A Comparison of PNN and SVM for Stock Market Trend Prediction using Economic and Technical Information

Salim Lahmiri
Dept. of Computer Science
University of Quebec at Montreal
Montreal, Canada

ABSTRACT
Probabilistic Neural Networks (PNN) and Support vector machines (SVM) are employed to predict stock market daily trends: ups and downs. The purpose is to examine the effect of macroeconomic information and technical analysis indicators on the accuracy of the classifiers. In addition, the study aims to study their joint effect on the classification performance when used together. First, Granger tests were performed to identify causal relationships between the input variables and the predicted stock returns. Then, lagged returns to be considered in the input space are identified by use of autocorrelation function. Finally, the hit ratio of predictions by PNN and SVM were compared. It is found that macroeconomic information is suitable to predict stock market trends than the use of technical indicators. In addition, the combination of the two sets of predictive inputs does not improve the forecasting accuracy. Furthermore, the prediction accuracy improves when trading strategies are considered.

General Terms
Pattern Recognition, Time Series.

Keywords
Probabilistic Neural Networks, Support Vector Machines, Classification, Stock Market

1. INTRODUCTION
Forecasting stock market behavior is a very difficult task since its dynamics are complex and non-linear. For instance, stock return series are generally noisy and may be influenced by many factors; such as the economy, business conditions, and political events to name a few. Indeed, empirical finance shows that publicly available data on financial and economic variables may explain stock return fluctuations in the United States [1]-[8]. For instance, a number of applications have been proposed to forecast stock market returns with macroeconomic variables with the use of neural networks [9-12] and Bayesian networks and support vector machines [13][14]. On the other hand, technical indicators have been also used to predict stock market movements using neural networks [15-21], adaptive fuzzy inference system [22][23], and fuzzy logic [24-26]. The literature shows that economic variables and technical indicators have achieved success in predicting the stock market. However, none of the previous studies have compared the performance of the economic information and technical indicators in terms of prediction accuracy. Indeed, it is not known what type of information leads to better forecasts. In addition, one wonders what would be the effect of combining the two types of information on the prediction accuracy. Unlike the literature, only predictive variables that show strong evidence of causal relationship with return series are considered. In addition, lagged returns to be considered in the input space are statistically determined. The purpose of this study is to predict stock market trends (ups and downs) with macroeconomic variables and technical indicators separately and jointly in order to compare the performance of the classifiers. For instance, two classifiers are employed. They are the probabilistic neural networks (PNN) and support vector machines (SVM).

The PNN provides a general solution to pattern classification problems based on Bayesian theory. It is chosen because of its ability to classify a new sample with the maximum probability of success given a large training set using prior knowledge [27]. The PNN combines the simplicity, speed and transparency of traditional statistical classification models and the computational power and flexibility of back-propagated neural networks [28]. On the other hand, SVM are expressed in the form of a hyper-plane that discriminates between positive and negative instances. This is achieved by maximizing the distance between the two classes (positive and negative instances) and the hyper-plane. The SVM are applied in this study since they can avoid local minima and have superior generalization capability [29].

The rest of the paper is organized as follows. The data and pre-processing, and classifiers are presented in section 2. In Section 3, the results are provided. Section 4 contains the concluding remarks.

2. METHODOLOGY

2.1 Data and pre-processing
The initial sample consisted of the S&P500 daily returns as well as economic variables, all from January 11, 2000, to January 31, 2008, with no missing values. The economic variables were obtained from the FRED database of the Saint-Louis Federal Bank [30]. Given that the objective of the study was to predict future daily stock returns (R_{t+1}), the continuously compounded rate of return is computed on a daily basis as follows:

$$R_t = \log\left(\frac{p_t}{p_{t-1}} \right)$$

where p is the price of the S&P500 index. Figure 1 exhibits the S&P500 return series (R_t). Tables 1 and 2 show the economic variables and technical indicators that were employed for the prediction task. The choice of these primary predictive inputs is based on their frequent use in the literature [1-26].

