23rd ABCM International Congress of Mechanical Engineering
December 6-11, 2015, Rio de Janeiro, RJ, Brazil

QUALITATIVE ANALYSIS OF ARTIFICIAL NEURAL NETWORKS APPLIED TO FORECAST TRENDS IN THE BRAZILIAN STOCK MARKET

Gustavo José Corrêa Gonçalves
Eduardo Paciencia Godoy
GASI – Group of Automation and Integrated Systems, UNESP - Universidade Estadual Paulista
Av. Três de Março, 511, CEP 18087-180, Sorocaba, São Paulo, Brazil
gustavojcgoncalves@hotmail.com ; epgodoy@sorocaba.unesp.br

Abstract. *The interest over the stock market has increased in the last decade, once it can offer considerable profits, from very short to long term. However, the stocks graphical analysis is not an exact technique, so subjective is present in it, depending on the person who performs the analysis. Thus, the use of Artificial Neural Networks is prominent when applied to predict movements of stocks traded on the stock exchange. This work proposes the use of Nonlinear Autoregressive Neural Networks with Exogenous Inputs to predict trends over daily close price of preferred shares of Itaú Bank. Furthermore, the influences of configuration parameters of the network over the predictions are analyzed.*

Keywords: *Artificial Neural Networks, NARX, Stock Market*

1. INTRODUCTION

The stock market always have been seen with some fear by people, once it can cause serious financial loss. However, the possibility of obtaining high profits have been attracting lots of investors, specially when it is compared to other type of investiments in Brazil. (Herling, 2012). So, forecast the moviments of the market always have been an interesting subject among speculators. (Dhar *et al.*, 2010; Li and Liu, 2009) and of major importance for market researchers and regulators (Shen and Xing, 2009; Lee, 2004).

Predict the future price of financial assets is not an easy task (Iuhasz *et al.*, 2012). Once the financial markets are controlled by people, their investment decisions are strongly influenced by emotions (Dhar *et al.*, 2010). Thus, macroeconimic decisions, politics and other nature influences plays an important role over the stock market fluctuations (Yixin *et al.*, 2010; Shen and Xing, 2009). So, every financial asset data contains noisy signals with nonlinear characteristics (Yixin *et al.*, 2010; Yu *et al.*, 2009; Sun and Li, 2010). Moreover, the variables with great influence over the stock market have a complex and changeable relationship with each other (Coelho *et al*, 2008).

Several statistical techniques to forecast data from time series have been used, as such the GARCH model (*Generalized Autoregressive Conditional Heterocedasticity*), the SV model (*Stochastic Volatility*) and the ARIMA model (*Autoregressive Integrated Moving Average*) (Shen and Xing, 2009). However, the results were not good enough to guide people's investment decisions (Yixin *et al.*, 2010; Coelho *et al*, 2008) mostly because the classic statistical models could not deal well with dynamic systems and complex modeling systems (Li and Liu, 2009). Besides, these models didn't have the learning capability or the extrapolation skill, which made then very limited in the stock market forecast application (Shen and Xing, 2009).

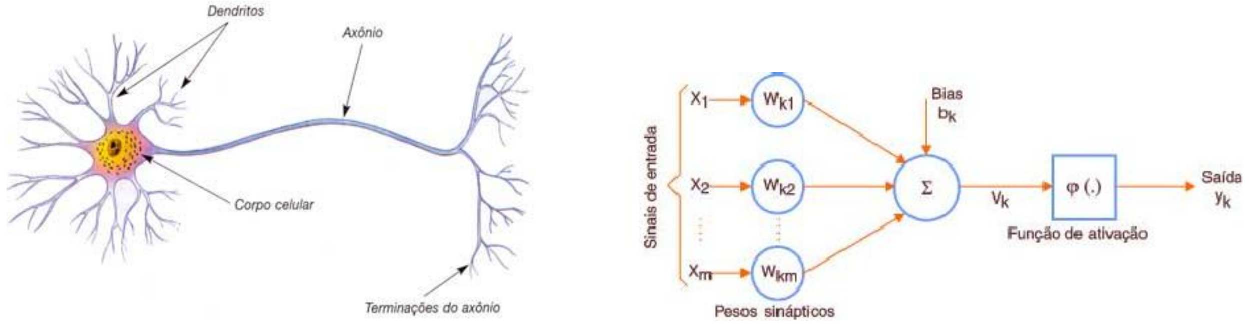
Due to the incapability of those classic statistical methods in solving the problem, researchers used Artificial Intelligence (AI) techniques, trying to find a better solution to this problem (Herling, 2012). The most used AI techniques are the Artificial Neural Networks (ANN), Fuzzy Logic (FL) and Genetic Algorithms (GA) (Shen and Xing, 2009). The ANN are used in problems which demands learning skills, especially in dynamic environments (Nguyen *et al.*, 2013; Sun and Li, 2010). So, the number of papers which used these AI techniques to forecast the financial markets has increased in the last few years (Wurges and Borba, 2010), which had good results (Li and Liu, 2009).

