Chapter 11
Stock Market Forecasting by Using Support Vector Machines

K. Liagkouras and K. Metaxiotis

Abstract Support Vector Machine (SVM) is a well established technique within machine learning. Over the last years, Support Vector Machines have been used across a wide range of applications. In this paper, we investigate stock prices forecasting by using a support vector machine. Forecasting of stock prices is one of the most challenging areas of research and practice in finance. As input parameters to the SVM we utilize some well-known stock technical indicators and macroeconomic variables. For evaluating the forecasting ability of SVM, we compare the results obtained by the proposed model with the actual stocks movements for a number of constituents of FTSE-100 in London.

Keywords Support vector machines · Stock price forecasting · Technical indicators · Macroeconomic variables

11.1 Introduction

Forecasting is the process of utilizing historical data for predicting future changes. Although managers and businesses around the world use forecasting to help them take better decisions for future, one particular business domain has been benefited the most from the development of different forecasting methods: the stock market forecasting. Forecasting of stock prices has always been an important and challenging topic in financial engineering, due to the dynamic, nonlinear, complicated and chaotic in nature movement of stock prices.

The emergence of machine learning [10] and artificial intelligence techniques has made it possible to tackle computationally demanding models for stock price

K. Liagkouras (✉) · K. Metaxiotis
Decision Support Systems Laboratory, Department of Informatics, University of Piraeus, 80, Karaoli & Dimitriou Str. 18534 Piraeus, Greece
e-mail: kliagk@unipi.gr

K. Metaxiotis
e-mail: kmetax@unipi.gr

G. A. Tsihrintzis and L. C. Jain (eds.), *Machine Learning Paradigms*, Learning and Analytics in Intelligent Systems 18,
https://doi.org/10.1007/978-3-030-49724-8_11

forecasting [7]. Among the various techniques that have been developed over the last years we distinguish approaches that stem from artificial neural networks (ANNs) [9, 27] and support vector machines (SVMs) [21, 29], because they have gained an increasing interest from academics and practitioners alike. ANN is a computing model whose layered structure resembles the structure of neurons in the human brain [2]. A number of studies examine the efficacy of ANN in stock price forecasting. Below, we provide a concise presentation of recent research finding in the field.

Adhikari and Agrawal [1] propose a combination methodology in which the linear part of a financial dataset is processed through the Random Walk (RW) model and the remaining nonlinear residuals are processed using an ensemble of feedforward ANN (FANN) and Elman ANN (EANN) models. The forecasting ability of the proposed scheme is examined on four real-world financial time series in terms of three popular error statistics. Pan et al. [25] presented a computational approach for predicting the Australian stock market index using multi-layer feed-forward neural networks. According to the authors their research is focused on discovering an optimal neural network or a set of adaptive neural networks for predicting stock market prices.

According to Ou and Wang [23] an important difficulty that is related with stock price forecasting is the inherent high volatility of stock market that results in large regression errors. According to the authors [23] compared to the price prediction, the stock direction prediction is less complex and more accurate. A drawback of ANNs is that the efficiency of predicting unexplored samples decreases rapidly when the neural network model is overfitted to the training data set. Especially this problem is encountered when we are dealing with noisy stock data that may lead ANNs to formulate complex models, which are more prone to the over-fitting problem.

Respectively, a considerable number of studies utilize approaches that are based on SVM for stock price forecasting [34]. Rosillo et al. [26] use support vector machines (SVMs) in order to forecast the weekly change in the S&P 500 index. The authors perform a trading simulation with the assistance of technical trading rules that are commonly used in the analysis of equity markets such as Relative Strength Index, Moving Average Convergence Divergence, and the daily return of the S&P 500. According to the authors the SVM identifies the best situations in which to buy or sell in the market.

Thenmozhi and Chand [29] investigated the forecasting of stock prices using support vector regression for six global markets, the Dow Jones and S&P500 from the USA, the FTSE-100 from the UK, the NSE from India, the SGX from Singapore, the Hang Seng from the Hong Kong and the Shanghai Stock Exchange from China over the period 1999–2011. The study provides evidence that stock markets across the globe are integrated and the information on price transmission across markets, including emerging markets, can induce better returns in day trading.

