Journal of Economics, Finance and Administrative Science
ISSN: 2077-1886
jguillen@esan.edu.pe
Universidad ESAN
Perú

Hedayati Moghaddam, Amin; Hedayati Moghaddam, Moein; Esfandyari, Morteza
Stock market index prediction using artificial neural network
Journal of Economics, Finance and Administrative Science, vol. 21, núm. 41, diciembre, 2016, pp. 89-93
Universidad ESAN
Surco, Perú

Available in: http://www.redalyc.org/articulo.oa?id=360748513007

- How to cite
- Complete issue
- More information about this article
- Journal's homepage in redalyc.org

Scientific Information System
Network of Scientific Journals from Latin America, the Caribbean, Spain and Portugal
Non-profit academic project, developed under the open access initiative

Journal of Economics, Finance and Administrative Science 21 (2016) 89–93

Journal of Economics, Finance and Administrative Science

www.elsevier.es/jefas

Article

Stock market index prediction using artificial neural network

Amin Hedayati Moghaddam [a], Moein Hedayati Moghaddam [b,*], Morteza Esfandyari [c]

[a] Department of Chemical and Petroleum Engineering, Sharif University of Technology, Tehran, Iran
[b] Faculty of Managing and Accounting, College of Farabi, University of Tehran, Qom, Iran
[c] Department of Chemical Engineering, Faculty of Engineering, University of Bojnord, ,Bojnord, Iran

ARTICLE INFO

Article history:
Received 27 October 2015
Accepted 1 July 2016

JEL classification:
C5

Keywords:
NASDAQ
ANN
Prediction

ABSTRACT

In this study the ability of artificial neural network (ANN) in forecasting the daily NASDAQ stock exchange rate was investigated. Several feed forward ANNs that were trained by the back propagation algorithm have been assessed. The methodology used in this study considered the short-term historical stock prices as well as the day of week as inputs. Daily stock exchange rates of NASDAQ from January 28, 2015 to 18 June, 2015 are used to develop a robust model. First 70 days (January 28 to March 7) are selected as training dataset and the last 29 days are used for testing the model prediction ability. Networks for NASDAQ index prediction for two type of input dataset (four prior days and nine prior days) were developed and validated.

© 2016 Universidad ESAN. Published by Elsevier España, S.L.U. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

Predicción del índice del mercado bursátil utilizando una red neuronal artificial

Códigos JEL:
C5

Palabras clave:
NASDAQ
ANN
Predicción

RESUMEN

En este estudio se investigó la capacidad de previsión del índice bursátil diario NASDAQ, por parte de la red neuronal artificial (RNA). Se evaluaron diversas RNA proalimentadas, que fueron entrenadas mediante un algoritmo de retropropagación. La metodología utilizada en este estudio consideró como *inputs* los precios bursátiles históricos a corto plazo, así como el día de la semana. Se utilizaron los índices bursátiles diarios de NASDAQ del 28 de enero al 18 de junio de 2015, para desarrollar un modelo robusto. Se seleccionaron los primeros 70 días (del 28 de enero al 7 de marzo) como conjuntos de datos de entrenamiento, y los últimos 29 días para probar la capacidad del modelo de predicción. Se desarrollaron y validaron redes para la predicción del índice NASDAQ, para dos tipos de conjuntos de datos de *input* (los cuatro y los nueve días previos).

© 2016 Universidad ESAN. Publicado por Elsevier España, S.L.U. Este es un artículo Open Access bajo la licencia CC BY-NC-ND (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

In studying some phenomenon, developing a mathematical model to simulate the non-linear relations between input and output parameters is a hard task due to complicated nature of these phenomenons. Artificial intelligent systems such as artificial neural networks (ANN), fuzzy inference system (FIS), and adaptive neuro-fuzzy inference system (ANFIS) have been applied to model a wide range of challenging problems in science and engineering. ANN displays better performance in bankruptcy prediction than conventional statistical methods such as discriminant analysis and logistic regression (Quah & Srinivasan 1999). Investigations in credit rating process showed that ANN has better prediction ability than statistical methods due to complex relation between financial and other input variables (Hájek, 2011). Bankruptcy prediction (Alfaro, García, Gámez, & Elizondo, 2008; Lee, Booth, & Alam, 2005; Baek & Cho, 2003), credit risk assessment (Yu, Wang, & Lai, 2008; Angelini, Di Tollo, & Roli, 2008), and security market applications

