

A COMPUTATIONAL APPROACH TO FORECASTING EXCHANGE RATES

¹DIPTI RANJAN MOHANTY, ²DR. SUSANTA KUMAR MISHRA

¹PhD Research Scholar, IBCS - Faculty of Management Studies, SOA University, Bhubaneswar, India

²Faculty in Finance and Center Head, Koustuv Business School, Bhubaneswar, India

E-mail: ¹ydiptimohanty@gmail.com, ²susanta65@gmail.com

ABSTRACT

It is a major endeavor to forecast the spot prices in the FX market. The spot price prediction presents a theoretical as well as computational challenge. This paper attempts to blend Local Linear Wavelet Technique with Artificial Neural Network to forecast the currency exchange market.

Keywords: *FX rates prediction, Currency exchange rates forecasting, Local Linear Wavelet Neural Network*

1. INTRODUCTION

Currency exchange (FX) rate is a crucial element of capital markets and international transactions. It is widely regarded as one of the most volatile and chaotic financial time series. The FX rate forecasting helps hedge risks and take speculative positions. A better forecasting not only helps hedge risks of businesses but also helps capital markets players better manage positions and risks.

The nature of complexity of FX rates, accentuates the need for a better forecasting model. There are a couple of models currently in use for forecasting; 1- Statistical models like ARMA-Autoregressive Moving Average, exponential smoothing, ARIMA-Autoregressive Integrated Moving Average, GARCH- Generalized Autoregressive Conditional Heteroskedastic and 2- Soft computing methods like Artificial Neural Network (ANN), Fuzzy Set Theory and Rough Set etc. Among contemporary models, sophisticated artificial neural networks e.g. Dynamic Filter-Weights Neural Network [2, 3], Local Linear Wavelet Neural Network (LLWNN) [4] and Functional Link Artificial Neural Network (FLANN) [5] have been developed and extensively used for forecasting financial time series data like stock market, electricity market and currency exchange market. Ahmed Emam [6] used ANN to forecast currency exchange rate for a short period

of time. Yuehui Chen and others [7] have applied LLWNN model to predict stock market indices where the parameters are optimized by using estimation of distribution algorithm. In this paper, we propose a Local Linear Wavelet Neural Network to predict the currency exchange rate data for one day, one month, six months and twelve months in advance.

This paper is an attempt to introduce a model to forecast exchange rates. However, unlike contemporary models which are grounded in the analysis of supply and demand for a particular currency with respect to another currency as the baseline (usually USD), this model uses a pure computational approach. This model does not intend to substitute or replace the existing models. This model is instead proposed as an additional dimension to the prediction of exchange rates.

The key research question that this paper addresses is whether a computational approach like LLWNN can provide a more accurate forecast of exchange rates. The outcome of this research, described in more detail in subsequent sections, suggests that this model indeed provides a more accurate forecast.

The outcome of this paper should only be used in conjunction with models that actually use the demand and supply of currency. This is a key limitation of the model in that it shall not be able to



accurately account for sudden changes in market liquidity of currency owing to market factors.

2. WAVELET NEURAL NETWORK

Wavelet neural network (WNN) combines neural network and the wavelet technique. The disadvantage of WNN is that when the dimension increases, the number of hidden layers increases to an undefined number. Similar is the problem with neurons in the hidden layers. As a result, calculation becomes more complicated. But, the advantage of LLWNN is that it needs less hidden layer units. Also, connection weights between the hidden layer units and output units are replaced by a local linear model.

According to the wavelet transformation theory, following form wavelets

$$\psi = \left\{ \psi_i = \left| a_i \right|^{-1/2} \psi \left(\frac{x - b_i}{a_i} \right) : a_i, b_i \in R^n, i \in z \right\} \tag{1}$$

$$x = (x_1, x_2, \dots, x_n),$$

$$a_i = (a_{i1}, a_{i2}, \dots, a_{in}),$$

$$b_i = (b_{i1}, b_{i2}, \dots, b_{in}),$$

are a family of functions generated from one single function $\psi(x)$ by the operation of dilation and translation. $\Psi(x)$, localized in time space and frequency space, is termed a mother wavelet. Also, a_i is the scale parameter and b_i is the translation parameter.