So, this paper proposes the use of the ANN to forsee trends in the stock market. Once the technical analysis, used by speculators to try to predict price movements, are too subjective, the use of an ANN can reduce its subjectivity. Also, several tests were conducted in order to evaluate the influence of the ANN settings over the predictions.

2. ARTIFICIAL NEURAL NETWORKS

The Artificial Neural Networks are networks with multiple layers composed by neurons (Yixin *et al.*, 2010) which resemble its structure to the human brain, in order to achieve brain's parallel processing skill (Dhar *et al.*, 2010). The ANN neurons are the basis to the whole network and they are responsible for the basic processing of the knowledge (Coelho *et al.*, 2008).

Fugure 1a shows a scheme of the biological neuron. With about 10.000 synaptic contacts, the neurons are stimulated through the dendrites. Inside it, chemical and electrical signals run through the cell body and go towards the axon, since certain excitation threshold is overcome. Thus, the axon is divided in several ramifications, which are chemically connected to other neuron's dendrites, transmitting the information through the synapse (Haykin, 2008).



(a) (b)

Figure 1. (a) Representation of biological neuron (Estevam, 2013). (b) Model of an artificial neuron (Mecatrônica Atual, 2006).

Figure 1b shows an artificial model of the neuron, which dendrites are represented by the input signals X_1, X_2, ... X_m. The relation between the dendrites of one neuron and the axon of another neuron is measured by the synaptic weight, generically expressed by W_{km}. The bigger this relationship, the higher the importance of a certain input to the neuron. All the input data are processed at the same time by the neuron through the sum (∑). The value of b_k is the biological neuron's excitation threshold, so that the electrical signal only will be propagated if this value is reached or exceeded. Lastly, the activation function $\varphi(.)$ will modify the value before send it to the output, which represents a ramification of the biological neuron's axon. Thereby, the artificial neuron can be mathematically modeled, as follows in Eq. (1) and Eq. (2) (Haykin, 2008), in which V_k is the weighted input of the neuron and Y_k is the neuron output:

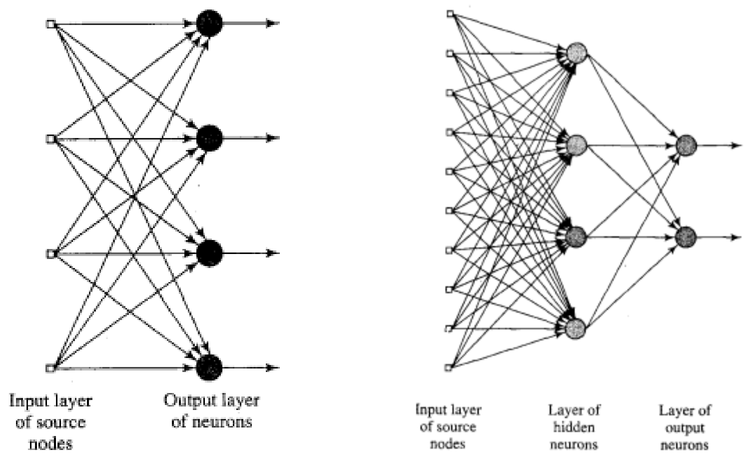
$$V_k = \sum_{j=1}^{m}(W_{kj}\, X_j) \tag{1}$$

$$Y_k = \varphi\,(V_k + b_k) \tag{2}$$

2.1 Feedforward networks

The feedforward neural networks are the most used, once it can be applied to several type of problems such as functions approximation and predictions (Razavi and Tolosn, 2011). Its constructive characteristic makes information always to be propagated forward, so, output data are not considered in other places in the network (Haykin, 2008).

