

Exchange Rate Prediction Using Neural-Genetic Model

R. Mechgoug and A. Titaouine

Department of Genie Electric, University Mohamed Kheidre, 07000 Biskra, Algeria

Abstract: Neural network have successfully used for exchange rate forecasting. However, due to a large number of parameters to be estimated empirically, it is not a simple task to select the appropriate neural network architecture for exchange rate forecasting problem. Researchers often overlook the effect of neural network parameters on the performance of neural network forecasting. The performance of neural network is critically dependant on the learning algorithms, the network architecture and the choice of the control parameters. Even when a suitable setting of parameters (weight) can be found, the ability of the resulting network to generalize the data not seen during learning may be far from optimal. For these reasons, it seems logical and attractive to apply genetic algorithms. Genetic algorithms may provide a useful tool for automating the design of neural network. The empirical results on foreign exchange rate prediction indicate that the proposed Hybrid Model exhibits effectively improved accuracy when is compared with some other time series forecasting models.

Key words: Prediction, time series, foreign exchange, neural network, genetic algorithm

INTRODUCTION

Exchange rate forecasting is an important and challenging task for both academic researchers and business practitioners. Various theoretical models including both econometrics and time series approaches have been suggested to model and forecast exchange rates.

During last 25 years, many different nonlinear models have been proposed in the literature to model and forecast exchange rates. The GARCH Model has been used in the past for volatility estimation in US dollar foreign exchange markets Baillie and Bollerslev (1989) and in the European Monetary System. Initial studies into explanatory power of out-of-sample forecasts gave out disappointing results West and Cho (1995). Jorion (1995) found that volatility forecasts for several major currencies from the GARCH Model were outperformed by implied volatilities generated from the Black and Scholes Option Pricing Model (Black and Scholes, 1973). These studies used the squared daily return as the variable to be forecast. Since, the exchange rate may move around a lot during the day, it has been established that one can significantly improve the forecasting power of the GARCH Model by using sumofintraday squared returns Andersen and Bollerslev (1998). This measure is referred to as integrated or realized volatility.

Some researchers claimed that exchange rates are rather unpredictable and so Random Walk Model is better predictor Chang and Osler (1999), Meese and Rose (1990) and Gencay (1999). Kuan and Liu (1995) estimate and

select feedforward and recurrent networks to evaluate their forecasting performance of five exchange rates against USD. The networks performed differently for different exchange rate series. Yao and Tan (2000) show that if technical indicators and time series data are fed to neural networks to capture the underlying process of the movement in currency exchange rates, then useful prediction can be made. Gradojevic and Yang (2006) construct a neural network that never performs worse than a linear model but always performs better than the Random Walk Model when predicting Canadian dollar/dollar exchange rate. Kiani and Kastens (2008) have successfully employed neural networks to forecast the exchange rate.

The networks of neurons have particular's properties which make him desirable to resolve a wide family of problems. However, they also have some limitations and inconveniences which do not allow their individual application in some cases (Bonissone *et al.*, 1999). Various forms for combining the neural network and genetic algorithms. The study of network neurons and genetic algorithms combination can be summarize in three general aspects: learn genetically the network neurons, genetic optimization of network topology and optimization of the control parameters (Bonissone *et al.*, 1999). The combination of neural network and genetic algorithms allows the merging all advantages of both individual approaches and having more the capacity to surmount difficulties and the limitations which characterize every approach (Castilo and Melin, 2002). In this research, researchers are interested in three combinations together,

i.e., researchers are going to make the learning, the optimization of topology and the selection of the control parameters by the genetic algorithms.

This study employs the popular multi-layer feedforward neural networks with back-propagation learning algorithms the were the topology and the control parameters are selected by reel genetics algorithms for predicting daily currency exchange rates of US Dollar (USD), against four other currencies such as Canadian Dollar (CAD), Great British Pound (GBP), Japanese Yen (JPY) and EUR using their historical exchange rates from 02/01/2003 to 30/06/2011was found on the web site: <http://pacific.commerce.ubc.ca/xr/>.

A total of 8 years historical exchange rates data, for each of four currency rates were collected and used as inputs to build the prediction models in the study and then additional 6 months exchange rates data were used to evaluate the models. The prediction results of the method is compared with Dhamija and Bhalla (2010) based on two evaluation indicators such as Normalized Mean Square Error (NMSE) and Mean Absolute Error (MAE).

