Computer Science • 21(4) 2020 https://doi.org/10.7494/csci.2020.21.4.3474

Kishore Kumar Sahu
Sarat Chandra Nayak
Himansu Sekhar Behera

FORECASTING CURRENCY EXCHANGE RATE TIME SERIES WITH FIREWORKS ALGORITHM-BASED HIGHER ORDER NEURAL NETWORK, WITH SPECIAL ATTENTION TO TRAINING DATA ENRICHMENT

Abstract *Exchange rates are highly fluctuating by nature; thus, they are difficult to forecast. Artificial neural networks (ANNs) have proven to be better than statistical methods. Inadequate training data may lead the model to reach sub-optimal solutions, resulting in poor accuracy (as ANN-based forecasts are data-driven). To enhance forecasting accuracy, we suggests a method of enriching training datasets through exploring and incorporating virtual data points (VDPs) by an evolutionary method called the fireworks algorithm-trained functional link artificial neural network (FWA-FLN). The model maintains a correlation between current and past data, especially at the oscillation point on the time series. The exploration of a VDP and forecast of the succeeding term go consecutively by FWA-FLN. Real exchange rate time series are used to train and validate the proposed model. The efficiency of the proposed technique is related to other similarly trained models and produces far better prediction accuracy.*

Keywords exchange rate, virtual data point, interpolation, artificial neural network, fireworks algorithm, functional link neural network

Citation Computer Science 21(4) 2020: 463–488

1. Introduction

The foreign exchange market plays an important role in the financial and economic sector. Due to its elevated liquidity, it is difficult to analyze exchange rates. The enormous amount of trading, the quantity of data produced, and the continuous operation of the market make it hard to manage [27,29]. It is a complicated black box scheme with restricted data about its inner mechanism [4].Any event having an effect on one currency value will propagate through the market by changing other exchange rates [2, 19, 21]. Earlier studies [19, 20, 23, 37] demonstrate that actual FOREX rates ought to mimic random walks. Later the concept was contradicted by Clark and West [8, 9] in their hypothesis. Machine-learning methods [18, 28] are commonly used in exchange rate prediction, because they make machines learn to acknowledge unknown trends that can affect exchange rate rises and falls. Various researchers demonstrated the modeling of financial markets through machine-learning methods such as artificial neural networks (ANNs) [48], support vector machines (SVMs) [25], fuzzy neural networks (FNNs) [26], higher-order neural networks (HONNs) [38–40], and genetic algorithms (GAs) [6, 35].

An effective ANN-based forecast depends on searching unknown weights. To increase the efficiency of a network, several research-recommended meta-heuristic approaches such as artificial fish swarm algorithms (AFSs) [44], shuffled frog-leaping (SFL) [12, 14], particle swarm optimization (PSO) [16], differential evolution (DE) [13], differential harmony searches (DHS) [15, 17], chemical reaction optimization (CRO) [41, 42], and so on for ANN training. Although the back propagation algorithm is very common for ANN training, its susceptibility to local optima and slow convergence rate are the major drawbacks associated with it. Meta-heuristics have been proven to be a better alternative. The better performance of PSO over back propagation-based ANN training have been established [11, 45, 46]. A hybridization of PSO and the adaptive radial basis function for exchange rate prediction has been carried out by Sermpinis et al. [43]; the proposed hybrid model outperformed others in terms of efficiency and accuracy.

The fireworks algorithm (FWA) is a meta-heuristic that simulates the phenomenon of fireworks explosions at night [47]. Like other nature-inspired optimization, it is also a population-based evolutionary algorithm. It tries to find the best fit solution in the search space through the explosion of fireworks. Several applications of FWA are found in the literature for solving real data-mining problems. In the meantime, there are a few improved and enhanced versions of FWA that have been proposed; their superiority has been established. However, its application toward financial time series is limited.

Usually, ANN-based forecasts are data-driven models. To gain knowledge from data through the training process, they require an ample number of training instances. The inadequate volume of training data makes the model partially trained and may reduce its generalization and approximation ability. For many real-life situations, insufficiency in the training data and/or weak correlation among the data points are common phenomena. Arbitrary fluctuations in financial data make it complicated to predict from past data. In a financial time series, future values are predicted by observing the behavior of the previous data. Previous data points on a time series (particularly those at a fluctuation point) may not contribute to the detection of the prediction trend. Rather, these data points have an adverse affect on the forecasting trend, which leads to producing sub-optimal solutions. The far-off data points in a training sample are to be replaced with close-enough data point for improving the prediction accuracy of the model.

A few deterministic approaches [1, 3] found in the literature have attempted to augment the training volume by adding artificial training examples. However, this method may add noise to the training data and reduce its forecasting accuracy. Taeho Jo [24] proposed a virtual term-generation scheme to generate these terms in between two consecutive data points of time-series data. The effects of this scheme on multivariate time series prediction were validated using the back-propagation learning algorithm. The occurrence of new artificial data points along with the existing data may preserve the inclination to a certain extent. Several deterministic as well as stochastic methods have been proposed in the literature for ordinary time series analysis and were found effective. However, a financial time series behaves quite differently than an ordinary time series. The existing methods generate new training data by following the local interpolation of successive data points. Therefore, they may not be efficient enough to realize the rise and fall that constantly occurs in the case of stock movement. A few attempts have been found toward financial forecasting by applying deterministic and stochastic methods [31, 33]. The authors in [33] explored and incorporated virtual data positions (termed VDPs) to an original financial time series. They developed models from an extended time series where the model parameters were fine-tuned by back-propagation and genetic algorithms separately. The superiority of GA-based training along with the VDPs has been established. Nayak et al. [31, 34] proposed two approaches where the VDPs are explored by evolutionary optimization techniques. However, the computation of such VDPs, the complexity of MLP, and the GA-based training increases the computational burden.

The objective here is to enhance the accuracy of an ANN-based forecast by enriching a training data-set with VDPs.To beat the computational overhead stated in[40], we adopted FWA and FLN; thus, we developed a model called FWA-FLN-VDP. Due to the faster convergence ability of FWA and the lower complexity and better generalization capability of FLN, FWA-FLN-VDP possesses lower computational complexity. A performance comparison of models with and without the adoption of VDPs has been carried out. The calculations of VDPs are not performed manually; rather, they have been derived by the FWA-FLN model. The data required for the exploration/search are presented to the model; the model then evaluates an output that is treated as a VDP for the subsequent training data. The same model is made to be used for VDP exploration as well as for subsequent data-point prediction. The proposed model is validated on forecasting six real exchange rate series from South Asian countries.

Based on the above study, this article aims to achieve the objectives as follows:
- Adoption of higher-order neural network (i.e., FLN) with lower computational complexity as contrast to multi-layer neural networks.
- Hybridization of FWA and FLN to design robust forecasting model (FWA-FLN).
- Employ FWA-FLN to explore VDPs and predict next data point for exchange rate time series with improved accuracy.
- Extensive simulation studies on six real exchange rate time series.
- Statistical significance test for rejecting null hypothesis of no variance between models.

This article is organized into six sections. An introduction about exchange rate forecasting, ANN, the impact of training data on ANN, and training data enrichment are presented in Section 1. The utilized methods and materials are explained in Section 2. The proposed FWA-FLN-VDP approach is explained in Section 3. Section 4 elaborates the results, analysis, and statistical test. Section 5 concludes the research.

2. Methods and materials

This section discusses base methods such as FWA and FLN as well as the data for experimentation. The detail about these is beyond the scope of this article. An extensive statistical analysis of exchange rate series is carried out here. The base papers are cited wherever required.

