Forecasting stock market prices

Miroslav Janeski, Slobodan Kalajdziski

Faculty of Electrical Engineering and Information Technologies, Skopje, Macedonia
miroslav.janeski@gmail.com,skalaj@feit.ukim.edu.mk

Abstract. In recent years, use of data mining and machine learning techniques in finance for such tasks as pattern recognition, classification, and time series forecasting have dramatically increased. However, the large numbers of parameters that must be selected to develop a good forecasting model have meant that the design process still involves much trial and error. The objective of this paper is to select the optimal parameters for designing of a neural network model for forecasting economic time series data. There is proposed a neural network based forecasting model for forecasting the stock market price movement. The system is tested with data from one Macedonian Stock, the NLB Tutunska Banka stock. The system is shown to achieve an overall prediction rate of over 60%. A number of difficulties encountered when modeling such forecasting model are discussed.

Keywords: neural network, backpropagation, financial forecasting, time series, stock forecasting

1 Introduction

Forecasting is a process that produces a set of outputs by given a set of variables. It assumes that some aspects of the past patterns will continue into the future. In general, forecasting process is working with historical data. Main idea is to discover patterns in the past and to use them in current moment. If we have a function which can map t-2 data to forecast t-1 data, then we can use it to predict t+1 data with t data. Forecasting financial time series is even more complicated task because there are a various sets of input data that can be analyzed. Methods for forecasting financial markets, especially stock market prices can be grouped into two types: technical and fundamental analysis.

Fundamental analysis of a market is consisting of analyzing overall state of the economy and overall state of the market where the observed companies are operating. With this analysis the stock price can be predicted reasonably. Fundamental analysis has very good performance for long term prediction, but there is also a huge disadvantage. This analysis is difficult to formalize for automated decision support because very often is highly subjective.

Technical analysis is a process for forecasting the future direction of prices of a stock market through the analysis of past market data, especially stock prices and volume. Unlike the fundamental analysis, technical analysis is used for short term prediction. Although this analysis is widely used in forecasting it is not recommended because is highly subjective and statistically not valid.

Despite the above two analysis, there are several data mining techniques that offer very good performance in forecasting financial time series. In recent years, artificial neural networks [13] (ANN's) have become another important technique for predicting stock prices. A NN is an interconnected network of simple processing nodes with a different weight associated with each connection. With a proper network topology and appropriate weights between the connections, a NN can be trained to approximate any function mapping between its input(s) and output(s) by using an appropriate learning algorithm such as backpropagation (BP). In [5] is shown how a forecasting model can produce buy/sell signals, in [6,7 and 8] are shown best practices for data preprocessing, especially time series data, and in [2 and 4] are shown best practices in designing a good forecasting model.

2 Designing a neural network based forecasting system

The object of this paper is to summarize recommendations from the recent research [2 and 14] and to experimentally select the optimal parameters for designing a neural network based model for forecasting time series, especially predicting the daily closing price from one Macedonian Stock, the NLB Tutunska Banka stock. The model shall provide framework for testing different input parameters and different network topology parameters. These variations will be tested with training data, and then the optimal parameters will be selected for the testing data.

Fig.1 illustrates the proposed architecture of neural network. On the left (input) side of the neural network are different time series which will be input variables in the forecasting model. On the output side is one output node which represent the predicted time series, in this case closing price of the above stock.

2.1 Neural network overview

Recent research [1 and 2] activities in artificial neural networks have shown that ANN's have powerful capabilities in classification and regression problems. There are several advantages [2] that make them attractive in forecasting financial time series. First of all they are data – driven self adaptive methods, which means that ANN's can be used in field where we know a little or nothing about the relationships along the analyzed data. Second, they can generalize. ANN's can often correctly infer the unseen part of the data even if the data has a high noise coefficient. As forecasting is performed via prediction of future behavior (unseen part) from examples of past behavior, it is an ideal application for neural networks. And the third advantage is that the ANN's are universal approximators. They can approximate any kind of function, regardless it is linear or non linear. Despite the advantages there are disadvantages in the using of ANN's. One of the main disadvantages is the black box nature of their solutions, excessive training times, difficulty in replicating stable solutions, over fitting and the large number of parameters that must be experimentally selected to generate a good forecast.