Figure 2a shows a feedfoward neural network structure with only the output layer:



(a) (b)

Figure 2. (a) Feedforward neural network with a single layer. (b) Feedforward network fully connected, with a single hidden layer. (Haykin, 2008)

Although this configurations of ANN are able to solve that type problems, it is proven that the same network with an additional hidden layer is more efficient when applied to more complex problems. Although, the use of more than one hidden layer makes the network less efective (Razavi and Tolosn, 2011), unlike what would seem logical at first. Above, Fig. 2b shows the structure of a feedforward network with a single hidden layer:

2.2 Predictive networks

The predictive neural networks are divided in two groups: the non-recurrent networks and the recurrent networks (Haykin, 2008). The recurrence of a network relates to wheather the network receives an output feedback from itself.

2.2.1 Non-recurrent predictive network

Although the feedforward networks are effective to several problems, when the application is the prediction of time series, specific neural networks has to be used. Figure 3 shows a structural modification of the feedforward network:

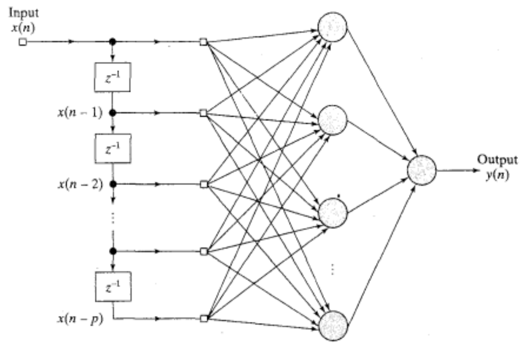


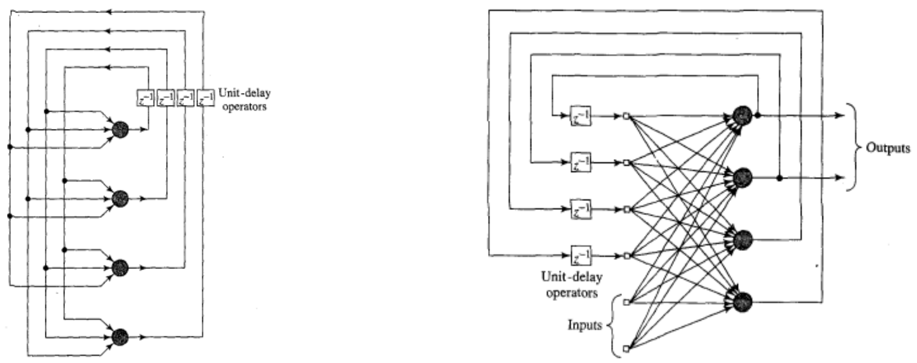
Figure 3. Focused time delay feedforward network (Haykin, 2008).

This modification, which adds time delays over the inputs, aims to train the network using p past values. The unit time delays, represented by z^{-1}, makes this network, also known as FTDNN (*Focused Time Delay Neural Network*), be able to solve time varying problems more efficiently (Kim and Shin, 2007).

2.2.2 Recurrent predictive networks

The recurrent predictive networks use its own predicted values to feed back the neurons of the input layer. The presence of the feed back has a great influence over the network learning skill and performance (Haykin, 2008).

Figure 4a shows a non-autoregressive recurrent network, which a neuron is never fed back with its own output. Figure 4b presents an autoregressive neural network, which each neuron is fed back for its own output and for other neurons outputs:



(a) (b)

Figure 4. (a) Non-autoregressive recurrent network without hidden layers. (b) Autoregressive neural network with exogenous inputs. (Haykin, 2008)

Thus, this autoregressive network uses inputs which are out of the feedback cycle. This inputs are called exogenous inputs. So, this network is called NARX (*Nonlinear Autoregressive with Exogenous Inputs*).

The exougenous inputs of the NARX network are variables which has direct influence or reflect the behavior of the variable to be predicted. So, the exogenous inputs are used to increase the predictive capacity of the network (Haykin, 2008).

3. METHODOLOGY

Setting an ANN goes through several stages: choose the type of the network, the number of layers, the number of neurons in which layer (Nguyen *et al.*, 2013), the activation function of the neurons of each layer, the data which will be used for training, the training method, the learning rate, the stop training criteria and the data which will be used for validation.