2.1. Fireworks algorithm

FWA is an optimization technique that mimics the explosion procedure of fireworks [47]. It tries to select a definite number of positions in the search space for the explosion of fireworks to produce a set of sparks. Positions with qualitative fireworks are considered for the subsequent generation. The mechanism continues iteratively up to the desired optimal or reaching the halting criterion. This process is mainly comprised of three steps: setting up N fireworks at N selected positions/locations, obtaining the positions/locations of the sparks after explosions and estimating them, and stopping upon reaching an optimal location or selecting N other locations for the next generation of explosions. The explosion of fireworks can be considered to be the search process of a local space. According to basic FWA, with following parameters:
- x_i represents the elementary fireworks,
- A_i represents the amplitude of explosion,
- s_i represents number of sparks

are defined as follows:

$$A_i = \hat{A} \cdot \frac{f(x_i) - f_{min} + \varepsilon}{\sum_{j=1}^{p} \left(f(x_j) - f_{min} \right) + \varepsilon} \tag{1}$$

$$s_i = \frac{m \cdot f_{max} - f(x_i) + \varepsilon}{\sum_{j=1}^{p} \left(f_{max} - f(x_j) \right) + \varepsilon}, \tag{2}$$

where $\hat{A}$ is the maximum explosion amplitude, f_{max} and f_{min} are the maximum and minimum objective function values among the p fireworks, respectively, m is a controlling parameter for the total number of sparks generated by the fireworks, and ε is a constant used to avoid a zero division error. Bounds are imposed on s_i to overcome the devastating effects of marvelous fireworks as follows:

$$s_i = \begin{cases} s_{max}, \ if \ s_i > s_{max} \\ s_{min}, \ if \ s_i < s_{min} \\ s_i, \ otherwise \end{cases} \tag{3}$$

The location of each spark x_j generated by x_i is calculated by setting z directions randomly and setting component x_j^k for each dimension k based on x_i^k, where $1 \leq j \leq s_i, \ 1 \leq k \leq z$.

The setting of x_j^k can be done in two ways (as follows):

1. For most sparks, a displacement is added to x_j^k as:

$$x_j^k = x_i^k + A_i \cdot rand\,(-1, 1) \tag{4}$$

   where $rand(-1, 1)$ generates uniformly distributed random numbers within a range of $(-1, 1)$.

2. To maintain diversity, an explosion coefficient based on Gaussian distribution is applied to x_j^k for a few specific sparks as follows:

$$x_j^k = x_i^k \cdot Gaussian(1, 1) \tag{5}$$

When a new location falls out of the search space, it is mapped to the potential space as follows:

$$x_j^k = x_{min}^k + \left|x_j^k\right| \ \% \ \left(x_{max}^k - x_{min}^k\right) \tag{6}$$

where "%" is the modulo operator.

The next step is the selection of another N location for the fireworks explosion. This step always keeps the current best location x^* for the next generation. The remaining $N - 1$ locations are considered on the basis of their distance from other locations. The distance between location x_i and other locations (K) can be calculated as the sum of the Euclidean distance between them and as follows:

$$Distance \ (x_i) = \sum_{j \in K} \|x_i - x_j\| \tag{7}$$

Location x_i is selected for the next generation based on a probability value as follows:

$$prob \ (x_i) = \frac{Distance \ (x_i)}{\sum_{j \in K} Distance \ (x_j)} \tag{8}$$

Based on the above concepts, the basic FWA is formulated and represented in Algorithm 1.

Algorithm 1. FWA Framework

Select N locations randomly for fireworks;

while (stopping criteria == false)

Set off N fireworks at N locations

for each fireworks element x_i

Calculate number of sparks s_i using Eq. 3

Obtain locations of s_i sparks of firework x_i using Eq. 4.

end for

for k = 1: m

Select firework element x_j randomly

Generate a specific spark using Eq. 5.

end for

Select best location x^* and keep it for the next generation

Select remaining *N-1* locations randomly based on a probability using Eq. 8.

end while

Since the sparks suffer from the power of the explosion, they move along z directions simultaneously. This makes FWA achieve a faster convergence. Also, it avoids premature convergence with the two types of spark-generation methods and specific location-selection method [47]. The advantages of FWA over standard PSO and its improved variants are demonstrated in [47].

2.2. Functional link neural network

Pao introduced the FLN model [36]. It is basically considered to be a neural network of a single layer. This model is computationally effective, as the single-layer involves fewer computations. FLN non-linearity is generated by expanding input patterns using nonlinear basis functions such as trigonometric, power series, or Chebyshev--type nonlinear functions.

Another benefit of FLN is that it reduces the trouble of determining the network structure. The FLN-based forecasting is shown in Figure 1.

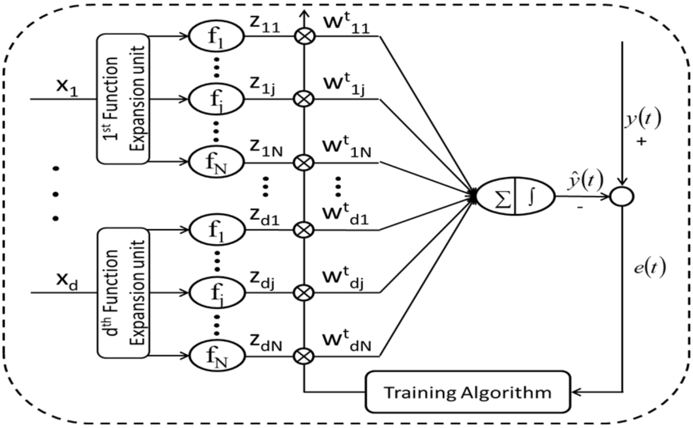

Figure 1. FLN-based forecasting model

2.3. Data

The proposed FWA-FLN-VDP approach is validated on predicting the future prices of six real exchange rate time series. The exchange rates time series are the Indian Rupee to US Dollar (INR/USD), Chinese Yuan to USD (CNY/USD), Pakistan Rupee to USD (PKR/USD), Bangladesh Taka to USD (BDT), Nepalese Rupee to USD (NPR/USD), and Sri Lankan Rupee to USD (LKR/USD). The data is collected from https://www.exchangerates.org.uk/ during the period of February 7, 2019, through August 5, 2019. Each dataset consists of around 150 data points forming time series. Before the experimentation, we conduct a statistical analysis; the statistics are summarized in Table 1.

Table 1

Descriptive statistics from exchange rate time series

STATISTICS	INR/USD	PKR/USD	BDT/USD	CNY/USD	NPR/USD	LKR/USD
Minimum	68.4103	138.6879	83.6815	6.6872	109.3439	174.1359
1st Quartile	69.0967	140.2255	84.0908	6.7164	110.5007	175.7850
Mean	69.6291	147.3221	84.2992	6.8055	111.4913	176.6901
Median	69.4358	141.6612	84.3424	6.7811	111.1750	176.4256
3rd Quartile	69.9720	156.9623	84.4915	6.8848	112.3570	177.7124
Inter Quartile range	0.8752	16.7368	0.4007	0.1684	1.8563	1.9274
Maximum	71.3828	163.5410	85.0451	7.0497	114.3323	180.0727
Standard deviation	0.7640	8.2301	0.2609	0.0883	1.2569	1.4862
Variance	0.5837	67.7350	0.0681	0.0078	1.5799	2.2089
Skewness	0.5081	2.3403	-0.1626	1.9776	-1.7208	-0.6131
Kurtosis	5.3777	16.6110	5.9754	15.4469	19.2963	6.9212
JB statistic (h=1)	40.9516	1.3034e+03	54.8735	1.0732e+03	1.6992e+03	106.2006

We see that the minimum, mean, standard deviation, and variance of CNY/USD is much smaller, followed by INR/USD. For others, these values are found to be higher.

The BDT/USD, NPR/USD, and LKR/USD series are negatively skewed. Six time series are shown in Figure 2. Similarly, the log-returns of all data are plotted in Figure 3.