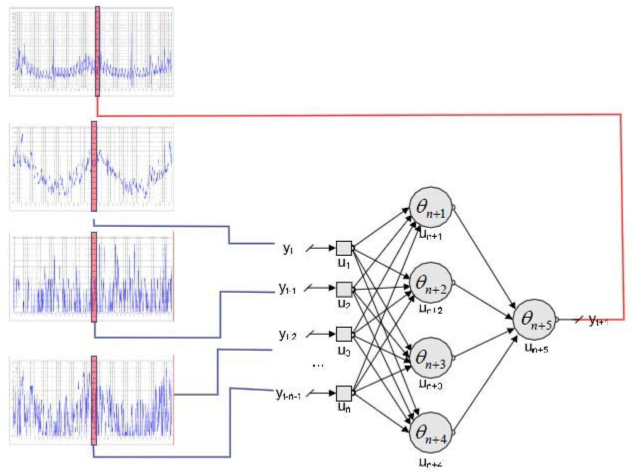

Fig. 1. Neural network architecture [14]

According to a recent survey [12] made by Crone/Kourentzes and [13] there is a decade's long research interest in artificial neural networks (ANN) that has lead to several successful applications. Besides their success, both in theoretical and practical works, there are still a lot of modeling issues [9 and 10] that are critical to design a reliable forecasting model. Kaastra [4] and [3 and 5] are proposing similar procedures for designing a neural network forecasting model which are consisted of several main tasks: variable selection, data preprocessing ,training and testing sets, neural network paradigm and neural network training. Every step has its own meaning and own part in the overall performance of the forecasting system.

2.2 Variable selection

Success in designing a neural network depends on a clear understanding of the problem. Our problem is to predict the daily closing price, so the traders can use it to make their trading more successful. Same as the mentioned type of analysis: fundamental and technical, there are fundamental and technical data that can be used as input variables. Fundamental data are mainly represented via fundamental parameters, like local and global economy and market parameters, which despite technical data are harder to interpret as an input variables for short term prediction. Technical data are in general prices and volume from historical trading on the stock market. In the [5] is proposed "keep it simple and short" KISS method which uses only technical data from every day trading and some derived technical data. In our approach we will use the same input variables which are:

O_i	the opening price of day i
O_{i-1}	the opening price of day i–1
H_{i-1}	the daily high price of day i–1
L_{i-1}	the daily low price of day i–1
C_{i-1}	the closing price of day i–1
C_{i-2}	the closing price of day i–2
V_i	the trading volume of day i
M_{i-1}	the 5-day momentum of day i-1(C_{i-1}–C_{i-6})

The last one, the 5 day momentum is used as a derived technical attribute, which is the difference between C_{i-1} and C_{i-6}. The output of the system will be the daily closing price of the stock. The historical data can be found on Macedonian Stock Exchange web site (http://www.mse.org.mk).

2.3 Data Preprocessing

The next step in designing a neural network is data preprocessing. Once the attributes are selected they should be adequately preprocessed in order to achieve higher quality and reliable predictions. Data preprocessing refers to analyzing and transforming the input and output attributes to minimize noise, highlight important relationships, and detect trends and seasons of the variables to assist the neural network in learning the relevant patterns. In recent research by Zhang [7 and 8] there are several techniques for detrending and deseasonalization the time series. It is shown that these techniques can influents on the overall forecasting performance of the system. In our approach these techniques will be left out of the analysis. Also, Sven Crone in his research [6] is experimenting with various scaling methods. His results show that scaling offers better performance than normalization. In our approach attributes from the data sets are scaled in interval [-1, 1], appropriate to the thresholds of the activation function which will be explained latter.

2.4 Training and testing sets

Defining the training and testing sets and strategy is probably the most important issue in designing a neural network. Training set is usually larger than the testing set and is used to train the neural network. The neural network model that has the best results [4] on the training set shall be used on testing set. In the nature of the time series is the time as a dimension which must be considered in the analysis. Therefore, a moving-windows approach [11] was adopted for training and testing.

Defining the appropriate size of the training set is the main issue in the moving-windows approach. If the training set is too small, neural network will convergence easily, but it will not predict with acceptable accuracy, on the other side if the training set is too large it will be more difficult for the network to convergence. Because time series are time depended, if the training set is too large (in meaning of time period) it may have learned some patterns that may no longer be effective because the market conditions are have already changed.

The benefit of the moving windows approach is that the observed market and stock may change over the time, so the constant moving the training and testing sets can avoid these changes. That means the trained network that was optimal in the past may not be optimal any more. Besides the recommendations [4 and 5] for using the moving-windows approach there are no defined rules how to define the training set size. The testing period was from May 2009 to May 2010. Because the period has 13 months the training set sizes with which was tested were less than a13 months. We tested with 12 months, 6 months and 3 months period, like a training set sizes. The appropriate training sets are described in [tables1, 2 and 3].