NEURAL NETWORK

Time Series Forecasting Model assumes that there is an underlying process from which data are generated and the future value of a time series is solely determined by the past and the current observation. Neural network are able to capture the underlying pattern within a time series even when the underlying law governing the value of time series is unknown or too complex to describe. Therefore, for a time series forecasting problem, the inputs of network are the past lagged observations of the time series and the output is the future value:

$$y_t = f(y_{t-d}, y_{t-2*d}, \dots, y_{t-N*d}) \quad (1)$$

Where:

N = The number of input

d = Time lagged

The NN architectures considered here is a feedforward network or Multi-Layer Perceptron (MLP). In which the signal flow is from input layer towards output layer and:

	high ₁	low ₁	high ₂	low ₂	high ₃	low ₂
set	S ₂	S ₂	S ₂	S ₂	S ₂	S ₂

the training phase is effected with Back Propagation algorithm (BP). The topology of neural predictor requires the determination:

- Number of Hidden Layers (NHL)
- Number of nodes (Nhl) for each layer l
- The activation functions (f^l + f^{out}) each layer
- Presence or absence of the bias (b)
- The learning coefficient (α)
- Momentum coefficient (η)
- The input number (N)
- The times that separate two successive input (d)

The GA has been used to determine the appropriate set of the parameter.

OPTIMIZING NEURAL NETWORK STRUCTURE

The neural Network Predictor (NP) is defined by genetic encoding in which the chromosome composed of the different characteristics of the NP. Researchers consider a specific representation of the parameters related to the neural predictor structure; the chromosome is constituted of 17 genes. The Number of Hidden Layers, (NHL) the numbers of Neurons in each layer 1 (Nhl) and other genes representing the activation function type for the hidden layers and the output layer (f^l, f^{out}) the number of neurons in the input layer (N), the presence or absence of bias (b), lag (d), the learning coefficient (α), momentum coefficient (η), the variation interval of the synaptic weights for the 2 hidden layer and the output layer.

The generation of the initial population: The chromosome in the initial population is produced by a stochastic generation and shown in Fig. 1. Each allele is defined in a subset S_i, i = {1, ..., 7} shown in the second row of the Table 1.

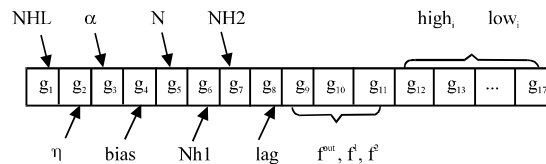


Fig. 1: Chromosome representation

Table 1: The subset of each gene

Allele	Set
NHL	S ₁
η	S ₂
α	S ₂
b	S ₃
N	S ₄
Nh1	S ₅
Nh2	S ₅
d	S ₆
f ^l	S ₇
f ^e	S ₇
f ^{out}	S ₇

Since, it is sufficient in most applications that a network has few hidden layers, the value of Nhl has been allowed to vary from 1-2 while the maximum value for Nhl is calculate using the flowing equation:

$$\max_Nhl = \frac{(N+1)}{2} + \sqrt{n_data} \quad (2)$$

where, n_data represent the number of the training data. All allele are defined in the following subsets:

$$S_1 = \{1, 2\}, S_2 = \{0, 1\}, S_3 = \{0, 1\}, S_4 = \{2, 3, \dots, 30\}, \\ S_5 = \{3, 4, \dots, 40\}, S_6 = \{2, 3, \dots, 10\}, S_7 = \{1, 2, 3\}$$

where, S₇ means that the activation function can assume the value 1 ≡ f₁ for the sigmoid, 2 ≡ f₂ for the hyperbolic tangent, 3 ≡ f₃ for the pure linear. S₃ means that 0 ≡ absence of the bias, 1 ≡ the presence of the bias. Low, high present the interval variation of the synaptic weights in the two hidden layers and output layer.

Fitness function: To evaluate the goodness of an individual, the neural is trained with fixed number of allele and then evaluated according to determinate parameters. The parameters which seem to describe better the goodness of a neural configuration are the mean square error MSE at the end of training. Clearly, it is desirable to obtain for MSE as low possible:

$$f(x) = \frac{\sum_{i=1}^N (y_i - a_i)^2}{N} \quad (3)$$

a_i is the output from the network.

Genetics operators

Selection: Selection of a pair of individuals from the population. There variety type of selection methods in these researches, researchers used remainder selection that assigns parents deterministically from the integer part of each individual’s scaled value and then uses roulette selection on the remaining fractional part (Bonissone *et al.*, 1999).

