VERSLAS: TEORIJA IR PRAKTIKA
BUSINESS: THEORY AND PRACTICE
ISSN 1648-0627 print / ISSN 1822-4202 online

2012 13(4): 324–332
doi:10.3846/btp.2012.34

FORECASTING OMX VILNIUS STOCK INDEX − A NEURAL NETWORK APPROACH

Audrius Dzikevičius[1], Neringa Stabužytė[2]

Vilnius Gediminas Technical University, Saulėtekio al. 11, LT-10223 Vilnius, Lithuania
E-mails:[1]audrius.dzikevicius@vgtu.lt (corresponding author); [2]n.stabuzyte@gmail.com

Received 15 June 2012; accepted 10 September 2012

Abstract. Predicting a stock market is a challenging task for every investor. Stock market contains difficult relations and its behavior is heavily forecasted. As the investment's profitability is directly related to the market's predictability, the need for more accurate and sophisticated forecasting techniques arises. The academic literature is showing a growing interest in implementing non- linear techniques in a time series prediction. The paper goes through the process of creating a time series prediction model for OMX Vilnius stock index using artificial neural network approach. A multi layer perceptron model is applied in order to make periodical daily and monthly forecasts for both the actual index future value and the direction of the index. The neural network is trained using back-propagation method, several topologies are analyzed and the most suitable is selected. The method accuracy is compared to several traditional statistical methods (moving averages and linear regression).

Keywords: stock, index, forecast, neural network, back-propagation, layer, error.

JEL Clasification: G15, G17

OMX VILNIUS AKCIJŲ INDEKSO PROGNOZAVIMAS NAUDOJANT DIRBTINIUS NEURONŲ TINKLUS

Audrius Dzikevičius[1], Neringa Stabužytė[2]

Vilniaus Gedimino technikos universitetas, Saulėtekio al. 11, LT-10223 Vilnius, Lietuva
El. paštas: [1]audrius.dzikevicius@vgtu.lt; [2]n.stabuzyte@gmail.com

Įteikta 2012-06-15; priimta 2012-09-10

Santrauka. Akcijų rinkos prognozavimas – sunki užduotis kiekvienam investuotojui. Akcijų rinkoje egzistuojantys sąryšiai yra sudėtingi ir sunkiai numatomi. Kadangi investavimo pelningumas yra tiesiogiai susijęs su rinkos nuspėjamumu, kyla poreikis naudoti modernesnes ir tikslesnes prognozavimo priemones. Užsienio mokslo darbuose vis daugiau dėmesio skiriama netiesiniams laiko eilučių prognozavimo modeliams. Šiame straipsnyje aprašomas OMX Vilnius indekso prognozavimo, naudojant dirbtinius neuronų tinklus, modelis. Daugiasluoksnis neuronų modelis taikomas kitos dienos ir kito mėnesio būsimoms indekso reikšmėms bei indekso krypčiai nuspėti. Neuronų tinklas apmokomas naudojant atgalinio sklidimo algoritmą. Analizuojamos kelios tinklo struktūros ir išrenkama pati tinkamiausia. Metodo tikslumas palyginamas su keliais tradiciniais statistikos metodais (slankiuoju vidurkiu ir tiesine regresija).

Reikšminiai žodžiai: akcija, indeksas, prognozė, atgalinio sklidimo algoritmas, sluoksnis, paklaida.

1. Introduction

Stock market prediction brings a lot of discussion between academia. First of all negotiations arise whether future prices can be forecasted or not. One of the first theories against ability to forecast the market is Efficient Market Theory (EMH). It states that current prices "fully" reflect all available information so there is no possibility to earn any excess profit (Fama 1970). Another important statement was made several years later announcing, that stocks take a random and unpredictable path, stock prices have the same distribution and are independent from each other, so past movement cannot be used to predict the future (Malkiel 1973).This idea stands for Random Walk Theory. According to these statements no one investor could profit from the market without additional unpublicized information or undertaking additional risk. But these theories are facing critics and negotiations that during the time prices are maintaining some trends so it is possible to outperform the market by implementing appropriate forecasting models and strategies.