Table 1. 12 + 1 month training and testing sets

Cycle	*Training set*		*Testing set*	
	Period	No. of patterns	Period	No. of patterns
1	May 09 - Apr 10	201	May 10	16

Table 2. 6 + 1 month training and testing sets

Cycle	*Training set*		*Testing set*	
	Period	No. of patterns	Period	No. of patterns
1	May 09 - Oct 09	86	Noe 09	20
2	Jun 09 - Noe 09	98	Dec 09	19
3	Jul 09 - Dec 09	100	Jan 10	16
4	Aug 09 - Jan 10	104	Fev 10	20
5	Sept 09 -Fev 10	109	Mar 10	21
6	Oct 09 - Mar 10	115	Apr 10	19
7	Noe 09 - Apr 10	115	May 10	16

After preliminary experiments we have concluded that best performance is achieved when training set size is three month period, so three month period training set size is used for further experiments. On fig.3 are illustrated experimental results.

Table 3. 3 + 1 month training and testing sets

Cycle	*Training set*		*Testing set*	
	Period	No. of patterns	Period	No. of patterns
1	May 09 - Jul 09	37	Aug 09	15
2	Jun 09 - Aug 09	44	Sept 09	15
3	Jul 09 - Sept 09	42	Oct 09	19
4	Aug 09 - Oct 09	49	Noe 09	20
5	Sept 09 -Noe 09	54	Dec 09	19
6	Oct 09 - Dec 09	58	Jan 10	16
7	Noe 09 - Jan 10	55	Fev 10	20
8	Dec 09 - Feb 10	55	Mar 10	21
9	Jan 10 - Mar 10	57	Apr 10	19
10	Feb 10 - Apr 10	60	May 10	16

Another issue in the training process is the overtraining of the network which means that the network is trained too much, or it is too complex. Overtraining can be avoided if the mean square error is minimized during training. That means that the number of epochs should be decreased when MSE is increasing, which is done during the experiments.

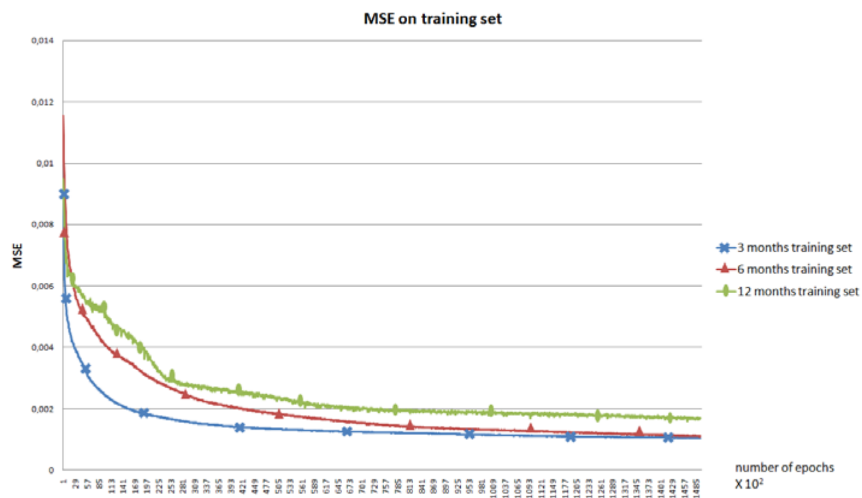

Fig.3. MSE on different training set sizes

2.5 Neural network paradigms

Defining neural network parameters is related with the problem of the observed domain. In general it is consist of defining the number of hidden layer, number of hidden neutrons, number of output neutrons, which are architectural considerations, and the dynamics of the neural network which is a transfer or activation function.

2.5.1 Neural network topology

Because the number of input and output variables are already defined with the attribute selection process the only thing remaining is the number of hidden layers and the number of hidden nodes. In theory, a neural network with one hidden layer with a sufficient number of hidden neurons is capable of approximating any continuous function. In practice [1 and 12], networks with one or two hidden layers are used in 90% of the cases. Large number of hidden layers than two can make the neural network time consuming and over fitted, which means that the network has too few degrees of freedom and memorize general patterns. In our system we used one hidden layer with different number of hidden neurons which was part of the experiments. So we have the following topology: 8 – k -1; k – number of hidden neutrons.