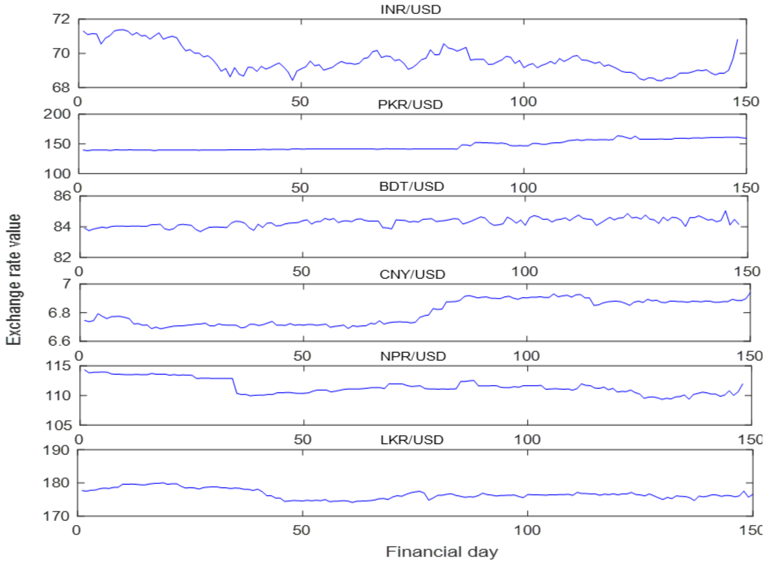

Figure 2. Exchange rate series

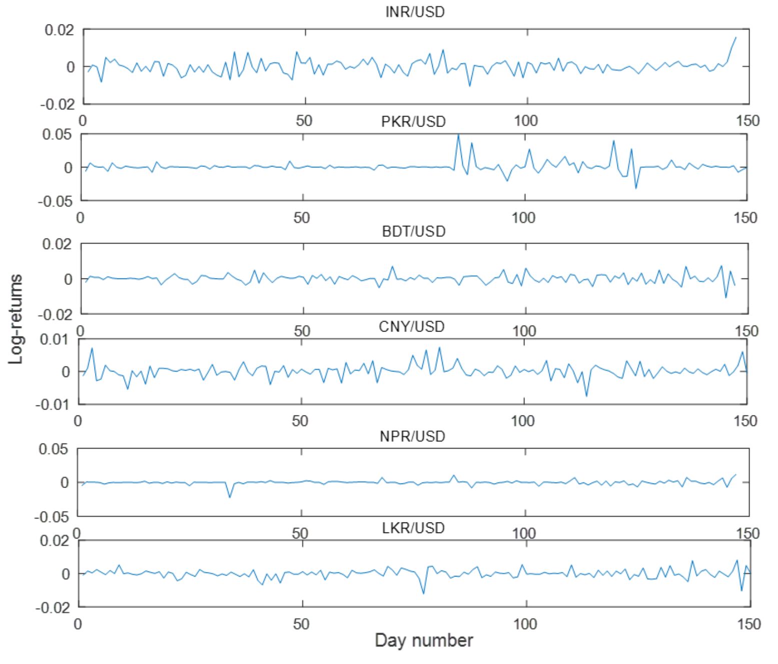

Figure 3. Log-returns

Usually, financial time series show randomness and possess no serial correlation nor heteroskedasticity. To be sure, we tested six time series for randomness [5, 7, 10]; these results are summarized in Table 2.

Table 2
Computed p-values from randomness tests

Exchange rate series	Log-returns	Log-return2
Box and Pierce		
INR/USD	0.9868	0.9638
PKR/USD	0.0125	0.4703
BDT/USD	8.0381	0.0596
CNY/USD	0.9489	1.0000
NPR/USD	0.8247	1.0000
LKR/USD	0.2149	0.7118
Cox and Stuart		
INR/USD	0.4825	0.2418
PKR/USD	1.000	0.2198
BDT/USD	1.000	0.0010
CNY/USD	1.000	0.6442
NPR/USD	0.3491	0.0010
LKR/USD	0.0853	0.0209
Difference sign		
INR/USD	0.4095	0.2143
PKR/USD	0.0508	0.0028
BDT/USD	0.1869	0.0021
CNY/USD	0.1036	0.0028
NPR/USD	0.2482	0.0024
LKR/USD	1.0000	0.0028
Runs test		
INR/USD	0.5290	0.4985
PKR/USD	0.9706	0.1149
BDT/USD	0.9241	0.0441
CNY/USD	0.3242	1.0000
NPR/USD	0.3623	0.2681
LKR/USD	0.7433	0.0057

All of the tests were conducted on the log-returns and squared log-returns of the time series. The time series are tested for no serial correlation [22]; these results are summarized in Table 3. This evidence is in support of the auto-correlation function (ACF) in Figure 4 and partial auto-correlation function test (PACF) in Figure 5 for six currency pairs. The histograms of the daily log-returns against the theoretical normal distribution are plotted in Figure 6. These observations are in support of the fact that exchange rate time series are nonlinear in nature and do not exhibit serial correlations (thus, they are difficult to forecast).

