STOCK MARKET INDEX PREDICTI ON USING MACHINE LEARNING TECHNIQUES. A CASE STUDY OF BUCHAREST STOCK EXCHANGE

[i]HAKOB GRIGORYAN

Bucharest University of Economic Studies
E-mail: Grigoryanhakob90@yahoo.com

Abstract—Financial markets are complex systems and their behavior is characterized as non-stationary and volatile. In the presented paper, the potential of the machine learning techniques, in particular, adaptive network-based fuzzy inference system (ANFIS) model and Support Vector Machines (SVM) are investigated in forecasting task of stock market data. The performance of proposed models are evaluated with test data of Bucharest Stock Exchange (BVB) and results obtained are compared in terms of well known evaluation metrics. The comparative analysis leads to the conclusion that both models can be successfully applied to forecast the high-frequency stock data while SVM outperforms the ANFIS model in prediction accuracy.

Index Terms— ANFIS, Fuzzy neural network, Support Vector Machines, Time series prediction, stock market

I. INTRODUCTION

In the last few decades, financial data forecasting has become an attractive topic of research for many scholars in time series domain. For large companies and investors which conduct huge amount of transactions in different markets, being able to accurately predict market price tendencies can result in high profits with small investment risks. Due to the complex and non-linear nature of high frequency financial data, the modeling of prediction systems is a challenging task [1]. Although many prediction models have been presented to forecast the behavior of different financial systems based on various econometric methods, most of them failed to catch non-linear patterns in financial data and have shown less significant results [2,3].

In recent years, machine learning techniques have become popular methods in time series analysis because of their ability to learn and detect complex relationships in non-stationary data and successfully perform classification and prediction tasks. Artificial neural networks (ANNs) are the most widely used machine learning algorithms in different fields of time series domain classified as data driven techniques with the ability to approximate any function regardless of its linearity [4]. However, ANNs also suffer from several shortcomings such as model over-fitting and unstable performance on noisy data [5].

Recent machine learning algorithms aim to overcome the black-box nature of ANNs. The adaptive network-based fuzzy inference system (ANFIS) introduced by Jang [6] is a Takagi-Sugeno type inference model in which a fuzzy system is implemented in the framework of adaptive networks. ANFIS as a model for stock price prediction has been reported by many researchers and showed promising results compared to traditional linear methods and other neural networks [7,8]. Support Vector Machines (SVM) is another well known family of learning algorithms established on the theory of the structural risk minimization principle proposed by Vapnik [9]. SVMs have proved to be accurate techniques able to achieve high generalization performance on noisy data and resistant to the over-fitting problem [10]. An extension of SVM for regression task has been successfully applied to financial forecasting problem and showed high degree of accuracy in prediction of stock prices[11-14].

Although ANFIS and SVM techniques have been examined in several studies of financial data analysis, still there is no clear conclusion in the literature which model performs better in prediction task of noisy stock data. The objectives of the present paper are twofold: 1) to investigate the predictability of stock market index of Bucharest Stock Exchange (BVB) with ANFIS and SVM techniques and 2) to conduct a comparative analysis in order to draw a conclusion about superiority of each method based on different evaluation metrics.

The remainder of the paper is arranged as follows. Sections 2 and 3 present general overview of the adaptive network-based fuzzy inference system (ANFIS) model and Support Vector Machines (SVM). In Section 4, research methodology is described. Finally, experimental results and conclusions are summarized in Section 5.

II. ADAPTIVE NETWORK-BASED FUZZY INFERENCE SYSTEM (ANFIS)

The adaptive network-based fuzzy inference system (ANFIS) is a special architecture of neuro-fuzzy systems which combines both the reasoning capabilities of fuzzy logic systems and learning capabilities of neural network. The main idea of

ANFIS model is to correctly associate the input values with the target values by applying a combined learning algorithm and using input-output data sets. As an adaptive network, ANFIS is a multilayer feed-forward network composed of several nodes connected by directed links and described by the node function. The architecture of simple ANFIS model which uses two variables as inputs and one output is depicted in fig 1. Each node in the network performs a particular function on its incoming signals to generate a single node output. In ANFIS model, two type of nodes are used, namely, adaptive and fixed based on parameter sets. In fig 1, adaptive nodes are denoted by squares, and fixed nodes are denoted by circles.