However, the parameter which influence the most the network performance are the training data (Dhar *et al.*, 2010). An ANN which was not accordingly trained will not be able to do good predictions. Thus, if a ANN is overly trained, it can only memorize the data and not learn it. If that happens, the ANN will not be able to generalize and perform good predictions (Dhar *et al.*, 2010). On the other hand, if the training data poorly represents the system behavior, the ANN will not be capable of learning its function (Vieira, 2013).

Taking all theses factors into account, the settings used during this project are presented.

3.1 Historical training data

The prediction universe of the trained neural network was defined in five days. So, it will only be able to predict short-term trends.

Based on that, the daily closing price of the preferred shares of the Itaú bank (negotiation code ITUB4), Bradesco bank (negotiation code BBDC4) and the BOVESPA Index (negotiation code IBOV), which is the brazilian stock market main index, were used for training.

The used price series starts at 01/01/2011 and it ends at 10/20/2014, totaling 956 data points of the serie. The data source was the financial specialized website Yahoo! Finanças (http://finance.yahoo.com), which offers the historical series in a format more accessible than the market itself (BM&F BOVESPA).

It was used 630 data points for the training, 135 data points for validation and 135 data points for test, randomly selected. For the evaluation performed in this work, the latest 56 data points of each series were used.

All data was manipulated using the software MATLAB®, R2013a version.

3.2 Structure of the Neural Network

Based on the results of several works related to the prediction of time series using NARX networks (Soman, 2008; Mitrea *et al.*, 2009; Jr and Barreto, 2006), this was the structure chosen to develop this work.

The Neural Network Toolbox™ which is available within MATLAB® was used to create, train and test the NARX network presented in this work. Figure 5 shows one of several NARX networks that were used, with five delays and ten neurons in the hidden layer, in which *y(t)* is the predicted variable and *x(t)* is the exogenous inputs.

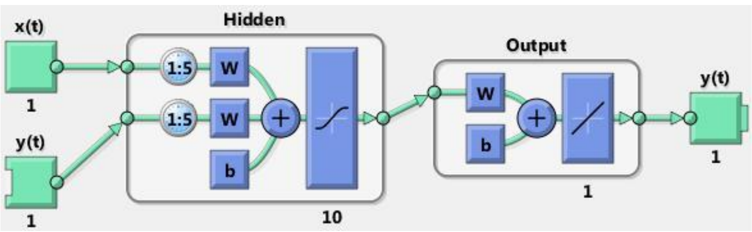


Figure 5. NARX network with five time delays. Taken from the software MATLAB®.

In all experiments, the neural network goal was to predict the closing price of the ITUB4 shares within a horizon of five days. For this porpouse, the BBDC4 data or IBOV data was used as exogenous inputs, as indicated in each result.

According with studies made for this work, the ITUB4 and BBDC4 shares have similar movements regarding to the shape and intensity. It can be explained because them both belongs to the same business area, the financial one. Therefore, macroeconomic changes will affect both companies in similar ways, which will be reflected in the prices of its shares.

The BOVESPA Index was used as a exogenous input once it has great influence over the brazilian stocks. So, the use of more than one exogenous input (not at once) aimed to compare which one would be better to the predictions.

The number of hidden layers was fixed in one and the number of neurons in the hidden layer was variable, so the influence of this parameter over the predictions could be verified. The activation function used at the hidden layer was the hyperbolic tangent, while the linear function was used for the output layer.

3.3 Neural network training

Regarding possible training methods, the most used is the Error Backpropagation (EB) (Yixin *et al.*, 2010), which brings excellent results in training several types of neural networks (Li, 2010).

The EB algorithm seeks the minimum point of the error gradient in relation to the synaptic weights. So, for a certain neuron k of the layer j of a neural network, it is known that the output error, $e_k(n)$, in a certain epoch n, is the difference between the desired value for the output, $d_k(n)$, and the actual value calculated by the network, $y_k(n)$:

$$e_k(n) = d_k(n) - y_k(n) \tag{3}$$

In order to measure this error at each training epoch, is defined the Mean Square Error (MSE), $\mathcal{E}(n)$, as shown in Eq. (4):

$$\mathcal{E}(n) = \frac{1}{2} e_k{}^2(n) \tag{4}$$

As training epochs advance, the synaptic weights, w_{kj}, are updated in order to reach the minimum value of the MSE function gradient, in which η is the learning rate of the network. Thus:

$$w_{kj}(n+1) = w_{kj}(n) + \Delta w_{kj}(n) \tag{5}$$

$$\Delta w_{kj}(n) = -\eta \frac{\partial \mathcal{E}}{\partial \mathrm{w}} \tag{6}$$