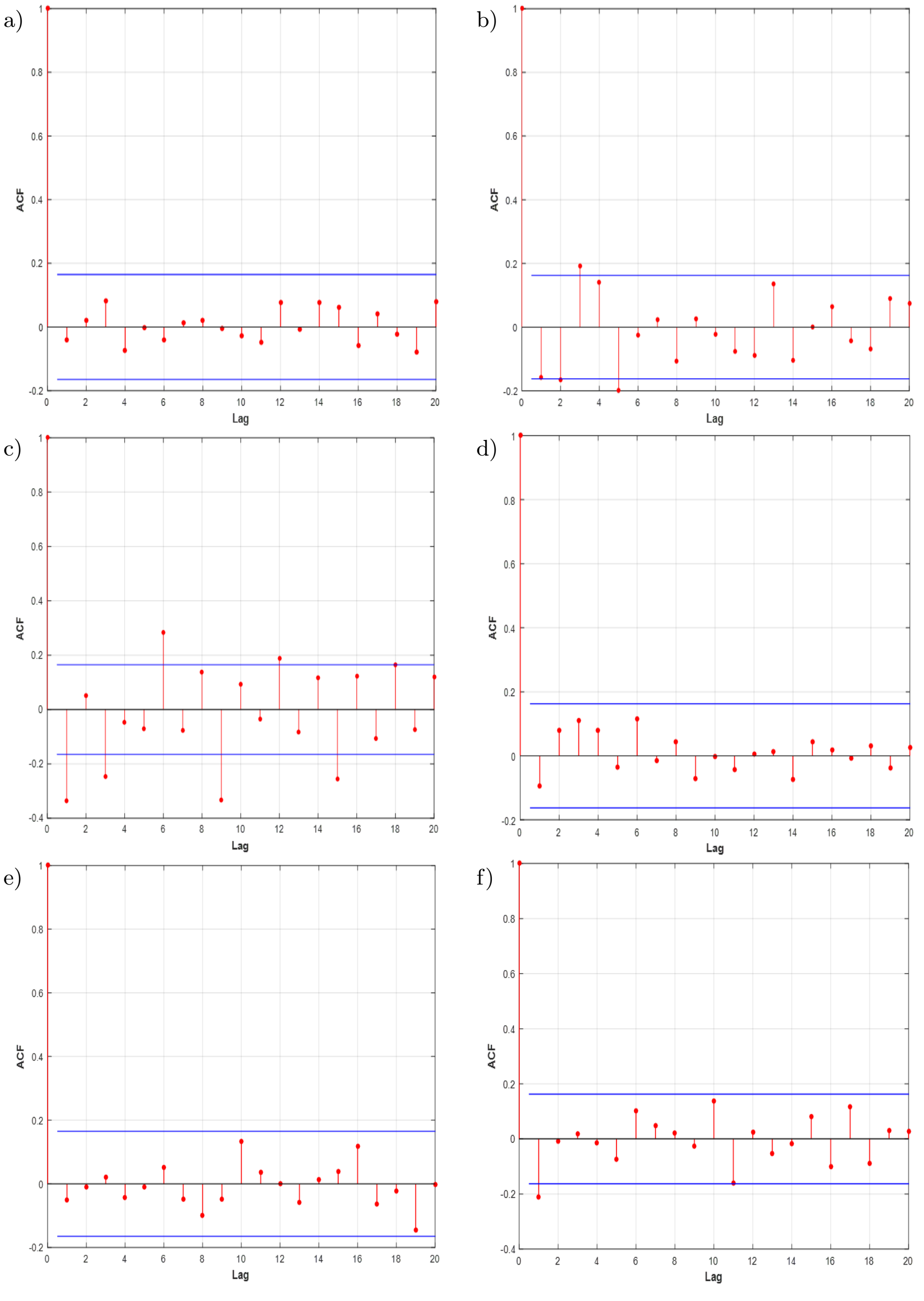

Figure 4. Auto-correlation function (ACF) for six currency pairs: a) INR/USD; b) PKR/USD; c) BDT/USD; d) CNY/USD; e) NPR/USD; f) LKR/USD

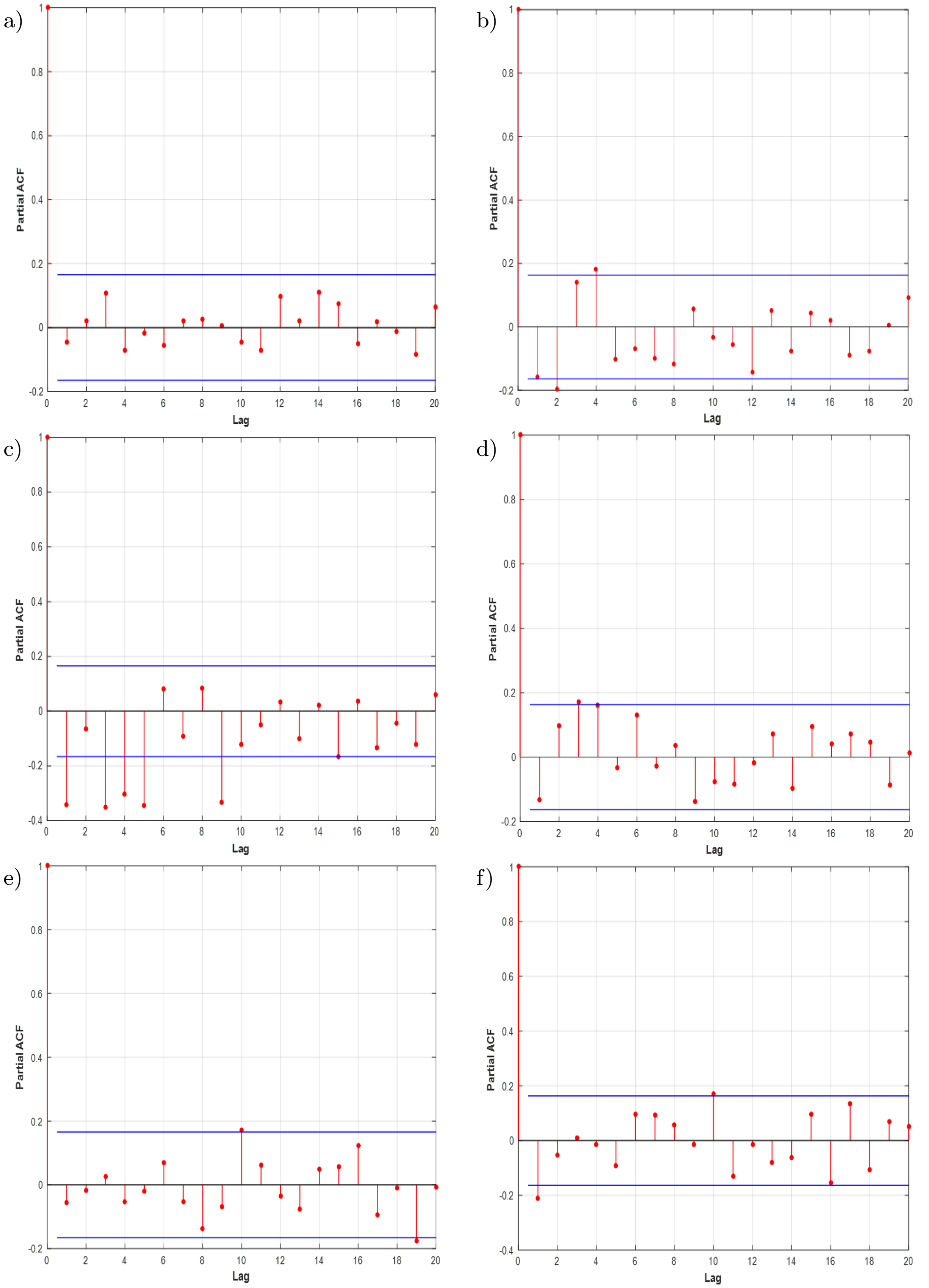

Figure 5. Partial auto-correlation function (ACF) for six currency pairs: a) INR/USD; b) PKR/USD; c) BDT/USD; d) CNY/USD; e) NPR/USD; f) LKR/USD

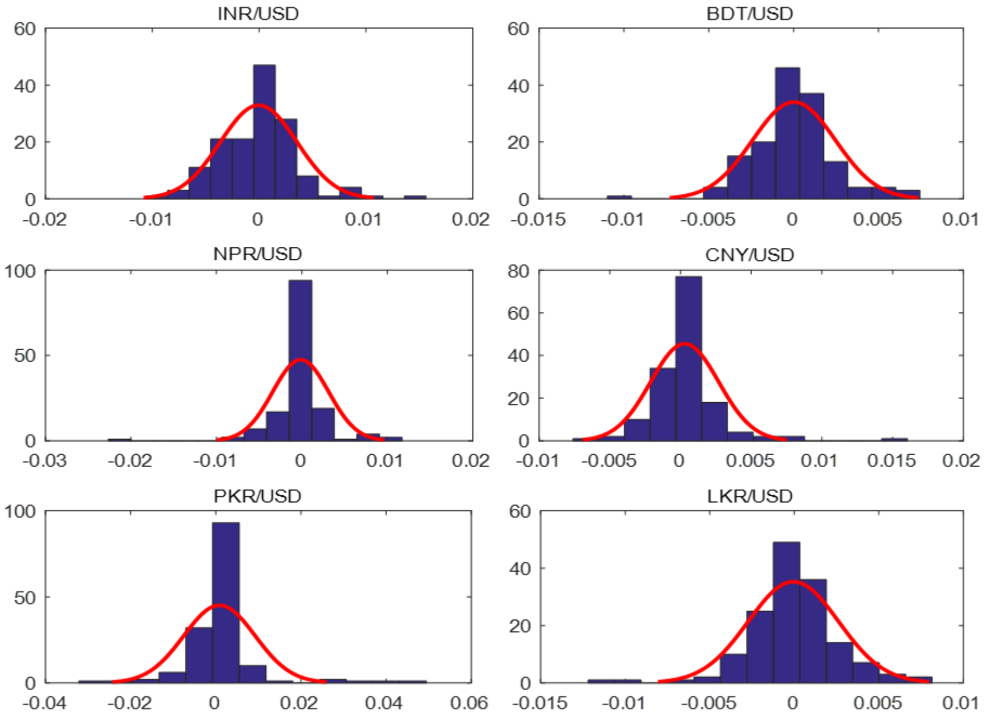

Figure 6. Histogram of log-returns

Table 3
Computed p-values from tests for no serial correlation

Exchange rate series	Durbin and Watson test value
INR/USD	1.7447e-11
PKR/USD	9.3620e-12
BDT/USD	1.9664e-11
CNY/USD	9.4633e-12
NPR/USD	1.7286e-11
LKR/USD	9.4812e-12

3. FWA-FLN-based VDP exploration and forecasting

This section describes the proposed FWA-FLN-based VDP exploration method and forecasting. Let us start the description of the proposed approach from the following simple example. A sliding window of a fixed size is used to generate the training data from the original time series.