* Corresponding author.
E-mail address: moein_hedayati67@yahoo.com (M.H. Moghaddam).

http://dx.doi.org/10.1016/j.jefas.2016.07.002
2077-1886/© 2016 Universidad ESAN. Published by Elsevier España, S.L.U. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

Nomenclature

ANN	artificial neural networks
BPNN	back propagation neural network
RBFNN	radial basis function neural network
FIS	fuzzy inference system
ANFIS	adaptive neuro-fuzzy inference system
MLP	multi-layer perceptron
PNN	probabilistic neural network
GFNN	genetic algorithm based fuzzy neural network
x	input parameter
w_P	connection weight of neuron P
u_P	input combiner
b_P	bias
φ	activation function
y_P	output of the neuron
SCG	scaled conjugate gradient
LM	Levenberg-Marquardt
OSS	one step secant
GDA	gradient descent with adaptive learning rate
GDM	gradient descent with momentum
y(k)	stock price at time k
D(k)	day of week
R^2	determination coefficient
MSE	mean square error
$y_{exp.}$	experimental value
$y_{pred.}$	predicted value

are the other economical areas that ANN has been widely applied. Objective of this study is to investigate the ability of ANN in forecasting the daily NASDAQ stock exchange rate.

2. Background

Guresen, Kayakutlu, and Daim (2011) investigated the performance of multi-layer perceptron (MLP), dynamic ANN, and hybrid ANN models in forecasting the market values. Chen, Leung, and Daouk (2003) used probabilistic neural network (PNN) to predict the direction of Taiwan stock index return. They reported that PNN has higher performance in stock index than generalized methods of moments-Kalman filter and random walk forecasting models. Kuo, Chen, and Hwang (2001) developed a decision support system through combining a genetic algorithm based fuzzy neural network (GFNN) and ANN for stock market. The proposed system was evaluated using the data of Taiwan stock market. Qiu, Liu, and Wang (2012) developed a new forecasting model on the basis of fuzzy time series and C-fuzzy decision trees to predict stock index of shanghai composite index. Atsalakis and Valavanis (2009) developed an adaptive neuro-fuzzy inference controller to forecast next day's stock price trend. They reported the potential ability of ANFIS in predicting the stock index.

3. Artificial intelligent systems used in forecasting

3.1. Artificial neural network

A neural network is a bio-inspired system with several single processing elements, called neurons. The neurons are connected each other by joint mechanism which is consisted of a set of assigned weights.

MLP is a common approach in regression-type problems. MLP network has three layers: input layer, output layer, and hidden layer. Neuron takes the values of inputs parameters, sums them up according to the assigned weights, and adds a bias. By applying the transfer function, the value of the outputs would be determined. The number of neurons in input layer corresponded to the number of input parameters. The architecture of a typical MLP is presented in Figure 1.

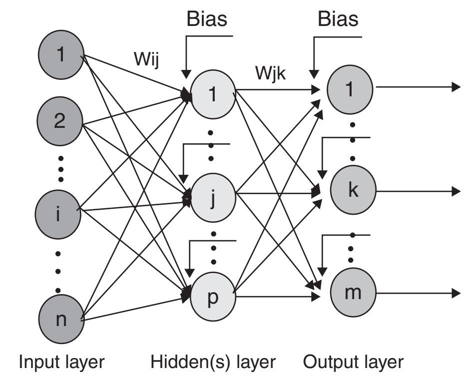

Figure 1. Multi-layer feed forward neural network structure.
Elaborated by the authors.