Gavrishchaka and Banerjee [6] investigate the limitations of the existing models for forecasting of stock market volatility. According to the authors [6] volatility models that are based on the support vector machines (SVMs) are capable to extract information from multiscale and high-dimensional market data. In particular, according to the authors the results for SP500 index suggest that SVM can efficiently work with high-dimensional inputs to account for volatility long-memory and multiscale effects and is often superior to the main-stream volatility models.

Özorhan et al. [24] examine the problem of predicting direction and magnitude of movement of currency pairs in the foreign exchange market. The authors make use of Support Vector Machine (SVM) with a novel approach for input data and trading strategy. In particular, the input data contain technical indicators generated from currency price data (i.e., open, high, low and close prices) and representation of these technical indicators as trend deterministic signals. Finally, the input data are also dynamically adapted to each trading day with genetic algorithm. The experimental results suggest that using trend deterministic technical indicator signals mixed with raw data improves overall performance and dynamically adapting the input data to each trading period results in increased profits.

Gupta et al. [8] presents an integrated approach for portfolio selection in a multi-criteria decision making framework. The authors use Support Vector Machines for classifying financial assets in three pre-defined classes, based on their performance on some key financial criteria. Next, they employ Real-Coded Genetic Algorithm to solve the multi-criteria portfolio selection problem.

According to Cortes and Vapnik [4] the SVMs often achieves better generalization performance and lower risk of overfitting than the ANNs. According to Kim [12] the SVMs outperform the ANNs in predicting the future direction of a stock market and yet reported that the best prediction performance that he could obtain with SVM was 57.8% in the experiment with the Korean composite stock price index 200 (KOSPI 200). Two other independent studies, the first by Huang et al. [11] and the second by Tay and Cao [28] also verify the superiority of SVMs over other approaches when it comes to the stock market direction prediction. Analytically, according to Huang et al. [11] a SVM-based model achieved 75% hit ratio in predicting Nihon Keizai Shimbun Index 225 (NIKKEI 225) movements.

A potential research limitation concerns the testing environment of the aforementioned studies. In particular, for the majority of the examined studies the testing was conducted within the in-sample datasets. Even in the cases that the testing was conducted in an out-of-sample testing environment, the testing was performed on small data sets which were unlikely to represent the full range of market volatility. Another difficulty in stock price forecasting with the SVMs lies in a high-dimensional space of the underlying problem. Indeed, the number of stock markets constituents can range from as few as 30–40 stocks for a small stock market, till several hundreds of stocks for a big stock market, which leads to a high dimensional space [20]. Furthermore, the bigger the examined test instance the bigger the requirements in terms of memory and computation time.

This paper is organized as follows. In Sect. 11.2, we provide an overview of the SVMs and describe how they are integrated in our model. In Sect. 11.3, we identify factors that influence the risk and volatility of stock prices. In Sect. 11.4, we present the proposed model for forecasting stock prices with SVMs. Finally, in Sect. 11.5, we discuss the experimental results and conclude the paper.

11.2 Support Vector Machines

Support Vector Machines were originally developed by Vapnik [30]. In general SVMs are specific learning algorithms characterized by the capacity control of the decision function and the use of kernel functions [31]. In its simplest form a Support Vector Machine is a supervised learning approach for discriminating between two separable groups $\{(\mathbf{x}; y)\}$, where the scalar target variable y is equal to either $+1$ or -1. The vector input variable $\mathbf{x}$ is arbitrary and it is commonly called "separating hyperplane" or otherwise plane in $\mathbf{x}$-space which separates positive and negative cases.

For the linearly separable case, a hyperplane separating the binary decision classes in the three-attribute case is given by the following relationship:

$$y = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3, \tag{11.1}$$

where y is the outcome, x_i are the attribute values, and there are four weights w_i. The weights w_i are determined by the learning algorithm. In Eq. (11.1), the weights w_i are parameters that determine the hyperplane. The maximum margin hyperplane can be represented by the following equation in terms of the support vectors:

$$y = b + \sum a_i y_i \mathbf{x}(i) \cdot \mathbf{x}, \tag{11.2}$$

where y_i is the class value of training example $\mathbf{x}(i)$. The vector $\mathbf{x}$ represents a test example and the vectors $\mathbf{x}(i)$ are the support vectors. In this equation, b and a_i are parameters that determine the hyperplane. Finally, for finding the support vectors and determining the parameters b and a_i a linearly constrained quadratic programming problem is solved.