Crossover: That part of reproductive process in which two chromosomes exchanges some of their corresponding genes (Bonissone *et al.*, 1999). The crossover used is heuristic crossover that returns a child that lies on the line containing the two parents, a small distance away from the parent with the better fitness value in the direction away from the parent with the worse fitness value. You can specify how far the child is from the better parent by the parameter ratio which appears when you select

Heuristic. The default value of ratio is 1.2. If parent₁ and parent₂ are the parents and parent₁ has the better fitness value, the function returns the child:

$$child = parent_2 + R * (parent_1 - parent_2) \quad (4)$$

where, R is the ratio.

Mutation: It is an operation that applied for every gene of the new individuals with probability of P_m. The mutation used is adapt feasible mutation which randomly generates directions that are adaptive with respect to the last successful or unsuccessful generation. The feasible region is bounded by the constraints and inequality constraints. A step length is chosen along each direction so that linear constraints and bounds are satisfied.

EXPERIMENTAL RESULTS AND DISCUSSION

Data collection: The data used in this study is the foreign exchange rate of four different currencies against US Dollar (USD) from 02/01/ 2003 to 30/06/2011 found on the web site: <http://pacific.commerce.ubc.ca/xr>. We considered exchange rate European EUR, British Pound (GBP), Japanese Yen (JPY) and Canadian Dollar (CAD). The data used to train the four forecasters cover a period of 8 years from 02/01/2003 to 31/12/2010 whereas, the test data covers 7 months from 04/01/2011 to 30/06/2011 (Fig. 2a and b).

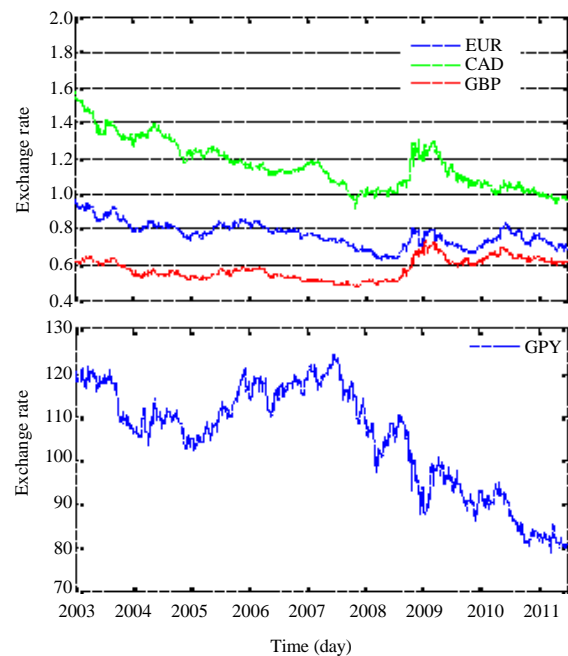


Fig. 2: Historical exchange rates for; a) EUR, GBP, CAD and b) JPY against USD

Evaluation indicators: The forecasting performance of the above model is evaluated against a two evaluation indicators, namely, Normalized Mean Square Error (NMSE), Mean Absolute Error (MAE). These criteria are defined in the following equation:

$$NMSE = \frac{\sum_i (y_i - a_i)^2}{\sum_i (y_i - \bar{a}_i)^2}$$

$$MAE = \frac{1}{N} \sum_i |y_i - a_i|$$

y_i and a_i are the actual and predicted values, respectively. NMSE and MAE measure the deviation between actual and forecast value. Smaller values of these evaluation indicators signify higher accuracy in forecasting.

Simulation results: The major purpose of this study is to select architecture, control parameters and weights variation interval of neural network by using a real genetic algorithm. The neural network parameters selected are: number of hidden layers, number of nodes for each layer l, activation functions for each layer, presence or absence of the bias, learning and momentum coefficients, input number, time that separate two successive inputs which

are critical parameters in the design of neural network. These parameters are selected genetically by minimizing the mean square error (Table 2).

A Neural Network Model was trained with four inputs representing the four exchange rates, one or two hidden layers and an output unit to predict the exchange rate. The final set of weights to which a network settles down depends on initial weights chosen. The training was terminated at iteration number between 1000 and 10000. The network that yielded the best result out of the genetic algorithm optimization is shown in Table 3. Researchers measured the performance metrics on the test data to investigate how well the neural network forecasting model captured the underlying trend of the movement of each currency against USD.

The comparative diagrams showing the output forecast by Neural Network Model and actual time series >143 days for four currencies exchange rate are shown in Fig. 3-6 show the forecasting error of EUR/USD, GBP/USD, CAD/USD and JPY/USD.

