Stock Market Trend Prediction Based on the LS-SVM Model Update Algorithm

Ivana Marković[1], Miloš Stojanović[2], Miloš Božić[3], and Jelena Stanković[1]

[1] Faculty of Economics, Trg Kralja Aleksandra Ujedinitelja 11, Niš, Serbia
{ivana.markovic,jelenas}@eknfak.ni.ac.rs
[2] College of Applied Technical Sciences, Aleksandra Medvedeva 20, Niš, Serbia
milosstojanovic10380@yahoo.com
[3] Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, Niš, Serbia
milos1bozic@gmail.com

Abstract. The paper proposes a trend prediction model based on an incremental training set update scheme for the BELEX15 stock market index using the Least Squares Support Vector Machines (LS-SVMs) for classification. The basic idea of this updating approach is to add the most recent data to the training set, as become available. In this way, information from new data is taken into account in model training. The test results indicate that the suggested model is suitable for short-term market trend prediction and that prediction accuracy significantly increases after the training set has been updated with new information.

Keywords: Stock market trend prediction, Least Squares Support Vector Machines (LS-SVMs), Model update.

1 Introduction

The stock market index, as a hypothetical portfolio of selected stocks, is commonly used to measure overall market or particular sector performance [1]. Recent studies [2], indicated that trading strategies guided by predictions regarding the direction of change in the prices could be more effective and could generate a greater yield in comparison to the precise predictions of the level of financial instrument prices. As a result, the world's largest financial markets are now turning to trading in stock market indices more and more often. Consequently, predicting the direction of the movement of the price of financial instruments has now become a current area of academic research.

In numerous studies, the algorithms of machine learning proved to be quite effective in predicting the direction of movement of the value of stock indices and contributed to the increase in yield and reduction in the risk involved in trading. Some of the more frequently adopted methods include the following: Artificial Neural Networks (ANNs) [3], linear and multi-linear regression (LR, MLR) [4], genetic algorithms (GAs) [4], and Support Vector Machines (SVMs) [5]. According to [1], the most widely used methods for stock market trend prediction include approaches based on

A. Madevska Bogdanova and D. Gjorgjevikj, *ICT Innovations 2014*,
Advances in Intelligent Systems and Computing 311, DOI: 10.1007/978-3-319-09879-1_11

SVMs. In [6], it was further indicated that in most cases the LS-SVMs, and SVMs outperform other machine learning methods, since in theory they do not require any previous a priori assumptions regarding data properties. Moreover, they guarantee an efficient global optimal solution.

As a result of the fact that the financial market is a complex, evolving and dynamic system whose behavior is pronouncedly non-linear, non-stationary and stochastic [5], mining the stock market tendency is a challenging task. Evolving and non-stationary as characteristics imply that the distribution of financial time series changes over a period of time. Thus, to obtain systematically good predictions under such circumstances, it may be necessary to update the underlying models.

The existing stock market trend prediction systems usually focus on several aspects: feature selection, the selection of prediction model and feature evaluation. The problem of model updating, however, has so far not been studied in sufficient detail, particularly in the field of stock market trend prediction. Model updating strategies that correspond to time-evolving systems, including the stock rate index, can usually be undertaken from two perspectives: as incremental learning systems [7, 8, 9, 10], where the respective models are updated online as new instances become available during the training phase, and as batch learning systems [11, 12], where a collection of training instances can be updated prior to model re-training. In this paper, the second model update approach is considered, where the new data over a given time period are added to the initial training set and the respective model is then re-trained. In [11] and [12] similar concepts are presented, but for a different subject matter. To our knowledge, the proposed approach of model updating has so far not been used for stock market trend index prediction.

In this paper LS-SVMs will be used to create a prediction model, but any classification technique is suitable for the application of the proposed model updating algorithm. The problem of stock index trend prediction is modeled as a binary classification problem. Experimental results, benchmarking the standard and updated model, show that prediction accuracy can be increased after updating the initial training set with new available data.

The proposed algorithm offers a systematic approach for model updating based on new instances as they become available.

