International Journal of Academic Research in Accounting, Finance and Management Sciences
Vol. 4, No.4, October 2014, pp. 307–316
E-ISSN: 2225-8329, P-ISSN: 2308-0337

www.hrmars.com

Forecasting Euro and Turkish Lira Exchange Rates with Artificial Neural Networks (ANN)

Olcay ERDOGAN[1]
Ali GOKSU[2]

[1,2]International Burch University, 71210 Sarajevo, Bosnia and Herzegovina
[1]E-mail: olcay.erdogan@bosnasema.ba, [2]E-mail: ali.goksu@ibu.edu.ba

Abstract *Forecasting of financial data has been a field of research since the efficiency of prediction is essential for strategical decision making. Forecasting exchange rates is not a simple task because it is influenced by many factors and linear models are not able to capture nonlinear relationships in the data. Therefore ANNs have been used in financial forecasting problems since it is capable of handling complex data. The aim of this study is to consider predictive accuracy of ANNs with normalized back propagation using the historical Euro and Turkish Lira (EUR/TRY) exchange rates. The data is obtained from CBRT (Central Bank of the Republic of Turkey) over the period 2010-2013. Several factors affect the accuracy of neural network in the implementation process. Various structures are built by changing the number of neurons, transfer functions and learning algorithms to acquire higher performance. This empirical research has been a comparative study of accuracy in different ANN architectures also in different time horizons. The results are evaluated by MSE (Mean Squared Error) values of each case and it has been found out that ANNs can closely forecast the future EUR/TRY exchange rates.*

Key words ANN, Backpropagation, Exchange Rate Forecasting, Financial Time Series

DOI: 10.6007/IJARAFMS/v4-i4/1361 URL: http://dx.doi.org/10.6007/IJARAFMS/v4-i4/1361

1. Introduction

The currency exchange market (FOREX) has its own irregularity. Foreign exchange rates are influenced by many economic, political and psychological factors. Forecasting exchange rates is not a simple task because of these unstable factors. Investors trade currencies to benefit from fluctuations of exchange rates. Efficient predictions are quite important for decision making of future investments. Researchers have developed many models for forecasting time series like HMM (The Hidden Markov Model), GLAR (Generalized linear auto-regression), ARIMA (Auto Regressive Integrated Moving Averages. These linear methods are easy to implement and they are also relatively simple to understand. But linear models are not able to capture any nonlinear relationships in the data (Zhang, 2003).

The first forecasting approach is reported in (Theil, 1966). The Box–Jenkins forecasting (Box and Jenkins, 1970) based on auto-regression integrated moving-average (ARIMA) models is specified as an effective method. The forecasting of financial activities reported are interest rate forecasting exchange rate forecasting stock market forecasting and bankruptcy prediction. Since the linear models are not able to analyze nonlinear relationships ANN is a well suited model for the time series data. New types of neural networks are developed to get higher accuracy; these models are RNN, FLANN, FLANN-KR, CFLANN, PSN, RBF and many others. These models have superior efficiency and forecasting seems promising with these newly introduced models.

Therefore, we utilized ANN (Artificial Neural Networks) model with back propagation to see its applicability in financial prediction problems and if it outperforms existing models since ANNs classify objects rather simply, it can take complex data as input, derives rules based on those data, and makes decisions. Our aim is to train the ANN to predict the EUR/TRY exchange rate. We tried to increase the accuracy by using appropriate transfer functions, neuron number or learning algorithms. Empirical analysis is performed with MATLAB using the dataset obtained from Central Bank of the Republic of Turkey (CBRT).

2. The challenges of analyzing financial time series

Forecasting exchange rates is a common financial problem that is receiving attention although it has difficulties. Because of the following factors the situation requires function approximators, like ANN, that can hold nonlinear functions. Table 1. shows the forecasting studies and models employed for those works.

Table 1. Models and Currencies Used in Reviewed Works

No	Author	Year	Currency	Models
1	El Shazly and El Shazly	1997	BP, DM, JPY	MLP
2	Zhang and Hu	1998	GBP/USD	ARCH, GARCH, MLP
3	Zhang	2003	BP/USD	ARIMA and ANN(hybrid)
4	Sermpinis et al.	2012	EUR/USD	RNN, PSN, ARMA, MLP
5	Zhang and Berardi	2001	BP/USD	MLP
6	Nag and Mitra	2002	DM/USD, JY/USD, USD/BP	GANN, FGANN, GARCH, ARCH
7	Onaran	2010	USD/TRY	MLP
8	McNelis	2005		GARCH
9	Özkan	2013	TRY/USD, TRY/EUR	PPP, ANN
10	Hua et al	2010	US/JPY, US/BP, US/IR	FLANN, FLANN-KR
11	Majhi et al	2009	BP, USD, JY	FLANN, CFLANN
12	Gradojevic and Yang	2006	CAN/USD	Random Walk, ANN, linear
13	Sewell and Taylor	2012	USD/DEM, USD/JPY, GBP/USD, USD/CHF, DEM/JPY, GBP/CHF	HMM
14	Dhamija and Bhalla	2011	BP/USD, DM/USD, JPY/USD, IR/USD, EUR/USD	MLP, RBF