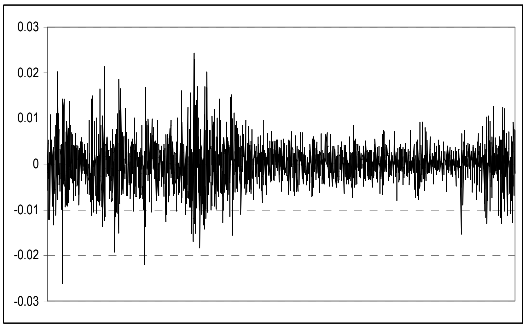

Fig. 1. Return series of the S&P500 Index.

Table 1. List of macroeconomic variables

Variables	Description
DBAA	Moody's Seasoned Baa Corporate Bond Yield
DTB3	3-Month Treasury Bill: Secondary Market Rate
DTB6	6-Month Treasury Bill: Secondary Market Rate
DFEDTAR	Federal Funds Target Rate
DFF	Effective Federal Funds Rate
DEXCAUS	Canada / U.S. Foreign Exchange Rate
DEXJPUS	Japan / U.S. Foreign Exchange Rate
DEXSZUS	Switzerland / U.S. Foreign Exchange Rate
DEXUSEU	U.S. / Euro Foreign Exchange Rate
DEXUSUK	U.S. / U.K Foreign Exchange Rate
DTWEXB	Trade Weighted Exchange Index: Broad
DTWEXM	Trade Weighted Exchange Index: Major Currencies

Table 2. List of technical indicators

Indicators	Description
MA(t)	moving average
X1(t)	[X(t) - lowest price(t)] / [(highest price(t) - lowest price(t)]
X2(t)	[X(t) - MA(t)] / MA(t)
X3(t)	X(t) - MIN[X(t),…,X(t-5)]
X4(t)	X(t) - MAX[X(t),…,X(t-5)]
MPM(t)	[(highest price(t)-lowest price(t))-(highest price(t-2)-lowest price(t-2))]/2
BR(t)	volume(t)/[(highest price(t) - lowest price(t))/2]
EMV(t)	MPM(t)/BR(t)
MOM(t)	closing price (t) - closing price (t-6)
ROC(t)	closing price (t) / closing price (t-6)
DIS(t)	[closing price (t) / MA(t-6)]*100
STOD(t)	[closing price(t) - low price(t,t-6)]/ [high price(t, t-6) – low price(t,t-6)]

Unlike previous studies [9-26] and in order to improve the coherence of the input-output mapping, Granger tests [31] were performed to identify causal relationships between predictive inputs and future stock returns. As a result, only statistically causal inputs are retained and the number of dimension in the input space is reduced. For instance, the relevant inputs were selected by first running Granger causality tests on the macro economic variables and technical indicators. Since correlation does not necessarily imply causation, the Granger test was employed to investigate whether x (the macro variables or the technical indicators) causes y (the stock returns in this work). This simple statistical approach is about seeing how the current y can be explained by past values of x and then whether adding k lagged values of x can improve the explanation. In other words, y is said to be Granger-caused by x if x helps forecast y. The Granger test allows considering only inputs that have a high statistical causal effect on future stock returns. The test is based on bivariate regressions of the form:

$$y_t = \beta_{y,0} + \beta_{y,1} y_{t-1} + \ldots + \beta_{y,k} y_{t-k} + \delta_{x,1} x_{t-1} + \ldots + \delta_{x,k} x_{t-k} + \eta_t$$

$$x_t = \beta_{x,0} + \beta_{x,1} x_{t-1} + \ldots + \beta_{x,k} x_{t-k} + \delta_{y,1} y_{t-1} + \ldots + \delta_{y,k} y_{t-k} + \upsilon_t$$

where η and υ represent Gaussian disturbances. Then, F-statistics are computed as the Wald statistics [28] for the joint hypothesis:

$$\delta_1 = \delta_2 = \ldots = \delta_k = 0$$

The F-statistics allows testing whether the coefficients on the lagged x's are statistically significant in explaining the dependant y. In this study, the number of lags, k, was set to 5 and the retained statistical significance is 5%. Tables 3 and 4 provide the obtained results from the Granger causality tests for macroeconomic variables and technical indicators respectively.