The rest of this paper is organized as follows: Section 2 presents the basic theory of LS-SVMs for classification. Section 3 presents the used updating methodology. Section 4 gives data set analysis and presents the experimental results. Finally, Section 5 provides the conclusions.

2 Least Squares Support Vector Machines for Classification

The Least Squares Support Vector Machines, proposed by Suykens in [13], includes a set of linear equations which are solved instead of a Quadratic Programming (QP) for classical SVMs. Therefore, LS-SVMs are more time-efficient than standard SVMs, but with lack of sparseness.

Let's study a training group of a total of N examples $T=\{\boldsymbol{x}_i, y_i\}_{i=1}^{N}$. In the learning phase, the model is formed based on the known training data ($\boldsymbol{x}_1$, y_1), ($\boldsymbol{x}_2$, y_2), …, ($\boldsymbol{x}_N$, y_N), where $\boldsymbol{x}_i$ are the input vectors, and y_i are the labels of binary classes that were assigned to them. Each input vector consists of numeric features, while $y_i \in \{-1, +1\}$.

According to [13] LS-SVMs for binary classification were defined as follows:

$$\min_{w,b,e} J_{LS}(w,b,e) = \tfrac{1}{2} w^T w + \gamma \tfrac{1}{2} \sum_{k=1}^{N} e_k^2 \tag{1}$$

with the equality conditions:

$$y_k[w^T \varphi(x_k) + b] = 1 - e_k, \quad k = 1,..., N \tag{2}$$

where φ is a non-linear function that maps input vectors in some higher dimensional feature space. The weight vector of the hyper plane is marked by a $\boldsymbol{w}$, while b is the scalar shift, that is, weight threshold. The variable e_k represents the allowed errors of classification, while the parameter γ controls the process, that is, the relationship between the complexity of the model and the accepted error of classification.

After solving the optimization problem defined by (1) and (2), a solution can be found in [13], the function of the separation of LS-SVM classifications is defined as:

$$y(x) = sign\left[\sum_{k=1}^{N} \alpha_k y_k K(x, x_k) + b\right] \tag{3}$$

where α_k represent the support vectors (Lagrange multipliers), and b is a constant. $K(x, x_k)$ represents the Kernel function, which is defined by the dot product between $\boldsymbol{x}$ and $\boldsymbol{x}_k$.

As presented in [14], on the basis of twenty different groups of data, the best general prediction rate was given by LS-SVM classifiers with a RBF (Radial basis function) kernel. In addition, according to [15] in cases where the number of examples for classification is much greater than the number of features, the use of the RBF kernel is also recommended. Accordingly, the RBF kernel was used, defined by:

$$K(x, x_k) = e^{\frac{-\|x - x_k\|^2}{\sigma^2}} \tag{4}$$

When training the LS-SVM model it is necessary to determine the value of parameter γ, as well as the parameters of the selected kernel, in this case the width σ. One of the ways to determine these parameters is the k fold Cross - Validation procedure in combination with a Grid - Search, described in more detail in next section.

3 The Model Update Algorithm

Most machine learning based models implemented for stock market trend prediction use a fixed-size training set in a learning phase. In other words, forecasts for several days, weeks, months or a year are made by a prediction model trained with the same training set known before model construction.

However, in stock market trend prediction that includes the constant input of new data, the generalization capability of the predictor is expected to improve following the completion of the learning process, i.e. with new available data. Thus, a dynamic update of the model is crucial for maintaining and improving the performance of the prediction model.

In the proposed model updating algorithm, an initial prediction model is re-trained on the basis of new incoming data: every P new example, when it becomes available, is added to the initial training set, and the model is then re-trained. The algorithm requires a parameter P to specify the number of training instances that are added to the initial training set, i.e. the time horizon of model re-training. The adaptation of the model to the time-evolving environment can be determined by the changes in the value of P, that is, the current scope of the model.

The optimal value of parameter P is dependent on the observed data time series and mainly based on heuristic methods. In general, large numbers of training instances that are added to the initial training set are preferred for stationary processes, while smaller numbers of instances are preferred in non-stationary environments.

The model update algorithm can be seen in Figure 1.