Once is not possible to directly relate the MSE with the synaptic weights, it is done indirectly, once the MSE is a function of the error of the network, which is a function of the network's output, which depends on the output of a certain neuron k, u_k, which varies according to this neuron's synaptic weights. Therefore:

$$\mathcal{E} = f(e(y(u(w)))) \tag{7}$$

$$\nabla\mathcal{E}(w) = \frac{\partial \mathcal{E}}{\partial \mathrm{w}} = \frac{\partial \mathcal{E}}{\partial \mathrm{e}} \frac{\partial \mathrm{e}}{\partial \mathrm{y}} \frac{\partial \mathrm{y}}{\partial \mathrm{u}} \frac{\partial \mathrm{u}}{\partial \mathrm{w}} \tag{8}$$

Despite being a very effective method, the EB algorithm can have some problems, such as train the network in a way that the training stops at a local minimum point of the MSE gradient function (Li, 2010). Thus, the Levenberg-Marquardt (LM) training algorithm, which is also based in the descendent gradient method such as the EB algorithm, is a great option to train a network, in way to avoid training errors that can occour using the EB algorithm (Costa, 2012). So, the LM algorithm was chosen as default to train the tested neural networks.

The learning rate, η, sets the speed in which the network will apdate its synaptic weights, as shown in Eq. (6), covering the surface of MSE function. Thus, setting a small value for η makes the learning process slower and, in some cases, more efficient. A high value for η makes the learning process faster, but it can cause instability and avoid finding the minimum point of the MSE function (Haykin, 2008). Therefore, a low learning rate was used, with a default value of 0,01. For the training stop criteria, the default MATLAB® values were maintained.

3.4 Evaluation criteria

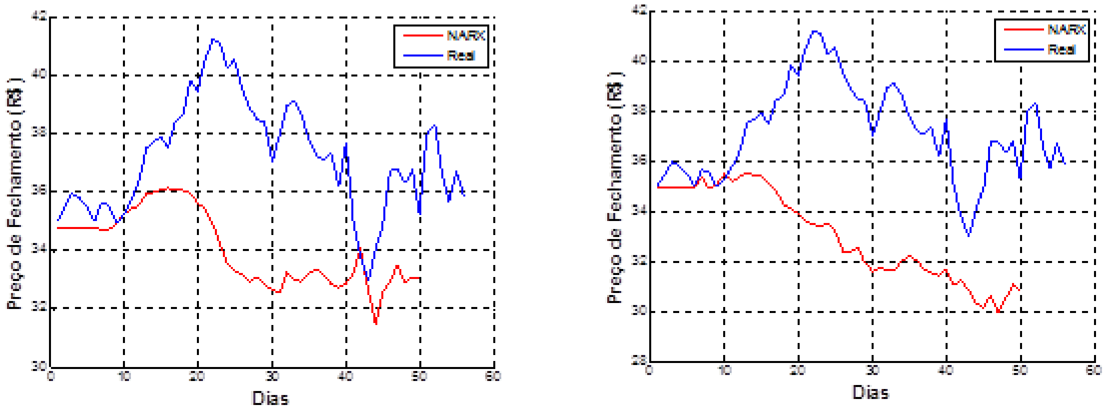
According to the Eq. (3) and Eq. (4), the MSE varies with the neural network output. However, the neuron outputs are influenced by the synaptic weights between the neurons, according to Eq. (1). So, before start the network training, it is necessary to establish the initial values for the synaptic weights in order to calculate the MSE and, then, begin the process towards the downward gradient of this function. Thus, the EB-based training algorithms randomly creates initial synaptic weights. It makes each training of the network being unique.

As the training of a neural network produces different networks, even when the settings and data are kept constant, it is necessary to train several different networks in order to find the best of them. So, ten neural networks were trained using the same settings and data inputs in order to verify which one was better. To do so, it was verified linear regression, Mean Square Error and correlation, besides the outputs of each network compared to the actual market data.

4. RESULTS

The performed tests were conducted to verify the efficiency of the NARX network in predicting stock market trends and also the influence of the network settings over results. In all graphs, the closing price are at y-axis and the time (days) is in x-axis. The blue line represents the actual closing price of the ITUB4 shares. The red line represents the NARX network output.

