network and does not improves the prediction performance. Certain technical indicators have greater effect on the prediction performance than others. Inclusion of certain fundamental/economic factors to the inputs as indicated in the discussion section does not necessarily improve performance. However, suitable combinations of technical and fundamental parameters give better results. The FLANN-RLS model is computationally much more efficient than its LMS counterpart and also take less training samples to train and hence may be used for online prediction. In all, the FLANN based stock market prediction model is an effective approach both computationally as well as performance wise to foresee the market levels both in short and medium terms future.

References

Azhar, S., Badros, G. J., Glodjo, A., Kao, M.-Y., & Reif, J. H. (1994). Data compression techniques for stock market prediction. In *Proceedings of data compression conference, DCC'94* (pp. 72–82).

Badawy, F. A., Abdelazim, H. Y., & Darwish, M. G. (2005). Genetic algorithms for predicting the Egyptian stock market. In *Proceedings of third international conference on information and communications technology* (pp. 109–122).

Chen Y., Dong X., & Zhao Y. (2005). Stock index modeling using EDA based local linear wavelet neural network. In *Proceedings of international conference on neural networks and brain* (Vol. 3, pp. 1646–1650).

Hassan, Md. R., Nath, B., & Kirley, M. (2007). A fusion model of HMM, ANN and GA for stock market forecasting. *Expert Systems with Applications, 33*(1), 171–180.

Hiemstra Y. (1994). A stock market forecasting support system based on fuzzy logic. In *Proceedings of twenty-seventh Hawaii international conference on system sciences, vol. III: Information systems: Decision support and knowledge-based systems* (Vol. 3, pp. 281–287).

Kim, H.-J., & Shin, K.-S. (2007). A hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns in stock markets. *Applied Soft Computing, 7*(2), 569–576.

Kim, K.-J. (2006). Artificial neural networks with evolutionary instance selection for financial forecasting. *Expert Systems with Applications, 30*, 519–526.

Kimoto T., Asakawa K., Yoda M. & Takeoka M. (1990). Stock market prediction system with modular neural networks. In *Proceedings of international joint conference on neural network (IJCNN)* (Vol. 1, pp. 1–6).

Kuo, R. J., Chen, C. H., & Hwang, Y. C. (2001). An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network. *Fuzzy Sets and Systems, 118*, 21–45.

Lee, R. S. T. (2004). IJADE stock advisor: An intelligent agent based stock prediction system using hybrid RBF recurrent network. *IEEE Transactions on Systems, Man and Cybernetics, Part A, 34*(3), 421–428.

Masters, T. (1993). *Practical neural network recipes in C++*. New York: Academic Press.

Oh, K. J., & Kim, K-J (2002). Analyzing stock market tick data using piecewise non linear model. *Expert System with Applications, 22*, 249–255.

Ornes C. & Sklansky J. (1997). A neural network that explains as well as predicts financial market behavior. In *Proceedings of computational intelligence for financial engineering, the IEEE/IAFE 1997* (pp. 43–49).

Pao, Y. H. (1989). *Adaptive pattern recognition & neural networks*. Reading, MA: Addison-Wesley.

Pao, Y. H., Phillips, S. M., & Sobajic, D. J. (1992). Neural Net Computing and intelligent control systems. *International Journal of Control, 56*(2), 263–289.

Patra, J. C., Pal, R. N., Chatterji, B. N., & Panda, G. (1999). Identification of non-linear & dynamic system using functional link artificial neural network. *IEEE Transactions on System, Man & Cybernetics – Part B; Cybernetics, 29*(2), 254–262.

Saad, E. W., Prokhorov, D. V., & Wunsch, D. C. (1998). Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks. *IEEE Transactions of Neural Network, 9*(6), 1456–1470.

Schumann M. & Lohrbach T. (1993). Comparing artificial neural networks with statistical methods within the field of stock market prediction. In *Proceedings of twenty-sixth Hawaii international conference on system sciences* (Vol. 4, pp. 597–606).

Sheta, A. (2006). Software effort estimation and stock market prediction using Takagi-Sugeno fuzzy models. In *Proceedings of IEEE international conference on fuzzy systems* (pp. 171–178).

Tan Clarence, N. W., & Wittig Gerhard, E. (1993). A Study of the parameters of a back propagation stock price prediction model. In *Proceedings of first New Zealand international two-stream conference on artificial neural networks and expert systems* (pp. 288–291).

Tan H., Prokhorov D. V. & Wunsch D. C., II (1995). Conservative thirty calendar day stock prediction using a probabilistic neural network. In *Proceedings of computational intelligence for financial engineering, the IEEE/IAFE 1995* (pp. 113–117).

Tan T. Z., Quek C. & Ng G. S. (2005). Brain inspired genetic complimentary learning for stock market prediction. In *Proceedings of IEEE congress on evolutionary computation* (Vol. 3, pp. 2653-2660).

Taylor, S. (1986). *Modeling financial time series*. John Wiley & Sons.

Tsaih, R., Yenshan, H., & Lai Charles, C. (1998). Forecasting S&P 500 stock index futures with a hybrid AI system. *Decision Support Systems, 23*, 161–174.

Wang, Y. (2002). Predicting stock price using fuzzy grey prediction system. *Expert System with Applications, 22*, 33–39.

Wang, Y. (2003). Mining stock prices using fuzzy rough set system. *Expert System with Applications, 24*, 13–23.

Yamashita T., Hirasawa K. & Hu J. (2005). Application of multi-branch neural networks to stock market prediction. In *Proceedings of IEEE international joint conference on neural networks (IJCNN '05)* (Vol. 4, pp. 2544–2548).