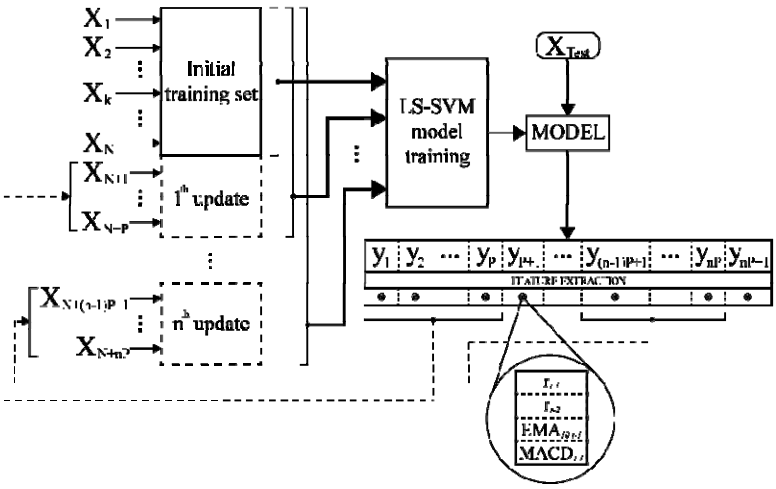

Fig. 1. Model update algorithm

The first step in the proposed algorithm is model-training based on the initial training set. The optimal (γ, σ) pair is determined on the basis of the initial training set T=($\boldsymbol{x}$, y) using a grid-search with k-fold cross validations, as mentioned in section 2. The training set is randomly subdivided into k disjoint subsets of approximately equal

size and the LS-SVM model is built k times with the current pair (γ, σ). Each time, one of the k subsets is used as the test set and the other $k - 1$ subsets are combined to form a training set. After k iterations, the average hit rate is calculated for the current pair (γ, σ). The entire process is repeated with an update of the parameters (γ, σ) until the given stopping criterion is reached, in this case the maximization of the hit rate, although other criterions can be used, depending on the nature of the classification problem. The parameters (γ, σ) are updated exponentially in the given range using predefined equidistant steps, according to the grid-search procedure. After obtaining the optimal (γ, σ) combination, the LS-SVM final forecasting model is formed according to (3) and (4).

The model is then employed for the prediction of the stock market trend for one step ahead.

The first step is the selection of a test instance $\boldsymbol{x}_t$ from the test set t=($\boldsymbol{x}_t$, y_t). It should be noted that at the moment of applying the model on the current test vector $\boldsymbol{x}_t$, the value of the associated target value y_t is unknown.

After that, based on the value of the parameter P, it is necessary to update the initial training set T=($\boldsymbol{x}$, y) for the next prediction step with P past instances from t=($\boldsymbol{x}_t$, y_t), which are known at the moment. Before the selection of the next $\boldsymbol{x}_t$ for the next step, the initial training set is updated by adding stock data from the previous P steps, which are known at that moment, and the model is re-trained. The update and re-training are performed in every *P-th* iteration of the test loop until the given number of instances in the test set is reached.

The training set in the proposed algorithm includes data which were observed after the model was initially constructed, as well as the initial data. The updating model algorithm was designed to make full use of the information, as soon as it becomes available.

4 The Experiment and Results

4.1 The Data Used in the Experiment

The value of the Belex15 index determines the price of the most liquid stocks traded on the regulated market of the Belgrade Stock Exchange. The series consists of six sizes which are determined for each day: the closing price, the change in the value of the index in relation to the previous trading day, in percentages, the opening price, highest price, lowest price and the trading volume.

The available data were divided into two groups. The first group consisted of 1811 records required for the training model, from October 26, 2005 to December 31, 2012. For the second group of data, data from January 3, 2013 to December 31, 2013 were used. A total of 253 days of trading were selected that represent whole trading year. The data from the first group were assigned to the training set, while the data from the second were used for the test set.