For the nonlinearly separable case, a high-dimensional version of Eq. (11.2) is given by the following relationship:

$$y = b + \sum a_i y_i K(\mathbf{x}(i), \mathbf{x}), \tag{11.3}$$

The SVM uses a kernel function $K(\mathbf{x}(i), \mathbf{x})$ to transform the inputs into the high-dimensional feature space. There are some different kernels [32] for generating the inner products to construct machines with different types of nonlinear decision surfaces in the input space. Choosing among different kernels the model that minimizes the estimate, one chooses the best model. Figure 11.1 illustrates how a kernel function works. In particular, with the use of a kernel function K, it is possible to compute the separating hyperplane without explicitly carrying out the map into the feature space [33].

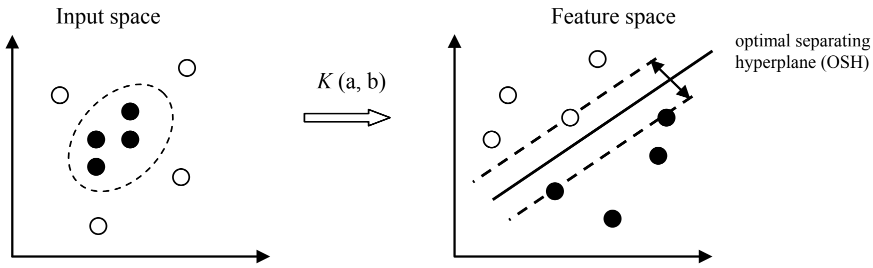

Fig. 11.1 Kernel functions in SVMs

11.3 Determinants of Risk and Volatility in Stock Prices

In stock prices there are two main sources of uncertainty. The first source of risk has to do with the general economic conditions, such as interest rates, exchange rates, inflation rate and the business cycle [14, 15]. None of the above stated macroeconomic factors can be predicted with accuracy and all affect the rate of return of stocks [13]. The second source of uncertainty is firm specific. Analytically, it has to do with the prospects of the firm, the management, the results of the research and development department of the firm, etc. In general, firm specific risk can be defined as the uncertainty that affects a specific firm without noticeable effects on other firms.

Suppose that a risky portfolio consists of only one stock (let say for example *stock* 1). If now we decide to add another stock to our portfolio (let say for example *stock* 2), what will be the effect to the portfolio risk? The answer to this question depends on the relation between *stock* 1 and *stock* 2. If the firm specific risk of the two stocks differs (statistically speaking *stock* 1 and *stock* 2 are independent) then the portfolio risk will be reduced [16]. Practically, the two opposite effects offset each other, which have as a result the stabilization of the portfolio return.

The relation between *stock* 1 and *stock* 2 in statistics is called correlation. Correlation describes how the returns of two assets move relative to each other through time [17]. The most well known way of measuring the correlation is the correlation coefficient (r). The correlation coefficient can range from -1 to 1. Figure 11.2, illustrates two extremes situations: Perfect Positive correlation ($r = 1$) and Perfect Negative correlation ($r = -1$).

Another well-known way to measure the relation between any two stocks is the covariance [18]. The covariance is calculated according to the following formula:

$$\sigma_{\mathrm{X},\Upsilon} = \frac{1}{\mathrm{N}} \sum_{t=1}^{\mathrm{N}} \left(X_t - \bar{X}\right)\left(Y_t - \bar{Y}\right) \tag{11.4}$$

There is a relation between the correlation coefficient that we presented above, and the covariance. This relation is illustrated through the following formula:

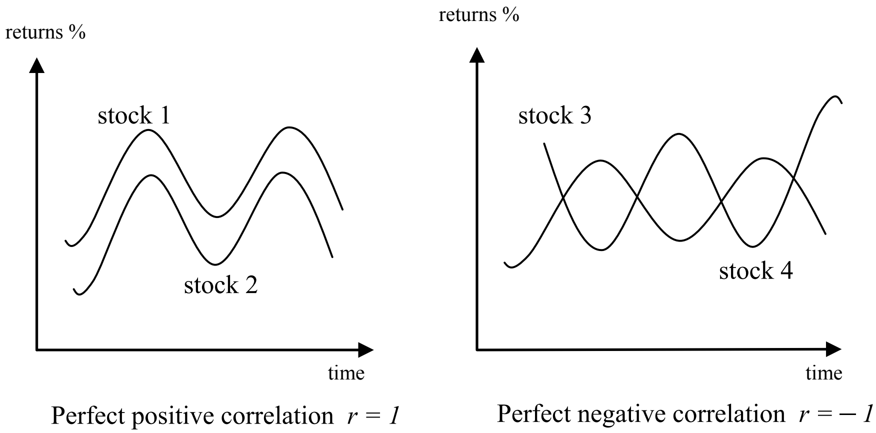

Fig. 11.2 Correlation between stocks

$$r_{X,\Upsilon} = \frac{\sigma_{X,\Upsilon}}{\sigma_X \sigma_Y} \tag{11.5}$$

The correlation coefficient is the same as the covariance, the only difference is that the correlation coefficient has been formulated in such way that it takes values from −1 to 1. Values of the correlation coefficient close to 1 mean that the returns of the two stocks move in the same direction, and values of the correlation coefficient close to −1 mean that the returns of the two stocks move in opposite directions. A correlation coefficient $r_{X,\Upsilon} = 0$ means that the returns of the two stocks are independent. We make the assumption that our portfolio consists 50% of *stock* 1 and 50% of *stock* 2. On the left part of the Fig. 11.2 because the returns of the two stocks are perfectly positively correlated the portfolio return is as volatile as if we owned either *stock* 1 or *stock* 2 alone. On the right part of the Fig. 11.2 the *stock* 3 and *stock* 4 are perfectly negatively correlated. This way the volatility of return of *stock* 3 is cancelled out by the volatility of the return of *stock* 4. In this case, through diversification we achieve risk reduction.

The importance of the correlation coefficient is indicated by the following formula:

$$\sigma_p^2 = w_1^2\sigma_1^2 + w_2^2\sigma_2^2 + 2w_1w_2r_{1,2}^2\sigma_1\sigma_2 \tag{11.6}$$

Equation 11.6 give us the portfolio variance for a portfolio of two stocks 1 and 2. Where w are the weights for each stock and $r_{1,2}$ is the correlation coefficient for the two stocks [19]. The standard deviation of a two—stocks portfolio is given by the formula:

$$\sigma_p = (w_1^2\sigma_1^2 + w_2^2\sigma_2^2 + 2w_1w_2r_{1,2}^2\sigma_1\sigma_2)^{1/2} \tag{11.7}$$

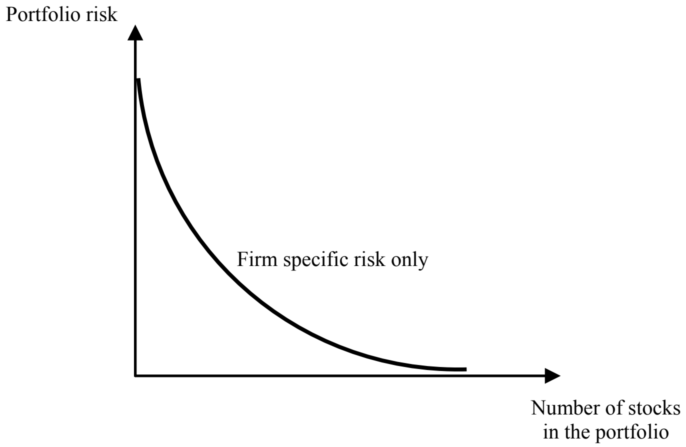

Fig. 11.3 Firm specific risk

From Eq. 11.7, it is obvious that the lower the correlation coefficient $r_{1,2}$ between the stocks, the lower the risk of the portfolio will be.

Obviously, if we continue to add stocks that are negatively correlated into the portfolio the firm-specific risk will continue to reduce. Eventually, however even with a large number of negatively correlated stocks in the portfolio it is not possible to eliminate risk [20]. This happens because all stocks are subject to macroeconomic factors such as inflation rate, interest rates, business cycle, exchange rates, etc. Consequently, no matter how well we manage to diversify the portfolio [22] it is still exposed to the general economic risk.