Table 3. Granger tests for macro variables

Null Hypothesis	F-Stat.	Probability
DAAA does not Granger Cause *R*	0.95143	0.4465
DBAA does not Granger Cause *R*	1.16856	0.32223
DEXCAUS does not Granger Cause *R*	1.06268	0.37919
DEXJPUS does not Granger Cause *R*	1.44641	0.20448
DEXSZUS does not Granger Cause *R*	2.57318	**0.02498**
DEXUSEU does not Granger Cause *R*	1.4195	0.21401
DEXUSUK does not Granger Cause *R*	1.23676	0.2892
DFEDTAR does not Granger Cause *R*	1.0858	0.36616
DFF does not Granger Cause *R*	3.07175	**0.00911**
DTB3 does not Granger Cause *R*	0.26302	0.93331

DTB6 does not Granger Cause *R*	1.21703	0.29847
DTWEXB does not Granger Cause *R*	2.08736	0.06423
DTWEXM does not Granger Cause *R*	1.72344	0.12589

Table 4. Granger tests for technical indicators

Null Hypothesis	F-Statistic	Probability
BR does not Granger Cause *R*	0.24284	0.94345
DIS does not Granger Cause *R*	1.58049	0.16224
EMV does not Granger Cause *R*	2.22486	**0.0494**
MOM does not Granger Cause *R*	1.00126	0.41542
MPM does not Granger Cause *R*	1.48907	0.1901
ROC does not Granger Cause *R*	0.98104	0.42785
STOD does not Granger Cause *R*	0.68897	0.63181
X1 does not Granger Cause *R*	0.95859	0.44194
X2 does not Granger Cause *R*	1.58049	0.16224
X3 does not Granger Cause *R*	1.04415	0.38987
X4 does not Granger Cause *R*	2.64313	**0.02173**

Granger causality tests show strong evidence that DEXSZUS, DFF, EMV, and X4 cause changes in S&P500 returns. For instance, the null hypothesis is rejected with probability 0.02498 for DEXSZUS, with probability 0.00911 for DFF, with probability 0.0494 for EMV, and with probability 0.02173 for X4. Thus, two macroeconomic variables (DEXSZUS and DFF) and two technical indicators (EMV and X4) are selected to be fed to the classifiers as inputs. For example, the PNN and SVM are trained and tested with these resulting data. The first 80% of the data (1620 observations) were used for model training while the last 20% (405 observations) were used for out-of-sample forecasting. Cross validation is not considered since the goal of this study is to model time series and no future observations are used to predict past ones. The variable to be predicted is stock market future trends. Then, the output becomes:

$$y_t = \{-1 \quad if \quad R_t < 0; +1 \quad if \quad R_t > 0\}$$

In order to test what type of information (macroeconomic variables versus technical indicators) provide higher accuracy rate, and whether the combination of the two type of information helps improving the prediction accuracy, the following prediction models (PM) are considered:

Model.1:

$$y_t = f(R_{t-1}, R_{t-2} \ldots, R_{t-k}, DEXSZUS_{t-1}, DFF_{t-1})$$

Model.2:

$$y_t = f(R_{t-1}, R_{t-2}, \ldots, R_{t-k}, EMV_{t-1}, X_7)$$

Model.3:

$$y_t = f(R_{t-1}, R_{t-2}, \ldots, R_{t-k}, DEXSZUS_{t-1}, DFF_{t-1}, EMV_{t-1}, X_7)$$

where *t* is time script and *k* is a lag order which is determined using the auto-correlation function τ given by:

$$\tau_k = \left[\sum_{t=k+1}^{T} \left(R_t - \overline{R}\right)\left(R_{t-k} - \overline{R}\right) \right] \left[\sum_{t=1}^{T} \left(R_t - \overline{R}\right)^2 \right]^{-1}$$

where $\overline{R}$ is the sample mean of R_t. The appropriate *k* is determined following the methodology of [32][33]. Indeed, it is important to include past returns to predict future market directions if the return series are auto-correlated. In other words, history of the returns helps predicting future returns. Figure 2 shows the auto-correlation function up to 10 lags. Since τ is nonzero *for k=1 and k=2*, it means that the series is serially correlated. In addition, the values of the auto-correlation function die quickly in the first three lags which is a sign that the series obeys a low-order autoregressive (AR) process; for example an AR(2). Therefore, the number of lags to be included in the prediction models is up to *k*=2. Finally, the following prediction models (PM) are implemented and simulated:

PM.1:

$$y_t = f(R_{t-1}, R_{t-2}, DEXSZUS_{t-1}, DFF_{t-1})$$

PM.2:

$$y_t = f(R_{t-1}, R_{t-2}, EMV_{t-1}, X_7)$$

PM.3:

$$y_t = f(R_{t-1}, R_{t-2}, DEXSZUS_{t-1}, DFF_{t-1}, EMV_{t-1}, X_7)$$

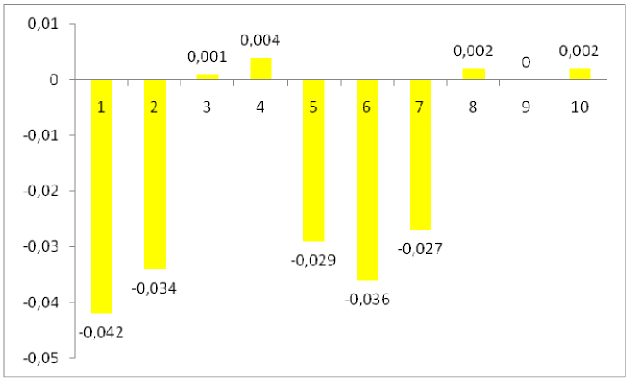

Fig. 2. Auto-correlation function τ_k of S&P500 returns at different lags $k \in [1,10]$.

2.2 The classifiers

2.2.1 Probabilistic neural networks

The PNN was proposed by Specht [28]. It is built based upon the Bayesian method of classification. Indeed, the PNN employs Bayesian decision-making theory based on an estimate of the probability density of the data. The probabilistic neural network requires only a single presentation of each pattern. The PNN employs an exponential activation function rather than the sigmoid function that is commonly used in the multi-layer perceptron. Then, PNN can identify nonlinear decision boundaries that approach the Bayes optimal [34]. The basic network topology consists of four layers. The first layer is the inputs layer. In the second layer, the probability density function (PDF) of each group of patterns is directly estimated from the set of training samples using [35] window approximation method. The third layer performs the summation of all PDFs. Finally, the Bayesian decision is made in the fourth layer. In sum, the network structure of PNN is similar to back-propagation neural network; but the main difference is that the transfer function is replaced by exponential function and all training samples are stored as weight vectors. For instance, the PDF is assumed to follows a Gaussian distribution. Then, the PDF for a feature vector X to be of a certain category A is given by:

$$f_A(X) = \left(1/(2\pi)^{0.5p}\sigma^p\right)m^{-1}\sum_{i=1}^{m}\exp\left((X - X_{Ai})'(X - X_{Ai})/2\sigma^2\right)$$

where, p is the number of patterns in X, m is the number of the training patterns of category A, i is the pattern number, and σ is the smoothing factor of the Gaussian curves used to construct the PDF. The value of σ is optimized during training based on the clearest separation of classes with the highest classification rate [36].

2.2.2 Support vector machines

Support Vector Machines (SVM) is a supervised statistical learning technique introduced by Vapnik [29]. It is one of the standard tools for machine learning successfully applied in many different real-world problems. For instance, they have been successfully applied in financial time series trend prediction [14][37]. The SVM were originally formulated for binary classification. The SVM seek to implement an optimal marginal classifier that minimizes the structural risk in two steps. First, SVM transform the input to a higher dimensional space with a kernel (mapping) function. Second, SVM linearly combine them with a weight vector to obtain the output. As result, SVM provide very interesting advantages. They avoid local minima in the optimization process. In addition, they offer scalability and generalization capabilities. For instance, to solve a binary classification problem in which the output $y \in \{-1,+1\}$ SVM seek for a hyper-plane $w.\Phi(x)+b = 0$ to separate the data from classes +1 and −1 with a maximal margin. Here, x denotes the input feature vector, w is a weight vector, Φ is the mapping function to a higher dimension, and b is the bias used for classification of samples. The maximization of the margin is equivalent to minimizing the norm of w [38]. Thus, to find w and b, the following optimization problem is solved:

$$Minimize: \quad \|w\|^2 + C\sum_{i=1}^{n}\xi_i$$

$$s.t \quad y_i\left(w \cdot \Phi(x_i) + b\right) \geq 1 - \xi_i \quad \xi_i \geq 0 \quad i = 1,\ldots,n$$

where C is a strictly positive parameter that determines the tradeoff between the maximum margin and the minimum classification error, n is the total number of samples, and generalization and ξ is the error magnitude of the classification.