Researchers provide many models for stock market forecasting. They include various fundamental and technical analysis techniques. Fundamental analysis involves evaluating all the economy as a whole, analyzing exogenous macroeconomic variables, the root is based on expectation. On the contrary, technical analysis is using historical data, such as price and volume variables, preprocessing this data mathematically and making future forecasts rooted in statistics.

Financial time series forecasting brings a lot of challenges because of its chaotic, difficult, unpredictable and nonlinear nature. The most traditional methods are made under assumption that relation between stock price and certain variables is linear. There is evidence that these techniques, such as moving average, do not have acceptable accuracy (Dzikevičius *et al.* 2010). Most popular linear dependencies are simple moving averages, exponential moving averages and linear regression.

One of the newest approaches to forecast dynamic stock market nature is looking for non- linear techniques such as artificial neural networks (ANN). These methods, inspired by human brain, have an ability to find non- linear patterns, to learn from past and generalize. Neural networks are widely used in physical sciences but the popularity is rising in the financial field as well. The main research paper target is to evaluate the neural network ability to forecast stock market behavior by implementing a multi- layer perceptron (MLP) model to predict stock market index OMX Vilnius (OMXV) future movements (actual value and direction of the index). The model's accuracy is compared with several traditional linear models (moving average and linear regression).

The organization of this paper is as follows. The second section provides a brief review of previous researches, the third parts describes data and chosen methodology, the fourth part presents empirical results. The last section provides a brief summary and conclusions.

2. Literature review

The born year of neural network method can be called the year 1958, when the first neural network structure was defined. It was called perceptron (Rosenblatt 1958). Another important date is the year 1986. The authors introduced the 'back-propagation' learning algorithm that still nowadays is the most popular and will be discussed in a more detailed way in the next section (Rumelhart *et al.* 1986).

Nowadays modern ANN use of field is really wide: it includes biological, physical science, industry, finance, etc. There are four main reasons of such increasing popularity of use (Zhang *et al.* 1998). First of them is that oppositely from the other traditional methods ANN have very few assumptions, because they are learning from examples and capturing functional relationships. The second advantage is generalization – the ability to find the unseen part of population from a noisy data. Thirdly, ANN are very good functional approximations and the last one is non-linearity. On the other hand, these models also have some weaknesses: they need training, a large data set and a time for experimenting with the most suitable topology and parameters.

There is a really wide list of ANN applications in finance field. ANN approach can be used to forecast inflation rate (Catik, Karacuka 2012), estimating credit risk (Boguslauskas, Mileris 2009), evaluating foreign direct investment (Plikynas, Akbar 2005), etc. Zhang *et al.* (1998) provides a detailed summary of modeling issues of ANN use in forecasting.

The ability of forecasting stock market is also broadly discussed and results are quite acceptable. A variety of stock market indexes is analyzed using neural network approach. It includes such indexes as BEL 20 (Belgium stock market), BSE Sensex (Bombay stock market index), S&P CNX Nifty 50 (India stock market index), ISE National 100 (Istanbul stock market index), KLCI (Malaysia stock market index), IGBM (Madrid stock market index), TSE (Taiwan stock market index), Tepix (Iran stock market index) , etc. The summary of these researches is provided in Table 1. As it is seen from Table 1, there is a list of evidences that ANN can be used successfully in stock market prediction. The majority of these researches described below select inputs (variables) as lagged values of the dependant variable on different periodicity. Some of them combine both historical data and macroeconomic, fundamental data.

There are evidences that Lithuania's stock market index OMXV was analyzed before. It includes implementing a set of GARCH models for this index (Teresienė 2009), the effect of macroeconomic variables on the index was analyzed by Tvaronavičienė and Michailova (2006), in 2009 by Pilinkus and one year later by Baranauskas (2010).

Table 1. Summary of previous stock market index modeling issues using ANN forecasting