In Fig. 11.3 we can see that the firm specific risk can be eliminated if we add a large number of negatively correlated stocks into the portfolio. The risk that can be eliminated by diversification except from firm specific risk is called non systematic risk or diversifiable risk.

In Fig. 11.4 we can see that no matter how well diversified is the portfolio there is no way to get rid of the exposure of the portfolio to the macroeconomic factors. These factors related to the general economic risk are called market risk or systematic risk or non diversifiable risk.

11.4 Predictions of Stock Market Movements by Using SVM

11.4.1 Data Processing

The forecasting process requires the following steps: input of selected data, data pre-processing, training and solving support vectors, using test data to calculate

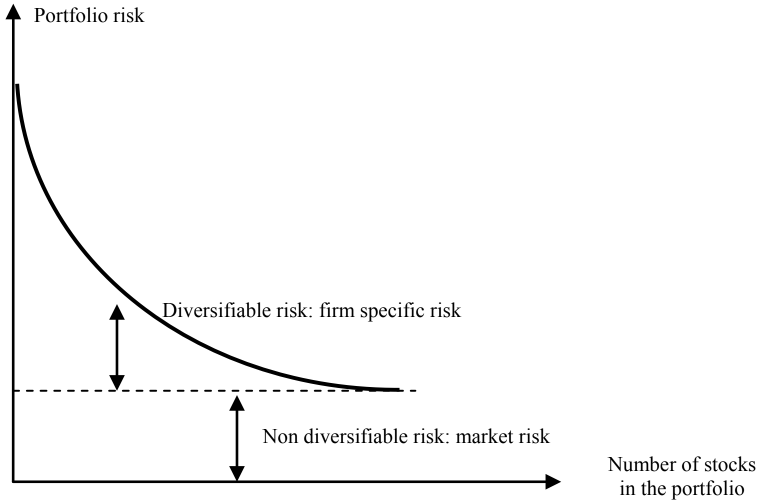

Fig. 11.4 Market risk or non diversifiable risk

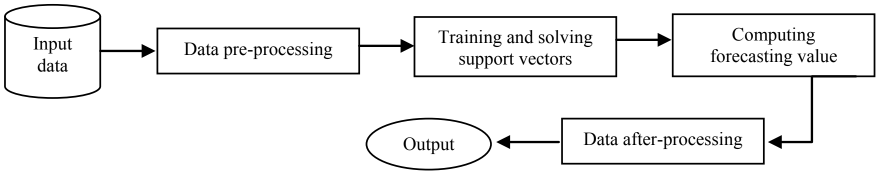

Fig. 11.5 The Forecasting process with SVM

forecasting values, data after-processing, and results analysis. Figure 11.5 illuminates the entire process.

For the purposes of this study, we used a dynamic training pool as proposed by Zhang [35]. Essentially, the training window will always be of the same constant size and 1, 5, 10, 15, 20, 25 and 30 days ahead predictions will be performed by using rolling windows to ensure that the predictions are made by using all the available information at that time, while not incorporating old data. Figure 11.6 illustrates how the dynamic training pool is implemented for the purposes of this study.

For the purposes of the present study we used the daily closing prices of 20 randomly selected constituents of FTSE-100 in London between, Jan. 2, 2018 and Dec. 31, 2018. There are totally 252 data points in this period of time. During the pre-processed phase the data are divided into 2 groups: training group and testing group. The 200 data points belong to training data and the remaining 52 data points are testing data. As shown in Fig. 11.6 we apply a dynamic training pool, which means that the training window will always be of the same constant size (i.e. 200 data points) and one-day-ahead predictions will be performed by using rolling windows.