Suppose that the training and test dataset generated by the sliding window of size three is shown as follows:

$$\underbrace{\begin{matrix} x(i) & x(i+1) & x(i+2) & \vdots \\ x(i+1) & x(i+2) & x(i+3) & \vdots \\ x(i+2) & x(i+3) & x(i+4) & \vdots \end{matrix}}_{Training\ data} \quad \underbrace{\begin{matrix} x(i+3) \\ x(i+4) \\ x(i+5) \end{matrix}}_{Target}$$

$$\underbrace{x(i+3) \quad x(i+4) \quad x(i+5)}_{Test\ data} \ \vdots \ \underbrace{x(i+6)}_{Target}$$

The training and test data after the incorporation of VDPs are shown as follows. A fractional value-added such as$x(i+*.5)$ represents a VDP.

$$\underbrace{\begin{matrix} x(i) & x(i+0.5) & x(i+1) & x(i+1.5) & x(i+2) & x(i+2.5) & \vdots \\ x(i+.5) & x(i+1) & x(i+1.5) & x(i+2) & x(i+2.5) & x(i+3) & \vdots \\ x(i+1) & x(i+1.5) & x(i+2) & x(i+2.5) & x(i+3) & x(i+3.5) & \vdots \\ x(i+1.5) & x(i+2) & x(i+2.5) & x(i+3) & x(i+3.5) & x(i+4) & \vdots \\ x(i+2) & x(i+2.5) & x(i+3) & x(i+3.5) & x(i+4) & x(i+4.5) & \vdots \end{matrix}}_{Training\ data} \quad \underbrace{\begin{matrix} x(i+3) \\ x(i+3.5) \\ x(i+4) \\ x(i+4.5) \\ x(i+5) \end{matrix}}_{Target}$$

$$\underbrace{x(i+2.5) \quad x(i+3) \quad x(i+3.5) \quad x(i+4) \quad x(i+4.5) \quad x(i+5) \ \vdots}_{Data\ needed\ for\ exploration} \quad \underbrace{x(i+5.5)}_{VDP\ to\ be\ explored}$$

The FWA-FLN model is used to explore the VDPs as above and the prediction of the next data points alternatively. The VDPs are merged into the original training data, followed by normalization. A location (individual) of FWA can be viewed as a potential weight and bias vector for FLN. In the beginning, a set of such locations is initialized; for each of these locations, two types of explosion are carried out (as discussed in Section 2). Subsequently, the training samples are produced to the FWA--FLN model, the estimations are generated, and the fitness of the individuals in the population is evaluated. The exploration and exploitation of the search space are achieved by these explosion methods. The locations are then evaluated in terms of error-signal generation. The location with the lowest error signal is considered to be the best location. The selection process is then carried out with the inclusion of this best location and remaining locations as described in Section 2. The same process continues for a finite number of iterations; finally, the best individual is identified and considered as the optimal weight and bias vector for the FLN model. Now, the test data (along with the optimal weight and bias vector) is presented to the model. Based on this, the model generates an estimation. At this point, the model must decide whether to use this estimation value as a VDP or forecasted output. Here, a counter value is checked; if the counter value is odd, then the dataset goes for an exploration of the VDP along with the trained weight and bias vector. The output of the model is considered to be the VDP and merged to the subsequent training data set. If the counter value is even, the target signal is produced at the output neuron, and the error value is calculated by comparing it with the estimated signal. This error value is recorded for computing the performance of the proposed model, and the estimated value is used as the forecasted output. The iteration value for subsequent training is then set to a lower value, as the model is now adaptively trained. Based on the above explanation, the high-level FWA-FLN-based VDP exploration and forecasting is presented by Algorithm 2.

Algorithm 2: FWA-FLN-based VDP exploration and forecasting

1. Set *MaxIteration*, *PopSize*, and FWA parameters
2. Initialization of population of *PopSize*
3. Form *TrainData* and *TestData* /*using sliding window from original time series*/
4. **While** (more *TestData*)
   (a) Set *Counter* to 1
   (b) Normalization of *TrainData* and *TestData* /*using sigmoid method*/
   */*Training phase*/*
   (c) Set *Iteration* to 0
   (d) **While** (*Iteration* $< =$ *MaxIteration*)
   Present *TrainData*, and FWA *individual* to FLN model
   Evaluate *individuals* for fitness and calculate *error* signal
   Apply FWA search operators for exploration and exploitation
   Select best fit *individual* and update population
   Iteration $=$ *Iteration* $+ 1$
   (e) **End while**
   */*Test phase*/*
   (f) Present *TestData*, and best-fit *individual* to FLN model
   (g) **If** *Counter* $=$ Odd
   Estimate and preserve *VDP*
   Incorporate *VDP* to form next *TrainData*
   (h) **Else**
   Estimate and preserve *error* /*the error will be used to measure the performance of the model*/
   Incorporate *VDP* to form next *TrainData*
   (i) *Counter* $=$ *Counter* $+1$
5. **End while**
6. Set *MaxIteration* to a small value /*for adaptive learning*/

As mentioned earlier, the *TrainData* and *TestData* are generated from the original time series using a sliding window technique. The size of the window is experimental. The FWA parameters are chosen as suggested in [47]. The data is normalized using sigmoid normalization [30]. Based on a counter value, model estimation is used as the VDP and forecasted value alternatively. After the first training and test dataset, the value of *MaxIteration* is set to a small value; this in effect reduces the training time of the model. In this way, the model has trained adaptively [30].

4. Experimental results

4.1. Analysis of results from FWA-FLN approach

First, we analyze the performance of the FWA-FLN model without the VDP approach. As discussed, a sliding window of a fixed size was used to form the training sets from the original time series. The data is then normalized and fed to the FWA-FLN model. The model was simulated 25 times for each training set; the average is considered for the comparative analysis. The mean absolute percentage error (MAPE) is evaluated as the performance index (Eq. 9).

$$MAPE = \frac{1}{No.of\ pattern} \sum_{i=1}^{No.of\ pattern} \frac{|Actual_i - Estimated_i|}{Actual_i} \cdot 100\% \tag{9}$$

For a comparative performance analysis of the FWA-FLN-based forecasting, we developed another three hybrid models: differential evolution-based FLN (DE-FLN), genetic algorithm-based FLN (GA-FLN), and particle swarm optimization-based FLN (PSO-FLN). We considered these because these are popular and widely applied evolutionary optimization techniques for ANN training. The training datasets for these models are the same as those of FWA-FLN. As discussed, adaptive training was conducted for the four models. The MAPE statistics from one-day-ahead, one-week-ahead, and one-month-ahead forecasting are summarized in Tables 4–6, respectively. The lowest MAPE values are highlighted in boldface.

It can be observed from Table 4 that the FWA-FLN model generated the lowest MAPE values for six time series. Compared to the others, its performance is better. In the case of one-week-ahead prediction, the MAPE values of FWA-FLN are all better except for NPR/USD, where PSO-FLN performed better. Similar results were obtained for one-month-ahead forecasting except for PKR/USD and LKR/USD, where GA-FLN and PSO-FLN were found to be the best performers, respectively. Considering a different forecasting horizon, it was found that the performance of each model degrades. The MAPE values from one-month-ahead prediction are found to be much higher than those from one-day-ahead and one-week-ahead prediction. However, the overall performance of FWA-FLN is found to be superior. The forecasted exchange rates of the FWA-FLN and the actual exchange rates for each financial day are compared and the plots are shown in Figures 7.