The ANFIS architecture can be described by assuming that the fuzzy inference system contains two fuzzy if-then rules of Takagi and Sugeno's defined as:

Rule 1: If x is $A_1$and y is B_1; $f_1 = p_1 x + q_1 y + r_1$,

Rule 2: If x is $A_2$and y is B_2; $f_2 = p_2 x + q_2 y + r_2$,

where $A_i$and B_i are appropriate fuzzy sets, p_i, q_i and r_i are linear parameters (i=1 or 2).

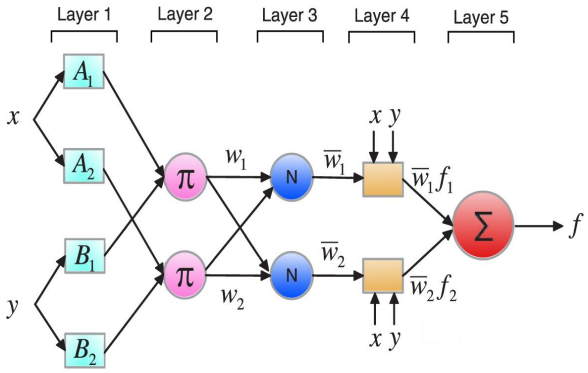


Fig 1. The architecture of ANFIS model

Five layers of ANFIS structure with their associated nodes are given below:

Layer 1: The first layer is the input layer which converts the inputs into a fuzzy set with membership functions (MFs). This layer has two inputs and two MFs associated to each one and contains four adaptive nodes. The output of a node $\mu_{A_i}(x)$ is the MF of A_i. For fuzzy sets, MFs are any continuous and piecewise differentiable functions between 0 and 1.

$$O_{1,i} = \mu_{A_i}(x), \quad \text{for } i = 1,2 \quad (1)$$

The Gaussian and bell-shaped functions are the popular MFs given as:

$$\mu_{GaussA_i}(x) = exp\left[-\frac{1}{2}\left(\frac{x - c_i}{a_i}\right)^2\right] \quad (2)$$

$$\mu_{gbellA_i}(x) = \frac{1}{1 + \left[\left(\frac{x - c_i}{a_i}\right)^2\right]^{b_i}} \quad (3)$$

where a_i, b_i and c_i are premise parameters of the MFs.

Layer 2: Every node in this layer is a fixed node labeled by π, where incoming signals are multiplied and sent to output by the following equation:

$$O_{2,i} = w_i = \mu_{Ai}(x) \times \mu_{Bi}(y), \quad \text{for } i = 1,2 \quad (4)$$

Layer 3: As in the previous layer, every node in this layer is a fixed node. This layer is called the normalization layer labeled by N. The Output of this layer normalizes the firing strength of all rules by the following equation:

$$O_{3,i} = \overline{w_i} = \frac{w_i}{w_1 + w_2}, \quad \text{for } i = 1,2 \quad (5)$$

Layer 4: Every node in the fourth layer is an adjustable node. The node function computes the contribution of each i-th rule based on the consequent parameters and then multiply them by the related normalized weight such that:

$$O_{4,i} = \overline{w_i} f_i = \overline{w_i}(p_i x + q_i y + r_i), \quad \text{for } i = 1,2 \quad (6)$$

where p_i, q_i, r_i are called consequent parameters.

Layer 5: This layer consists of a single fixed node labeled by Σ. The output node computes the overall output as the summation of all incoming signals and equation is given by:

$$O_{5,i} = f_{output} = \sum_i \overline{w_i} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i}, \quad \text{for } i = 1,2 \quad (7)$$

The ANFIS systems are trained by using supervised learning techniques. The learning rule for ANFIS is a hybrid algorithm combining gradient descent and the least-squares method [6]. The gradient descent method is used to tune premise non-linear parameters $\{a_i, b_i c_i\}$ while the least-squares method is used to identify set of consequent linear parameters $\{p_i, q_i, r_i\}$.

III. SUPPORT VECTOR MACHINES (SVM)

Support vector machines (SVM) are linear learning algorithms originated from the studies of statistical learning theory developed by Vapnik [9]. The extension of SVMs for regression (SVR) uses linear function to solve the regression problem by mapping the given data into a high-dimensional feature space via a nonlinear mapping. The illustrative diagram of non-linear SVR is presented in fig 2. The main idea of SVR is to find a linear hyperplane that can separate classes by the maximum margin between support vectors.