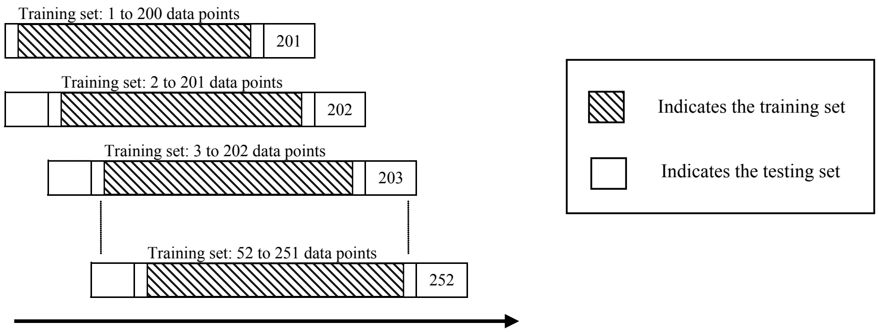

Fig. 11.6 The dynamic training pool for the case of 1-day ahead prediction

In this paper we treat the problem of stock price forecasting as a classification problem. The feature set of a stock's recent price volatility, index volatility, mean absolute error (MAE), along with some macroeconomic variables such as Gross National Product (GNP), interest rate, and inflation rate, are used to predict whether or not the stock's price 1, 5, 10, 15, 20, 25 and 30 days in the future will be higher (+1) or lower (−1) than the current day's price.

11.4.2 The Proposed SVM Model

For the purposes of this study we use the following radial kernel function:

$$K(x_i, x_k) = \exp\left(-\frac{1}{\delta^2}\sum_{j=1}^{n}\left(x_{ij} - x_{kj}\right)^2\right) \tag{11.8}$$

where δ is known as the bandwidth of the kernel function [12]. This function classifies test examples based on the example's Euclidean distance to the training points, and weights closer training points more heavily.

11.4.3 Feature Selection

In this study we use six features to predict stock price direction. Three of these features are coming from the field of macroeconomics and the other three are coming from the field of technical analysis. We opted to include three variables from the field of macroeconomics as it is well-known that macroeconomic variables have an influence on stock prices. For the purposes of this study we use the following macroeconomic variables: (a) Gross National Product (GNP), (b) interest rate, and (c) inflation rate.

Table 11.1 Features used in SVM

Feature name	Description	Formula
σ_s	The stock price volatility is calculated as an average over the past *n* days of percent change in a given stock's price per day	$\frac{\sum_{i=t-n+1}^{t} \frac{C_i - C_{i-1}}{C_{i-1}}}{n}$
σ_i	The index volatility is calculated as an average over the past *n* days of percent change in the index's price per day	$\frac{\sum_{i=t-n+1}^{t} \frac{I_i - I_{i-1}}{I_{i-1}}}{n}$
MAE	The mean absolute error (MAE) measures the average magnitude of the errors in a set of forecasts, without considering their direction	$\frac{1}{n}\sum_{i=1}^{n}\lvert y_i - x_i\rvert$
GNP%	Gross national Product (GNP). The formula for calculating the percent change in GNP rate looks like this	$\frac{GNP_i - GNP_{i-1}}{GNP_{i-1}} \times 100$
Interest rate%	Interest rate is the cost of borrowing money. The formula for calculating the percent change is interest rate is given by the following relationship	$\frac{IR_i - IR_{i-1}}{IR_{i-1}} \times 100$
Inflation rate%	Inflation rate is the percentage increase in general level of prices over a period. The formula for calculating the inflation rate is given by the following relationship	$\frac{CPI_i - CPI_{i-1}}{CPI_{i-1}} \times 100$

[a]Where C_i is the stock's closing price at time *i*. Respectively, I_i is the index's closing price at time *i*. In MAE y_i is the prediction and x_i the realized value. IR_i stands for interest rate at time *i*. Finally, CPI_i stands for Consumer Price Index at time *i*. We use these features to predict the direction of price change 1, 5, 10, 15, 20, 25 and 30 days ahead

According to a study by Al-Qenae et al. [3] it is found that an increase in inflation and interest rates have negative impact on stock prices, whereas an increase in GNP has positive effect on stock prices.

Respectively, we use the following three technical analysis indicators: (a) price volatility, (b) sector volatility and (c) mean absolute error (MAE). More details about the selected features are provided in Table 11.1.

11.5 Results and Conclusions

Figure 11.7 illustrates the mean forecasting accuracy of the proposed model in predicting stock price direction 1, 5, 10, 15, 20, 25 and 30 days ahead.