In Fig. 6a, in which BBDC4 data was used as exogenous input, the short-term prediction was efficient.



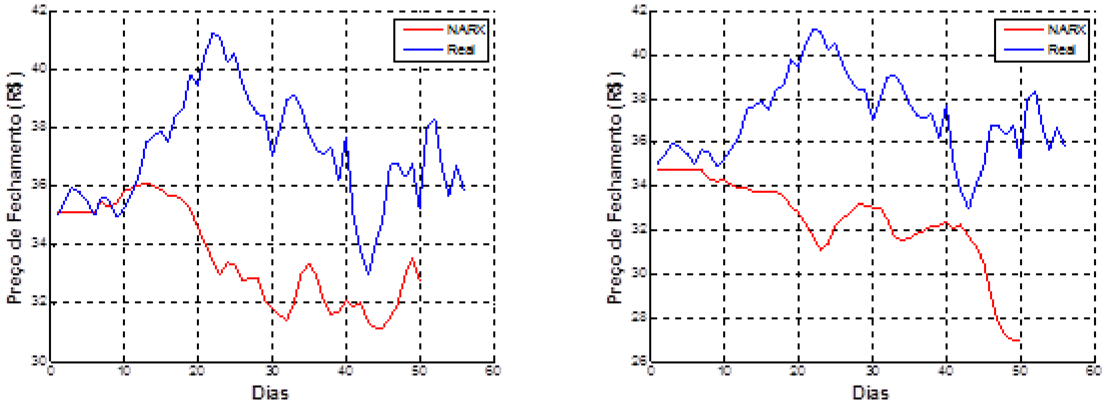
(a) (b)

Figure 6. (a) NARX network trained using BBDC4, 10 neurons in the hidden layer and η = 0,01. (b) NARX network trained using BBDC4, 10 neurons in the hidden layer and η = 0,01.

Figure 6a shows that after the 13th day, the network has correctly predicted the end of the uptrend for the next five days and has indicated a downtrend, which has actually occurred. From the 40th day, the network has indicated a minor uptrend, but then it started just following the price movements, which may be caused by an unexpected movement.

Figure 6b shows the NARX network output when trained with the very same parameters. As can be seen, the results were different, proving what was previously said about the EB algorithm and its synaptic weights inicialization. This new network was also successful in predicting the inicial downtrend, in spite of not being able to predict the exact closing price. These results reinforces the necessity to look for the best network among those which have been trained with the same parameters.

In Fig. 7a, the network was trained using 20 neurons in the hidden layer. It was capable of predict the downtrend, but results showed that, considering all trained networks, an increase on the number of neurons in the hidden layer cause worse results when compared to the first case. At Fig. 7b, have been used five neurons in the hidden layer:

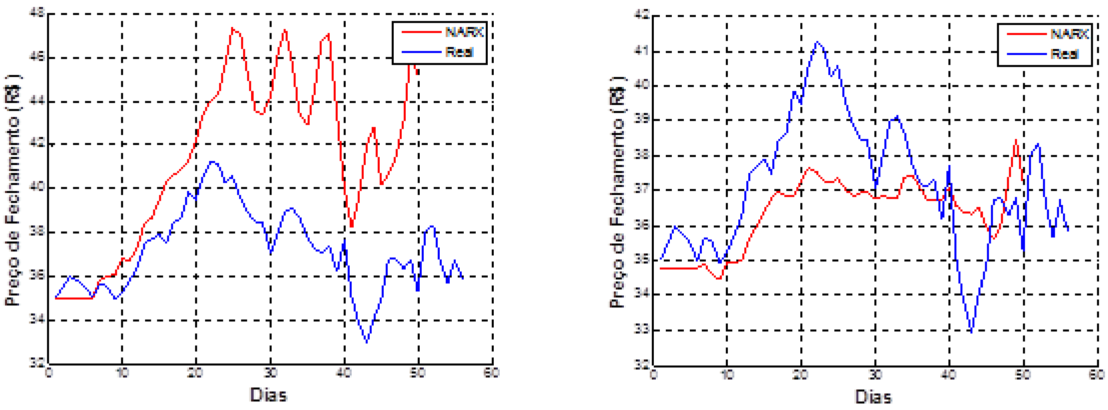


(a) (b)

Figure 7. (a) NARX network trained using BBDC4, 20 neurons in the hidden layer and η = 0,01. (b) NARX network trained using BBDC4, 5 neurons in the hidden layer and η = 0,01.