In mathematical terms, the performance of neuron P can be described as follows:

$$u_P = \sum_{i=1}^{n} w_{Pi} x_i \quad (1)$$

$$y_P = \varphi(u_P + b_P) \quad (2)$$

where $x_1, \ldots, x_n$ are the input parameters; $w_{P1}, \ldots, w_{Pn}$ are the connection weights of neuron P; u_P is the input combiner; b_P is the bias; φ is the activation function; and y_P is the output of the neuron.

In this study feed forward artificial neural networks that were trained by the back propagation algorithm has been used.

There are several learning techniques such as scaled conjugate gradient (SCG), Levenberg-Marquardt (LM), one step secant (OSS), gradient descent with adaptive learning rate (GDA), gradient descent with momentum (GDM) etc. that are using for training and developing the constructed models.

4. Predicting NASDAQ index

The methodology used in this study considered the short-term historical stock prices as well as the day of week as inputs. The overall procedure is governed by the following equation:

$$y(k) = f(y(k-1), y(k-2), y(k-3), \ldots, y(k-n), D(k)) \quad (3)$$

where y(k) is the stock price at time k, n is the number of historical days, and D(k) is the day of week.

Daily stock exchange rates of NASDAQ from January 28, 2015 to 18 June, 2015 are used to develop a robust model. First 70 days (January 28 to March 7) are selected as training dataset and the last 29 days are used for testing the model prediction ability.

For constructing the model, training, and testing procedure MATLAB software R2010a was used. The performance of ANNs was evaluated using the determination coefficient (R^2) and the mean square error (MSE) of the modeled output. R^2 was determined as follows:

$$R^2 = 1 - \frac{\sum \left(y_{\exp.} - y_{pred.}\right)^2}{\sum \left(y_{\exp.} - \bar{y}\right)^2} \quad (4)$$

MSE represents the average squared difference between the predicted values estimated from a model and the actual values. MSE was determined by the following equation:

$$MSE = \frac{\sum (y_{pred.} - y_{\exp.})^2}{M} \quad (5)$$

where $y_{exp.}$ and $y_{pred.}$ were experimental and predicted values, respectively, and M was the total number of data.

5. Result and discussion

In this section several networks for NASDAQ index prediction for two input dataset (four prior days and nine prior days) were developed and validated. Then the optimized network structure for both type of dataset was selected according to their abilities in prediction.

5.1. Four prior working days

In Table 1 the values of R^2 for different training algorithms and transfer function of a BPNN with 20-40-20 neurons in hidden layers have been shown. In experiments 1 through 3, networks were trained by LM, in experiments 4 through 6 by OSS, and in experiment 7 by GDA method. As is shown, applying OSS training method and TANGSIG transfer function resulted in an optimized trained network according to the values of R^2 of validation dataset.

Table 1
The prediction ability of a BPNN with different training and transfer function.

No.	Training function	Transfer function	R^2			
			Train	Test	Validation	Total
1	LM	TANSIG	0.9925	0.9869	0.8864	0.974
2	LM	PURELIN	0.9457	0.9675	0.9027	0.9395
3	LM	LOGSIG	0.9989	0.9698	0.7339	0.9475
4	OSS	LOGSIG	0.9166	0.9133	0.8669	0.9069
5	OSS	PURELIN	0.7016	0.8824	0.8230	0.7675
6	OSS	TANSIG	0.9386	0.8917	0.9408	0.9267
7	GDA	LOGSIG	0.9016	0.8308	0.8497	0.8649

Elaborated by the authors.

Networks with transfer function of TANSIG or PURELIN and training functions of GDA were not able to generate a robust model (not shown). Accordingly, in the next experiments in the current study OSS and TANSIG were selected as training method and transfer function, respectively.