Table 4
MAPE values from one-day-ahead forecasting (without VDP)

Exchange rate series	FWA-FLN	DE-FLN	GA-FLN	PSO-FLN
INR/USD	**0.03268**	0.04929	0.04150	0.03852
PKR/USD	**0.03863**	0.03977	0.05415	0.05004
BDT/USD	**0.04876**	0.09835	0.07122	0.07383
CNY/USD	**0.03275**	0.05705	0.05973	0.10063
NPR/USD	**0.03625**	0.07258	0.10107	0.05387
LKR/USD	**0.26390**	0.29436	0.30557	0.26887

Table 5
MAPE values from one-week-ahead forecasting (without VDP)

Exchange rate series	FWA-FLN	DE-FLN	GA-FLN	PSO-FLN
INR/USD	**0.05108**	0.05963	0.06177	0.06650
PKR/USD	**0.06360**	0.08575	0.13414	0.07588
BDT/USD	**0.06275**	0.10833	0.08535	0.07795
CNY/USD	**0.04028**	0.07355	0.07223	0.13160
NPR/USD	0.05655	0.07633	0.15225	**0.05528**
LKR/USD	**0.26896**	0.32437	0.32055	0.28645

Table 6
MAPE values from one-month-ahead forecasting (without VDP)

Exchange rate series	FWA-FLN	DE-FLN	GA-FLN	PSO-FLN
INR/USD	**0.14824**	0.32960	0.32610	0.25655
PKR/USD	0.26432	0.28500	**0.19634**	0.37536
BDT/USD	**0.13271**	0.18730	0.18566	0.17027
CNY/USD	**0.24315**	0.28735	0.27299	0.30365
NPR/USD	**0.16063**	0.27500	0.24524	0.25055
LKR/USD	0.52895	0.83243	0.83213	**0.52643**

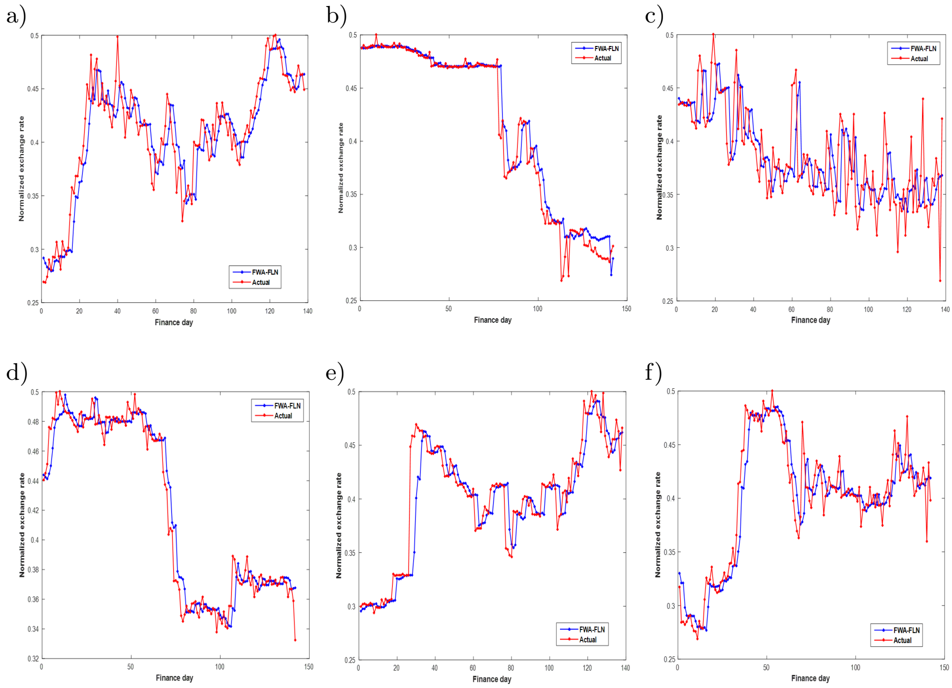

Figure 7. Rates forecasted by FWA-FLN vs. actual exchange rates for six currency pairs: a) INR/USD; b) PKR/USD; c) BDT/USD; d) CNY/USD; e) NPR/USD; f) LKR/USD

4.2. Analysis of results from FWA-FLN-VDP approach

Now, we present the results from the FWA-FLN-VDP approach. As presented in Algorithm 2, the VDPs are estimated by the FWA-FLN model. It may be noted that the same model was used for the exploration of the VDPs as well as the prediction of the next data point. The MAPE values obtained from one-day-ahead, one-week-ahead, and one-month-ahead forecasting are summarized in Tables 7–9, respectively. The lowest MAPE values are highlighted in boldface. From the one-day-ahead forecasting results, it can be observed that FWA-FLN-VDP achieved the lowest MAPE values for the INR/USD, BDT/USD, CNY/USD, and NPR/USD datasets. For PKR/USD and LKR/USD, the lowest MAPE values are obtained by DE-FLN-VDP and PSO-FLN-VDP, respectively. However, in the case of one-week-ahead forecasting, the FWA-FLN-VDP generated the lowest MAPE values for all datasets. A similar trend was shown for one-month-ahead forecasting (with the exception of PKR/USD). The estimated exchange rates by the FWA-FLN-VDP model are plotted in Figure 8. These plots show the closeness of the estimated values to the actual ones. The model follows the trend of the exchange rate time series accurately.

Table 7
MAPE values from one-day-ahead forecasting with VDP

Exchange rate series	FWA-FLN-VDP	DE-FLN-VDP	GA-FLN-VDP	PSO-FLN-VDP
INR/USD	**0.02938**	0.04375	0.03627	0.03505
PKR/USD	0.03588	**0.02994**	0.05087	0.04854
BDT/USD	**0.03975**	0.09566	0.07036	0.07085
CNY/USD	**0.03005**	0.05499	0.05575	0.09961
NPR/USD	**0.02895**	0.07011	0.10000	0.05038
LKR/USD	0.25739	0.29030	0.29235	**0.25305**

Table 8
MAPE values from one-week-ahead forecasting with VDP

Exchange rate series	FWA-FLN-VDP	DE-FLN-VDP	GA-FLN-VDP	PSO-FLN-VDP
INR/USD	**0.03775**	0.05032	0.05387	0.05855
PKR/USD	**0.05936**	0.08113	0.11006	0.06853
BDT/USD	**0.04487**	0.10070	0.07953	0.07200
CNY/USD	**0.03675**	0.07163	0.06955	0.10088
NPR/USD	**0.05023**	0.07385	0.13605	0.05264
LKR/USD	**0.25899**	0.31227	0.30038	0.26829

Table 9
MAPE values from one-month-ahead forecasting with VDP

Exchange rate series	FWA-FLN-VDP	DE-FLN-VDP	GA-FLN-VDP	PSO-FLN-VDP
INR/USD	**0.09775**	0.30894	0.31105	0.22014
PKR/USD	0.22465	0.26953	**0.18537**	0.33530
BDT/USD	**0.09975**	0.15965	0.15483	0.14382
CNY/USD	**0.20377**	0.25228	0.25373	0.30311
NPR/USD	**0.11537**	0.25385	0.20052	0.21854
LKR/USD	**0.48623**	0.75245	0.75266	0.50061