Author	Publishing data (Year)	Index ticker	Object	Data set	Method applied	Results
Lendasse, De Bodt, Wertz, Verleysen	2000	BEL 20	Forecasting the tendency of BEL 20	2600 daily index data	MLP with 1 hidden layer and 5 hidden neurons	Average 65.30% accurate approximations of the sign
Thenmozhi	2006	BSE SENSEX	Forecasting daily returns of BSE SENSEX	3667 daily returns of the index	MLP with 1 hidden layer and 4 hidden neurons	96.6% accuracy of testing data
Desai, Joshi, Juneja, Dave	2011	S&P CNX Nifty 50	Forecasting the daily direction of S&P CNX Nifty 50	Daily index data 01.09.2009–30.04.2011	MLP with 1 hidden layer and 20 hidden neurons	ANN based investment strategy outperformed "buy and hold" strategy
Kara, Boyacioglu, Baykan	2011	ISE National 100	Forecasting the daily direction of ISE National 100 index	2733 daily index data	MLP with 1 hidden layer with various numbers of hidden neurons	Average accuracy 75.74%
Aris, Mohamad	2008	KLCI	Forecasting the direction of KLCI index	5254 daily index data	MLP with 1 hidden layer and 1,2 or 4 hidden neurons	ANN model outperforms moving average
Fernandez-Rodriguez, Gonzalez-Martel, Sossvila-Rivero	2000	IGBM	Forecasting future IGBM index value and sign prediction	Daily index data 02.10.1991–15.10.1997	Feedback network	Sign predictions range 54–58%, trading strategy based on ANN outperforms "buy and hold" strategy in "bear" and "stable" market episodes
Chen, Leung, Daouk	2003	TSE	Forecasting the direction of TSE index after 3, 6, and 12 months	Daily index data 1982–1992	Probabilistic neural network, Generalized methods of moments, random walk	PNN outperforms GNN and random walk
Panahian	2011	TEPIX	Forecasting future trend of TEPIX index	Daily index data 2007–2010	MLP with 1 hidden layer and 3 hidden neurons and multiple regression	ANN model outperformed multiple regression model

Traditional statistical methods of forecasting stock market using moving averages were discussed by Dzikevičius and Šaranda (2010) on the OMX Baltic Benchmark and S&P 500 index. The results revealed that every market is specific and needs a detailed analysis to find the most appropriate parameters for every forecasting technique. In our research ANN model forecasting accuracy is also compared with moving averages approach.

3. MLP model, data and methodology

3.1. MLP structure

Artificial neural networks were inspired by biological science – more exactly by human brain structure. Human brain and nervous system is composed of small cells called neurons. First of all, human body receives a signal from environment, and then the signal is transformed by the receptors into an electric impulse and goes to neuron. Evaluation of the signal is analyzed inside the network and impulse is sent out as an effect. Neurons are connected through the synapses and are able to communicate through them. Learning is a process of adjusting old synapses or adding new ones by using some functions. A simplified structure of this process is provided in Figure 1. The process of transferring inputs to the outputs can be expressed mathematically. Practically it can be implemented using various computer programs software.

Fig. 1. Simplified structure of an artificial neural network

The basic ANN structure consists of artificial neurons that are grouped into layers. A structure of one neuron (perceptron) is presented in Figure 2.

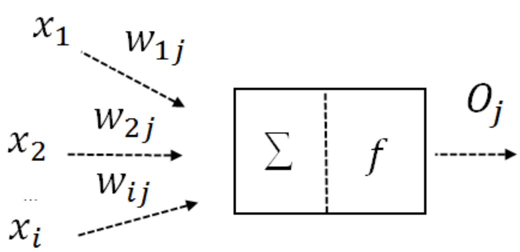

Fig. 2. Structure of one artificial neuron (perceptron)

Here X_1, X_2, …X_i, e called neuron's inputs. Every connection has its weight attached W_{ij} where j is the number of the neurons and i stands for the i-th input. Weights can be both positive and negative. The neuron sums all the signals it receives by multiplying every input by its associated weight:

$$h_j = \sum\left(W_{ij} \cdot X_i\right) \cdot \tag{1}$$

This h_j is often called the summing node. This output h_j goes to the next step through an activation function (sometimes it is called a transfer function f):

$$O_j = f\left(h_j\right) = f(\sum(W_{ij} \cdot X_i)) \cdot \tag{2}$$

Activation function f in most cases is non-linear. It gives the final output O_j . The activation function is chosen according to specific needs: the most popular function is sigmoid (logistic) function:

$$f\left(x\right) = \frac{1}{1+e^{-x}}, \tag{3}$$

and hyperbolic tangent function:

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \cdot \tag{4}$$

These functions are most widely used because of their easy differentiability but other variants are also possible. Linear (identity) function can be used as well.