Economic models do not yet exist to be and satisfactory models because the rates affected by hundreds of parameters. Even the psychological and sociological factors which can not be measured, affects the results (Lendasse, De Bodt, Wertz and Verleysen, 2000).

The macroeconomic time series is very noisy. There exists sudden decrease and increase in data. The reason for the noise is the collection techniques of data that affect the economy (Yıldız, 2006).

Traditional macroeconomic time series models are linear. Time series have a chaotic behaviour (Torkamani, Mahmoodzadeh, Pourroostaei and Lucas, 2007) However, many economist researchers recommend the use of nonlinear models for predicting the future rates, because of the failure of linear models the use of artificial neural networks will be advantageous for prediction (Lendasse, De Bodt, Wertz and Verleysen, 2000).

3. Artificial Neural Networks (Ann) Model

Neural networks is a mathematical model inspired by biological neural networks. Researchers are designing ANNs to solve a variety of problems in pattern recognition, prediction, optimization, associative memory, and classification. Multi Layer perceptron (MLP) is a feed forward neural network which is commonly used in forecasting problems. An MLP consists of an input layer and an output layer, with one or more hidden layers in between (http://neuroph.sourceforge.net/tutorials/MultiLayerPerceptron.html). Usually the input units only hold the input signal without any processing. Computation occurs in neurons in the hidden layers and the output layer.

3.1. An Artificial Neuron

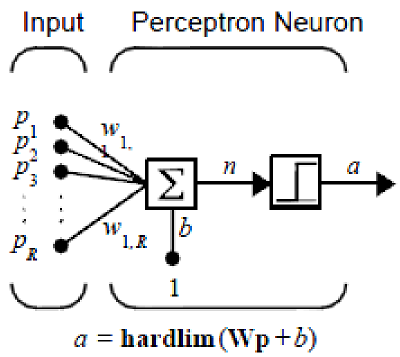


Figure 1. Summing and Threshold Process in a Perceptron
(http://radio.feld.cvut.cz/matlab/toolbox/nnet/percep10.html)

A neuron is a processing unit in a neural network. A single layer feed-forward network may have one or more output neurons, each of which is connected with a weight w to all of the inputs i. The input p is multiplied by the weight w, and then it is sent to the summer. The other input, 1, is multiplied by a bias b and then sent to the summer. The summer output n goes into a transfer function which produces the output "a" as in the Figure 1.

$$a = \frac{1}{1+e^{-n}} \tag{1}$$

The network output can be written as:

$$n = \sum_{i=1}^{m} w_i p_i + b \tag{2}$$

$$a = \varphi\left(\sum_{i=1}^{m} w_i p_i + b\right) \tag{3}$$

Where m is the number of inputs. The node that generates an output according to a transfer function called the activation function φ (Kröse and Van Der Smagt, 1996).

3.2. Perceptron Learning Rule

If an input vector is presented and the output of the neuron is correct

a = t

e = t – a = 0

Where "a" denotes output and "t" target value, then the weight vector w is not altered. If the neuron output is 0 and should have been 1

a = 0

t = 1

e = t – a = 1

Then the input vector p is added to the weight vector w, weigh is adjusted.

If the neuron output is 1 and should have been 0

a = 1

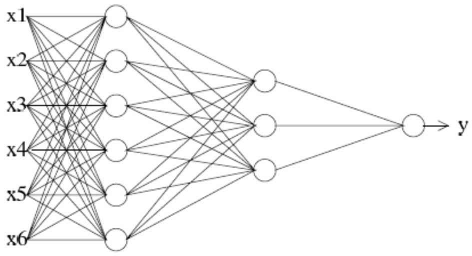
t = 0

e= t – a = –1

The input vector p is subtracted from the weight vector w (Fausett, 1994).

3.3. Multilayer Perceptron

MLPs are feed-forward NNs, generally trained with a back-propagation algorithm. They are the most commonly used types of artificial networks in financial time-series forecasting. The training of the MLP network is processed on a three layered architecture (Du and Swamy, 2006). A sample MLP is shown on the figure 2.