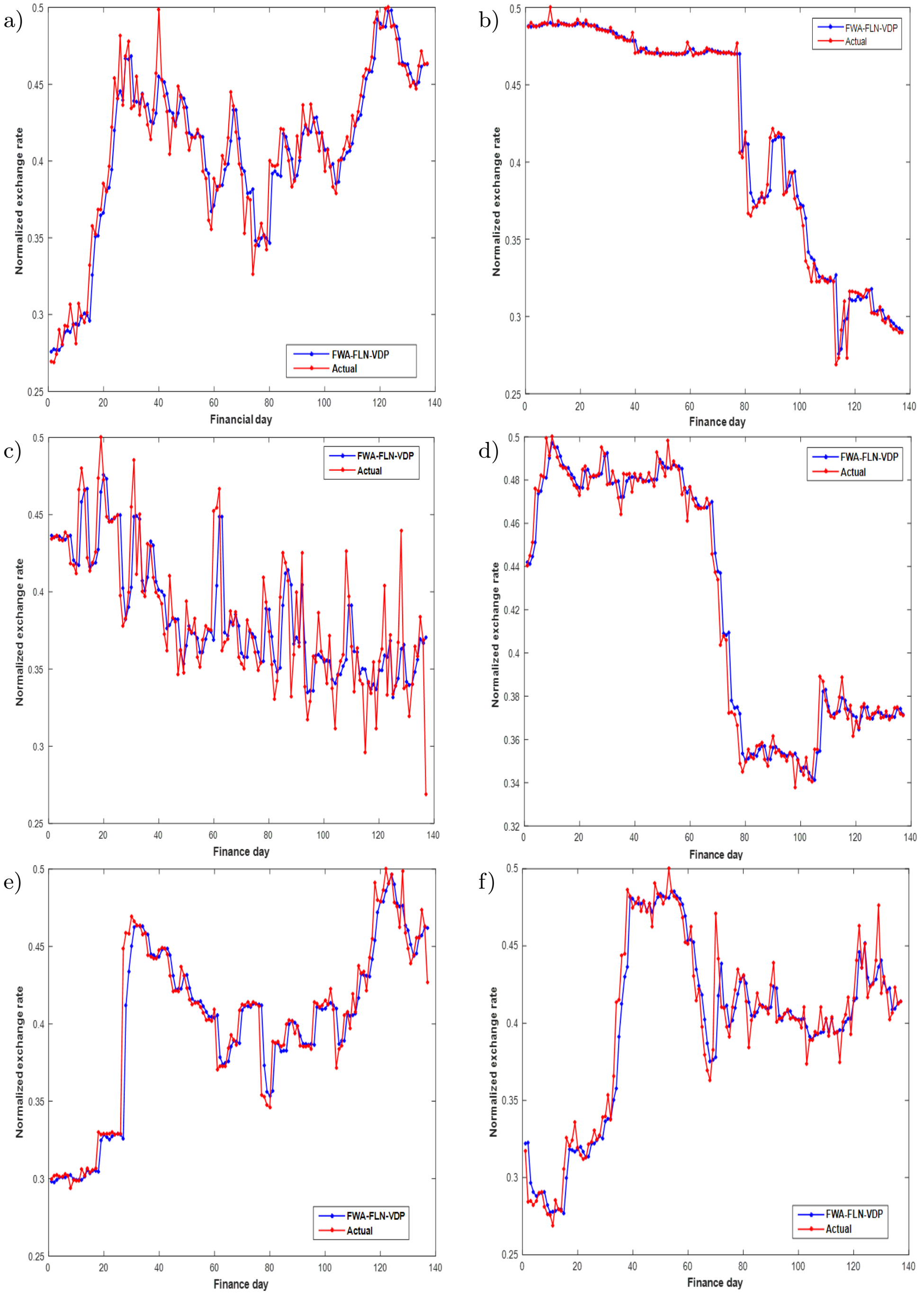

Figure 8. Forecasted by FWA-FLN-VDP v/s actual exchange rates for six currency pairs: a) INR/USD; b) PKR/USD; c) BDT/USD; d) CNY/USD; e) NPR/USD; f) LKR/USD

4.3. Further analysis

Let us discuss the advantage of the VDP approach over non-VDP. Considering a particular dataset, the MAPE gain by a VDP-approach model over a non-VDP model is calculated as follows:

$$\text{MAPE gain} = \frac{\text{MAPE of VDP model} - \text{MAPE of non VDP model}}{\text{MAPE of non VDP model}} \cdot 100\% \qquad (10)$$

The MAPE gains by VDP-based models over non-VDP-based models are shown in Figures 9 and 10. It can be clearly observed that the adaptation of the VDP approach reduced the forecasting error significantly.

Considering the six datasets and three forecasting horizons, the average performance gain of FWA-FLN-VDP over DE-FLN-VDP is 38.15738%. Similarly, its performance gain over GA-FLN-VDP and PSO-FLN-VDP are 35.92564 % and 33.34695%, respectively.

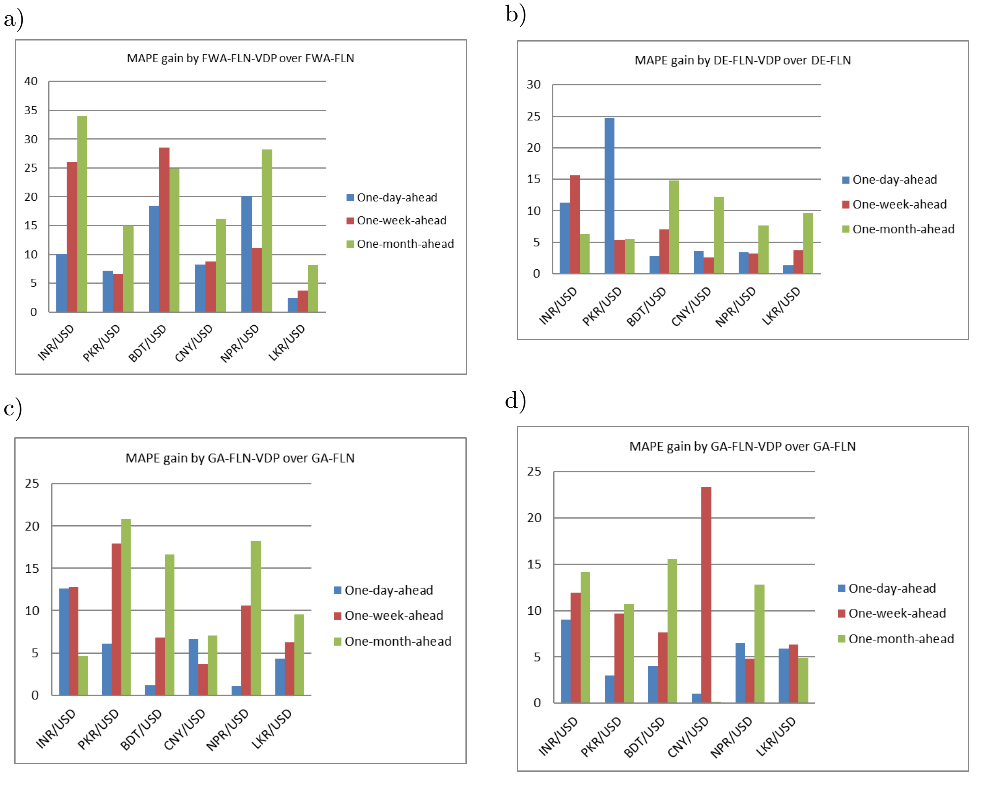

Figure 9. Performance Gain of models: a) gain by FWA-FLN-VDP over FWA-FLN; b) gain by DE-FLN-VDP over DE-FLN; c) gain by GA-FLN-VDP over GA-FLN; d) gain by PSO--FLN-VDP over PSO-FLN

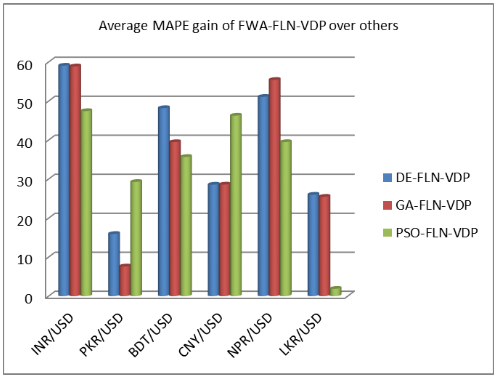

Figure 10. Average Performance gain by FWA-FLN-VDP over others

4.4. Statistical significance test

Wilcoxon's signed-rank test and Deibold-Mariano's test [32] are conducted for checking the statistical significance of the proposed model. 16. Wilcoxon signed–rank test is an alternate to t-test, used to perform repeated calculation on a sample space of pared values. It assures the consistency of the mean rank of the sample space in consideration. Diebold-Mariano test is used to confirm the indifference in the forecast values of two comparable models. Considering a 5% significance level, if the statistic falls beyond $\pm$ 1.965, then the null hypothesis of no difference will be rejected. For the sake of space, we presented the statistics from the INR/USD time series only. The computed statistics from the Wilcoxon signed-rank test are summarized in Table 10, and those from the Deibold-Mariano test are summarized in Table 11. These results are in support of the fact that the proposed FWA-FLN-VDP method is significantly different from the other methods under consideration.