4.2 Feature Selection and Model Formation

For stock market trend prediction, features are usually selected from a group of technical or fundamental indicators. In this study, the technical indicators as input features were used to predict the stock market trend. In our previous study [16], we established the basis for the formation of a standard LS-SVM model for predicting the trend of the Belex15 index. There, the process of features selection was studied in more detail, along with the characteristics of the time series. The conducted analyses selected two lagged values of the logarithmic return that were statistically determined based on the values of the auto-correlational coefficients. The Exponential Moving Average (EMA), as the moving average of the closing price calculated using a smoothing factor to place a higher weight on recent closing prices, was then also selected based on its features. This indicator can be used to calculate the values backwards to an almost infinite number of steps (for example, EMA5, EMA100 or EMA200), which is an important characteristics of modeling time series. The EMA feature is consequently adjusted with respect to the time horizon, thus the selected period for calculating the EMA transformation consisted of the previous 10 days. The Moving Average Convergence-Divergence (MACD), as the indicator that measures the strength and direction of the trend and momentum, was added to the current model as it was determined in [17] to be effective in optimizing the investment strategies on emerging markets.

The detailed mathematical formulations for the applied transformations and indicators are given in Table 1.

Table 1. Input features

Features	Formula
Closing price	CP_t, t= 1,2, ... N
Logarithmic return	$r_t = \log CP_t - \log CP_{t-1}$
EMA_N	$EMA_N = r_t * k + EMA_{t-1} * (1-k)$; $k = 2/(N+1)$
MACD	$MACD = EMA_{12} - EMA_{26}$
r_{t-1}	$r_{t-1} = \log CP_{t-1} - \log CP_{t-2}$
r_{t-2}	$r_{t-2} = \log CP_{t-2} - \log CP_{t-3}$

The abovementioned transformations contribute to the stationary nature of the series, which additionally increases the effectiveness of the machine learning algorithm.

In the proposed model, the variable to be predicted is the future trend of the stock market. The feature which serves as a label for the class is a categorical variable used to indicate the movement direction of the logarithmic return on the Belex15 index over time t. If the logarithmic return over time t is larger than zero, the indicator is 1. Otherwise, the indicator is −1. Figure 2 shows the trend fluctuations. It can be determined that in reality the market price trend does not constantly follow a straight line; it is volatile, and the line fluctuates up and down repeatedly, rendering it challenging for prediction.

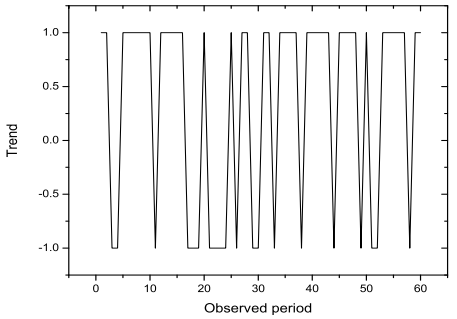

Fig. 2. Trend fluctuations

Based on the previous analysis, the following prediction model was created

$$y_t = LS-SVM(r_{t-1}, r_{t-2}, EMA_{10t-1}, MACD_{t-1}) \tag{5}$$

In order to form the LS-SVM models, LS-SVMlab [18] was used.

4.3 Experimental Results

As a general measure for the evaluation of the prediction effect, the Hit Ratio (HR) was used, which was calculated based on the number of properly classified results within the test group:

$$HR = \frac{1}{m}\sum_{i=1}^{m} PO_i \tag{6}$$

where PO_i is the prediction output of the *i-th* trading day. PO_i equals 1 if is actual value, for the *i-th* training day, otherwise, PO_i equals 0, and *m* is the number of data in the test group [19].

Table 2 shows a comparison of the hit rates obtained using the model updating algorithm (MU-LS-SVMs) with different step sizes $P = \{1, 2, 5, 10, 20\}$ with the Random walk (RW) benchmark model and the LS-SVM model without update. The RW uses the current value to predict the future value, assuming that the latter in the following period (y_{t+1}) will be equal to the current value (y_t). Step sizes are defined based on the definition of the short time stock market periods [20] and previous analyzes of the available time series [16].