Table 2: The control parameters of the GA used for optimizing the neural predictor

Parameters	p-values
No. of generation	100.0
Size of population	50.0
Size of chromosome	17.0
Probability of crossover	1.0
Probability of mutation	0.1

Table 3: Neural network predictor's optimum structure f received from GA

Parameters neural network	EUR/USD	GBP/USD	CAD/USD	JPY/USD
No of hidden layers	2	2	1	1
Learning coefficient	0.7578	0.3524	0.9307	0.3475
Momentum coefficient	0.8421	0.8858	0.5457	0.4944
The presence/absence of the bias	Presence (1)	Absence (0)	Presence (1)	Presence (1)
Number of the input	20	17	10	6
Lag time	1	1	4	3
Number of neurons in the hidden layer1	16	29	12	14
Number of neurons in the hidden layer 2	16	8	7	9
The activation function in the hidden layer1	Pure linear 3	Pure linear 3	Pure linear 3	Pure linear 3
The activation function in the hidden layer2	Pure linear 3	sigmoid 1	sigmoid 1	Pure linear 3
The activation function in the output layer	hyperbolic 2	Pure linear 3	Pure linear 3	Pure linear 3
Int _{out}	[0.0725, 0.2798]	[0.0901, 0.3240]	[0.1634, 0.4703]	[0.0240, 0.3264]
Int ₁	[0.1035, 0.6864]	[0.0326, 0.5631]	[0.06360, 0.8137]	[0.0097, 0.3710]
Int ₂	[0.1106, 0.3346]	[0.0773, 0.6963]	[0.2002, 0.9350]	[0.04284, 0.4252]

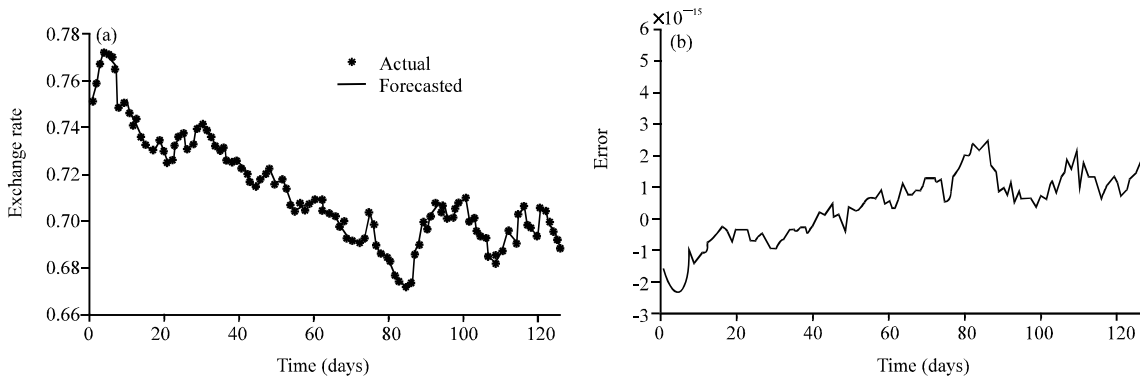


Fig. 3: a) Actual and forecasted exchange rate EUR/USD and b) Forecasting error EUR/USD

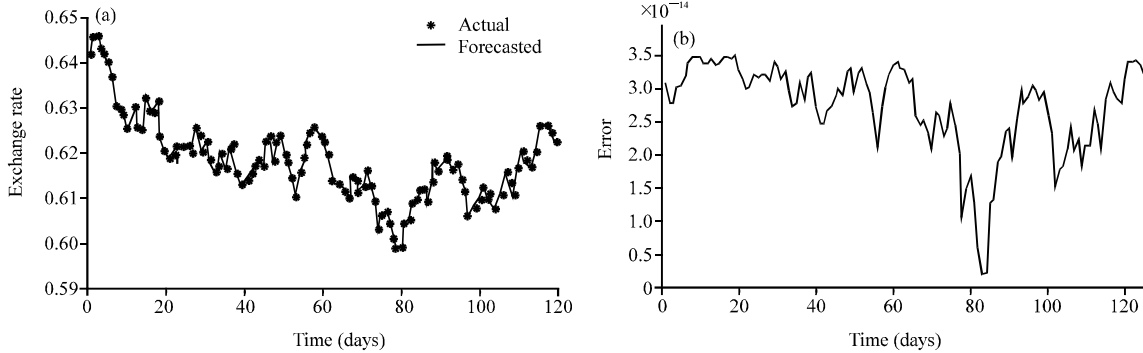


Fig. 4: a) Actual and forecasted exchange rate GBP/USD and b) Forecasting error GBP/USD