The conditions ensure that no training example should be within the margins. The number of training errors and examples within the margins is controlled by the minimization of the term:

$$\sum_{i=1}^{n}\xi_i$$

The solution to the previous minimization problem gives the decision frontier:

$$f(x) = \sum_{x_i} y_i\alpha_i\Phi(x_i)\cdot\Phi(x) + b$$

where each α_i is a Lagrange coefficient. As mentioned before, the role of the kernel function is to implicitly map the input vector into a high-dimensional feature space to achieve better separability. In this study the polynomial kernel is used since it is a global kernel. For instance, global kernels allow data points that are far away from each other to have an influence on the kernel values as well [39].

$$K(x, x_i) = \Phi(x_i)\Phi(x) = ((x_i \cdot x) + 1)^d$$

where the kernel parameter d is the degree of the polynomial to be used. In this study, d is set to 2. Finally, the optimal decision separating function can be obtained as follows:

$$y = sign\left(\sum_{i=1}^{n} y_i \alpha_i K(x_i, x) + b\right)$$

3. RESULTS

The accuracy is calculated based on the number of correct classifications (Hit Ratio). For example, 100% accuracy is where all records are properly classified and 0% accuracy is where none are properly classified. The results obtained from simulation are shown in Figure 3. They show that the PNN achieved its highest and lowest accuracy with technical indicators (54%) and macroeconomic information (53%) respectively. In addition, the simulations show that the best performance (64%) is obtained with support vector machines when macroeconomic information is fed to the classifier. On the other hand, the lowest performance (49%) is achieved by support vector machines when technical indicators are used as inputs to the classifier. The findings suggest that the performance of each classifier depends on the type of input. For instance, PNN performs best with technical indicators, whilst SVM perform best with economic information.

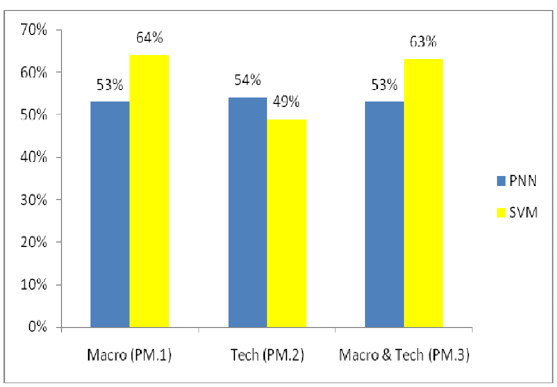

Fig. 3. Classification accuracy

Finally, the performance of both PNN and SVM decreases when macroeconomic information and technical indicators are fed to the classifiers. Then, the combination of macroeconomic information and technical indicators does not help to improve the prediction accuracy. Although the prediction accuracy is low, the overall results are interesting since some previous studies reported that stock prices are approximately close to the random walk process and; consequently; an accuracy of 56% in the predictions is a satisfying result for stock forecasting [40]-[43]. However, it is possible to improve the accuracy if we consider two investment strategies. The first strategy is to predict stock market ups by more than a predetermined rate, and the second strategy is to detect downs by less than another predetermined rate. Indeed, investors act as if they extrapolate a positive price trend by overbuying winners [44]. Therefore, they are likely to follow a momentum investment strategy and buy winners [45]-[47]. Then, in order to improve the prediction of the future trend in S&P500, two trading strategies are defined. For instance, the output of PNN is defined according to two strategies as follows:

Strategy.1:

$$y_{i,t} = \{0 \quad if \quad R_{i,t} < 0.5\% ; 1 \quad if \quad R_{i,t} \geq 0.5\%\}$$

Strategy.2:

$$y_{i,t} = \{1 \quad if \quad R_{i,t} \leq -0.5\% ; 0 \quad if \quad R_{i,t} > -0.5\%\}$$

The first strategy is designed to detect S&P500 ups by more than 0.5%. This strategy is designed for an aggressive investor who is a risk-taker. On the other hand, the second strategy is designed to detect S&P500 downs by less than 0.5%. It is designed for a risk-averse investor who dislikes losses. The thresholds 0.5% and -0.5% are arbitrarily chosen. For the classification task, the PNN is considered since it employs Bayesian decision-making theory which is more suitable to estimate the probability of future gains (strategy 1) and losses (strategy 2). The forecasting performance of the PNN given each strategy is shown in Figure 4.