A structure of several perceptrons and their connections is called a multi – layer perceptron model (MLP). It is typically composed of several layers of neurons. In the first layer the external information is received and it is called an input layer. The last layer is the output layer where the answer of the problem is achieved. These two layers are separated by one or more layers (called hidden layers). If all the nodes are connected from lower to higher layers, the ANN is called a fully connected network. The network is called a feedforward if no cycles or loops of connections exist. There are other types of ANN, but our research focuses on the most traditional feedforward MLP which structure is provided in Figure 3.

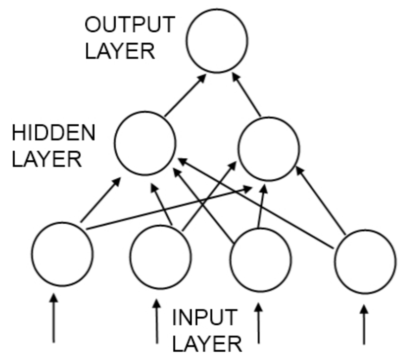

Fig. 3. A typical feedforward neural network (MLP)

The amount of neurons in every layer may have different specifics and the number of hidden layers can be also from zero to tens or more. The MLP structure depends on the nature of every specific data.

3.2. The Back-propagation algorithm

In order to find the most appropriate weights W_{ij} the ANN requires learning procedure. Supervised learning is the most common type. The aim of it is to provide neural network with many previous examples so that it could find the best approximation. The network must be provided with input values together with corresponding output values. Through the iterative process the network is adjusting values while the acceptable approximation is reached. The process is called supervised learning, and a set of examples – training set. The literature provides more other types of learning algorithms, but this research focuses on back-propagation learning algorithm.

The idea of this algorithm is to adjust the weights in a way that error between desired output (target) d_i and actual output y_i ould be reduced:

$$E = \frac{1}{2}\sum(y_i - d_i)^2 . \quad (5)$$

First of all, partial derivatives of the error according to the weights are calculated: $\frac{\eth E}{\eth w_{ij}}$ for all output neurons and hidden neurons. The size of weight changes can be determined by learning rate α (values between 0 and 1), and the weights are adjusted according to the formula until the convergence of error function is reached:

$$W_{new} = W_{old} - \alpha \cdot \frac{\eth E}{\eth W_{old}} \cdot \quad (6)$$

The learning rate controls the speed of the convergence: if it is small the process becomes slower and using a large value of α the error function E ay not converge.

3.3. Steps in implementing MLP model

Before implementing the MLP method several characteristics must be decided:

- Input selection. It is necessary to choose such variables that have a prediction power to the output. In time series forecasting the most common inputs are various periods lagged values of the dependable variable.
- The number of outputs. This selection is directly related to the problem of what the object of forecasting is.
- Data preprocessing (normalization). In order to make the training process easier, data can be scaled. If the network uses such activation functions as sigmoid or hyperbolic tangent, it is necessary to make the data from the same range. Data preprocessing can include logarithmic, linear transformation or statistical normalization.
- The NN architecture. A number of hidden layers and a number of neurons in each layer need to be decided. Previous researches reveal that in most cases one hidden layer is sufficient. The amount of hidden neurons is some kind of experiment to find the best results.
- Activation function. This function describes a relationship between inputs and outputs in the one neuron and the whole network. The literature reveals that the most common activation function used is sigmoid function but it is advisable to test several of them.
- The NN training. Using back-propagation training algorithm learning rates, momentum and number of iterations parameters can be chosen. The best practice also comes from the experience by testing several parameters combination of every specific data.
- The training and testing data. Once the learning process is done by providing the network with training data (examples), the accuracy of this model should be evaluated by providing it with new data. The network makes forecasting using new inputs (testing set) and the accuracy is evaluated by comparing the actual value with the output. All the current data should be provided in two proportions. The most widely used proportion is 90% of training data and 10% of testing data.
- Performance measures. The most frequently used tools for evaluating forecasting accuracy are: Mean Absolute Deviation (MAD), the Sum of Squared error (SSE), Mean Square error (MSE) and Mean Absolute Percentage Error (MAPE):

$$MAD = \frac{\sum|Y_t - F_t|}{N}, \quad (7)$$

$$SSE = \sum(Y_t - F_t)^2 , \quad (8)$$

$$MSE = \frac{\sum(Y_t - F_t)^2}{N}, \quad (9)$$

$$MAPE = \frac{1}{N}\sum\left|\frac{Y_t - F_t}{Y_t}\right| . \quad (10)$$

Where Y_t is an actual value and F_t value is forecasted. These formulas can be used for evaluating the accuracy of actual index forecasts. In order to evaluate the accuracy of forecasting index direction, Sign Prediction correctness (SP) metric is involved:

$$SP = \frac{\sum(Correct\ sign\ predictions)}{Total\ predictions} \cdot \quad (11)$$