Source: (Simon, 2012)

Figure 2. A 6-3-1 Multilayer Perceptron

3.4. Backpropagation

Backpropagation is an abbreviation for "backward propagation of errors", is a common method of training artificial neural networks. The network training is an iterative process.

The processes for developing the back-propagation neural network model are as follows. It starts with deciding the architecture and parameters, the weight initialization. It's followed by training till the number of iterations reached or the error is acceptable. Then the network with the minimum error is chosen to forecast the outcome (Jain, Mao and Mohiuddin, 1996).

The training of the network, which is the adjustment of its weights in the way that the network maps the input value of the training data to the corresponding output value, the learning algorithm simply tries to find those weights which minimize the error.

$$w^{new} = w^{old} + ep^{T} \tag{4}$$

Where p^T is the transpose of the input value. The training of a network is stopped when the mean squared error is at minimum (Philip, Taofiki and Bidemi, 2011).

Results can be evaluated using MSE. It's the most commonly used error metric with the formula:

$$MSE = \frac{1}{Q}\sum_{k=1}^{Q} e(k)^2 = \frac{1}{Q}\sum_{k=1}^{Q} (t(k) - a(k))^2 \tag{5}$$

Where Q represents the number of forecast values used in computing (Evans, 2007). The network parameters are then estimated by fitting the training data using the iterative procedure backpropagation of errors.

4. Empirical analysis

4.1. Data collection

Our inputs used in the implementation process are obtained from CBRT website over the period 01.01.2010 to 01.01.2013, shown in the table 2. (http://evds.tcmb.gov.tr/yeni/cbt-uk.html). We tried to select the parameters which have the greatest influence on fluctuations of EUR/TRY currencies. We have used four datasets formed to forecast 150 days, 3 months, 1 month and 1 week of EUR/TRY parity. The 150 days forecast dataset is segmented into two parts, %80 training set and %20 test set. To forecast 3 months horizon we've chosen last 60 workdays for test set, for 1 month forecast last 20 days and for 1 week forecast we've chosen last 5 days. Totally the dataset contains 756 working days.

Table 2. Parameters Effecting EUR/TRY Exchange Rates

Attribute Information
1-USD/TRY exchange rates (selling)
2-Bullion Gold Selling Price (TRY/Gr)
3- Weighted Average of Maximum Interest Rates Applicable to FX Deposits
4-Central Bank Reserves (Weekly(Friday), $ Millions)
5-Producer Price Index (2003=100)(TURKSTAT)(Monthly)
6-Consumer Price Index (2003=100) (TURKSTAT) (Monthly)
7-Istanbul Stock Exchange (ISE) Daily Trading Volume
8-Weighted Average of Maximum Interest Rate (Turkish Lira Deposits)
9-EUR/TRY exchange rates (selling)

4.2. Building the network

The network is asked to predict the next value in the time sequence, thus we have one output neuron. The number of inputs is the number of parameters except the target EUR/TRY parity. The most challenging part is the selection of number of hidden layer neurons. Researchers developed some rules; however it's about the ability of the designer. In a network having n inputs and m outputs, the number of neurons in hidden layer can be squareroot of (n·m). The result can be in the interval of 1.5 times the formula above to 2 times the formula (Yıldız, 2006).

All simulations run for 5000 iterations during the learning phase (epochs), learning rate is 0.2 and the other characteristics are shown in the Table 3. All standard experiments are done using 8-10-1 structure with tansig-logsig-purelin transfer functions, as the learning algorithm trainbr is preferred.

Table 3. Characteristics of the Neural Network

Characteristic	**MLP**
Learning algorithm	trainbr, trainlm, trainoss, trainscg,
Learning rate	0.2
Epoch	5000
Input neurons	8
Hidden neurons	5, 10, 20
Output neurons	1
Transfer functions	tansig, logsig, purelin tansig, purelin, purelin logsig, logsig, purelin

4.3. Implementation

The current values of the EUR/TRY exchange rate, together with other economical indicators are used to forecast the EUR/TRY exchange rate for a week, a month, 3 months and 150 days time. For forecasting next three months there is going to be 60 working days to predict to forecast the EUR/TRY exchange rate for next one month there is going to be 20 working days.

The evaluation is done for four different forecast horizons under the titles effects of used learning algorithm, effects of change in number of neurons in hidden layer and effects of used transfer functions. MLP with normalized backpropagation is used for training and efficiency of each situation is evaluated with MSE (Mean Squared Error) with the formula

$$MSE = \frac{1}{Q}\sum_{k=1}^{Q} e(k)^2 = \frac{1}{Q}\sum_{k=1}^{Q}(t(k)-a(k))^2 \qquad (6)$$

Where Q represents the number of forecast values used in computing (Evans, 2007). Then the results are analyzed for identifying the effects of changes in the network.

