Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange

Article *in* Expert Systems with Applications · May 2011

DOI: 10.1016/j.eswa.2010.10.027 · Source: DBLP

Yakup Kara
Selcuk University
22 PUBLICATIONS 488 CITATIONS
SEE PROFILE

Melek Acar Boyacioglu
Selcuk University
11 PUBLICATIONS 324 CITATIONS
SEE PROFILE

Omer Kaan Baykan
Selcuk University
16 PUBLICATIONS 233 CITATIONS
SEE PROFILE

Expert Systems with Applications 38 (2011) 5311–5319

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange

Yakup Kara[a], Melek Acar Boyacioglu[b,*], Ömer Kaan Baykan[c]

[a] *Selçuk University, Faculty of Engineering and Architecture, Department of Industrial Engineering, Campus 42031, Konya, Turkey*
[b] *Selçuk University, Faculty of Economics and Administrative Sciences, Department of Business Administration, Campus 42031, Konya, Turkey*
[c] *Selçuk University, Faculty of Engineering and Architecture, Department of Computer Engineering, Campus 42031, Konya, Turkey*

ARTICLE INFO

Keywords:
Artificial neural networks
Support vector machines
Prediction
Stock price index

ABSTRACT

Prediction of stock price index movement is regarded as a challenging task of financial time series prediction. An accurate prediction of stock price movement may yield profits for investors. Due to the complexity of stock market data, development of efficient models for predicting is very difficult. This study attempted to develop two efficient models and compared their performances in predicting the direction of movement in the daily Istanbul Stock Exchange (ISE) National 100 Index. The models are based on two classification techniques, artificial neural networks (ANN) and support vector machines (SVM). Ten technical indicators were selected as inputs of the proposed models. Two comprehensive parameter setting experiments for both models were performed to improve their prediction performances. Experimental results showed that average performance of ANN model (75.74%) was found significantly better than that of SVM model (71.52%).

1. Introduction

Predicting stock price index and its movement has been regarded as one of the most challenging applications of time series prediction. Even though there have been many empirical researches which deal with the issues of predicting stock price index, most empirical findings are associated with the developed financial markets. However, few researches exist in the literature to predict the direction of stock price index movement in emerging markets, especially in Turkish stock market. Accurate predictions of movement of stock price indexes are very important for developing effective market trading strategies (Leung, Daouk, & Chen, 2000). Thus, investors can hedge against potential market risks and speculators and arbitrageurs have opportunities to make profit by trading in stock index (Manish & Thenmozhi, 2005).

Stock market prediction is regarded as a challenging task of the financial time series prediction process since the stock market is essentially dynamic, nonlinear, complicated, nonparametric, and chaotic in nature (Abu-Mostafa & Atiya, 1996). In addition, stock market is 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, and psychology of investors etc. (Tan, Quek, & See, 2007).

ANN and SVM have been successfully used for modeling and predicting financial time series. Although ANN can be one of the very useful tools in time series prediction, several studies showed that ANN had some limitations in learning the patterns because stock market data has tremendous noise, non-stationary characteristics, and complex dimensionality. ANN often exhibit inconsistent and unpredictable performance on noisy data (Kim, 2003; Kim & Han, 2000; Manish & Thenmozhi, 2005). Therefore, predicting stock price movements is quite difficult.

It is of interest to study the extent of stock price index movement predictability using data from emerging markets such as that of Turkey. Since its establishment in December 1985, the Istanbul Stock Exchange (ISE) has presented an outstanding growth as an emerging market. The ISE is characterized with high volatility in the market returns. Such volatility attracts many local and foreign investors as it provides high return possibility (Armano, Marchesi, & Murru, 2005). The number of companies listed in the ISE increased to 343 in 2010 while it was 80 in 1986. Total trading volume reached to $ 316.326 billion and total market capitalization was $ 235.996 billion in 2009 (http://www.ise.org). The ISE National 100 Index, which is the main market indicator of the ISE, is a market capitalization-weighted index and represents at least 75% of the total market capitalization, traded value, number of shares traded and number of trades realized in the market (Bildik, 2001).

The core objective of this paper is to predict the direction of movement in the daily ISE National 100 Index using artificial

[*] Corresponding author. Tel.: +90 332 2234350; fax: +90 332 3522722.
E-mail address: melekacar@yahoo.com (M. Acar Boyacioglu).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.10.027

neural networks (ANN) and support vector machines (SVM). The major contributions of this study are to demonstrate and verify the predictability of stock price index direction using ANN and SVM and to compare the performance of these techniques.

The remainder of this paper is organized into six sections. Section 2 provides a brief overview of the theoretical literature. Section 3 describes the research data. Section 4 provides the prediction models used in this study. Section 5 reports the empirical results from the comparative analysis. Finally, Section 6 contains the concluding remarks.

2. Literature review

In recent years, there have been a growing number of studies looking at the direction of movements of various kinds of financial instruments. Both academic researchers and practitioners have made tremendous efforts to predict the future movements of stock market index or its return and devise financial trading strategies to translate the forecasts into profits (Chen, Leung, & Daouk, 2003). In the following section, we focus the review of previous studies on artificial neural networks (ANN) and support vector machines (SVM) applied to stock market prediction.