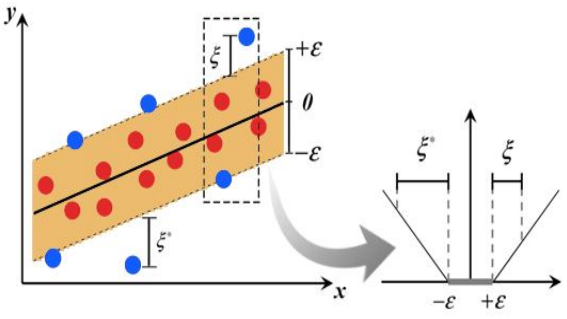


Fig 2. The basic concept of non-linear support vector regression

For the given data set $G = \{(x_i, y_i),\ i = 1, \ldots, n\}$, the regression function is formulated as follows:

$$f(x) = w \cdot \varphi(x) + b \quad (8)$$

where w is a weight vector, b is bias, and φ is the mapping form.

The coefficients w and b are estimated by minimizing the following equation

$$R_{reg}(f) = C\frac{1}{N}\sum_{i=1}^{N} L_\varepsilon(y_i, f(x_i)) + \frac{1}{2}\|w\|^2 \quad (9)$$

$$L_\varepsilon(y_i, f(x_i)) = \begin{cases} |y_i, f - (x_i)| - \varepsilon, & |y_i, f - (x_i)| \geq \varepsilon \\ 0, & otherwise \end{cases}$$

where C and ε are predefined parameters and L_ε is called the ε-insensitive loss function.

By introducing slack variables ξ_i, ξ_i^* the optimization problem (8) can be reformulated as:

$$minimize\ \ R(\omega, \xi_i, \xi_i^*) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \quad (10)$$

$$subject\ to \begin{cases} y_i - w^T x_i - b \leq \varepsilon + \xi_i \\ w^T x_i + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \\ C > 0 \end{cases}$$

By introduction Lagrange multipliers α_i and α_i^*, the convex optimization problem proposed in (10) can be solved using Karush-Kuhn-Tucker (KKT) conditions described in [15].

Finally, the general form of the SVR function can be given by:

$$f(x) = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)K(x_i x) + b \quad (11)$$

where $K(x_i x_j)$ is called a kernel function and any function that satisfies the Mercer's conditions can be used as a kernel function. In this study, the well known Gaussian radial basis function (RBF) is used as kernel function, defined as:

$$K(x, x') = exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right) \quad (12)$$

IV. RESEARCH DESIGN

The main goal of this study is to examine the capability of machine learning techniques in solving the problem of stock market index prediction. The adaptive network-based fuzzy inference system (ANFIS) model and Support Vector Machines (SVM) have been used to construct the prediction model. Most of the previous studies have used multivariate inputs by introducing macroeconomic variables and technical indicators. This study uses the simple model which considers the use of only one variable in construction of the prediction model.

The general forecasting model is constructed based on the following equation:

$$\hat{Y}_{(t+p)} = f\left(Y_{(t)}, Y_{(t-1)}, Y_{(t-d)}\right) \quad (13)$$

where $\hat{Y}_{(t+p)}$ is the predicted value of the target variable for the period p, $Y_t^{(d)}$ is the target variable at the moment of time t and d is the delay.

In our case, the prediction horizon is set to 1, and the delay d is set to 2 based on the methodology proposed by [16]. Consequently, the general forecasting model is given by:

$$\hat{Y}_{(t+1)} = f\left(Y_{(t)}, Y_{(t-1)}, Y_{(t-2)}\right) \quad (14)$$

The research data used in the experiments has been collected from the Bucharest Stock Exchange (BVB). The data set includes 300 daily observations of the main index of BVB called BET index for the period of May 27, 2015 to August 1, 2016.

The data is divided into two sets - training and testing. The training set contains 70% of the whole data (210 data points) and the resting 30% of data (90 data points) is used to validate the results. All time series data are normalized into the scale of [0,1].

The prediction models are implemented in Matlab and default values of settings are used in the experiments. For ANFIS model, the fuzzy subtractive clustering method was applied to separate the input-output training set. For building the SVM prediction model, the radial basis function (RBF) is used as kernel function.

The evaluation of the prediction performance of proposed models is done using two well known measures, namely Mean Absolute Error (MAE) and Root Mean Square Error (RMSE)

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|A(i) - P(i)| \quad (15)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(A(i) - P(i))^2} \quad (16)$$

where n is the number of data points, A represents the vector of actual value and P is the vector of predicted values.

V. RESULTS AND CONCLUSION

Each of the prediction model described in the last section has been used to conduct a comparative analysis in order to draw conclusions with respect to

the forecasting accuracy. Both methods have proved prediction ability of machine learning-based techniques in case of high-frequency data. The forecasting results of proposed models with the actual values in case of Bucharest Stock Exchange are displayed in fig 3, while the prediction accuracy is presented in table 1.