5. Experimental results

Each section below studies the effect of different architectures on forecasting results.

5.1. Effects of used Learning Algorithm

After trials we observed that the most accurate prediction is 1-week forecast with trainbr and the closest is with trainlm shown in the Table 4. Higher accuracies are recorded in the short term forecasts. The algorithms traingd, trainscg and trainoss weren't applicable to most of the structures, thus they're not shown on the experimental results. Figure 3 and also Table 5 shows the target and obtained output values for the best accuracy structure.

Table 4. Experiments with Chosen Learning Algorithms

Forecast horizon	Trans Func.	Training Algorithm	Epoch	Time	MSE	R^2
150 days	tansig-logsig-purelin	trainbr	36	2	0.018180	0.6187
	tansig-logsig-purelin	trainlm	26	0	0.028257	0.3887
3- Months	tansig-logsig-purelin	trainbr	86	3	0.01246	0.1812
	tansig-logsig-purelin	trainlm	51	2	0.09833	0.0520
1-Month	tansig-logsig-purelin	trainbr	51	1	0.003512	0.2126
	tansig-logsig-purelin	trainlm	74	1	0.003200	0.5480
1-Week	tansig-logsig-purelin	trainbr	60	1	0.001232	0.6558
	tansig-logsig-purelin	trainlm	33	0	0.002966	0.2621