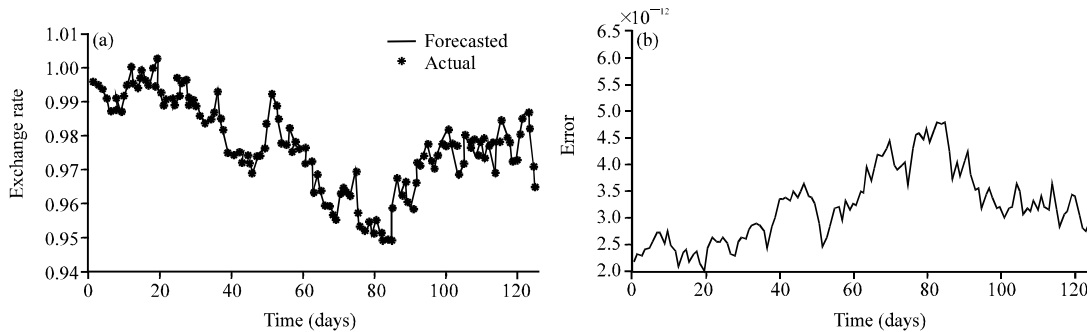


Fig. 5: a) Actual and forecasted exchange rate CAD/USD and b) Forecasting error CAD/USD

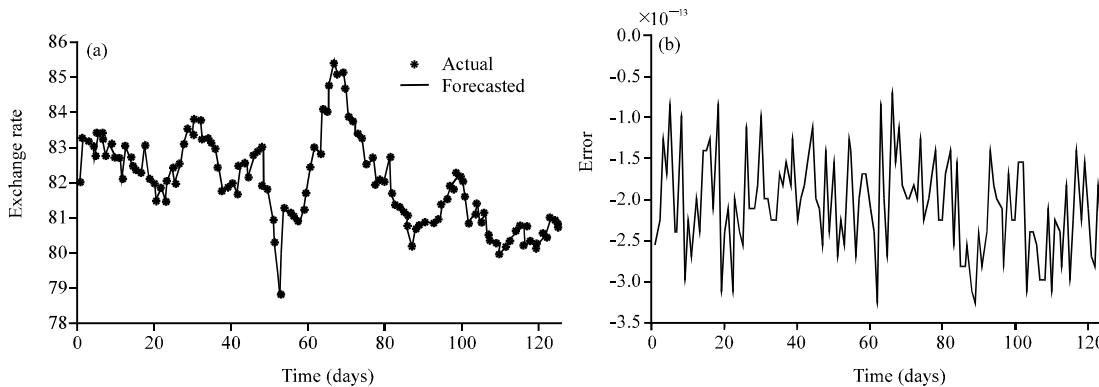


Fig. 6: a) Actual and forecasted exchange rate JPY/USD and b) Forecasting error JPY/USD

COMPARISON OF NEURAL-GENETIC, NEURAL NETWORK, CONDITIONAL HETEROSCEDASTIC MODELS

Table 4-7 show the evaluation indicators achieved by each model over a forecasting period. The results show that Neural-Genetic Model consistently performs better than Network and Conditional Heteroscedastic Models (GARCH, GARCH-M, EGARCH, TGARCH/GJR and IGARCH) in terms of the two evaluation indicators in all

Table 4: Comparative analysis of EUR/USD series

Models	NMSE	MAE×10 ²
GARCH (1,1)	0.99998	0.34930
GARCH (1,1)-M	0.93987	0.34918
EGARCH (1,1)	0.89833	0.30927
TGARCH/GJR (1,1)	0.89986	0.31414
IGARCH (1,1)	0.92064	0.34999
Neural network	0.88442	0.38644
Neural genetic	7.1053×10 ⁻⁵	5.5898×10 ⁻⁶

the currency exchange rates. This means these models are capable of predicting exchange rates more closely than

Table 5: Comparative analysis of GBP/USD series

Models	NMSE	MAE×10 ²
GARCH (1,1)	1.00011	0.27177
GARCH (1,1)-M	0.86999	0.26832
EGARCH (1,1)	0.89286	0.32988
TGARCH/GJR (1,1)	0.87995	0.26164
IGARCH (1,1)	0.80463	0.27175
Neural network	0.56853	0.2274
Neural genetic	6.0868×10 ⁻⁴	1.7504×10 ⁻⁵