The prediction sign is evaluated by taking a difference of two consequent forecasted future values and comparing it to the actual market index movement of the targets.

There are arguments in the literature that the prediction of market index sign in the future is more important than the prediction of actual value. In our research both cases are included.

3.4. Data and methodology

The research focuses on the predictability of OMX Vilnius Stock Index. OMXV – a capitalization weighted index that includes all the shares listed on the Main and Secondary lists. Daily historical data is taken from the official stock exchange site NASDAX OMX (Vilnius Stock exchange, http://www.nasdaqomxbaltic.com). The period investigated is 01.01.2000–30.04.2012. The data is analyzed through two periodicities: daily data (3605 of data points) and monthly data (150 data points). Statistical information about OMXV is provided in Table 2.

Table 2. Statistics of OMXV

N	Min	Max	Mean	St.dev.	Kurt.	Skew.
3605	63.18	591.44	249.15	151.22	-1.22	0.39

On every periodicity the data is divided into two sets: historical data (or training data) and forecasting data (testing data) that is known and used to evaluate the accuracy of predictions. The predictability is evaluated for the actual value forecasting and for the future movement (sign) forecasting. The proportion of historical and forecasting data is approximately taken as 90% and 10%. So, daily data consists of 3245 historical and 360 testing data sets. Monthly data: 135 historical data and 15 testing data.

We use such forecasting tools: Simple Moving Average (SMA), Multiple regression and several structures of MLP (learning algorithm is back-propagation) in order to make daily and monthly predictions for the actual value and the movement of the index. As the input or variables from the previous 4 period lagged values of the index are used (4 lagged daily values and 4 lagged monthly values). The accuracy is compared using (7)–(11) formulas.

The prediction using Simple Moving Average (SMA) is done under assumption, that forecasted value of the fifth period is equal to the average of last four periods:

$$SMA = \frac{\sum_{t=1}^{n} Y_t}{n} = A_t , \qquad (12)$$

$$F_{t+1} = A_t . \qquad (13)$$

Multiple regression forecast is calculated under assumption that the fifth period index value is dependent variable from the previous lagged four periods values.

As in some MLP cases sigmoid and hyperbolic tangent functions are used, the data is preprocessed using linear transformation to the interval [a,b]. In the sigmoid case: [0,1], hyperbolic tangent case: [-1;1].

$$X_n = \frac{(b-a)\cdot(x_0 - X_{\min})}{(X_{\max} - X_{\min})} + a . \qquad (14)$$

Calculations are done with MS Excel 2007 and MATLAB R2009B, Neural Network Toolbox.

4. Empirical results

4.1. Daily forecasting

The accuracy of forecasting future daily values was evaluated by traditional multiple regression, simple moving average and 12 MLP models containing different structures and transfer functions. The forecasting ability was evaluated for both the prediction of actual value and also the sign of future index movement. All MLP structures contained 1 hidden layer and the number of hidden neurons varied from 1 to 6. Learning rate α was chosen to be 0,1 because other combinations did not improve the results. Every MLP model used 3000 epochs (iterations) to train. The empirical results are provided in Table 3. As it is seen from the results, the lowest forecasting error for the actual index value was achieved by using MLP model with one hidden layer and 2 hidden neurons with the selection of log-sigmoid transfer function. The results are very similar to the multiple regression results. Nevertheless, this MLP model outperforms multiple regression method according to all types of errors calculated, the difference is very slow. For this reason standard deviation of absolute percentage error is calculated. For the multiple regression case the standard deviation of absolute percentage error is 0.01030. and for the MLP model (1 hidden layer with 2 hidden neurons and log sigmoid transfer function) it is 0.01028. So, the second case provides a little bit more stable forecasts. The graph of the absolute percentage error for this case is provided in Figure 4 showing the error for every of 350 predictions of the future index actual value.