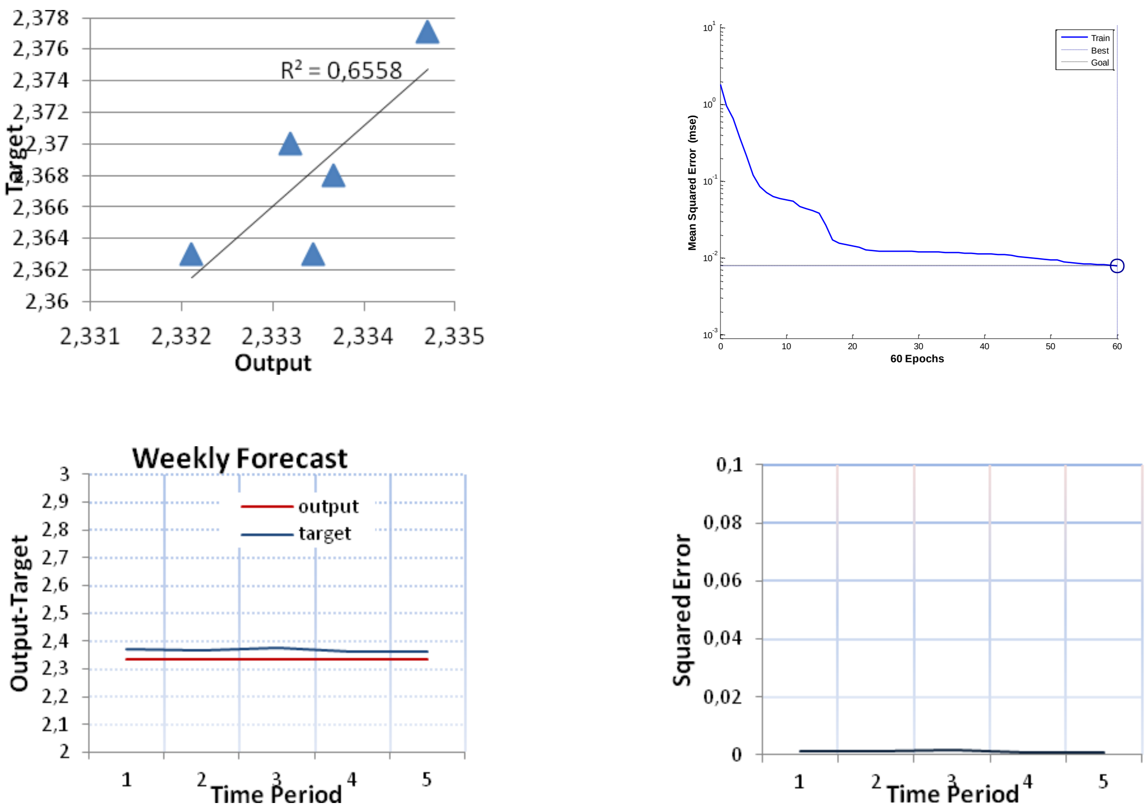


Figure 3. The Best Result of 1-Week Forecast with trainlm

Vol. 4 (4), pp. 307–316, © 2014 HRMARS

Table 5. Target-Output Comparision of Figure 3 with MSE=0.001232

USD	Bullion Gold	Interest rate FX	CB reserves	PPI	CPI	IMKB trading	Interest rate TRY	Target	Network Output
1.7979	99.2	0.25	100676	209.28	213.23	2277994	0.25	2.37	2.333188
1.7963	99.2	0.25	100676	209.28	213.23	2512238	0.25	2.368	2.333665
1.7934	99.2	0.25	100676	209.28	213.23	1957977	0.25	2.3771	2.334698
1.7915	99.2	0.25	100320	209.28	213.23	1969197	0.25	2.363	2.33344
1.7912	99.2	0.25	100320	209.28	213.23	3310331	0.25	2.360	2.332109

5.2. Effects of Change in Number of Neurons in Hidden Layer

We identified the number of inputs same as the number of parameters used in data collection and the output as one neuron which is the next day forecast. The structure with the neuron numbers 8-5-1, 8-10-1 and 8-20-1 are givenin Table 6. The neuron number of the hidden layer is increased for all forecast horizons. Figure 4 also Table 7 shows the target and obtained output values for the best accuracy structure.

Table 6. Experiments with Increasing Number of Neurons in Hidden Layer

No of experiments	Forecast horizon	Number of neuron	Time	Epoch	MSE	R^2
1	150 Days	8-5-1	1	24	0.022804	0.4916
2		8-10-1	3	37	0.024156	0.4245
3		8-20-1	1	29	0.015161	0.4676
1	3-Months	8-5-1	1	71	0.003835	0.3520
2		8-10-1	3	171	0.008801	0.5300
3		8-20-1	2	69	0.003390	0.0654
1	1-Month	8-5-1	0	37	0.002597	0.1631
2		8-10-1	1	53	0.003257	0.3310
3		8-20-1	2	68	0.007187	0.2053
1	1-Week	8-5-1	0	46	0.001330	0.7391
2		8-10-1	1	37	0.000484	0.3845
3		8-20-1	2	165	0.000267	0.2879

The following figure shows The Best Result of 1-Week Forecast with MSE=0.000267.