2.1. Artificial neural networks (ANN)

There exist vast literatures which concentrate on the predictability of the stock market. These studies used various types of ANN to predict accurately the stock price return and the direction of its movement. ANN has been demonstrated to provide promising results in predict the stock price return (Avci, 2007; Egeli, Ozturan, & Badur, 2003; Karaatli, Gungor, Demir, & Kalayci, 2005; Kimoto, Asakawa, Yoda, & Takeoka, 1990; Olson & Mossman, 2003; White, 1988; Yoon & Swales, 1991).. Leung et al. (2000) examined various prediction models based on multivariate classification techniques and compared them with a number of parametric and nonparametric models which forecast the direction of the index return. Empirical experimentation suggested that the classification models (discriminant analysis, logit, probit and probabilistic neural network) outperform the level estimation models (adaptive exponential smoothing, vector auto regression with Kalman filter updating, multivariate transfer function and multilayered feed forward neural network) in terms of predicting the direction of the stock market movement and maximizing returns from investment trading. Chen et al. (2003) attempted to predict the direction of return on the Taiwan Stock Exchange Index. The probabilistic neural network (PNN) is used to forecast the direction of index return. Statistical performance of the PNN forecasts is compared with that of the generalized methods of moments (GMM) with Kalman filter and random walk. Empirical results showed that PNN demonstrate a stronger predictive power than the GMM–Kalman filter and the random walk prediction models. Diler (2003) trained neural networks based on various technical indicators to estimate the direction of the ISE 100 Index. The technical indicators used are MA, momentum, RSI, stochastics K%, moving average convergence-divergence (MACD). The results of the study presented that the direction of the ISE 100 Index could be predicted at a rate of 60,81 %. Altay and Satman (2005) compared the forecast performances of neural network models with the Ordinary Least Square (OLS) regression model for ISE-30 and ISE-All Indexes. Although the prediction performance of neural network models for daily and monthly data failed to outperform the liner regression model, these models are able to predict the direction of the indexes more accurately. Cao, Leggio, and Schniederjans (2005) aimed to demonstrate the accuracy of ANN in predicting stock price movement for firms traded on the Shanghai Stock Exchange (SHSE). They compared the capital asset pricing model (CAPM) and Fama and French's 3-factor model to the predictive power of the univariate and multivariate neural network models. Their results showed that neural networks outperform the linear models compared.

Some researches tend to hybridize several artificial intelligence (AI) techniques to predict stock market returns (Baba & Kozaki, 1992; Chu, Chen, Cheng, & Huang, 2009; Hiemstra, 1995; Kim & Chun, 1998; Leigh, Purvis, & Ragusa, 2002; Oh & Kim, 2002; Pai & Lin, 2005; Saad, Prokhorov, & Wunsch, 1998; Takahashi, Tamada, & Nagasaka, 1998; Tan et al., 2007; Yudong & Lenan, 2009). Tsaih, Hsu, and Lai (1998) applied a hybrid AI approach to predict the direction of daily price changes in S&P 500 stock index futures. The hybrid AI approach integrated the rule-based systems and the neural networks technique. Empirical results demonstrated that reasoning neural networks (RN) outperform the other two ANN models (back propagation networks and perceptron). Empirical results also confirmed that the integrated futures trading system (IFTS) outperforms the passive buy-and-hold investment strategy.

2.2. Support vector machines (SVM)

Recently the support vector machines (SVM), has been also successfully applied to predict stock price index and its movements. Kim (2003) used SVM to predict the direction of daily stock price change in the Korea composite stock price index (KOSPI). This study selected 12 technical indicators to make up the initial attributes. The indicators are stochastic K%, stochastic D%, stochastic slow D%, momentum, ROC, Williams' %R, A/D oscillator, disparity5, disparity10, OSCP, CCI and RSI. In addition, this study examined the feasibility of applying SVM in financial prediction by comparing it with back-propagation neural network (BPN) and case-based reasoning (CBR). Experimental results proved that SVM outperform BPN and CBR and provide a promising alternative for stock market prediction. Manish and Thenmozhi (2005) used SVM and random forest to predict the daily movement of direction of S&P CNX NIFTY Market Index of the National Stock Exchange and compared the results with those of the traditional discriminant and logit models and ANN. In their study, they used the same technical indicators as input variables applied by Kim (2003). The experimental results showed that SVM outperform random forest, neural network and other traditional models. Huang, Nakamori, and Wang (2005), in their study, investigated the predictability of financial movement direction with SVM by predicting the weekly movement direction of NIKKEI 225 Index. To evaluate the prediction ability of SVM, they compared its performance with those of linear discriminant analysis, quadratic discriminant analysis and Elman backpropagation neural networks. The results of the experiment showed that SVM outperform the other classification methods. Manish and Thenmozhi (2006) investigated the usefulness of ARIMA, ANN, SVM, and random forest regression models in predicting and trading the S&P CNX NIFTY Index return. The performance of the three nonlinear models and the linear model are measured statistically and financially via a trading experiment. The empirical result suggested that the SVM model is able to outperform other models used in their study. Hsu, Hsieh, Chih, and Hsu (2009) developed two-stage architecture by integrating self-organizing map and support vector regression for stock price prediction. They examined seven major stock market indices. The results suggested that the two-stage architecture provides a promising alternative for stock price prediction. The review paper of Atsalakis and Valavanis (2009), which summarizes the related literature, can be useful for interested readers.

3. Research data

This section describes the research data and the selection of predictor attributes. The research data used in this study is the direction of daily closing price movement in the ISE National 100 Index. The entire data set covers the period from January 2, 1997 to December 31, 2007. The total number of cases is 2733 trading days. The number of cases with increasing direction is 1440 while the number of cases with decreasing direction is 1293. That is, 52.7% of the all cases have an increasing direction and 47.3% of the all cases have a decreasing direction. The historical data was obtained from the technical analysis module of Matriks gold 2.4.0., produced by Matriks Information Delivery Services Inc. The number of cases for each year is given in Table 1.

Some subsets were derived from the entire data set. The first subset was used to determine efficient parameter values for evaluated ANN and SVM models. This data set is called "parameter setting data set" and used in the *preliminary experiments*. The parameter setting data set is consisted of approximately 20% of the entire data set and is proportional to the number of increases and decreases for each year in the entire data set. For instance, the number of cases with increasing direction in the parameter setting data for 1997 is 30 and that of decreasing direction is 22. Using this sampling method, the parameter setting data set becomes more capable of representing the entire data set. This parameter setting data set was also divided into two equal-sized training (~10% of the entire) and holdout (~10% of the entire) sets. The training data was used to determine the specifications of the models and parameters while the holdout data was reserved for out-of-sample evaluation and comparison of performances among the two prediction models. The parameter setting data set yielded a total of 548 cases. The number of cases for each year in the parameter setting data set is given in Table 2.

Once the efficient parameter values are specified, prediction performances of ANN and SVM models can be compared to each other. This performance comparison was performed on the entire data set considering the parameter values specified using the parameter setting data set. That is, the prediction models must be re-trained using a new training data set which must be a new part of the entire data set and must be larger than the training subset of parameter setting data set. After re-training, out-of-sample evaluation of models must be carried out using a new holdout data set, which is the remaining part of entire data set. Therefore, the entire data set was re-divided into the training data set (~50% of entire) and the holdout data set (~50% of entire) for *comparison experiments*. This was also realized by considering the dispersion of increases and decreases in the entire data set. The number of cases in the resulting comparison data sets is given in Table 3.