As the results reveal, the simple moving average was the least accurate forecasting technique for future daily values.

The best accuracy for prediction of future index direction achieved was 53.06 % using MLP with one hidden layer and selection of 1 neuron (both transfer functions) and also with selection of the hyperbolic tangent transfer function and 5 neurons in 1 hidden layer. All forecasting techniques for index direction provide approximately 50% of correct predictions and that is quite a poor result.

4.2. Monthly forecasting

Monthly forecasting results are found to be really unacceptable. The main reason could be a cause of the lack of training set (historical data used to construct the model). Only 135 historical samples were used to predict 15 future values. The results are provided in Table 4.

The empirical results reveal that all MLP structures were unsuccessful in predicting future index values. The lowest forecast error for the actual value was achieved by

Table 3. Accuracy results for different daily forecasting techniques

Transfer function – hyperbolic tangent					**Transfer function – log sigmoid**				
1 hidden layer – 1 hidden neuron									
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
2.6718	7070.8355	19.6412	0.7451%	53.0556%	2.6718	7070.8355	19.6412	0.7451 %	53.0556%
1 hidden layer – 2 hidden neurons									
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
2.6618	7032.1291	19.5337	0.7414%	51.1111%	2.6607	7028.5305	19.5237	0.7412 %	51.6667%
1 hidden layer – 3 hidden neurons									
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
2.7500	7330.4234	20.3623	0.7672%	52.2222%	2.9236	11212.0085	31.9778	0.8225%	51.1111%
1 hidden layer – 4 hidden neurons									
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
2.8248	8313.9507	23.0943	0.7883%	50.8333%	4.7524	53881.3056	1496.8925	1.4120%	51.1111%
1 hidden layer – 5 hidden neurons									
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
2.8966	13886.7201	38.5742	0.8173%	53.0556%	2.9042	10427.2433	28.9646	0.8140%	52.5000%
1 hidden layer – 6 hidden neurons									
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
2.9082	10485.9830	29.1277	0.8167%	50.8333%	2.9393	10701.1447	29.7254	0.8255%	50.2778%
Multiple regression					**Simple moving average (4)**				
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
2.6635	7054.4705	19.5958	0.7420%	51.1111%	3.9180	14640.9342	40.67	1.09%	49.17%