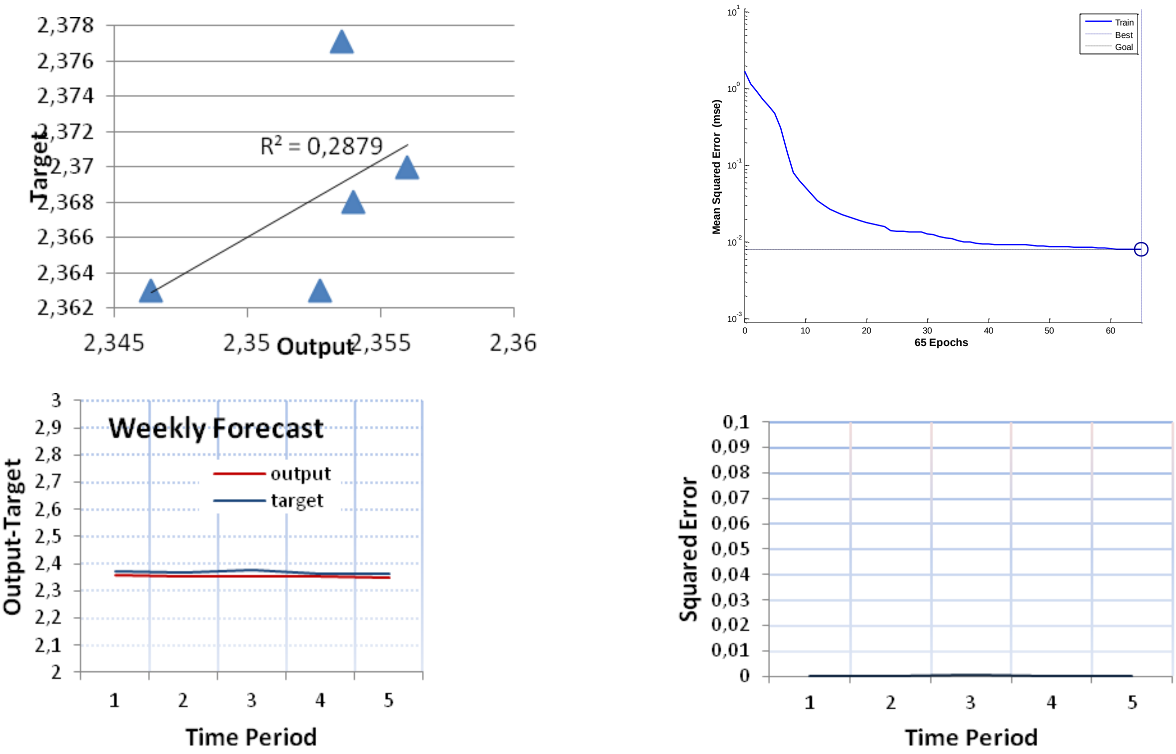


Figure 4. The Best Result of 1-Week Forecast with MSE=0.000267

Table 7. Target-Output Coparision of Figure 5.5 with MSE=0.000267

USD	Bullion Gold	Interest rate FX	CB reserves	PPI	CPI	IMKB trading	Interest rate TRY	Target	Network Output
1.7979	99.2	0.25	100676	209.28	213.23	2277994	0.25	2.37	2.35598
1.7963	99.2	0.25	100676	209.28	213.23	2512238	0.25	2.368	2.353929
1.7934	99.2	0.25	100676	209.28	213.23	1957977	0.25	2.3771	2.353495
1.7915	99.2	0.25	100320	209.28	213.23	1969197	0.25	2.363	2.352723
1.7912	99.2	0.25	100320	209.28	213.23	3310331	0.25	2.360	2.346359

5.3. Effects of Used Transfer Functions

We applied different combinations of transfer functions and achieved results are given in Table 8. In the figure 5 and also Table 9 the Target-Output comparison of the best accuracy structure is given.

Table 8. Experiments Performed with Different Transfer Functions

No of experiment	Forecast horizon	Transfer Function	Time	Epoch	MSE	R^2
1	150 Days	tansig, logsig, purelin	1	36	0.018180	0.6187
2		tansig, tansig, purelin	0	36	0.041351	0.5856
3		logsig, logsig, purelin	1	46	0.029785	0.5003
1	3 Months	tansig, logsig, purelin	1	58	0.001548	0.0936
2		tansig, tansig, purelin	3	133	0.001658	0.0562
3		logsig, logsig, purelin	1	80	0.001552	0.1125
1	1-Month	tansig, logsig, purelin	1	72	0.004094	0.3169
2		tansig, tansig, purelin	1	62	0.003012	0.2893
3		logsig, logsig, purelin	1	59	0.003717	0.2246
1	1-Week	tansig, logsig, purelin	0	26	0.000627	0.2683
2		tansig, tansig, purelin	1	70	0.000518	0.3035
3		logsig, logsig, purelin	1	48	0.000989	0.3485

The following figure shows the most accurate of the tests which produced the minimum MSE.