Besides the large number of examples of neural network models in the recent research there is no formula for determination the number of hidden neurons. Some recommendations [4] are that the number has to be from one half to three times the number of input nodes. Also in [5] is provided the following formula (1) which was used in our experiments

$$k = i \, x \, N - 1; \; i = 1, 2, 3... \quad \textbf{(1)}$$

However, selecting the best number of hidden neurons involves experimentation. According to this we have tested with different number of hidden neurons, 7, 10, 14 and 21.

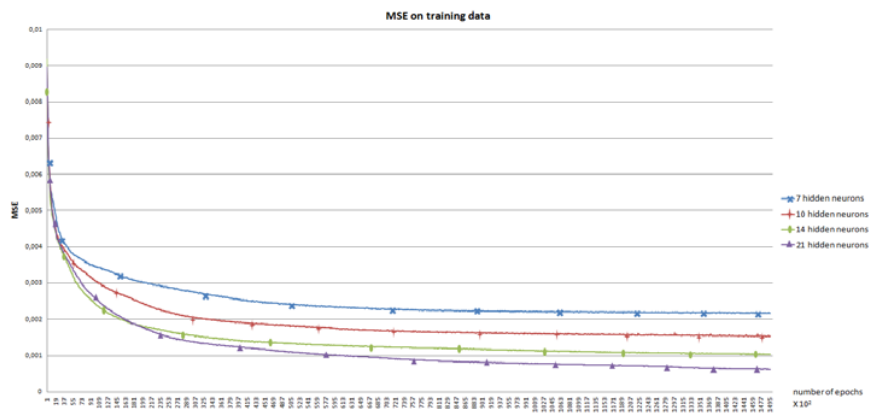

Fig. 4. MSE on different number of hidden neurons

Fig. 4 illustrates that the number of hidden neurons seriously has very high impact on the error rate of the neural network. The experiments have shown that neural network with 21 hidden neurons has lower error rate than the networks with less number of hidden neurons.

2.5.2 Transfer function

Transfer functions are mathematical formulas that determinate the output of a processing neuron. The large amount of neural networks models uses the sigmoid (S) function, besides the others like hyperbolic tangent, step, ramping, etc. In our system sigmoid function is used as a transfer function. Selection of the transfer function is often related with data scaling. To provide consistent system, data scaling in interval [-1, 1] was used same as the thresholds of the selected transfer function [-1, 1].

2.6 Training

Number of training iterations is another important issue in designing a neural network. There are two approaches in the literature [4] about selecting the right number of training iterations. First school says that researcher should only stop training until there is no improvement in the error function. The second school says that there should be series of training test iterations, where neural network is trained on different number of iterations and tested on test data. In our system we use the first approach. Firstly we started with 150,000 epochs, but on fig.4 we have shown that there is no big improvement in MSE after 80,000 epochs, which was therefore used for the testing set.

Other important issues in designing a neural network are connected with the backpropagation algorithm. These issues are learning rate and momentum. The recommendations are that the learning rate should be <0, 5 and momentum = 0, 4.

3 Experimental results and discussion

In general the performance of the neural network is measured by its MSE error, because is one of the most used in recent research [12]. MSE error of the neural network was especially important in the training phase where the optimal parameters were selected according to the MSE error movement. From the results of the training phase (fig.4) we tested with two neural network architectures. One is 8 – 14 – 1 and the second is 8 – 21 – 1. MSE errors on different neural network architectures are shown in table 4. The results was that 8-14-1 architecture has minimum MSE error, therefore was selected for further testing.

Table 4. MSE errors on testing phase

# of hidden neutrons	*14*	*21*
MSE	0,046	0,086

However in the testing phase the actual value or usefulness of the neural network is measured by its ability to make accurate predictions of the future market prices. In our approach we use an average prediction rate. We recorded the predictions of the daily closing price of NLB Tutunska Banka stock over the 10 months test period (table 3) and compared them with the actual values. For each testing cycle we generated average prediction rate, and also an average prediction rate of the whole testing period.

Table 5 depicts the experimental results. Column labeled “Average Prediction Rate” show the results for neural network accuracy in different cycles. The average prediction rate of the system is over 60%, which is fairly good. In the cycles we have ones with very high accuracy and with very low accuracy. Main reason for these oscillations is the difference in the training set size. Besides the fact that training size set was limited to three month period, obviously it was not enough. In training sets we have sets that are two times larger than the other ones. In fact, that is from the nature of the time series. In some periods time series are more frequent, in other they are not.

This approach was effective [5] in stock exchange where stock price values were more constant during the time (in a manner of frequency).