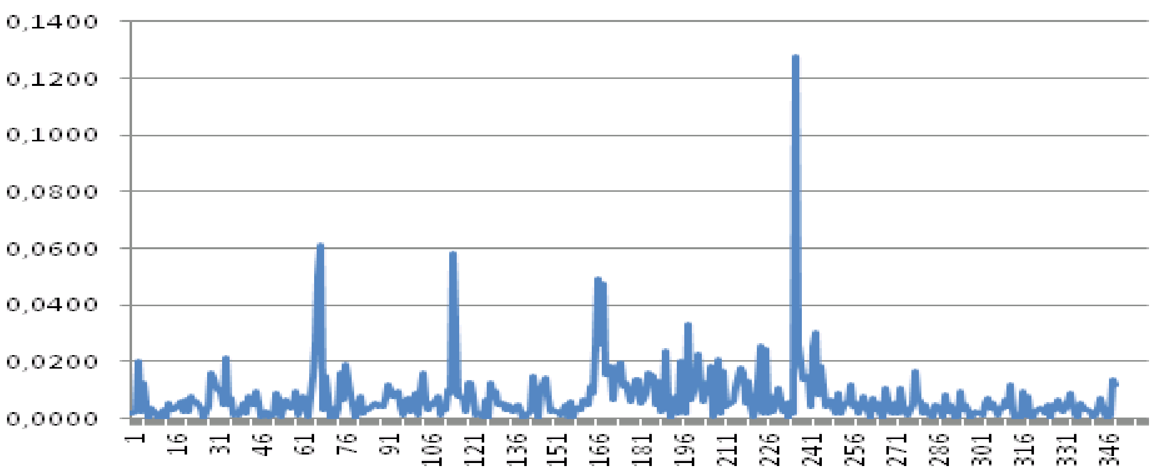

Fig. 4. Absolute percentage error for the 350 future daily forecasts

Table 4. Accuracy results for different monthly forecasting techniques

Transfer function – hyperbolic tangent					**Transfer function – log sigmoid**				
1 hidden layer – 1 hidden neuron									
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
17.2318	9213.7714	614.2514	5.0885%	40.0000%	17.2318	9213.7714	614.2514	5.0886%	40.0000%
1 hidden layer – 2 hidden neurons									
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
20.3298	9379.2678	625.2845	8.8388%	46.6667%	20.3298	9379.26.76	625.2845	5.8388%	46.6667%
1 hidden layer – 3 hidden neurons									
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
19.1361	10211.3395	680.7560	5.5538%	46.6667%	26.6590	17170.8006	1144.7200	8.0470%	46.6667%
1 hidden layer – 4 hidden neurons									
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
25.5821	14978.8270	998.5885%	7.5084%	33.3333%	27.8724	18768.8735	1251.2582	7.9776%	53.3333%
1 hidden layer – 5 hidden neurons									
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
31.8335	29343.0403	1956.0403	8.8157%	40.0000%	31.0626	21620.1964	1441.3464	8.8771%	33.3333%
1 hidden layer – 6 hidden neurons									
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
32.0707	26318.8587	1754.5906	9.1244%	46.6667%	35.3390	31104.7844	2073.6523	9.9803%	40.0000%
Multiple regression					**Simple moving average (4)**				
MAD	SSE	MSE	MAPE	Correct sign (SP)	MAD	SSE	MSE	MAPE	Correct sign (SP)
15.2200	6309.3059	420.6204	4.4712%	46.6667%	19.9212	11502.7355	766.8490	5.9461%	46.6667%

using multiple regression but the results are really poor. The highest accuracy for predicting index direction is approximately 53% by using MLP with 1 hidden layer, 4 hidden neurons and selection of log sigmoid transfer function. It is quite the same result as for making predictions for daily future index movements, but other forecasting techniques provided less than 50% of correct predictions.

5. Conclusions and further investigation

In comparison of forecasting future OMXV index using daily and monthly basis, the daily predictions are several times more accurate. In making daily predictions the lowest forecasting error was achieved by using MLP with 1 hidden layer and 2 hidden neurons with the selection of log sigmoid transfer function. The best index direction movement forecasting was also made by using several MLP model's topologies – 53.06%. In this case MLP model outperformed both traditional multiple regression and simple moving average methods.

Monthly predictions are found to be really poor. Multiple regression method outperforms moving average and all MLP structures in forecasting the actual value but nevertheless the results are inaccurate. The best direction forecast is done by MLP with 1 hidden layer, 4 neurons and log sigmoid transfer function – 53.33%.

Further investigations may be improved by adding more variables, using not only consequent lagged historical values. For the reason that this research uses only one type of neural network – feedforward network, also, more neural network types should be discussed including different algorithms of learning.

References

Aris, Z.; Mohamad, D. 2008. Application of artificial neural networks using Hijri Lunar transaction as extracted variables to predict stock trend direction, *Labuan e-Journal of Muamalat and Society* 2: 9–16.

Baranauskas, S. 2010. Portfolio formation and management according to macroeconomic indicators influence on OMXV, *Business: Theory and Practice* 11(3): 286–293.

Boguslauskas, V.; Mileris, R. 2009. Estimation of credit risk by artificial neural networks models, *Inzinerine Ekonomika – Engineering Economics* (4): 7–14.

Catik, A. N.; Karacuka, M. 2012. A comparative analysis of alternative univariate time series models in forecasting Turkish inflation, *Journal of Business Economics and Management* 13(2): 275–293. http://dx.doi.org/10.3846/16111699.2011.620135

Chen, A.; Leung, M. T.; Daouk, H. 2003. Application of neural networks to an emerging financial market: forecasting and trading the Taiwan stock index, *Computers & Operations Research* 30: 901–923. http://dx.doi.org/10.1016/S0305-0548(02)00037-0

Desai, J.; Desai, K.; Joshi, N.; Juneja, A.; Dave, A. R. 2011. Forecasting of Indian stock market index S&P CNX Nifty 50 using artificial intelligence, *Behaviourial & Experimental Finance eJournal* 79(3): 1–15.