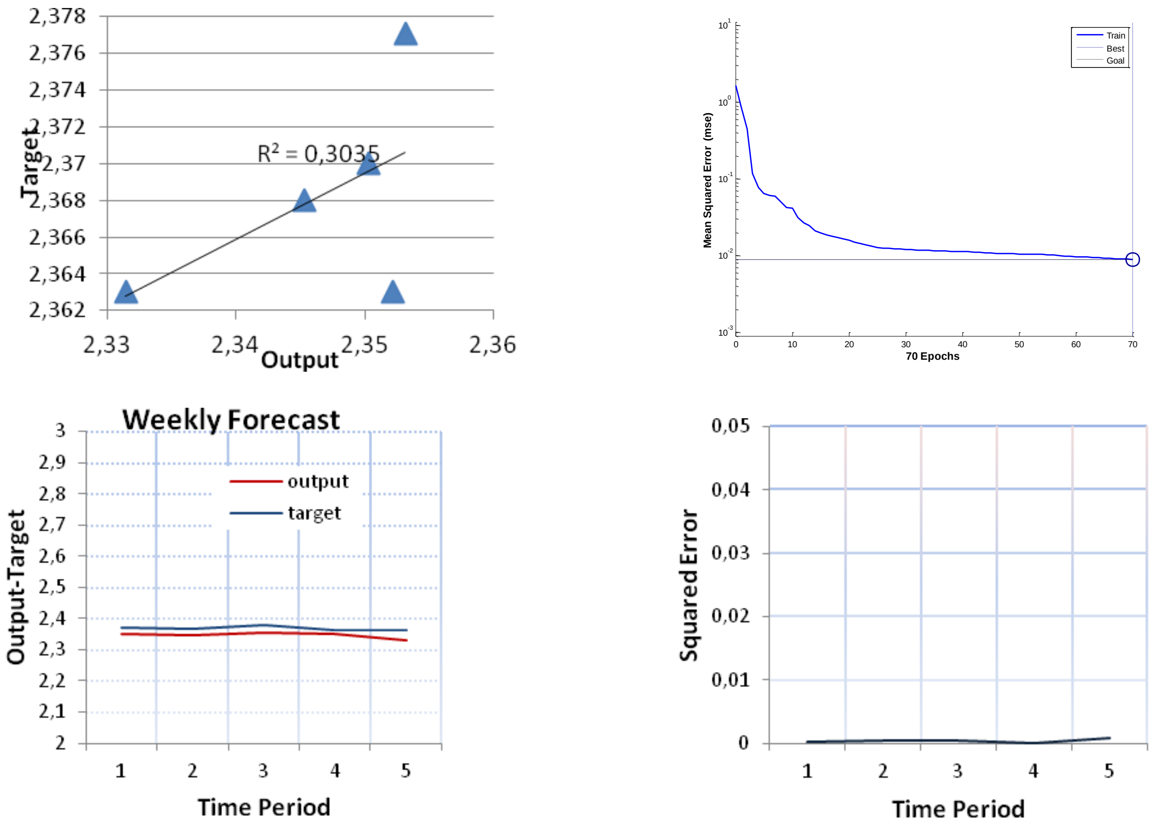


Figure 5. The Best Result of 1 Week Forecast with Transfer Functions tansig-tansig-purelin

Table 9. Target-Output Comparison of Figure 5.7 with MSE=0.000518

USD	Bullion Gold	Interest rate FX	CB reserves	PPI	CPI	IMKB trading	Interest rate TRY	Target	Network Output
1.7979	99.2	0.25	100676	209.28	213.23	2277994	0.25	2.37	2.350286
1.7963	99.2	0.25	100676	209.28	213.23	2512238	0.25	2.368	2.345289
1.7934	99.2	0.25	100676	209.28	213.23	1957977	0.25	2.3771	2.353137
1.7915	99.2	0.25	100320	209.28	213.23	1969197	0.25	2.363	2.35217
1.7912	99.2	0.25	100320	209.28	213.23	3310331	0.25	2.360	2.331452

6. Discussions and conclusions

The comparison of best results with respect to the forecast horizons shows that the 1-week forecasts are the most reliable. As in the given Table 10 and Table 11 we've observed that the network is efficient in short term forecasting much more than long horizon forecasts.

Table 10. Comparison of Best Accuracies wrt Forecast Horizon

Forecast horizon	Epoch	Time	MSE	R^2
150 Days	29	1	0.015161	0.4676
3 Months	58	1	0.001548	0.0936
1 Month	37	0	0.002597	0.1631
1 Week	165	2	0.000267	0.2879

Table 11. Target-Output Comparision of tansig-logsig-purelin structure with 8-10-1 neurons and trainbr algorithm for 1 week horizon

USD	Bullion Gold	Interest rate FX	CB reserves	PPI	CPI	IMKB trading	Interest rate TRY	Target	Network Output
1.7979	99.2	0.25	100676	209.28	213.23	2277994	0.25	2.37	2.350286
1.7963	99.2	0.25	100676	209.28	213.23	2512238	0.25	2.368	2.345289
1.7934	99.2	0.25	100676	209.28	213.23	1957977	0.25	2.3771	2.353137
1.7915	99.2	0.25	100320	209.28	213.23	1969197	0.25	2.363	2.35217
1.7912	99.2	0.25	100320	209.28	213.23	3310331	0.25	2.360	2.331452

Furthermore, we analyzed the effects of neuron number in the hidden layer and we observed that the increase is not directly proportional with the increase of the accuracy.

This study points out that the choice of the number of neurons effects the performance of the neural network. But increase in number of neurons doesn't increase the performance. Also, the experiment results showed that the success of a neural network is very sensitive to the chosen learning algorithms and transfer functions in the training process. During our study we observed that the network is efficient in short term forecasting much more than long horizon forecasts. When the outputs of the model were compared with the actual outputs, the curves showing the desired output and the actual output looked identical; which shows the success of the model.

Future work will be focused on enhancement of model's design and parameters used in dataset, also hybrid models can be developed for superior results. Also number of inputs can be increased by using a dataset over more years. The improvements in artificial intelligence technologies can provide us new ideas in the design of neural networks and make more efficient forecasts.