The mother wavelet is

$$i) \psi(x) = \frac{-x^2}{2} e^{-x^2/\sigma^2} \tag{2}$$

$$ii) \psi(x) = e^{-\left(\frac{x-c}{\sigma}\right)^2} \tag{3}$$

$$\text{where } x = \sqrt{p_1^2 + p_2^2 + \dots + p_n^2} \tag{4}$$

Further, in the place of weight w_i , a linear model $v_i = w_{i0} + w_{i1}x_1 + \dots + w_{in}x_n$ is introduced. $\tag{5}$

The standard output of WNN is

$$f(x) = \sum_{i=1}^M w_i \psi_i(x)$$

$$= \sum_{i=1}^M w_i \left| a_i \right|^{-1/2} \psi \left(\frac{x - b_i}{a_i} \right) \tag{6}$$

where ψ_i is the wavelet activation of i^{th} unit of hidden layer and w_i is the weight connecting the i^{th} unit of the hidden layer to the output layer unit.

In our model the output in the output layer is

$$Y = \sum_{i=1}^M (w_{i0} + w_{i1}x_1 + \dots + w_{in}x_n) \psi_i(x)$$

$$= \sum_{i=1}^M (w_{i0} + w_{i1}x_1 + \dots + w_{in}x_n) \left| a_i \right|^{-1/2} \psi \left(\frac{x - b_i}{a_i} \right) \tag{7}$$

3. A POPULAR LEARNING ALGORITHM

As all the parameters are chosen by hit and trial method and optimized by back propagation learning algorithm, the output may face local minima problems. So to obtain the maximum accuracy and to avoid the local minima problem which are inherent in back propagation, a derivative free global optimization technique i.e. Particle Swarm Optimization Algorithm is used to initialize and optimize the model parameters.

PSO or Particle Swarm Optimization was presented by Kennedy and Eberhart [11]. This is a search optimization method. It initializes randomly and works iteratively to find the optimal solution. This is achieved by the adjustment of the path of each individual towards its best position and towards the best particle of the entire population at each generation.

PSO model has a key upper hand owing to its ease of implementation and capability for a quick convergence to an optimal solution. For a particle moving in a multidimensional search space let S_i denotes the position of the i^{th} particle and v_i denotes the velocity at the sampling instant k , and the dimension j denotes the number of parameters to be optimized which if too high will result in allowing the particles to fly past good solutions. On the other hand, if v_{max} is too small, particles end up in local solutions only.

PSO, as with GA, has premature convergence and thus results in an inaccurate solution. In order to avoid this problem, a different approach is used to vary the inertia weight centered on variance of the population fitness. The revised parameters at instant $k + 1$ can be considered as

$$v_i(k+1) = \omega_i v_i(k) + c_1 \text{rand}() + (pbest_i - s_i(k)) + c_2 \text{rand}() (gbest_i - s_i(k)) \quad (18)$$

$$\text{and } S_i(k+1) = S_i(k) + v_i(k+1) \quad (19)$$

When c_1 and c_2 are constants identified as acceleration co-efficient, $\text{rand}()$ is an uniformly distributed random number in the range $[0,1]$, and ω_i is the inertia weight for the i^{th} particle.

A suitable selection of inertia weight ω and constants c_1 and c_2 is crucial in providing a balance between the global and local search in the flying space. The particle velocity at any instant is limited to v_{max} .