By observing Fig. 11.7, it is evident that the best mean forecasting accuracy of the proposed model is obtained for predicting stock price direction 1-day ahead. Furthermore, the forecasting accuracy falls drastically when the horizon increases. Indeed, the mean forecasting accuracy of the proposed model is slightly better than simple random guessing when it comes to predicting stock price direction 30-days ahead. This latest finding comes in support of the Efficient Markets Hypothesis [5], which posits that stock prices already reflect all available information and therefore technical analysis cannot be used successfully to forecast future prices. According

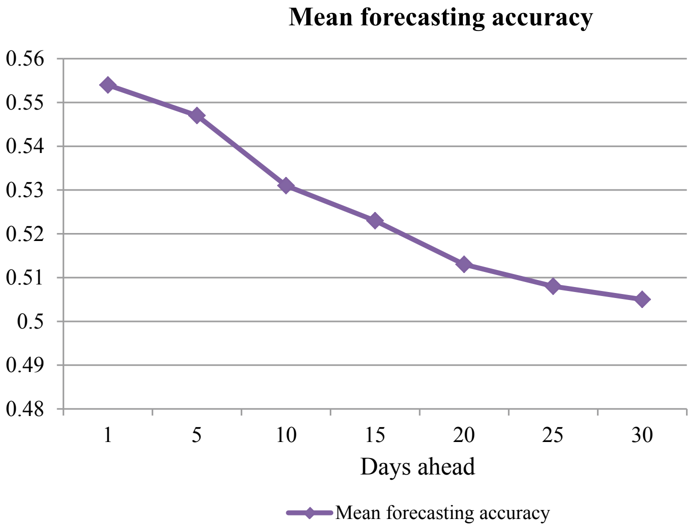

Fig. 11.7 Mean forecasting accuracy of the proposed model in predicting stock price direction

to the Efficient Markets Hypothesis, stock prices will only respond to new information and since new information cannot be predicted in advance, stock price direction cannot be reliably forecasted. Therefore, according to the Efficient Markets Hypothesis, stock prices behave like a random walk. To conclude the proposed model can be helpful in forecasting stock price direction 1–5-days ahead. For longer horizons, the forecasting accuracy of the proposed model falls drastically and it is slightly better than simple random guessing.

References

1. R. Adhikari, R.K. Agrawal, A combination of artificial neural network and random walk models for financial time series forecasting. Neural Comput. Appl. **24**(6), 1441–1449 (2014)
2. R. Aghababaeyan, N. TamannaSiddiqui, Forecasting the tehran stock market by artificial neural network. Int. J. Adv. Comput. Sci. Appl. Spec. Issue Artif. Intell. (2011)
3. R. Al-Qenae, C. Li, B. Wearing, The Information co earnings on stock prices: the Kuwait Stock Exchange. Multinatl. Financ. J. **6**(3 & 4), 197–221 (2002)
4. C. Cortes, V. Vapnik, Support-vector networks. Mach. Learn. **20**(3), 273–297 (1995)
5. E.F. Fama, Efficient capital markets: a review of theory and empirical work. J. Financ 25(2). in *Papers and Proceedings of the Twenty-Eighth Annual Meeting of the American Finance Association*, New York, N.Y, 28–30 December, 1969 (May, 1970), pp. 383–417
6. V.V. Gavrishchaka, S. Banerjee, Support vector machine as an efficient framework for stock market volatility forecasting, CMS **3**:147–160 (2006)
7. M. Gilli, E. Schumann, Heuristic optimisation in financial modelling. Ann. Oper. Res. **193**(1), 129–158 (2012)
8. P. Gupta, M.K. Mehlawat, G. Mittal, Asset portfolio optimization using support vector machines and real-coded genetic algorithm. J. Glob. Optim. **53**, 297–315