It was observed that the reduction in the number of neurons caused efficient networks more frequently than when 20 neurons were used, once the results had less oscillations.

Using a learning rate 100 times higher, η = 1, the results can be seen in Fig. 8a. It was possible to infer that the use of a high learning rate makes the training very difficult, which brings inefficient networks for most cases. As can be seen, the neural network was not capable to predict the price trends.



(a) (b)

Figure 8. (a) NARX network trained using BBDC4, 10 neurons in the hidden layer and η = 1. (b) NARX network trained using BBDC4, 10 neurons in the hidden layer and η = 0,0001.

On the other hand, when η is reduced in 100 times, fixed in 0,0001, the trainings occured normally. But the trained networks could not make efficient predictions, which is shown in Fig. 8b.

Lastly, the exogenous input, which helps the network with the predictions, was changed. In this case, the IBOVESPA Index was used. It was possible to infer that the use of IBOV was not so efficient as the BBDC4 shares. The network could not predict any trend within the ITUB4 shares, as can be seen in Fig. 9.

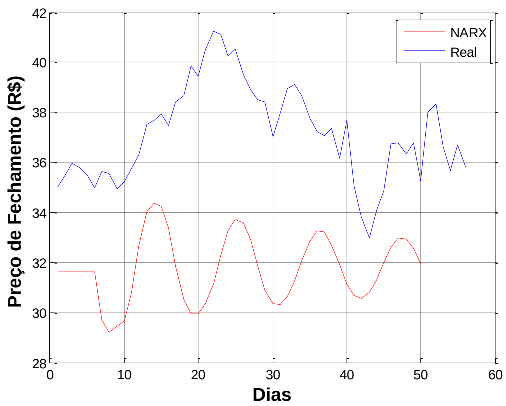


Figure 9. NARX network trained using IBOV, 10 neurons in the hidden layer and η = 0, 01.

5. CONCLUSIONS

The results showed that the NARX network is capable to foresee trends in price of the shares negotiated at the stock market. However, the exactly price prediction was not reached.

Based on the experiment results, it was possible to verify that the exogenous input plays an important role on NARX network predictions. So, it is very important to evaluate the relation among the regular input data and the exogenous inputs data, even before defining the settings for the neural network.

Regarding the network characteristcs, it was observed that the increase of the number of neurons in the hidden layer damaged the results, unlike was observed with the increase in this number. This statement is valid only for this work, once the increment on the number of inputs or the increase of the total data points used for training could make the problem too complex. If that is the case, the increase on the number of neurons could be good or even necessary to the network be able to learn the function.

Lastly, the variation of the learning rate directly influenced the predictive ability of the network. Using a very high rate, the network was not able to correctly learn the function. A very small rate, although better, was not efficient either.

Therefore, it is possible to affirm that the use of NARX networks applied to foresee trends in the stock market is possible, since it is trained accordingly. However, despite efficient, the structures of the neural networks presented here does not eliminate the subjectivity present in graphical analysis of stocks. So, there is the need of understating the basic concepts of the technical analisys in order to use the NARX network predictions efficiently, supplementing the analysis.

6. REFERENCES

Coelho, L. S, Santos, A. A. P and Costa JR., N. C. A., 2008. "Podemos Prever a Taxa de Cambio Brasileira? Evidência Empírica utilizando Inteligência Computacional e Modelos Econométricos". *Gestão & Produção*. São Carlos, v. 15, n. 3, dez.

Costa, A., 2012. "Previsão de Séries Temporais Financeiras por meio de Redes Neurais Dinâmicas e Processos de Transformação de Dados: Abordagem Empírico-Comparativa". 94 f. Master's thesis. - FEB, UNESP, Bauru.

Dhar, S, Mukherjee, T. and Ghoshal, A. K. 2010. "Performance Evaluation of Neural Network Approach in Financial Prediction: Evidence from Indian Market". In: *Proceedings of International Conference on Communication and Computational Intelligence.* p. 597-602.

Estevam, A. 2014. "Cérebro e Memória". 03 Nov. 2014 <http://www.rc.unesp.br/biosferas/mat0002.php>.