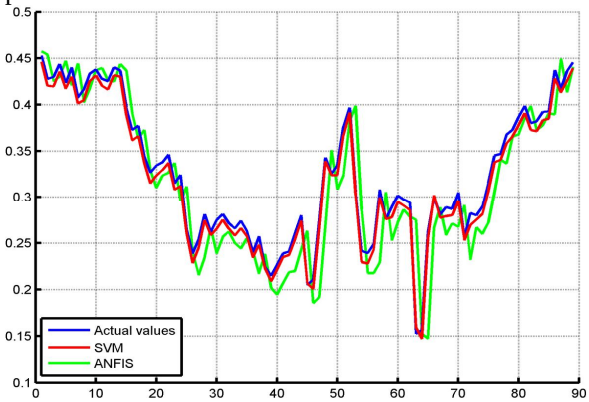


Fig 3. Prediction results of proposed models versus actual values

Model	RMSE	MAE
SVM	**0.0078984**	**0.0074**
ANFIS	**0.035983**	**0.0276**

Table 1. Forecasting accuracy

The prediction results given in Fig 3 show that forecasted values from SVM model are more closer to the actual values rather than the values predicted by ANFIS model. Meanwhile, information in table 3 provided by the evaluation metrics indicates that the SVM model is superior to ANFIS model in forecasting accuracy. Thus, the comparative analysis leads to the conclusion that prediction model based on SVM technique provides more accurate results compared to the results obtained from ANFIS model in prediction task of high-frequency data.

REFERENCES

[1] Abu-Mostafa, Y. S., & Atiya, A. F. (1996). Introduction to financial forecasting. Applied Intelligence, 6(3), 205-213.
[2] Adebiyi, A. A., Adewumi, A. O., & Ayo, C. K. (2014). Comparison of ARIMA and artificial neural networks models for stock price prediction. Journal of Applied Mathematics, 2014.
[3] Cocianu, C. L., & Grigoryan, H. (2015). An Artificial Neural Network for Data Forecasting Purposes. Informatica Economica, 19(2), 34.
[4] Leshno, M., Lin, V. Y., Pinkus, A., & Schocken, S. (1993). Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. Neural networks, 6(6), 861-867.
[5] Tu, J. V. (1996). Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. Journal of clinical epidemiology, 49(11), 1225-1231.
[6] J-S. R. Jang, "ANFIS: Adaptive-Network-Based Fuzzy Inference System", IEEE Trans. Syst., Man, Cybern., vol. 23, pp. 665-685, 1993.
[7] Esfahanipour, A., & Aghamiri, W. (2010). Adapted neuro-fuzzy inference system on indirect approach TSK fuzzy rule base for stock market analysis. Expert Systems with Applications, 37(7), 4742-4748.
[8] Fahimifard, S. M., Homayounifar, M., Sabouhi, M., & Moghaddamnia, A. R. (2009). Comparison of ANFIS, ANN, GARCH and ARIMA techniques to exchange rate forecasting. Journal of Applied Sciences, 9(20), 3641-3651.
[9] Vapnik, V. N., & Vapnik, V. (1998). Statistical learning theory (Vol. 1). New York: Wiley.
[10] Mukherjee, S., Osuna, E., & Girosi, F. (1997, September). Nonlinear prediction of chaotic time series using support vector machines. In Neural Networks for Signal Processing [1997] VII. Proceedings of the 1997 IEEE Workshop (pp. 511-520). IEEE.
[11] Kim, K. J. (2003). Financial time series forecasting using support vector machines. Neurocomputing, 55(1), 307-319.
[12] Tay, F. E., & Cao, L. (2001). Application of support vector machines in financial time series forecasting. Omega, 29(4), 309-317.
[13] Cocianu, C. L., & Grigoryan, H. (2016). MACHINE LEARNING TECHNIQUES FOR STOCK MARKET PREDICTION. A CASE STUDY OF OMV PETROM. Economic Computation & Economic Cybernetics Studies & Research, 50(3).
[14] Grigoryan, H. (2016). A Stock Market Prediction Method Based on Support Vector Machines (SVM) and Independent Component Analysis (ICA). Database Systems Journal, 7(1), 12-21.
[15] Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. Statistics and computing, 14(3), 199-222.
[16] Gujarati, D. N. (2009). Basic econometrics. Tata McGraw-Hill Education.

★★★

[i] "This paper was co-financed by The Bucharest University of Economic Studies during the PhD program"