Ten technical indicators for each case were used as input variables. Many fund managers and investors in the stock market generally accept and use certain criteria for technical indicators as the signal of future market trends (Kim, 2003). A variety of technical indicators are available. Some technical indicators are effective under trending markets and others perform better under no trending or cyclical markets (Tsaih et al., 1998). In the light of previous studies, it is hypothesized that various technical indicators may be used as input variables in the construction of prediction models to forecast the direction of movement of the stock price index (Chen et al., 2003). We selected ten technical indicators as feature subsets by the review of domain experts and prior researches (Armano et al., 2005; Diler, 2003; Huang & Tsai, 2009; Kim, 2003; Kim & Han, 2000; Manish & Thenmozhi, 2005; Yao, Chew, & Poh, 1999). Table 4 summarizes the selected technical indicators and their formulas.

Table 2
The number of cases in the parameter setting data set (~20% of entire).

Year	Training (~10% of entire)			Holdout (~10% of entire)		
	Increase	Decrease	Total	Increase	Decrease	Total
1997	15	11	26	15	11	26
1998	12	12	24	12	12	24
1999	13	10	23	13	10	23
2000	11	14	25	11	14	25
2001	12	13	25	12	13	25
2002	12	13	25	12	13	25
2003	13	11	24	13	11	24
2004	14	11	25	14	11	25
2005	15	11	26	15	11	26
2006	13	12	25	13	12	25
2007	13	13	26	13	13	26
Total	143	131	274	143	131	274

Table 3
The number of cases in the comparison data sets.

Year	Training (~50% of entire)			Holdout (~50% of entire)		
	Increase	Decrease	Total	Increase	Decrease	Total
1997	73	53	126	73	53	126
1998	62	62	124	62	62	124
1999	67	52	119	66	51	117
2000	56	68	124	55	68	123
2001	62	63	125	61	62	123
2002	62	65	127	61	64	125
2003	67	56	123	67	56	123
2004	71	54	125	71	53	124
2005	74	54	128	73	53	126
2006	66	60	126	65	59	124
2007	63	63	126	63	62	125
Total	723	650	1373	717	644	1360

Using the historical data, summary statistics for the selected indicators were calculated and given in Table 5.

The direction of daily change in the stock price index is categorized as "0" or "1". If the ISE National 100 Index at time t is higher than that at time $t-1$, direction t is "1". If the ISE National 100 Index at time t is lower than that at time $t-1$, direction t is "0". The original data were scaled into the range of $[-11]$. The goal of linear scaling is to independently normalize each feature component to the specified range. It ensures that the larger value input attributes do not overwhelm smaller value inputs, and helps to reduce prediction errors (Kim, 2003; Manish & Thenmozhi, 2005).

Table 1
The number of cases in the entire data set.

	Year											Total
	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	
Increase	146	124	133	111	123	123	134	142	147	131	126	1440
%	57.9	50	56.4	44.9	49.6	48.8	54.5	57	57.9	52.4	50.2	52.7
Decrease	106	124	103	136	125	129	112	107	107	119	125	1293
%	42.1	50	43.6	55.1	50.4	51.2	45.5	43	42.1	47.6	49.8	47.3
Total	252	248	236	247	248	252	246	249	254	250	251	2733

Table 4
Selected technical indicators and their formulas.

Name of indicators	Formulas
Simple 10-day moving average	$\frac{C_t+C_{t-1}+\cdots+C_{t-10}}{10}$
Weighted 10-day moving average	$\frac{((n)\times C_t+(n-1)\times C_{t-1}+\cdots+C_{10})}{(n+(n-1)+\cdots+1)}$
Momentum	$C_t - C_{t-n}$
Stochastic K%	$\frac{C_t-LL_{t-n}}{HH_{t-n}-LL_{t-n}} \times 100$
Stochastic D%	$\frac{\sum_{i=0}^{n-1} K_{t-i}\%}{n}$
RSI (Relative Strength Index)	$100 - \frac{100}{1+\left(\sum_{i=0}^{n-1} Up_{t-i}/n\right)/\left(\sum_{i=0}^{n-1} Dw_{t-i}/n\right)}$
MACD (moving average convergence divergence)	$MACD(n)_{t-1} + 2/n + 1 \times (DIFF_t - MACD(n)_{t-1})$
Larry William's R%	$\frac{H_n-C_t}{H_n-L_n} \times 100$
A/D (Accumulation/Distribution) Oscillator	$\frac{H_t-C_{t-1}}{H_t-L_t}$
CCI (Commodity Channel Index)	$\frac{M_t-SM_t}{0.015D_t}$

C_t is the closing price, L_t the low price, H_t the high price at time t, *DIFF*: $EMA(12)_t - EMA(26)_t$, *EMA* exponential moving average, $EMA(k)_t$: $EMA(k)_{t-1} + \alpha \times (C_t - EMA(k)_{t-1})$, α smoothing factor: $2/1 + k$, k is time period of k day exponential moving average, LL_t and HH_t mean lowest low and highest high in the last t days, respectively, $M_t : H_t + L_t + C_t/3;\ SM_t : \left(\sum_{i=1}^{n} M_{t-i+1}\right)/n,\ D_t : \left(\sum_{i=1}^{n} \mid M_{t-i+1} - SM_t \mid\right)/n, Up_t$ means the upward price change, Dw_t means the downward price change at time t.

Table 5
Summary statistics for the selected indicators.

Name of indicator	Max	Min	Mean	Standard deviation
Simple MA	57155.83	951.10	17867.86	14717.27
Weighted MA	57450.36	961.87	17897.53	14739.61
Momentum	159.30	61.51	101.94	9.65
Stochastic K%	99.34	6.86	56.81	24.73
Stochastic D%	97.30	11.81	56.82	19.34
RSI	96.25	14.40	54.49	13.09
MACD	2075.33	−2117.35	138.09	508.27
LW R%	.00	−100.00	− 41.72	30.25
A/D Oscillator	8E + 010	1E + 008	2E + 010	2E + 010
CCI	288.21	−323.22	12.93	86.99

4. Prediction models

4.1. Artificial neural networks

ANN has demonstrated their capability in financial modeling and prediction. In this paper, a three-layered feedforward ANN model was structured to predict stock price index movement. This ANN model consists of an input layer, a hidden layer and an output layer, each of which is connected to the other. At least one neuron should be employed in each layer of the ANN model. Inputs for the network were ten technical indicators which were represented by ten neurons in the input layer. Output of the network was two patterns (0 or 1) of stock price direction. The output layer of the network consisted of only one neuron that represents the direction of movement. The number of neurons in the hidden layer was determined empirically. The architecture of the three-layered feedforward ANN is illustrated in Fig. 1.