Dzikevičius, A.; Šaranda, S.; Kravčionok, A. 2010. The accuracy of simple trading rules, *Economics and Management* 15: 910–916.

Dzikevičius, A.; Šaranda, S. 2010. EMA Versus SMA usage to forecast stock markets: the case of S&P 500 and OMX Baltic Benchmark, *Business: Theory and Practice* 11(3): 248–255.

Fama, E. F. 1970. Efficient capital markets: A review of theory and empirical work, *Journal of Finance* 25(2): 383–417. http://dx.doi.org/10.2307/2325486

Fernandez-Rodriguez, F.; Gonsalez-Martel, C.; Sosvilla-Rivero, S. 2000. On the profitability of technical trading rules based on artificial neural networks: Evidence from the Madrid stock market, *Economics Letters* 69: 89–94.

Kara, Y.; Boyacioglu, M. A.; Baykan, O. K. 2011, 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. http://dx.doi.org/10.1016/j.eswa.2010.10.027

Lendasse, A.; De Bodt, E.; Wertz, V; Verleysen, M. 2000. Nonlinear financial time series forecasting – application to the BEL 20 stock market index, *European Journal of Economic and Social Systems* 14(1): 81–91. http://dx.doi.org/10.1051/ejess:2000110

Malkiel, B. G. 1973. *A random walk down the Wall Street: The time tested strategy for successful investing.* New York: W. W. Norton.

Math Works^{TM}, Neural Network Toolbox Documentation. http://www.mathworks.com/help/pdf_doc/nnet/nnet_ug.pdf

Panahian, H. 2011. Stock market index forecasting by neural networks models and nonlinear multiple regression modeling: study of Iran's capital market, *American Journal of Scientific Research* 18: 35–51.

Pilinkus, D. 2009. Stock market and macroeconomic variables: evidences from Lithuania, *Economics & Management* 14: 884–891.

Plikynas, D.; Akbar, Y.H. 2005. Explaining the FDI patterns in central and Eastern Europe: a neural network approach, *Ekonomika* 69: 78–103.

Rosenblatt, F. 1958. The perceptron: a probabilistic model for information storage and organization in the brain, *Psychological Review* 65(6): 386–408. http://dx.doi.org/10.1037/h0042519

Rumelhart, D. E.; Hinton, G. E.; Williams, R. J. 1986. Learning internal representations by back-propagating error, *Nature* 323: 533–536. http://dx.doi.org/10.1038/323533a0

Teresienė, D. 2009. Lithuanian stock market analysis using a set of GARCH models, *Journal of Business Economics and Management* 10(4): 349–360. http://dx.doi.org/10.3846/1611-1699.2009.10.349-360

Thenmozhi, M. 2006. Forecasting stock index returns using neural networks, *Delhi Business Review* 7(2): 59–69.

Tvaronavičienė, M.; Michailova, J. 2006. Factors affecting securities prices: theoretical versus practical approach, *Journal of Business Economics and Management* 7(4): 213–222.

Vilnius Stock Exchange. Statistical data on OMXV index history. http://www.nasdaqomxbaltic.com/ market/?pg=charts&lang=en&idx_main[]=OMXV&add_index=OMXBBPI&add_equity=LT0000128266&period=other&start_d=1&start_m=1&start_y=2000&end_d=30&end_m=4&end_y=2012.

Zhang, G.; Patuwo, B. E.; Hu, M. Y. 1998. Forecasting with artificial neural networks: the state of art, *International Journal of Forecasting* 14: 35–62 http://dx.doi.org/10.1016/S0169-2070(97)00044-7

Audrius DZIKEVIČIUS is an Associate Professor at the Department of Finance Engineering of Vilnius Gediminas Technical University (VGTU), defended doctoral dissertation "Trading Portfolio Risk Management in Banking" (2006) and was awarded the degree of Doctor of Social Sciences (Economics). In 2007 he started to work as an associate professor at the Department of Finance Engineering of VGTU. His research interests cover the following items: portfolio risk management, forecasting and modeling of financial markets, valuing a business using quantitative techniques.

Neringa STABUŽYTĖ was awarded Bachelor's degree (Statistics) in 2010. At the present time she continues her postgraduate studies on "Investment Management" in Vilnius Gediminas Technical University. The research interests cover the following items: investment, forecasting of financial markets.