The influence of the model update algorithm is clearly positive, since all updated models outperformed the model without update. It can be assumed that both MU LS-SVM_1 and MU LS-SVM_2 will outperform other models because of the observed strong autocorrelation factors in a time series for lag one and two. In addition, it can be seen that from other group of models, the best accuracy was achieved using the MU LS-SVM_{10} model, which further supports the validity of the selected parameters of the EMA features.

Table 2. Prediction accurancy of different prediction models

Model	Hit rate
RW	0.5000
LS-SVM	0.5396
MU-LS-SVM$_1$	0.5555
MU-LS-SVM$_2$	0.5555
MU-LS-SVM$_5$	0.5436
MU-LS-SVM$_{10}$	0.5476
MU-LS-SVM$_{20}$	0.5436

Furthermore, the comparison of the MU LS-SVM$_1$, RW and LS-SVM model on a temporal sequence basis which corresponds to the real frameworks of trading on the Belgrade stock exchange was studied, including the weekly, biweekly, monthly, bi-monthly, and quarterly work regime. This went on until entire trading year. The results are shown in Table 3.

Table 3. The models comparison results on the predefined time-sequence

Time Sequences	RW	LS-SVM	MU LS-SVM$_1$
0-5	0.6000	0.6000	0.6000
0-10	0.8000	0.8000	0.8000
0-20	0.7000	0.7000	0.7000
0-40	0.6000	0.6750	0.6500
0-60	0.6000	0.6167	0.6333
0-80	0.6125	0.6000	0.6125
0-100	0.6100	0.6300	0.6400
0-120	0.6083	0.6500	0.6583
0-140	0.5714	0.6286	0.6357
0-160	0.5438	0.5875	0.5938
0-180	0.5389	0.5889	0.5944
0-200	0.5200	0.5500	0.5550
0-220	0.5113	0.5520	0.5611
0-240	0.5125	0.5542	0.5625
0-252	0.5000	0.5397	0.5556

It can be noted that in the approximated first trading month, the rate of the hits is identical for all presented models. This can be explained by insufficient additional new training data and it is in favor of the previously noted strong correlation in the available data series. The longer the time period, the more dominant the prediction based on the proposed model update algorithm.

This algorithm extends computational time. The time needed to obtain the predictions increases for all the models that implement the update approach, compared to

the model trained with an initial training set (by approximately 150 seconds compared to 100 seconds). Nevertheless, an increase in computational time is compensated with an increase in the quality of the prediction results.

The results are obtained for one-day-ahead predictions using data over an extended period of time, one trading year, and exceed most of the time horizons presented in [5], [19], [21], [22], but are still in their mid-range. The results are reliable, based on all the currently available information, representing all the forms of model behavior.

5 Conclusion

A practical approach to building a dynamic model for the stock market trend prediction is proposed. Although the complexity of the calculations in the proposed algorithm is increased when compared to training only one forecasting model, it brings significant improvements in terms of stock market prediction accuracy. Every increase in precision is considered an exceptional contribution as it leads to an increase in the return and the decrease in the risk involved in trading.

As far as further research is concerned, first, in the proposed approach, prior information was not excluded. Since short periods of time were observed in the time series analyzed in this paper, the issue was not dealt with separately. In the case of the longer periods of time, the prediction model should not include all the available data. Thus, past information should be removed using a new methodology designed for that purpose.

Finally, most studies in this field deal with the prediction of market indices and the price of financial instruments on developed markets. It is important to emphasize that the prediction rate obtained in this study belongs to the stock index of emerging market of the Republic of Serbia, and that it gave competitive results.

References

1. Wang, Y., Choi, I.C.: Market Index and Stock Price Direction Prediction using Machine Learning Techniques: An empirical study on the KOSPI and HIS. ScienceDirect, 1–13 (2013)
2. Kumar, M., Thenmozhi, M.:Forecasting stock index movement: a comparison of support vector machines and random forest. In: Indian Institute of Capital Markets 9th Capital Markets Conference Paper (2006), SSRN: http://ssrn.com/abstract=876544, http://dx.doi.org/10.2139/ssrn.876544
3. Kara, Y., Boyacioglu, M., Baykan, Ö.K.: Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange. Expert Systems with Applications 38, 5311–5319 (2011)
4. Atsalakis, G.S., Valavanis, K.P.: Surveying stock market forecasting techniques – Part II: Soft computing methods. Expert Systems with Applications 36, 5932–5941 (2009)
5. Huang, W., Nakamori, Y., Wang, S.: Forecasting stock market movement direction with support vector machine. Computers & Operations Research 32, 2513–2522 (2005)
6. Phichhang, O., Wang, H.: Prediction of Stock Market Index Movement by Ten Data Mining Techniques. Modern Applied Science 3, 28–42 (2009)