The neurons of a layer are linked to the neurons of the neighboring layers with connectivity coefficients (weights). Using a learning procedure, these weights were adjusted to classify the given input patterns correctly for a given set of input–output pairs. The initial values of these weights were randomly assigned. The back-propagation learning algorithm was used to train the three-layered feedforward ANN structure in this study (Rumelhart, Hinton, & Williams, 1986). The relative percentage of root mean square (RMS%) was used to evaluate the performance of the ANN model. The gradient-descent method was used as the weight update algorithm to minimize RMS%. A tangent sigmoid transfer function was selected on the hidden layer. On the other hand, a logistic sigmoid transfer function was used on the output layer. That is, the outputs of the model will vary between 0 and 1. If the output value is smaller than 0.5, then the corresponding case is classified as a decreasing direction; otherwise, it is classified as an increasing direction in movement.

The number of neurons (n) in the hidden layer, value of learning rate (lr), momentum constant (mc) and number of iterations (ep) are ANN model parameters that must be efficiently determined. Ten levels of n, nine levels of mc and ten levels of ep were tested in the parameter setting experiments. As suggested in the literature, a small value of lr was selected as 0.1. The ANN parameters and their levels are summarized in Table 6.

The parameter levels evaluated in parameter setting yield a total of 10 × 10 × 9 = 900 treatments for ANN. Each parameter combination was applied to the training and holdout data sets and prediction accuracy of the models were evaluated. A training performance and a holdout performance were calculated for each parameter combination. The parameter combination that resulted in the best average of training and holdout performances was selected as the best one for the corresponding model. All experiments were conducted using neural networks toolbox of MATLAB software.

4.2. Support vector machines

Support vector machines (SVM) is a family of algorithms that have been implemented in classification, recognition, regression and time series. SVM originated as an implementation of Vapnik's (1995) Structural Risk Minimization (SRM) principle to develop binary classifications. SVM emerged from research in statistical learning theory on how to regulate generalization, and find an optimal trade off between structural complexity and empirical risk. SVM classify points by assigning them to one of two disjoint half spaces, either in the pattern space or in a higher-dimensional feature space (Khemchandani & Jayadeva Chandra, 2009).

The main idea of support vector machine is to construct a hyperplane as the decision surface such that the margin of separation between positive and negative examples is maximized (Xu, Zhou, & Wang, 2009). For a training set of samples, with input vectors $x_i \in R^d$ and corresponding labels $y_i \in \{+1, -1\}$, SVM learns how to classify objects into two classes.

For a two-class classification problem, assume that we have a set of input vectors $x_i \in R^d (i = 1, 2, \ldots, N)$ with corresponding labels $y_i \in \{+1, -1\}$ $(i = 1,2,\ldots,N)$. Here, +1 and −1 indicate the two classes. The goal is to construct a binary classifier or derive a decision function from the available samples which has a small probability of misclassifying a future sample. SVM maps the input vectors$x_i \in R^d$ into a high dimensional feature space $\Phi(x_i) \in H$ and constructs an Optimal Separating Hyperplane (OSH), which maximizes the margin, the distance between the hyperplane and the nearest data points of each classes in the space H. A separating hyperplane in the feature space corresponding to a non-linear boundary in the input space is illustrated in Fig. 2 (Hua & Sun, 2001).

The mapping $\Phi(\bullet)$ is performed by a kernel function $K(x_i,x_j)$, which defines an inner product in the space H. The resulting classifier is based on the decision function given in Eq. (1) and the quadratic programming problem to determine the coefficients α_i (Hua & Sun, 2001).

$$f(x) = sgn\left(\sum_{i=1}^{N} y_i\alpha_i \cdot K(x, x_i) + b\right) \tag{1}$$

Quadratic Programming Problem:

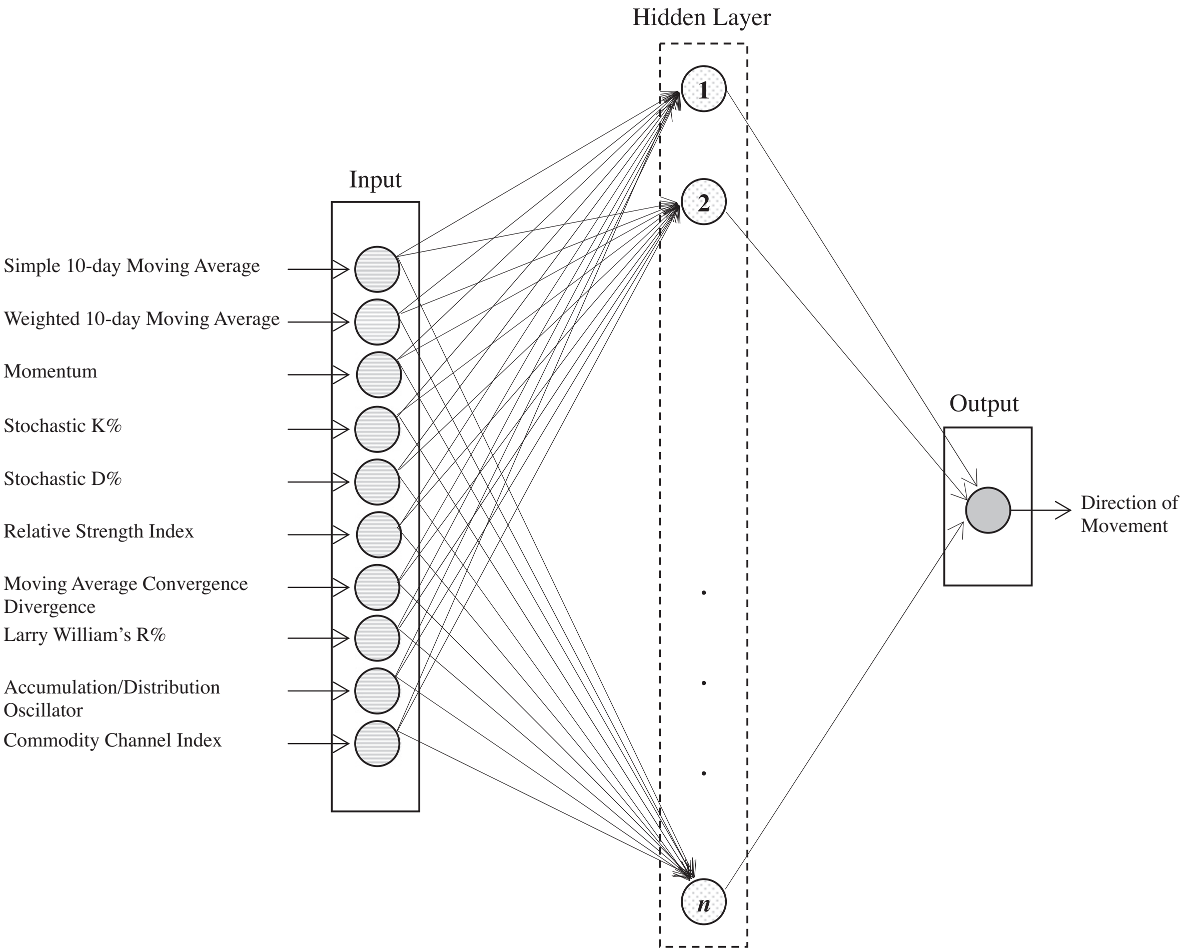

Fig. 1. The architecture of three layered feedforward ANN.

Table 6
ANN parameter levels tested in parameter setting.

Parameters	Level(s)
Number of neurons (n)	10, 20, …, 100
Epochs (ep)	1.000, 2.000, …, 10.000
Momentum constant (mc)	0.1, 0.2, …, 0.9
Learning rate (lr)	0.1

Table 7
SVM parameter levels tested in parameter setting experiments.

Parameters	Levels (*polynomial*)	Levels (*radial basis*)
Degree of kernel function (d)	1, 2, 3, 4	–
Gamma in kernel function (γ)	0, 0.1, 0.2, …, 5.0	0, 0.1, 0.2, …, 5.0
Regularization parameter (c)	1, 10, 100	1, 10, 100

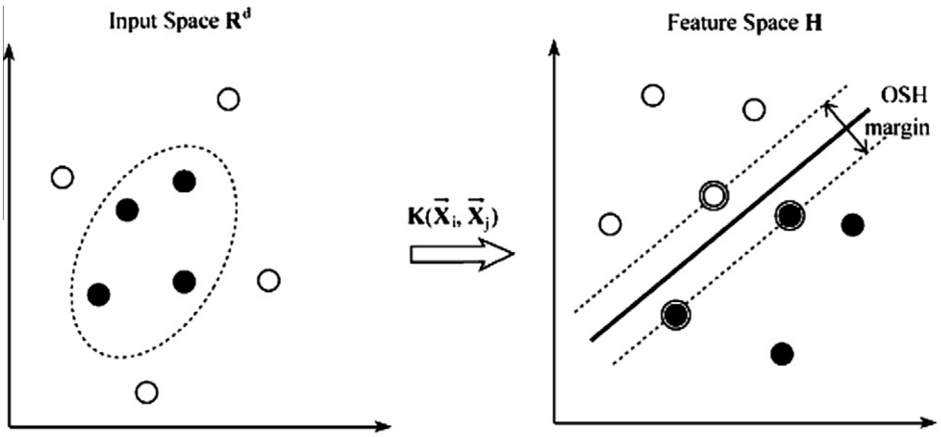

Fig. 2. A separating hyperplane in the feature space corresponding to a non-linear boundary in the input space (Hua & Sun, 2001).

$$\text{Maximize } \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j \cdot y_i y_j \cdot K(x_i, x_j) \tag{2}$$

$$\text{subject to } 0 \leqslant \alpha_i \leqslant c \tag{3}$$

$$\sum_{i=1}^{N}\alpha_i y_i = 0 \quad i = 1, 2, \ldots, N \tag{4}$$

where c is a regularization parameter which controls the trade off between margin and misclassification error.

There are two main kernel functions (polynomial and radial basis) used in SVM. These functions are given below:

$$\text{Polynomial Function} : K(x_i, x_j) = (x_i \bullet x_j + 1)^d \tag{5}$$

$$\text{Radial Basis Function} : K(x_i, x_j) = \exp(-\gamma\|x_i - x_j\|^2) \tag{6}$$

where d is the degree of polynomial function and γ is the constant of radial basis function.

The choice of kernel function is a critical decision for prediction efficiency. Both polynomial and radial basis functions were adopted in experiments. Several levels of the degree of polynomial function (d), gamma constant of radial basis function (γ) and regularization parameter (c) were tested in the parameter setting experiments. The SVM parameters and their levels are summarized in Table 7.

The parameter levels evaluated in parameter setting yield a total of 765 treatments for SVM. Each parameter combination was applied to the training and holdout data sets and prediction accuracy of the models were evaluated. A training performance and a holdout performance were calculated for each parameter combination. The parameter combination which resulted in the best average of training and holdout performances was selected as the best one for the corresponding model. All experiments were performed on MATLAB software using Chang and Lin (2001)'s library developed for SVM implementations.

5. Experimental results

At the first stage, the experiments for parameter setting were completed. A total of 900 parameter combinations for the ANN model were tested. The training performance of the ANN model for these parameter combinations was varied between 84.53% and 99.64%. On the other hand, the holdout performance of ANN model varied between 71.17% and 82.85%. It can be said that both the training and holdout performances of the ANN model are significant for parameter setting data set. However, it should be noted here that the best training performance and the best holdout performance were not obtained at the same parameter combination. Since we need one or a few parameter combinations for comparisons, we decided to calculate the average of training and holdout performances for each case. Then, we selected three parameter combinations which give the best three average performances. The best three parameter combinations and corresponding prediction accuracies are given in Table 8.

Three parameter combinations given in Table 8 are assumed to be the best ones in representing all cases in the entire data set. With these parameter combinations, we are now able to perform comparison experiments of the ANN model. The data sets summarized in Table 3 were applied to the ANN model with three different parameter combinations. The experiments were performed for each year separately and the results are given in Table 9.