Haykin, S. O., 2008. *Neural Networks and Learning Machines.* 3rd ed. Pearson Education.

Herling, L. H., Lima, M. V., Moritz, G. O. and Marangoni, P. H., 2012. "Testando a Hipótese de Eficiência de Mercado por meio de RNAs: Um Estudo de Caso com as Dez Principais Ações do Ibovespa do Primeiro Quadrimestre de 2011". *Future Studies Research Journal.* São Paulo, v.4, n.2, p. 164-184, jul/dez.

Iuhasz, G., Tirea, M. and Negru, V., 2012. "Neural Network Predictions of Stock Price Fluctuations". In: *International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*. 14th, Timisoara. p. 505-512.

Jr, J. M. P. M. and Barreto, G. A., 2006. "A New Look at Nonlinear Time Series Prediction with NARX Recurrent Neural Network". In: *Brazilian Neural Networks Symposium*, IX, Fortaleza.

Kim, H. and Shin, K., 2007. "A hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns in stock markets". *Applied Soft Computing*, v. 7, n. 2, p.569-576.

Li, F., 2010. "Research on the Estimating Model of the Stock Market Price Based on the LM-BP Neural Network". In: *International Conference on Genetic and Evolutionary Computing*, 4th, 2010, Shenzehen. p. 562-565.

Li, F. and Liu, C., 2009. "Application Study of BP Neural Network on Stock Market Prediction". In: *International Conference on Hybrid Intelligent Systems*, 9th, Shenyang. Anais. p. 174-178.

Mecatrônica Atual, 2006 "Reconhecimento de trajetória com redes neurais". 04 nov. 2014. <http://www.mecatronicaatual.com.br/educacao/1580-reconhecimento-de-trajetria-com-redesneu rais>.

Mitrea, C. A, Lee, C. K. M. and Wu Z., 2009. "A Comparison Between Neural Networks and Traditional Forecasting Methods – A case study". In: *International Journal of Engineering Business Management*, v. 1, n. 2 p. 19-24.

Nguyen, T, Khosravi, A, Nahavandi, S. and Creigton, D., 2013. "Neural Network and Interval Type-2 Fuzzy System for Stock Price Forecasting". In: *IEEE International Conference on Fuzzy Systems.* Hyderabad, p. 1-8.

Razavi, S. and Tolosn, B. A., 2011. "A New Formulation for Feedforward Neural Nets". *IEEE Trans. on Neural Networks.* v. 22 n. 10. p. 1588-1598.

Shen, W. and Xing, M., 2009 "Stock Index Forecast with Back Propagation Neural Network Optimized by Genetic Algorithm". *International Conference on Information and Computing Science*, 2nd, Manchester. p. 376-379.

Silva, R. S., 2010. "Previsão de venda de produtos em uma indústria de telecomunicação utilizando redes neurais artificiais". 2010. 78 f. Thesis Computação Aplicada, Universidade do Vale do Rio Dos Sinos, São Leopoldo.

Soman, P. C., 2008. "An Adaptive NARX Network Approach for Financial Time Series Prediction". 104 f. Master's thesis. Universidade Estadual de Nova Jersey, Nova Jersey.

Sun, B. and Li, T., 2010. "Forecasting and Identification of Stock Market based on Modified RBF Neural Network". *IEEE International Conference on Industrial Engineering and Engineering Management*, 17th, Xiamen. p.424-427.

Vieira, C., 2013. "Forecasting Financial Markets with Artificial Neural Networks". 2013. 37 f. Master's thesis. School of Economics & Management, Instituto Superior de Economia e Gestão, Lisboa.

Wuerges, A. and Borba, J., 2010. "Redes Neurais, Lógica Nebulosa e Algoritmos Genéticos: Aplicações e Possibilidades em Finanças e Contabilidade". *Revista de Gestão da Tecnologia e Sistemas*, v.7, n.1, p. 163-182.

Yixin, Z. and Zhang, J., 2010. "Stock Data Analysis Based on BP Neural Network". In: *International Conference on Communication Software and Networks*, 4th, 2010. Anais. IEEE Computer Society. p. 396-399.

Yu, L., Chen, H., Wang, S. and Lai, K., 2010. "Evolving Least Squares Support Vector Machines for Stock Market Trend Mining". *IEEE Trans. Evolutionary Computation.* v. 13, n. 1, p. 87-102.

7. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.