Since the PSO algorithm depends on the objective function to guide the search, the choice is made as

$$O_i = \frac{1}{NS} \sum_{k=1}^{NS} (\hat{y}(t) - y_d(t))^2 \quad (20)$$

NS= Represents the number of samples on which Root Mean Square Error is compared and the fitness of the i^{th} particle becomes

$$F_i = \frac{1}{1 + O_i} \quad (21)$$

For ω_i , start by finding the variance of the population fitness as

$$\sigma^2 = \sum_{i=1}^M \left(\frac{F_i - F_{\text{avg}}}{F} \right)^2 \quad (22)$$

F_{avg} = average fitness of the population

F_i = fitness of the i^{th} particle in the population, and

M = total number of particles

$$F = \begin{cases} \max\{F_i - F_{\text{avg}}\}, i=1,2,\dots,M & \text{if } \max\{F_i - F_{\text{avg}}\} > 1 \\ 1, & \text{if } \max\{F_i - F_{\text{avg}}\} < 1 \end{cases} \quad (23)$$

If σ is found to be large, the population will be a random searching mode, while for small σ the solution tends towards a premature convergence and will give the local best position of the particles. To circumvent this problem the inertia weight is changed in the following way:

$$\omega(k) = \lambda \omega(k-1) + (1-\lambda) \sigma^2 \quad (24)$$

A forgetting factor λ of 0.9 is used here.

Another alternative will be

$$\omega(k) = \lambda_1 \omega(k-1) + \text{rand}() / 2 \quad (25)$$

with $0 \leq \lambda_1 \leq 0.5$

Where $\text{rand}()$ is a random number between (0,1). Besides the influence of the past velocity of a particle on the current velocity is chosen to be random and the inertia weight is adapted randomly depending on the variance of the fitness value of a population.

The parameters used for LLWNN model with PSO as a learning algorithm are given in Table1.

Table 1 Parameters used for all three models (PSO)

Parameters	Values
Population size P	30
Maximum No. of generation G	200
Dimension or No. of variables D(LLWNN)	56
No. of consecutive generation for which no improvement is observed	15
Acceleration co-efficient C_1, C_2	2.1, 2.1
Lower bound	0.0001
Upper bound	1

4. INPUT DATA ANALYSIS

The Indian Rupee against US Dollar is taken as the experimental data. The total number of samples used is 9414, from 1st February 1973 to 16th July 2010. The used dataset is shown in Figure 3. Two important technical indicators, Moving Average and Exponential Moving Average are taken into consideration along with the exchange rate data. The used technical indicators and the formula are given in Table 2. In this model data are scaled between 0 and +1 as all the samples are divided by the maximum input value of the dataset. Scaling is not mandatory but it will help to improve the performance. The data is distributed into two, one dataset for training and the other for testing.

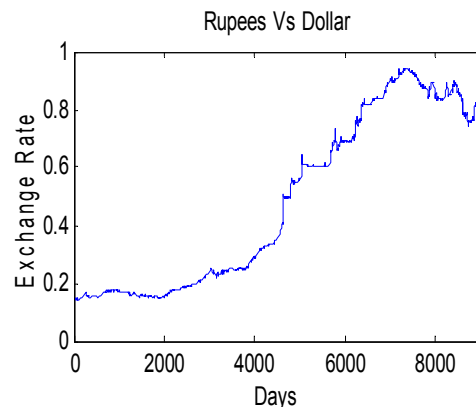


Fig. 3 Normalized Exchange Rate Data

Table 2: Technical Indicators with Formula

Statistical Parameter	Formulae
1.Moving Average	$\frac{1}{N} \sum_{i=1}^N x_i$, N=No. of Days, x _i =Hourly Price;
2.Exponential Moving Average	(P x A)+(Previous EMA x(1-A)); $A = \frac{2}{N+1}$; P=Current Price; A=Smoothing factor; N=Time Period;

The Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE) are used to measure the performance of both LLWNN and MLP models. The MAPE and RMSE are defined as

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100 \quad (26)$$

$$RSME = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (27)$$

where y_i is the predicted value and \hat{y}_i is the desired value. N is the total number of test data. Gradient descent algorithm is used to optimize the parameters of both the models. As LLWNN has given better results as compared to MLP model, the former is integrated with PSO to get more accurate results.