Table 9 shows that the average training and holdout performances of the ANN model for three different parameter

Table 8
Best three parameter combinations of ANN model.

No	*Lr*	*ep*	*mc*	*n*	Training	Holdout	Average
1	0.1	5000	0.7	30	98.18	82.48	90.33
2	0.1	7000	0.1	90	98.54	81.75	90.15
3	0.1	6000	0.4	90	98.18	81.75	89.96

Table 9
Prediction performance (%) of ANN model for best parameter combinations (*lr* = 0.1).

	Parameter combination (*ep*; *mc*; *n*)					
Year	(5000; 0.7; 30)		(7000; 0.1; 90)		(6000; 0.4; 90)	
	Training	Holdout	Training	Holdout	Training	Holdout
1997	100	76.98	100	79.37	100	79.37
1998	99.19	73.29	99.19	73.39	98.39	76.61
1999	99.16	75.21	99.16	76.92	99.16	75.21
2000	100	76.42	100	79.67	100	78.86
2001	100	69.11	100	70.73	99.20	71.54
2002	98.43	76.80	99.21	79.20	100	77.60
2003	100	69.92	99.19	68.29	100	73.98
2004	100	75.00	100	72.58	100	72.58
2005	100	78.57	100	76.98	100	76.98
2006	98.41	74.19	97.62	70.97	97.62	74.19
2007	97.62	75.40	97.62	73.81	97.62	76.19
Average	99.35	74.63	99.27	74.72	99.27	75.74

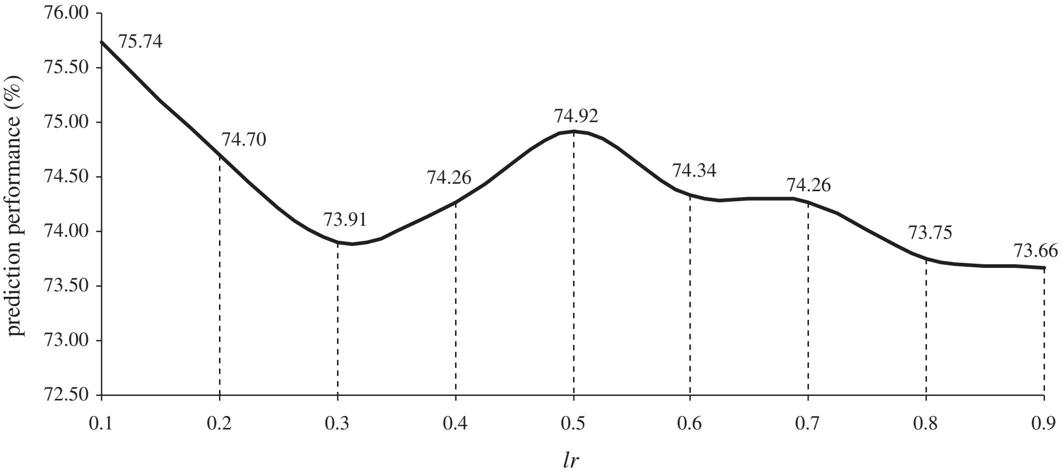

Fig. 3. Prediction performance (%) of ANN model for various *lr* values (*ep* = 6000; *mc* = 0.4; *n* = 90).

Table 10
Best three parameter combinations of SVM models.

No	Kernel function	d	γ	c	Training	Holdout	Average
1	Polynomial	3	2.6	100	99.27	75.18	87.23
2	Polynomial	3	2.5	100	98.91	75.18	87.04
3	Polynomial	3	3.2	100	100	73.72	86.86
4	Radial basis	–	2.5	100	100	76.64	88.32
5	Radial basis	–	3.0	100	100	76.28	88.14
6	Radial basis	–	3.1	100	100	75.91	87.96

Table 11
Prediction performance (%) of *polynomial* SVM model for best parameter combinations.

Parameter combination (d; γ; c)						
Year	(3; 2.6; 100)		(3; 2.5; 100)		(3; 3.2; 100)	
	Training	Holdout	Training	Holdout	Training	Holdout
1997	100	79.37	100	79.37	100	79.37
1998	100	69.36	100	69.36	100	69.36
1999	100	67.52	100	67.52	100	67.52
2000	100	76.42	100	76.42	100	76.42
2001	100	64.23	100	64.23	100	64.23
2002	100	78.40	100	78.40	100	78.40
2003	100	64.23	100	64.23	100	64.23
2004	100	66.94	100	66.94	100	66.94
2005	100	80.16	100	80.16	100	80.16
2006	100	73.39	100	73.39	100	73.39
2007	100	66.67	100	66.67	100	66.67
Average	100	71.52	100	71.52	100	71.52

Table 12
Prediction performance (%) of *radial basis* SVM model for best parameter combinations.

Parameter combination (γ; c)						
Year	(2.5; 100)		(3.0; 100)		(3.1; 100)	
	Training	Holdout	Training	Holdout	Training	Holdout
1997	100	71.43	100	70.63	100	69.84
1998	100	64.52	100	62.10	100	62.10
1999	100	67.52	100	64.96	100	64.96
2000	100	65.04	100	63.42	100	60.98
2001	100	59.35	100	60.98	100	59.35
2002	100	60.80	100	61.60	100	62.40
2003	100	59.35	100	56.10	100	55.29
2004	100	66.13	100	65.32	100	65.32
2005	100	66.67	100	65.08	100	64.29
2006	100	62.90	100	61.29	100	60.48
2007	100	60.32	100	59.52	100	59.52
Average	100	64.00	100	62.82	100	62.23

Table 13
t-Test results of model comparison.

Prediction model	N	Mean	Std. dev.	t	p
ANN	11	75.74	2.49	3.098	0.011
Polynomial SVM	11	71.52	6.18		

Table 14
Comparison of results with similar studies.

Diler (2003)	Altay and Satman (2005)	Our models		
BPN	BPN	OLS	BPN	SVM
60.81%	57.80%	55.00%	75.74%	71.52%

combinations are approximately the same. However, since its average holdout performance (75.74%) is relatively greater than the others, the performance of the third parameter combination (6000; 0.4; 90) is relatively better than others. Therefore, the prediction performance of this parameter combination can be adopted as the best of the ANN model and used in comparison with SVM models. Table 9 also shows that the prediction performances are different for each year. For the selected parameter combination, the best holdout performance (79.37%) was obtained in 1997 while the worst one (71.54%) was obtained in 2001. For the other two parameter combinations, the holdout performances in 2001 were generally lesser than the other years.