In Table 2 configurations of MLP are presented. The data achieved from 99 days of NASDAQ index were randomly divided into training set (60%), validation set (20%), and testing set (20%). On the basis of the preliminary study, the training method and transfer function were OSS and TANGSIG, respectively. The architecture of the neural network was optimized by applying different values for the number of hidden layers and number of neurons in each hidden layer. Sixteen networks with different architectures

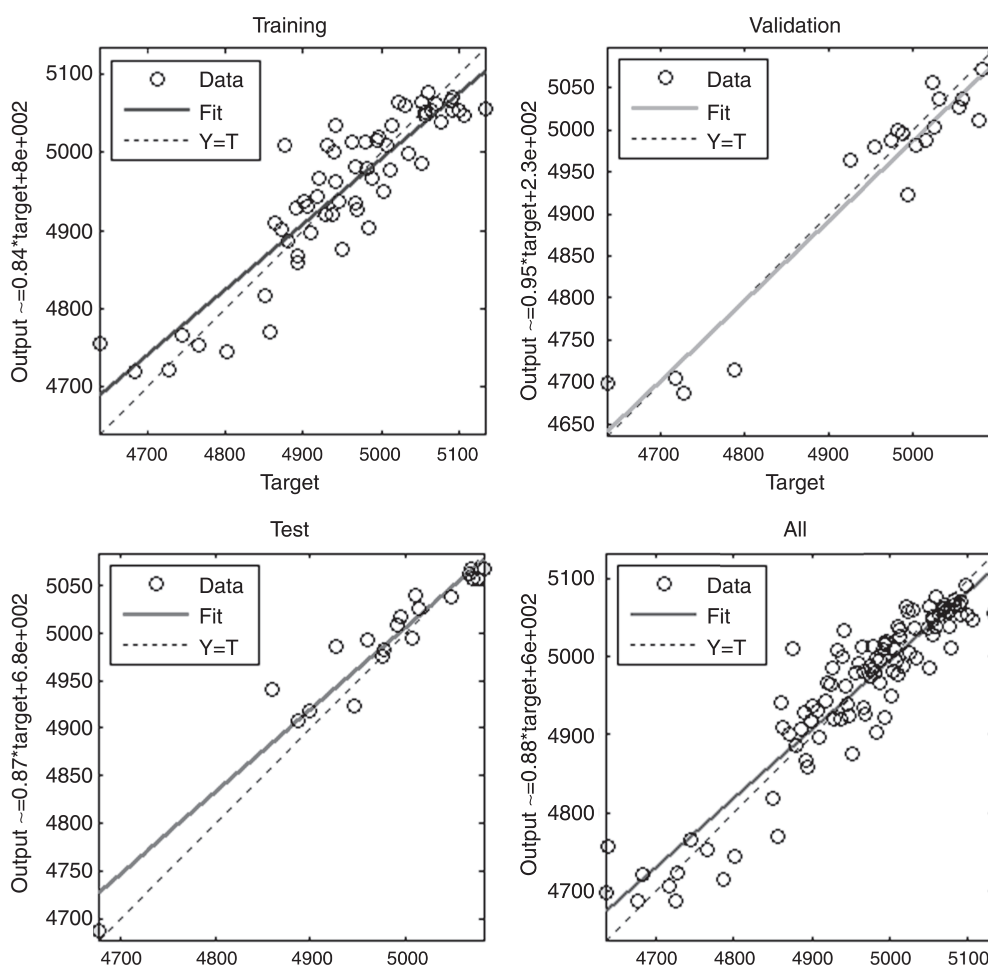

Figure 2. The predicted data against experimental data for training, validation, testing, and total data.
Elaborated by the authors.

Table 2
The R^2 value for BPNN with different structure for four prior days.

No.	Structure	R^2			
		Train	Test	Validation	Total
1	2	0.8177	0.9616	0.9493	0.8692
2	5	0.9250	0.9188	0.9605	0.9264
3	5-5	0.9229	0.9724	0.8631	0.9212
4	5-10	0.2185	−0.0165	−0.2065	0.0838
5	10-10	0.9534	0.9602	0.6811	0.9344
6	10-20	0.9059	0.9758	0.9108	0.9263
7	40-40	0.9003	0.9639	0.9616	0.9264
8	50-100	0.9576	0.9324	0.9393	0.9483
9	100-200	0.9390	0.9466	0.9533	0.9393
10	200-300	0.9267	0.9642	0.8822	0.9276
11	20-40-20	0.9386	0.8917	0.9408	0.9267
12	20-50-20	0.9403	0.9417	0.9077	0.9374
13	50-100-50	0.6837	0.8108	0.7785	0.7326
14	20-40-40-20	0.8990	0.8445	0.8093	0.8739
15	10-20-20-10	0.8977	0.9602	0.9015	0.9109
16	10-20-20-20-10	0.9304	0.9341	0.9456	0.9329

Elaborated by the authors.