References

1. Box, G.E.P., & Jenkins, G.M. (1970). *Time series analysis, forecasting and control*. San Francisco: Holden Day.
2. Dhamija, A.K., & Bhalla, V.K. (2010). Exchange rate forecasting: comparison of various architectures of neural networks. *Neural Computing and Applications*, *20*(3), 355–363. doi: 10.1007/s00521-010-0385-5.
3. Du, K.-L., & Swamy, M.N.S. (2006). *Neural networks in a soft computing framework*. London: Springer. Retrieved from http://site.ebrary.com/id/10228947.
4. El Shazly, M. and El Shazly, H. (1997). Comparing the forecasting performance of neural networks and forward exchange rates. *Journal of Multinational Financial Management*, 7, 345-356.
5. Evans, J. (2007) *Statistics, Data Analysis & Decision Modelling*. 3rd ed. Pearson Prentice Hall, 251-254.

6. Fausett, L.V. (1994). *Fundamentals of neural networks: Architectures, algorithms, and applications*. Englewood Cliffs, NJ; Delhi: Prentice-Hall; Dorling.

7. Gradojevic, N., & Yang, J. (2006). Non-linear, non-parametric, non-fundamental exchange rate forecasting. *Journal of Forecasting*, *25*(4), 227–245. doi:10.1002/for.986.

8. Hua, X., Zhang, D. and Leung, S.C.H. (2010). "Exchange rate prediction through ANN Based on Kernel Regression", paper presented at *Third International Conference on Business Intelligence and Financial Engineering*, China.

9. Jain, A.K., Mao, J. and Mohiuddin, K.M. (1996). Artificial Neural Networks: A Tutorial. *IEEE Computer*.

10. Kröse, B. and Van Der Smagt, P. (1996). *An Introduction to neural networks*. 8th ed. Amsterdam, 21-45.

11. Lendasse, A., De Bodt, E., Wertz, V., & Verleysen, M. (2000). Non-linear financial time series forecasting-Application to the Bel 20 stock market index. *European Journal of Economic and Social Systems*, *14*(1), 81–92.

12. Majhi, R., Panda, G., & Sahoo, G. (2009). Efficient prediction of exchange rates with low complexity artificial neural network models. *Expert Systems with Applications*, *36*(1), 181–189. doi:10.1016/ j.eswa.2007.09.005.

13. McNelis, P.D. (2005). *Neural networks in finance*. Amsterdam; Heidelberg [u.a.]: Elsevier Academic Press.

14. Nag, A.K., & Mitra, A. (2002). Forecasting daily foreign exchange rates using genetically optimized neural networks. *Journal of Forecasting*, *21*(7), 501–511. doi:10.1002/for.

15. Onaran, B. (2010). *Forecasting Exchange Rates Using Artificial Neural Networks*. MA. Istanbul Bilgi Üniverstesi

16. Özkan, F. (2013). Comparing the forecasting performance of neural network and purchasing power parity: The case of Turkey. *Economic Modelling*, *31*, 752–758. doi:10.1016/j.econmod.2013.01.010.

17. Philip, A.A., Taofiki, A.A., & Bidemi, A.A. (2011). Artificial Neural Network Model for Forecasting Foreign Exchange Rate. *World*, *1*. Retrieved from http://wcsit.org.

18. Simon, E. (2002). *Forecasting Foreign Exchange Rates with Artificial Neural Networks*. 2002 report.

19. Theil, H. (1966). *Applied Economic Forecasting*. Amsterdam: North-Holland Publishing Company.

20. Torkamani, M.A., Mahmoodzadeh, S., Pourroostaei, S., & Lucas, C. (2007). Chaos theory and application in foreign exchange rates vs. IRR (Iranian Rial). *International Journal of Human and Social Sciences*, *1*(3), 130–134.

21. Yıldız, Ö. (2006*). Using Artificial Neural Networks for Forecasting USD Exchange Rates*. MA. Eskisehir Osmangazi Üniversitesi.

22. Zhang, G.P., & Berardi, V.L. (2001). Time series forecasting with neural network ensembles: an application for exchange rate prediction. *Journal of the Operational Research Society*, 652–664.

23. Zhang, G. and Hu, M. (1998). Neural Network Forecasting of the British Pound/US Dollar Exchange Rate. *Omega, Int. J. Mgmt Sci.*, 26 (4), 495-506.

24. Zhang, G. Peter. (2003). *Neural networks in business forecasting*. Igi Global. Retrieved from http://books.google.com/books.

25. Central Bank of the Republic of Turkey (n.d.). *CBRT Electronic Data Delivery System*. Retrieved from: http://evds.tcmb.gov.tr/yeni/cbt-uk.html.

26. Neuroph (n.d.). *Multilayer Perceptron*. Retrieved from http://neuroph.sourceforge.net/tutorials/ MultiLayerPerceptron.html

27. Radio.feld.cvut.cz (2000). *Perceptrons (Neural Network Toolbox)*. Retrieved from: http://radio.feld. cvut.cz/matlab/toolbox/nnet/percep10.html.