Before adopting the parameter combination ($lr = 0.1$; $ep = 6000$; $mc = 0.4$; $n = 90$) as the best one, we performed an additional experiment to see the effect of lr on the quality of prediction performance. Fixing the other parameter values to their selected values, all experiments were re-conducted by changing lr value. Eight new values (0.2, 0.3, ..., 0.9) of lr were tested and the average holdout performances are given in Fig. 3.

Fig. 3 shows that the best average holdout performance is achieved at $lr = 0.1$. The prediction performance of the ANN model is decreasing as the value of lr is increasing. Based on the experimental results given above, we decided that the best parameter combination of ANN model is $lr = 0.1$, $ep = 6000$, $mc = 0.4$ and $n = 90$ with an average holdout performance of 75.74%.

The same experimental procedure was followed for the SVM models. The parameter setting data set was applied to the SVM models to determine their efficient parameter combinations. A total of 612 parameter combinations for polynomial SVM model and a total of 153 parameter combinations for radial basis SVM model were tested. The training performance of the polynomial SVM model varied between 52.19% and 100% while its holdout performance varied between 52.19% and 82.85%. On the other hand, the training performance of radial basis SVM model varied between 72.63% and 100% while its holdout performance varied between 72.26% and 82.48%. As adopted in the parameter setting of ANN model, the average of the training and holdout performances were calculated for each case. The best three parameter combinations and corresponding results are given in Table 10.

As shown in Table 10, parameter values and average prediction performances of the two SVM models are close to each other. The data sets summarized in Table 3 were applied to the SVM models with three different parameter combinations for each year separately and the results are given in Tables 11 and 12, respectively.

Table 11 shows that prediction performance of the polynomial SVM model is the same for all parameter combinations. The training performance is 100% for all years while the holdout performance varies between 64.23% and 79.37% with an average of 71.52%. Similar to the ANN model, Table 11 also shows that polynomial SVM model yielded its worst prediction performance in 2001.

The prediction performance of radial basis SVM model is different for three parameter combinations and is less than that of the polynomial SVM model. The best prediction performance is obtained at $\gamma = 2.5$ and $c = 100$ and varies between 59.35% and 71.43% with an average of 64.00%.

The results showed that the polynomial SVM gives a better prediction performance than the radial basis SVM model. Therefore, the performance of the ANN model is compared with the polynomial SVM model. The best results of the ANN model and polynomial SVM model in holdout performance is compared with a paired samples *t*-test. The results of *t*-test are given in Table 13.

Table 13 shows that the difference between mean performances of models is significant at $\alpha = 0.05$ significance level. That is, the performance of the ANN is significantly better than the performance of the SVM model. We can compare our results with the results of existing studies in the literature. Although there are many studies in the literature which aim to predict stock price index movement, it would be reasonable to compare our results with the studies which were performed on ISE National 100 index. To the best knowledge of the authors, there are two studies on ISE 100 index by Diler (2003) and Altay and Satman (2005). Diler (2003) proposed a neural network approach (BPN) while Altay and Satman (2005) proposed a BPN model and Ordinary Least Square (OLS) regression model. Our results are compared with the results of these studies in Table 14.

The accurate prediction performance of the BPN model of Diler (2003) is 60.81% while that of Altay and Satman (2005) is 57.80. However, our BPN model accurately predicted the direction of movement at a rate 75.74%. Although the models are similar, the performances are significantly different and our model is superior to other BPN models. On the other hand, our SVM model is also superior to the models of Diler (2003) and Altay and Satman (2005). The set of technical indicators adopted in our models can be considered as more appropriate for prediction. In addition, the detailed analysis of models' parameters and selection of efficient parameter values may result in higher prediction accuracy.

6. Conclusions

Predicting the direction of movements of the stock market index is important for the development of effective market trading strategies. It usually affects a financial trader's decision to buy or sell an instrument. Successful prediction of stock prices may promise attractive benefits for investors. These tasks are highly complicated and very difficult. This study attempted to predict the direction of stock price movement in the Istanbul Stock Exchange. Two prediction models were constructed and their performances were compared on the daily data from 1997 to 2007. Based on the experimental results obtained, some important conclusions can be drawn. First of all, it should be emphasized that both the ANN and SVM models showed significant performance in predicting the direction of stock price movement. Thus, we can say that both the ANN and SVM are useful prediction tools for this topic. The average prediction performance of the ANN model (75.74%) was found significantly better than that of the SVM model (71.52%). To the best knowledge of the authors, the prediction performance of the proposed models outperforms similar studies in the literature. However, prediction performances of our models may be improved by two ways. The first is to adjust the model parameters by conducting a more sensitive and comprehensive parameter setting, which can be a future work for interested readers. Second, different or additional input variables can be used in the models. Although we adopted ten technical indicators, some other macro economic variables such as foreign exchange rates, interest rates and consumer price index etc. can be used as inputs of the models. Nevertheless, ten technical indicators adopted here proved that they are useful in predicting the direction of stock price movement. Another important issue that should be mentioned here is the differences among the prediction performances for each year. It can be shown from the experimental results that the worst prediction performance was obtained in 2001. This is not a coincidence for us because Turkey had a devastating economical crisis in 2001. The crisis in the banking system had affected the stock market initially. The index had decreased to its lowest level since 1999. The ISE had lost 30% of its value. Under such circumstances of crisis, a decrease in the prediction performance of technical indicators can be considered acceptable.

References

Abu-Mostafa, Y. S., & Atiya, A. F. (1996). Introduction to financial forecasting. *Applied Intelligence, 6*(3), 205–213.

Altay, E., & Satman, M. H. (2005). Stock market forecasting: Artificial neural networks and linear regression comparison in an emerging market. *Journal of Financial Management and Analysis, 18*(2), 18–33.

Armano, G., Marchesi, M., & Murru, A. A. (2005). Hybrid genetic-neural architecture for stock indexes forecasting. *Information Sciences, 170*, 3–33.

Atsalakis, G. S., & Valavanis, K. P. (2009). Surveying stock market forecasting techniques – Part II: Soft computing methods. *Expert Systems with Applications, 36*(3), 5932–5941.