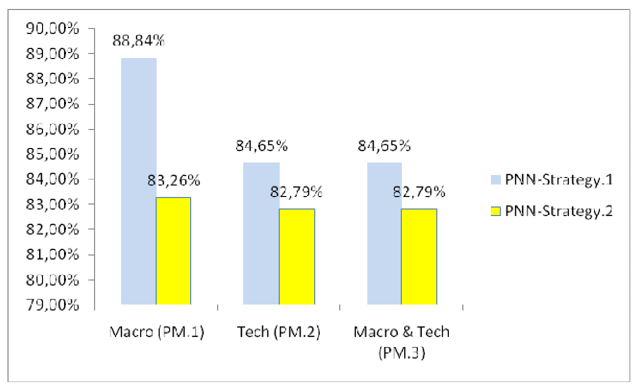

Fig. 4. Classification accuracy given strategies 1&2

The simulation results show that the accuracy increases in comparison with the results provided in Figure 3. In addition, the best accuracy is obtained with macroeconomic information (88.84%) when strategy 1 is adopted. On the other hand, the lowest accuracy is obtained with technical indicators (82.79%) when strategy 2 is adopted. As a result, macroeconomic information is more suitable for both risk-taker and risk-averse investors. In other words, economic information has proven its superiority in stock market prediction than technical indicators. Finally, in both strategies the combination of macroeconomic information and technical indicators does not help improving the performance of the prediction system.

4. CONCLUSION

The purpose of this study is to predict stock market trends (ups and downs) with macroeconomic variables and technical indicators separately and jointly using intelligent and statistical systems. Indeed, the contribution is to examine the effect of each type of predictive inputs; and also their combination; on the classification accuracy. Another contribution in this study is to consider a better selection of the predictive variables. For instance, unlike previous literature [14][48]-[52] that focus on the prediction of ups and downs of stock markets; Granger causality tests are performed to identify inputs (macroeconomic information and technical indicators) that statistically cause changes in stock returns. In addition, lagged returns -that will form the input space along with inputs selected by Granger tests- are statistically identified by use of auto-correlation function. In other words, the number of lagged returns as predictive variables is not set arbitrarily.

In this work the S&P500 returns are studied through probabilistic neural networks (PNN) and support vector machines (SVM). As shown in the experiments, PNN performs best with technical indicators, whilst SVM perform best with economic information. In addition, the combination of macroeconomic information and technical indicators does not help to improve the prediction accuracy. However, the best accuracy is obtained accuracy is obtained with SVM using economic information (64%). Although the prediction accuracy is low, the overall results are interesting since some previous studies reported that stock prices follow a random walk process. To improve the accuracy, two trading strategies are defined. The first trading strategy is designed for risk-taker investors to detect S&P500 ups. The second trading strategy is designed for risk-averse investors to detect S&P500 downs. As a result, the accuracy increased up to 88.84% and 83.26% for strategy 1 and strategy 2 respectively. In addition, the simulations show that for both strategies, economic information provides more accurate predictions than technical indicators. Furthermore, the combination of macroeconomic information and technical indicators does not help improving the performance of the prediction system in both trading strategies.

In sum, both PNN and SVM confirm that macroeconomic information is suitable to predict stock market trends than the use of technical indicators. This finding is explained by the fact that economic theory provides strong economic models relating stock market behaviour to the economy cycles and conditions [1-8]. On the other hand, technical analysis suffers from lack of theoretical foundations. For future researches, a large set of stock markets would be considered to validate the results. Also, the performance of the SVM with different polynomial orders would be investigated for each trading strategy.