9. E. Guresen, G. Kayakutlu, T.U. Daim, Using artificial neural network models in stock market index prediction. Expert Syst. Appl. **38**(8), 10389–10397 (2011)
10. C.J. Huang, P.W. Chen, W.T. Pan, Using multi-stage data mining technique to build forecast model for Taiwan Stocks. Neural Comput. Appl. **21**(8), 2057–2063 (2011)
11. W. Huang, Y. Nakamori, S.-Y. Wang, Forecasting stock market movement direction with support vector machine. Comput. Oper. Res. **32**(10), 2513–2522 (2005)
12. K.-J. Kim, Financial time series forecasting using support vector machines. Neurocomputing **55**(1), 307–319 (2003)
13. K. Liagkouras, K. Metaxiotis, A new probe guided mutation operator and its application for solving the cardinality constrained portfolio optimization problem. Expert Syst. Appl. **41**(14), 6274–6290 (2014). Elsevier
14. K. Liagkouras, K. Metaxiotis, Efficient portfolio construction with the use of multiobjective evolutionary algorithms: Best practices and performance metrics. Int. J. Inf. Technol. Decis. Making **14**(03), 535–564 (2015). World Scientific
15. K. Liagkouras, K. Metaxiotis, Examining the effect of different configuration issues of the multiobjective evolutionary algorithms on the efficient frontier formulation for the constrained portfolio optimization problem. J. Oper. Res. Soc. **69**(3), 416–438 (2018)
16. K. Liagkouras, K. Metaxiotis, Multi-period mean–variance fuzzy portfolio optimization model with transaction costs. Eng. Appl. Artif. Intell. **67**(2018), 260–269 (2018)
17. K. Liagkouras, K. Metaxiotis, Handling the complexities of the multi-constrained portfolio optimization problem with the support of a novel MOEA. J. Oper. Res. Soc. **69**(10), 1609–1627 (2018)
18. K. Liagkouras, K. Metaxiotis, A new efficiently encoded multiobjective algorithm for the solution of the cardinality constrained portfolio optimization problem. Ann. Oper. Res. **267**(1–2), 281–319 (2018)
19. K. Liagkouras, K. Metaxiotis, Improving the performance of evolutionary algorithms: a new approach utilizing information from the evolutionary process and its application to the fuzzy portfolio optimization problem. Ann. Oper. Res. **272**(1–2), 119–137 (2019)
20. K. Liagkouras, A new three-dimensional encoding multiobjective evolutionary algorithm with application to the portfolio optimization problem. Knowl.-Based Syst. **163**(2019), 186–203 (2019)
21. C.J. Lu, T.S. Lee, C.C. Chiu et al., Financial time series forecasting using independent component analysis and support vector regression. Decis. Support Syst. **47**(2), 115–125 (2009)
22. K. Metaxiotis, K Liagkouras, Multiobjective evolutionary algorithms for portfolio management: a comprehensive literature review. Expert Syst. Appl. **39**(14), 11685–11698 (2012). Elsevier
23. P. Ou, H. Wang, Prediction of stock market index movement by ten data mining techniques. Mod. Appl. Sci. **3**(12), 28 (2009)
24. M.O. Özorhan, I.H. Toroslu, O.T. Sehitoglu, A strength-biased prediction model for forecasting exchange rates using support vector machines and genetic algorithms. Soft. Comput. **21**, 6653–6671 (2017)
25. H. Pan, C. Tilakaratne, J. Yearwood, Predicting Australian Stock market index using neural networks exploiting dynamical swings and intermarket influences. J. Res. Pract. Inf. Technol. **37**(1), 43–55 (2005)
26. R. Rosillo, J. Giner, D. Fuente, (2013) The effectiveness of the combined use of VIX and Support Vector Machines on the prediction of S&P 500. Neural Comput. Applic. **25**, 321–332 (2013)
27. E.W. Saad, D.V. Prokhorov, D.C. Wunsch et al., Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks. IEEE Trans. Neural Netw. **9**(6), 1456–1470 (1998)
28. F.E.H. Tay, L. Cao, Application of support vector machines in financial time series forecasting. Omega **29**(4), 309–317 (2001)
29. M. Thenmozhi, G.S. Chand, Forecasting stock returns based on information transmission across global markets using support vector machines. Neural Comput. Applic. **27**, 805–824 (2016)

30. V.N. Vapnik, *Statistical Learning Theory* (Wiley, New York, 1998)
31. V.N. Vapnik, An overview of statistical learning theory. IEEE Trans. Neural Netw. **10**, 988–999 (1999)
32. L. Wang, J. Zhu, Financial market forecasting using a two-step kernel learning method for the support vector regression. Ann. Oper. Res. **174**(1), 103–120 (2010)
33. C. Wong, M. Versace, CARTMAP: a neural network method for automated feature selection in financial time series forecasting. Neural Comput. Appl. **21**(5), 969–977 (2012)
34. F.C. Yuan, Parameters optimization using genetic algorithms in support vector regression for sales volume forecasting. Appl. Math. **3**(1), 1480–1486 (2012)
35. Y. Zhang, Prediction of Financial Time Series with Hidden Markov Models, Simon Fraser University, 2004