Table 3
The R^2 value for BPNN with different structure for nine prior days.

No.	Structure	R^2			
		Train	Test	Validation	Total
1	5	0.8195	0.8480	0.8425	0.8274
2	10	0.8025	0.7113	0.8608	0.7882
3	5-5	0.8437	0.7762	0.8518	0.8280
4	5-10	0.8127	0.8554	0.8262	0.8180
5	20-20	0.8344	0.9116	0.7524	0.8389
6	20-30	0.8859	0.8690	0.8756	0.8707
7	50-100	0.8335	0.9028	0.8300	0.8292
8	200-300	0.8473	0.7899	0.8938	0.8506
9	300-400	0.8476	0.8685	0.7988	0.8435
10	20-50-20	0.8648	0.7937	0.8809	0.8372
11	20-40-20	0.9318	0.8827	0.9622	0.9262
12	50-100-50	0.8697	0.8367	0.8388	0.8552
13	20-40-40-20	0.7761	0.9205	0.8601	0.8188
14	20-50-50-20	0.8359	0.8462	0.8977	0.8443
15	10-20-20-20-10	0.8578	0.8348	0.6704	0.8107

Elaborated by the authors.

were generated, trained, and tested. R^2-values of training set, validation set, and total data were calculated, but only the R^2-value of validation was considered to select the optimized architecture of network. It is found that networks with four hidden layers and more were not able to be trained and to generate a robust model (these networks were not shown). As seen in Table 2, R^2 had desirable values (maximum value) when the number of hidden layers was 2 and the numbers of neurons in hidden layers were 40. It is worthwhile noting that any changes in number of neurons would influence the model proficiency. For example, as seen in Table 2 although a network with 5-5 had acceptable R^2 validation (0.8631) but a network with 5-10 neurons had poor prediction ability.

Figure 2 shows the predicted data generated by the optimized BPNN (two hidden layer with forty neurons) against the observed NASDAQ index for training, validation, testing, and total data. Figure 3 shows the real and predicted NASDAQ index values for four prior days in 99 days from 28 January to 18 June 2015.

5.2. Nine prior working days

Similar to four prior days, the values of R^2 for different training algorithms and transfer function of a MLP with 20-40-20 neurons in hidden layers have been generated and tested. Accordingly, applying OSS training method and LOGGSIG transfer function resulted in an optimized trained network according to the values of R^2 of validation dataset (0.9622).

In Table 3 several configurations of MLP are presented. The training method and transfer function were OSS and LOGSIG, respectively. It is found that a network with three hidden layers and 20-40-20 neurons in hidden layers was the optimized network. Figure 4 shows the real and predicted NASDAQ index values for nine prior days in 99 days from 28 January to 18 June 2015. Accordingly, there is no distinct difference between the prediction ability of the four and nine prior working days as input parameters.

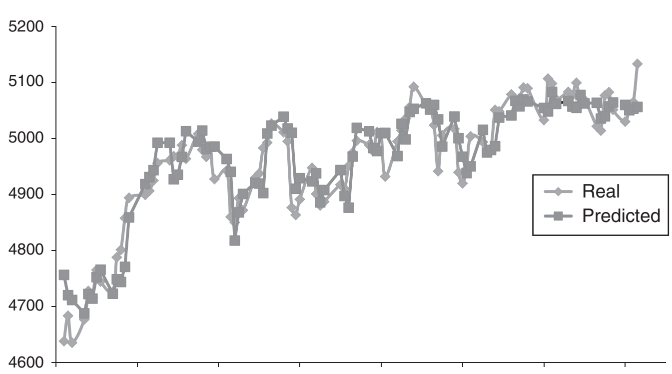

Figure 3. Real and predicted NASDAQ index values for four prior days.
Elaborated by the authors.