Avci, E. (2007). Forecasting daily and sessional returns of the ISE-100 index with neural network models. *Journal of Dogus University, 8*(2), 128–142.

Baba, N., & Kozaki, M. (1992). An intelligent forecasting system of stock price using neural networks. In *Proceedings of the international joint conference on neural networks, Baltimore, MD, USA* (pp. 371–377).

Bildik, R. (2001). Intra-day seasonalities on stock returns: Evidence from the Turkish Stock Market. *Emerging Markets Review, 2*, 387–417.

Cao, Q., Leggio, K. B., & Schniederjans, M. J. A. (2005). A comparison between Fama and French's model and artificial neural networks in predicting the Chinese stock market. *Computers & Operations Research, 32*, 2499–2512.

Chang, C. C., & Lin, C. J. (2001). LIBSVM: A library for support vector machines. <http://www.csie.ntu.edu.tw/∼cjlin/libsvm>.

Chen, A. S., Leung, M. T., & Daouk, H. (2003). Application of neural networks to an emerging financial market: Forecasting and trading the Taiwan Stock Index. *Computers & Operations Research, 30*(6), 901–923.

Chu, H. H., Chen, T. L., Cheng, C. H., & Huang, C. C. (2009). Fuzzy dual-factor time-series for stock index forecasting. *Expert Systems with Applications, 36*(1), 165–171.

Diler, A. I. (2003). Predicting direction of ISE national-100 index with back propagation trained neural network. *Journal of Istanbul Stock Exchange, 7*(25–26), 65–81.

Egeli, B., Ozturan, M., & Badur, B. (2003). Stock market prediction using artificial neural networks. In *Proceedings of the third Hawaii international conference on business, Honolulu, Hawai*.

Hiemstra, Y. (1995). Modeling structured nonlinear knowledge to predict stock market returns. In R. R. Trippi (Ed.), *Chaos and nonlinear dynamics in the financial markets: Theory, evidence and applications* (pp. 163–175). Chicago, IL: Irwin.

Hsu, S. H., Hsieh, J. J. P. A., Chih, T. C., & Hsu, K. C. (2009). A two-stage architecture for stock price forecasting by integrating self-organizing map and support vector regression. *Expert Systems with Applications, 36*(4), 7947–7951.

Huang, W., Nakamori, Y., & Wang, S. Y. (2005). Forecasting stock market movement direction with support vector machine. *Computers & Operations Research, 32*, 2513–2522.

Huang, C. L., & Tsai, C. Y. (2009). A hybrid SOFM-SVR with a filter-based feature selection for stock market forecasting. *Expert Systems with Applications, 36*(2), 1529–1539.

Hua, S., & Sun, Z. (2001). Support vector machine approach for protein subcellular localization prediction. *Bioinformatics, 17*(8), 721–728.

Karaatli, M., Gungor, I., Demir, Y., & Kalayci, S. (2005). Estimating stock market movements with neural network approach. *Journal of Balikesir University, 2*(1), 22–48.

Khemchandani, R., & Jayadeva Chandra, S. (2009). Knowledge based proximal support vector machines. *European Journal of Operational Research, 195*, 914–923.

Kim, K. (2003). Financial time series forecasting using support vector machines. *Neurocomputing, 55*, 307–319.

Kim, S. H., & Chun, S. H. (1998). Graded forecasting using an array of bipolar predictions: Application of probabilistic neural networks to a stock market index. *International Journal of Forecasting, 14*, 323–337.

Kim, K., & Han, I. (2000). Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert Systems with Applications, 19*, 125–132.

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

Leigh, W., Purvis, R., & Ragusa, J. M. (2002). Forecasting the NYSE composite index with technical analysis, pattern recognizer, neural network, and genetic algorithm: A case study in romantic decision support. *Decision Support Systems, 32*(4), 361–377.

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.

Manish, K., & Thenmozhi, M. (2005). Forecasting stock index movement: A comparison of support vector machines and random forest. In *Proceedings of ninth Indian institute of capital markets conference, Mumbai, India*. <http://ssrn.com/abstract=876544>.

Manish, K., & Thenmozhi, M. (2006). Support vector machines approach to predict the S&P CNX NIFTY index returns. In *Proceedings of 10th Indian institute of capital markets conference, Mumbai, India*. <http://ssrn.com/abstract=962833>.

Oh, K. J., & Kim, K. J. (2002). Analyzing stock market tick data using piecewise nonlinear model. *Expert Systems with Applications, 2*, 249–255.

Olson, D., & Mossman, C. (2003). Neural network forecasts of Canadian stock returns using accounting ratios. *International Journal of Forecasting, 19*(3), 453–465.

Pai, P. F., & Lin, C. S. (2005). A hybrid ARIMA and support vector machines model in stock price forecasting. *Omega, 33*, 497–505.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back- propagating errors. *Nature, 323*, 533–536.

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

Takahashi, T., Tamada, R., & Nagasaka, K. (1998). Multiple line-segments regression for stock prices and long-range forecasting system by neural networks. In *Proceedings of the 37th SICE annual conference, Chiba, Japan* (pp. 1127–1132).

Tan, T. Z., Quek, C., & See, Ng. G. (2007). Biological brain-inspired genetic complementary learning for stock market and bank failure prediction. *Computational Intelligence, 23*(2), 236–261.

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

Vapnik, V. N. (1995). *The nature of statistical learning theory*. New York: Springer.

White, H. (1988). Economic prediction using neural networks: The case of IBM daily stock returns. In *Proceedings of the IEEE international conference on neural networks, San Diego, CA, USA* (pp. 451–458).

Xu, X., Zhou, C., & Wang, Z. (2009). Credit scoring algorithm based on link analysis ranking with support vector machine. *Expert Systems with Applications, 36*, 2625–2632.

Yao, J., Chew, L. T., & Poh, H. L. (1999). Neural networks for technical analysis: A study on KLCI. *International Journal of Theoretical and Applied Finance, 2*(2), 221–241.

Yoon, Y., & Swales, G. (1991). Predicting stock price performance: A neural network approach. In *Proceedings of the 24th annual Hawaii international conference on systems sciences, Hawaii* (pp. 156-162).

Yudong, Z., & Lenan, W. (2009). Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network. *Expert Systems with Applications, 36*(5), 8849–8854.