5. OBSERVED OUTPUT

Table 3 demonstrates the comparison of MAPE for all type of experiments taken in this paper.

Table 3. Comparison of MAPE during Testing

Model	MAPE (1 day ahead)	MAPE (1 month ahead)	MAPE (6 months ahead)	MAPE (12 months ahead)
MLP	2.68	3.05	3.93	5.93
LLWNN	1.23	2.33	2.94	3.67
PSO based LLWNN	0.59	0.98	1.35	1.81

6. ANALYSIS OF OUTCOME

From the graphs and the tables given above, three observations can be made: (1) LLWNN model is producing better results as compared to the MLP model irrespective of any time horizon chosen out of the four, as discussed in the study. (2) The proposed LLWNN model when integrated with PSO, giving more accurate results than the traditional derivative based backpropagation learning algorithm. (3) Prediction accuracy is inversely proportional to time horizon: increase in time horizon shows decrease in prediction accuracy. Fig. 8 shows the convergence speed of all time horizons discussed in the study for Rupees Vs Dollar dataset.

7. SUMMARY AND CONCLUSION

This paper used an efficient hybrid forecasting model for the currency exchange time series data. Results of the experiments show that PSO based Local Linear Wavelet Neural Network model can forecast currency exchange rate time series data for one day, one month, six months and twelve months in advance. After comparing the result with other time series models like MLP, it is proved that PSO based LLWNN model is superior in terms prediction accuracy and convergence speed. MAPE obtained from LLWNN model indicates that it can be used as a feasible alternative for exchange rate prediction for all stake holders in currency exchange market.

REFERENCES

- [1] J. Hann, N. Kamber, (2001) "Data mining concepts and techniques". San Francisco; Morgan Kaufmann Publishers.
- [2] Sira-Ramirez Herbertt, Colina-Morles Elizer, Rivas-echeverria Francklin: (2000) "Sliding Mode-based adaptive learning in dynamical filter-weights neurons", International Journal of Control, Vol. 73, No. 8.
- [3] Chen Yuehui, Yang Bo, Dong Jiwen: (2005) "Time Series Prediction using a local linear wavelet neural network". Elsevier B.V.
- [4] Majhi Banshidhar, Shalabi Hasan and Fathi Mowafak: (2005) "FLANN Based Forecasting of S&P 500 Index" Information Technology Journal.
- [5] Emam Ahmed (2008) "Optimal Artificial Neural Network topology for Foreign Exchange Forecasting" ACM-SE, Auburn, AL, USA, 978-1-60558.
- [6] Yuehui Chen, Xiaohui Dong, Yaou Zaho (2005) " Stock Index Modeling using EDA



- based Local Linear Wavelet Neural Network”
IEEE, 0-7803-9422.
- [7] Chen Y H (2000) “Evolving the wavelet neural networks for system identification”-
Proceeding of international conference on electrical Engineering.
- [8] Ravichandran. K.S., Thirunavukarasu P., Nallaswamy R. and Babu R. (2005):
“Estimation on Return on Investment in share market through ANN”, Journal of Theoretical and Applied Information Technology.
- [9] Naeini Mahdi Pakdaman, Taremi Hamidreza and Hashemi Homa Baradaran, “Stock Market Value Prediction Using Neural Networks”,
IEEE 978-1-4244-7818, 2010.
- [10] Majhi Banshidhar, Shalabi Hasan and Fathi Mowafak, “FLAN Based Forecasting of S&P 500 Index” Information Technology Journal, 4(3): 289-292, Asian Network for Scientific Information 2005.
- [11] Eberhart Russell & Kennedy James (1995) “A New Optimizer Using Particle Swarm Theory”
Sixth International Symposium on Micro machine and Human Science, 0-7803-2676.