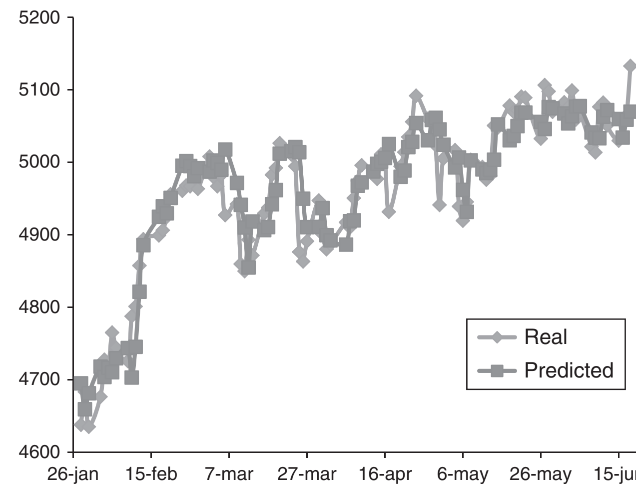

Figure 4. Real and predicted NASDAQ index values for nine prior days.
Elaborated by the authors.

6. Conclusion

The model uses the values of NASDAQ exchange rate of last four and nine working days as well as the day of week as the input parameters. For four prior working days, applying OSS training method and TANGSIG transfer function in a network with 20-40-20 neurons in hidden layers resulted in an optimized trained network with R^2 values of 0.9408 for validation dataset. For this dataset, the maximum R^2 values for the networks with OSS training method and TANGSIG transfer function would be obtained when the number of hidden layers was 2 and the number of neurons was 40-40. For nine prior working days a network with 20-40-20 neurons in hidden layers OSS training method and LOGSIG transfer function are the optimized network with validation R^2 of 0.9622. The model outputs show that there is no distinct difference between the prediction ability of the four and nine prior working days as input parameters.

References

Alfaro, E., García, N., Gámez, M., & Elizondo, D. (2008). Bankruptcy forecasting: An empirical comparison of AdaBoost and neural networks. *Decision Support Systems*, *45*, 110–122.

Angelini, E., Di Tollo, G., & Roli, A. (2008). A neural network approach for credit risk evaluation. *The Quarterly Review of Economics and Finance*, *48*, 733–755.

Atsalakis, G. S., & Valavanis, K. P. (2009). Forecasting stock market short-term trends using a neuro-fuzzy based methodology. *Expert Systems with Applications*, *36*, 10696–10707.

Baek, J., & Cho, S. (2003). Bankruptcy prediction for credit risk using an auto-associative neural network in Korean firms. In *Computational Intelligence for Financial Engineering, 2003. Proceedings. 2003 IEEE International Conference, 25-29. IEEE*.

Chen, A. S., 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.

Guresen, E., Kayakutlu, G., & Daim, T. U. (2011). Using artificial neural network models in stock market index prediction. *Expert Systems with Applications*, *38*, 10389–10397.

Hájek, P. (2011). Municipal credit rating modelling by neural networks. *Decision Support Systems*, *51*, 108–118.

Kuo, R. J., Chen, C., & Hwang, Y. (2001). An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network. *Fuzzy Sets and Systems*, *118*, 21–45.

Lee, K., Booth, D., & Alam, P. (2005). A comparison of supervised and unsupervised neural networks in predicting bankruptcy of Korean firms. *Expert Systems with Applications*, *29*, 1–16.

Qiu, W., Liu, X., & Wang, L. (2012). Forecasting shanghai composite index based on fuzzy time series and improved C-fuzzy decision trees. *Expert Systems with Applications*, *39*, 7680–7689.

Quah, T. S., & Srinivasan, B. (1999). Improving returns on stock investment through neural network selection. *Expert Systems with Applications*, *17*, 295–301.

Yu, L., Wang, S., & Lai, K. K. (2008). Credit risk assessment with a multistage neural network ensemble learning approach. *Expert Systems with Applications*, *34*, 1434–1444.