Table 6: Comparative analysis of USD/ JPY series

Models	NMSE	MAE×10 ²
GARCH (1,1)	0.89899	0.36216
GARCH (1,1)-M	0.84957	0.36010
EGARCH (1,1)	0.79654	0.32844
TGARCH/GJR (1,1)	0.80971	0.32207
IGARCH (1,1)	0.82374	0.36236
Neural network	0.74021	0.37706
Neural genetic	9.8296×10 ⁻⁴	1.3284×10 ⁻⁵

Table 7: Comparative analysis of USD/CAD series

Model	NMSE	MAE×10 ²
Neural genetic	1.5483×10 ⁻³	2.0568×10 ⁻⁴

Network and Conditional Heteroscedastic Models. The reason of better performance by Neural-Genetic Model is the improved genetic optimization which allows them to search efficiently in architecture, control parameters and weights variation interval space for solution. The results of the Conditional Heteroscedastic Models and neural are taken from (Dhamija and Bhalla, 2010).

CONCLUSION

Genetic algorithm was used in this study to optimize the parameters of a multilayer perceptron’s (the number of hidden layers, the numbers of neurons in each layer l, the activation function type for the hidden layers and the output layer, the number of neurons in the input layer, the presence or absence of bias, lag, the learning and momentum coefficient, the variation interval of the synaptic weights for the 2 hidden layer and the output layer).

To evaluate the effectiveness of the proposed approach, EUR/USD, GBP/USD, CAD/USD and JPY/USD exchange rates were predicted. Empirical results showed that the proposed evolutionary-connectionist hybrid model has better performance when is compared with some other time series forecasting models (Table 4, 5 and 7).

REFERENCES

Andersen, T.G. and T. Bollerslev, 1998. Answering the skeptics: Yes, standard volatility models do provide accurate forecasts. *Int. Econ. Rev.*, 39: 885-905.

Baillie, R.T. and T. Bollerslev, 1989. The message in daily exchange rates: A conditional variance tale. *J. Bus. Econ. Stat.*, 7: 297-305.

Black, F. and M. Scholes, 1973. The pricing of options and corporate liabilities. *J. Political Econ.*, 81: 637-654.

Bonissone, P.P., Y.T. Chen, K. Goebel and P.S. Khedkar, 1999. Hybrid soft computing systems: Industrial and commercial applications. *Proceedings of the IEEE Proceedings, Special Issue on Computational Intelligence, Volume 87, September, 1999, USA.*, pp: 1641-1667.

Castilo, O. and P. Melin, 2002. Hybrid intelligent systems for time series prediction using neural networks, fuzzy logic and fractal theory. *Proceedings of the IEEE Transactions on Neural Networks, Volume 13, November, 2002, USA.*, pp: 1395-10.1109/TNN.2002.804316.

Chang, P.H.K. and C.L. Osler, 1999. Methodical madness: Technical analysis and the irrationality of exchange-rate forecasts. *Econ. J.*, 109: 636-661.

Dhamija, A.K. and V.K. Bhalla, 2010. Financial time series forecasting: Comparison of neural networks and ARCH models. *Int. Res. J. Finance Econ.*, 49: 185-202.

Gencay, R., 1999. Linear, non-linear and essential foreign exchange rate prediction with simple technical trading rules. *J. Int. Econ.*, 47: 91-107.

Gradojevic, N. and J. Yang, 2006. Non-linear, non-parametric, non-fundamental exchange rate forecasting. *J. Forecast.*, 25: 227-245.

Jorion, P., 1995. Predicting volatility in the foreign exchange market. *J. Finance*, 50: 507-528.

Kiani, K.M. and T.L. Kastens, 2008. Testing forecast accuracy of foreign exchange rates: Predictions from feed forward and various recurrent neural network architectures. *Comput. Econ.*, 32: 383-406.

Kuan, C.M. and T. Liu, 1995. Forecasting exchange rates using feed forward and recurrent neural networks. *J. Applied Econ.*, 10: 347-364.

Meese, R.A. and A.K. Rose, 1990. Nonlinear, nonparametric, nonessential exchange rate estimation. *Am. Econ. Rev.*, 80: 192-196.

West, K.D. and D. Cho, 1995. The predictive ability of several models of exchange rate volatility. *J. Econ.*, 69: 367-391.

Yao, J. and C.L. Tan, 2000. A case study on using neural networks to perform technical forecasting of forex. *Neurocomputing*, 34: 79-98.