7. Jiang, J., Song, C., Zhao, H., Wu, C., Liang, Y.: Adaptive and Iterative Least Squares Support Vector Regression Based on Quadratic Renyi Entropy. In: Granular Computing, GrC 2008, pp. 340–345. IEEE Press, New York (2008)
8. Read, J., Bifet, A., Pfahringer, B., Holmes, G.: Batch-Incremental versus Instance-Incremental Learning in Dynamic and Evolving Data. In: Hollmén, J., Klawonn, F., Tucker, A. (eds.) IDA 2012. LNCS, vol. 7619, pp. 313–323. Springer, Heidelberg (2012)
9. Doomretni, C., Giunnoepulos, D.: Incremental Support Vector Machine Construction. In: Data Mining, ICDM 2001, pp. 589–593. IEEE Press, New York (2001)
10. Laskov, P., Gehl, C., Kruger, S., Muller, K.: Incremental Support Vector Learning, Analysis, Implementation and Applications. Journal of Machine Learning Research 7, 1909–1936 (2006)
11. Stojanović, M., Božić, M., Stajić, Z., Milošević, M.: LS-SVM model for electrical load prediction based on incremental training set update. Przegląd Elektrotechniczn 4, 195–199 (2013)
12. Guajardo, J.A., Weber, R., Miranda, J.: A model updating strategy for predicting time series with seasonal patterns. Applied Soft Computing 10, 276–283 (2010)
13. Suykens, J., Vandewalle, J.: Least Squares Support Vector Machines. Neural Processing Letters 9, 293–300 (1999)
14. Gestel, T.V., Suykens, A.K., Baesens, B., Viaene, S., Vanthienen, J., Dedene, G., Moor, B.D., Vandewalle, J.: Benchmarking Least Squares Support Vector Machine Classifiers. Machine Learning 54, 5–32 (2004)
15. Božić, M., Stajić, Z., Stojanović, M.: Short-term load forecasting using least square support vector machines. In: Infoteh Jahorina, pp. 326–329 (2010)
16. Marković, I., Stanković, J., Stojanović, M., Božić, M.: Stock exchange trend prediction of Belex15 index with LS-SVM classifier. In: XIII International Symposium Infoteh-Jahorina, pp. 739–742 (2014)
17. Eric, D., Andjelic, G., Redzepagic, S.: Application of MACD and RVI indicators as functions of investment strategy optimization on the financial market. Proceedings of the Faculty of Economics of Rijeka 27(1), 171–196 (2009)
18. Brabanter, K.D., Karsmakers, P., Ojeda, F., Alzate, C., Brabanter, J.D., Pelckmans, M.D.K., Vandewalle, B.J., Suykens, J.A.K.: LS-SVMlab Toolbox User's Guide. Technical report, ESAT-SISTA (2011)
19. Yuling, L., Guo, H., Hu, J.: An SVM-based Approach for Stock Market Trend Prediction. In: Neural Networks (IJCNN), pp. 1–7. IEEE Press, New York (2013)
20. Bradić-Martinović, A.: Stock market prediction using technical analysis. Economic Anal. 170, 15–145 (2006)
21. Lahmiri, S.: A Comparison of PNN and SVM for Stock Market Trend Prediction using Economic and Technical Information. International Journal of Computer Applications 29, 24–30 (2011)
22. Ni, L.-P., Ni, Z.-W., Gao, Y.-Z.: Stock trend prediction based on fractal feature selection and support vector machine. Expert Systems with Applications 38, 5569–5576 (2011)