5. REFERENCES

[1] Castanias R.P. Macro information and the Variability of Stock Market Prices. Journal of Finance 34 (1979), pp.439–50.

[2] Schwert G William. The Adjustment of Stock Prices to Information about Inflation. The Journal of Finance 36 (1981), pp.15-29.

[3] Schwert G William. Stock Returns and Real Activity: A Century of Evidence. Journal of Finance 14 (1990), pp.1237-1257.

[4] Fama EF. Stock Returns, Real Activity, Inflation and Money. American Economic Review 71 (1981), pp.71:545–65.

[5] Nai-Fu Chen, Roll R, Ross R. Economic Forces and The Stock Market. Journal of Business 59 (1986), pp.383-403.

[6] Hardouvelis Gikas A. Macroeconomic Information and Stock Prices. Journal of Economics and Business 1987;39:131-140.

[7] Darrat AF. Stock Returns, Money and Fiscal Deficits. Journal of Financial and Quantitative Analysis 25 (1990), pp.387–98.

[8] Flannery MJ, Protopapadakis AA. Macroeconomic Factors Do Influence Aggregate Stock Returns. The Review of Financial Studies 15 (2002), pp.751–82.

[9] Yiwen Y, Guizhong L, Zongping Z. Stock market trend prediction based on neural networks. Multiresolution Analysis and Dynamical Reconstruction. In Proceedings of the IEEE/IAFE/INFORMS Conference on Computational Intelligence for Financial Engineering, 2000.

[10] Wu X, Fung M, Flitman A. Forecasting stock market performance using hybrid intelligent system. Proceedings of the International Conference on Computational Science, 2001.

[11] Thawornwong S, Enke D. The adaptive selection of financial and economic variables for use with artificial neural networks. Neurocomputing 56 (2004), pp.205–232.

[12] Egeli B, Ozturan, M, Badur B. Stock market prediction using artificial neural networks. Proceedings of 3rd Hawaii International Conference on Business, 2003.

[13] Chen A S, Leung MT, Daouk H. Application of neural networks to an emerging financial market: Forecasting and trading the Taiwan Stock Index. Computers and Operations Research 30 (2003), pp.901–923.

[14] Huang W, Nakamori Y, Wang S-Y. Forecasting stock market movement direction with support vector machine. Computer and Operations Research 32 (2005), pp.2513–2522.

[15] Armano G, Marchesi M, Murru A. A hybrid genetic-neural architecture for stock indexes forecasting. Information Sciences 17 (2004), pp.3–33.

[16] Jaruszewicz M, Mandziuk J. One day prediction of NIKKEI Index considering information from other stock markets. Lecture Notes in Computer Science 3070 Springer (2004).

[17] Leigh W, Paz M, Purvis R. An analysis of a hybrid neural network and pattern recognition technique for predicting short-term increases in the NYSE Composite Index. Omega 30 (2002), pp.69–76.

[18] Lendasse A, De Bodt E, Wertz V, Verleysen M. Non-linear financial time series forecasting – Application to the Belgium 20 Stock Market Index. European Journal of Economical and Social Systems 14 (2000), pp.81–91.

[19] Motiwalla L, Wahab M. Predictable variation and profitable trading of US equities: A trading simulation using neural networks. Computer and Operations Research 27 (2000), pp.1111–1129.

[20] Bautista C C. Predicting the Philippine Stock Price Index using artificial neural networks. UPCBA Discussion 0107 (2001).

[21] Dong I, Duan C, Jang M-J. Predicting extreme stock performance more accurately. A paper written for ''Government 2001", 2003.

[22] Atsalakis G, Valavanis K. Neuro-fuzzy and technical analysis for stock prediction. Working paper, Technical University of Crete, 2006.

[23] Baek J, Cho S. Time to jump. Long rising pattern detection in KOSPI 200 future using an auto-associative neural network. Lecture Notes in Computer Science 2412. Springer, 2002.

[24] Dong M, Zhou X. Exploring the fuzzy nature of technical patterns of U.S. stock market. Proceedings of Fuzzy System and Knowledge Discovery 1 (2002), pp.324–328.

[25] Dourra H, Siy, P. Investment using technical analysis and fuzzy logic. Fuzzy Sets and Systems, 127 (2002), pp.221–240.