Table 5. System accuracy

Cycle	*Testing set* Period	# of patterns	Average Prediction Rate	MSE
1	Aug 09	15	37,69	0,0141
2	Sept 09	15	34,46	0,0733
3	Oct 09	19	66,34	0,1019
4	Noe 09	20	53,54	0,0887
5	Dec 09	19	58,05	0,0722
6	Jan 10	16	78,02	0,0351
7	Fev 10	20	84,39	0,0166
8	Mar 10	21	91,03	0,0075
9	Apr 10	19	70,75	0,0338
10	May 10	16	16,72	0,0102
Average Prediction Rate & MSE			60,98	0,0460

We have shown that for Tutunska Banka stock, we need more constant training set size than the proposed one. In future work we will test with same parameters for others stocks from Macedonian stock exchange and from other Balkan stock exchanges to check our hypothesis and to find the optimal mowing windows approach parameters.

Also, it is reasonable to expect that results can be improved in the following ways: including more technical, derived technical or even fundamental input attributes and fine tuning on network topology.

Finally, in this work we have used best practices in recent research and keep the simplicity in mind. Our approach uses the simple methods and recommendations to build neural network forecasting system which has an acceptable performance in accuracy and cost.

4 Conclusions

In this paper was elaborated a theoretical and practical analysis on the design and implementation of a neural based system for forecasting stock market prices, in detail the NLB Tutunska Banka Stock. Main objective was to determine the optimal parameters, especially the ones which still are generated via trial and error process. In this phase we have experimental results in defining the training set size in moving-windows approach, number of training epochs and number of hidden neurons. The design issues of neural networks for prediction tasks were discussed in detail and the design of forecasting system was presented. The empirical results showed that the proposed system was able to predict short-term price movement directions with an overall accuracy of over 60%. The system demonstrated that fairly good prediction accuracy can be achieved using a backpropagation network without the use of extensive market data or knowledge.

References

1. Zhang G.P., B.E. Patuwo, M.Y. Hu, "Forecasting with artificial neural networks: the state of the art", Int. J. Forecast. Vol.14 pp.35–62, 1998
2. Adya, M. & Collopy, F. "How effective are neural networks at forecasting and prediction? A review and evaluation". Journal of Forecasting, Vol.17, pp.481-495. 1998
3. JingTao Yao and Chew Lim Tan. "Guidelines for financial forecasting with neural networks". In Proceedings of International Conference on Neural Information Processing, pages 757-761, Shanghai, China, November 2001
4. Kaastra L. and M. Boyd. "Designing a neural network for forecasting financial and economic time series". Neurocomputing, Vol 10, Issue 3, pp 215-236 April 1996,Financial Applications, Part II
5. Philip M. Tsang, Paul Kwok, S.O. Choy, Reggie Kwan, S.C. Ng, Jacky Mak, Jonathan Tsang, Kai Koong, Tak-Lam Wong," Design and implementation of NN5 for Hong Kong stock price forecasting", Engineering Applications of Artificial Intelligence Vol.20 pp.453–461 (2007)
6. Crone S., Lessmann S., Stahlbock R.. "The Impact of Preprocessing on Data Mining: An Evaluation of Support Vector Machines and Artificial Neural Networks", European Journal of Operations Research, Vol. 173, No.3, pp. 781-800, 2005
7. Zhang, G. P. & QI, M. "Neural network forecasting for seasonal and trend time series". European Journal Of Operational Research, Vol.160, pp. 501-514. 2005
8. Zhang G. P. , D. M. Kline, "Quarterly time-series forecasting with neural networks", IEEE Transactions on Neural Networks, Vol.18/6, pp.1800-1814, 2007
9. Crone S.. "Stepwise Selection of Artificial Neural Network Models for Time Series Prediction", Journal of Intelligent Systems, Vol. 14, No. 2-3,pp. 99-122, 2005
10. Crone S., R. Dahwan. "Forecasting Seasonal Time Series with Neural Networks: A Sensitivity Analysis of Architecture Parameters", Proceedings of the International Joint Conference on Neural Networks, IJCNN'07, Orlando, USA, IEEE: New York, 2007
11. Skabar, A., Cloete, I., 2002. Neural networks, financial trading and the efficient market hypothesis. Australian Computer Science Communications 24 (1), 241−249
12. Kourentzes, N. Crone S. "Advances in forecasting with artificial neural networks", working paper, Lancaster University Management School, Lancaster UK 2010
13. Dorffner, G, "Neural Networks for Time Series Processing". *Neural Network World* Vol.4/96, pp. 447-468, 1996.
14. Crone S.F.. "Forecasting with Artificial Neural Networks", EVIC 2005 Tutorial Santiago de Chile, 15 December 2005