[26] Lam S S. A genetic fuzzy expert system for stock market timing. Proceedings of the IEEE Conference on Evolutionary Computation, 2001, pp. 410–417.

[27] P. Wasserman. Advanced Methods in Neural Computing. New York: Van No strand Reinhold, 1993.

[28] D. Specht. Probabilistic Neural Networks. Neural Networks **3** (1990), 109-118.

[29] Vapnik V N. The Nature of Statistical Learning Theory, Springer-Verlag, 1995.

[30] http://research.stlouisfed.org/fred2/

[31] Granger C W J. 1969. Investigating Causal Relations by Econometric Models and Cross-Spectral Methods. Econometrica 37 (1969), pp.424-438.

[32] William H Greene. Econometric Analysis. Prentice Hall; 5th edition, 2002.

[33] Peter J Brockwell, Richard A Davis. Introduction to Time Series and Forecasting. Springer; 8th Printing. Edition, (2002).

[34] E. Parzen. On Estimation of a Probability Density Function and Mode. Annals of Mathematical Statistics **33** (1962), 1065-1076.

[35] Z.R. Yang, M.B. Platt and H.D Platt. Probabilistic Neural Networks in Bankruptcy Prediction. Journal of Business Research **44** (1999), 67–74.

[36] C. Gaganis and F. Pasiouras, M. Doumpos. Probabilistic neural networks for the identification of qualified audit opinions. Expert Systems with Applications **32** (2007), 114–124.

[37] Cao, L. Support vector machines experts for time series forecasting. Neurocomputing 51 (2003), pp.321–339.

[38] N. Cristianini, J.S. Taylor, An Introduction to Support Vector Machines, Cambridge University Press, Cambridge, MA, 2000.

[39] Jing Sun, DeSheng Wen, GuangRui Li. An efficient guide stars classification algorithm via support vector machines. Second International Conference on Intelligent Computation Technology and Automation, 2009.

[40] Qian Bo, Khaled Rasheed. Stock market prediction with multiple classifiers. Applied Intelligence 26 (2007), pp.25–33.

[41] Walczak S. An empirical analysis of data requirements for financial forecasting with neural networks. Journal of Management Information Systems 17 (2001), pp.203–222.

[42] Tsibouris G, Zeidenberg M. Testing the efficient markets hypothesis with gradient descent algorithms. In: Refenes AP (ed) Neural networks in the capital markets. JohnWiley & Sons, Chichester, England, 1995.

[43] Baestaens D J E, Van Den Bergh W M, Vaudrey H. Market inefficiencies, technical trading and neural networks. In: Dunis C; (ed) forecasting financial markets, financial economics and quantitative analysis. John Wiley & Sons, Chichester, England, 1996, pp.254–260.

[44] T. Odean. Do investors trade too much? American Economic Review **89** (1999) 103-124.

[45] P.A. Andreasson. Explaining the price-volume relationship: The difference between price changes and changing prices. Organizational Behavior and Human Decision Processes **41** (1988), 371-389.

[46] W.F.M. DeBondt. Betting on trends: Intuitive forecasts of financial risk and return. International Journal of Forecasting **9** (1993), 355-371.

[47] H. Hong and J.C. Stein. A unified theory of underreaction, momentum trading, and overreaction in asset markets. Journal of Finance **54** (1999), 2143-2184.

[48] Kim, K. J. Financial time series forecasting using support vector machines. Neurocomputing 55 (2003), pp.307–319.

[49] Tay F E H, Cao L J. Application of support vector machines in financial time series forecasting. Omega 9 (2001), pp.309–317.

[50] Tay F E H, Cao L J. Improved financial time series forecasting by combining support vector machines with self-organizing feature map. Intelligent Data Analysis 5 (2001), pp.339–354.

[51] Sneha Soni, Shailendra Shrivastava. Classification of Indian Stock Market Data Using Machine Learning Algorithms. International Journal on Computer Science and Engineering 9 (2010), pp.2942-2946.

[52] Phichhang Ou, Hengshan Wang. Prediction of Stock Market Index Movement by Ten Data Mining Techniques. Modern Applied Science 3 (2009), pp.