5. Conclusion

In financial time series, previous data points (particularly at fluctuation points) may not contribute to the detection of inclination and have an adverse influence on the forecasting trend. The removal of far-off data points from the training sample and exploration of a few close enough points may improve the accuracy of the prediction. To improve ANN forecasting accuracy, this article suggests a method of enriching training data through the inclusion of VDPs from on-hand training data by an evolutionary method called FWA-FLN-VDP. Such VDPs are found to be capable of maintaining the correlation between the current and previous data, especially at the fluctuation points. The same model explored a VDP and predicted the next data point alternately. The proposed model synergies the faster convergence ability of FWA, the single-layer architecture and better generalization ability of FLN, and the contribution of VDPs for improved accuracy. The proposed method is applied for the prediction of six stock exchange rates. The compared experimental results and statistical significance test demonstrated the success of the proposed FWA-FLN-VDP approach as compared to other models. The computation of VDPs may increase the time complexity of the model; the exploration of mechanisms to handle this can be an extension. Also, the current work may be applied to other data-mining problems that lack sufficient training data.

References

[1] Abu-Mostafa Y.S.: Financial Applications of Learning from Hints. In: *Advances in neural information processing systems (NIPS 1994)*, pp. 411–418, The MIT Press, Cambridge, MA, 1995.

[2] Aiba Y., Hatano N., Takayasu H., Marumo K., Shimizu T.: Triangular Arbitrage in the Foreign Exchange Market. In: Takayasu H. (ed.), *The Application of Econophysics*, pp. 18–23, Springer, Tokyo, 2004.

[3] An G.: The effects of adding noise during backpropagation training on a generalization performance, *Neural Computation*, vol. 8(3), pp. 643–674, 1996.

[4] Anastasakis L., Mort N.: Exchange rate forecasting using a combined parametric and nonparametric self-organising modelling approach. *Expert Systems with Applications*, vol. 36(10), pp. 12001–12011, 2009.

[5] Bartels R.: The Rank Version of von Neumann's Ratio Test for Randomness, *Journal of the American Statistical Association*, vol. 77(377), pp. 40–46, 1982.

[6] Bhattacharyya S., Pictet O.V., Zumbach G.: Knowledge-intensive genetic discovery in foreign exchange markets. *IEEE Transactions on Evolutionary Computation*, vol. 6(2), pp. 169–181, 2002.

[7] Box G.E.P., Pierce D.A.: Distribution of Residual Autocorrelations in Autoregressive-Integrated Moving Average Time Series Models, *Journal of the American Statistical Association*, vol. 65(332), pp. 1509–1526, 1970.

[8] Clark T.E., West K.D.: Using out-of-sample mean squared prediction errors to test the martingale difference hypothesis, *Journal of Econometrics*, vol. 135(1–2), pp. 155–186, 2006.

[9] Clark T.E., West K.D.: Approximately normal tests for equal predictive accuracy in nested models, *Journal of Econometrics*, vol. 138(1), pp. 291–311, 2007.

[10] Cox D.R., Stuart A.: Some Quick Sign Tests for Trend in Location and Dispersion, *Biometrika*, vol. 42(1/2), pp. 80–95, 1955.

[11] Das G., Pattnaik P.K., Padhy S.K.: Artificial Neural Network Trained by Particle Swarm Optimization for Non-Linear Channel Equalization, *Expert Systems with Applications*, vol. 41(7), pp. 3491–3496, 2014.

[12] Dash R.: An improved shuffled frog leaping algorithm based evolutionary framework for currency exchange rate prediction, *Physica A: Statistical Mechanics and its Applications*, vol. 486, pp. 782–796, 2017.

[13] Dash R.: DECPNN: A hybrid stock predictor model using Differential Evolution and Chebyshev Polynomial neural network, *Intelligent Decision Technologies*, vol. 12(1), pp. 93–104, 2018.

[14] Dash R.: Performance analysis of a higher order neural network with an improved shuffled frog leaping algorithm for currency exchange rate prediction, *Applied Soft Computing*, vol. 67, pp. 215–231, 2018.

[15] Dash R., Dash P.K.: Efficient Stock Price Prediction Using a Self Evolving Recurrent Neuro-Fuzzy Inference System Optimized Through a Modified Differential Harmony Search Technique, *Expert Systems with Applications*, vol. 52, pp. 75–90, 2016.

[16] Dash R., Dash P.K.: Prediction of Financial Time Series Data Using Hybrid Evolutionary Legendre Neural Network: Evolutionary LENN, *International Journal of Applied Evolutionary Computation (IJAEC)*, vol. 7(1), pp. 16–32, 2016.

[17] Dash R., Dash P.K., Bisoi R.: A self adaptive differential harmony search based optimized extreme learning machine for financial time series prediction, *Swarm and Evolutionary Computation*, vol. 19, pp. 25–42, 2014.

[18] De Grauwe P., Markiewicz A.: Learning to forecast the exchange rate: Two competing approaches, *Journal of International Money and Finance*, vol. 32, pp. 42–76, 2013.

[19] Drożdż S., Kwapień J., Oświęcimka P., Rak R.: The foreign exchange market: return distributions, multifractality, anomalous multifractality and the Epps effect, *New Journal of Physics*, vol. 12(10), p. 105003, 2010.

[20] Engel C., Mark N.C., West K.D.: Factor Model Forecasts of Exchange Rates, *Econometric Reviews*, vol. 34(1-2), pp. 32–55, 2015.

[21] Fenn D.J., Howison S.D., McDonald M., Williams S., Johnson N.F.: The Mirage of Triangular Arbitrage in the Spot Foreign Exchange Market, *International Journal of Theoretical and Applied Finance*, vol. 12(08), pp. 1105–1123, 2009.

[22] Gibbons J.D., Chakraborti S.: *Nonparametric Statistical Inference: Revised and Expanded*, CRC Press, 2014.

[23] Ismailov A., Rossi B.: Uncertainty and deviations from uncovered interest rate parity, *Journal of International Money and Finance*, vol. 88, pp. 242–259, 2018.

[24] Jo T.: VTG schemes for using back propagation for multivariate time series prediction, *Applied Soft Computing*, vol. 13(5), pp. 2692–2702, 2013.

[25] Kamruzzaman J., Sarker R.A., Ahmad I.: SVM based models for predicting foreign currency exchange rates. In: *Third IEEE International Conference on Data Mining*, pp. 557–560, IEEE, 2003.

[26] Kuo R.J., Chen C., Hwang Y.: An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network, *Fuzzy Sets and Systems*, vol. 118(1), pp. 21–45, 2001.

[27] Li Q., Chen Y., Wang J., Chen Y., Chen H.: Web Media and Stock Markets: A Survey and Future Directions From a Big Data Perspective, *IEEE Transactions on Knowledge and Data Engineering*, vol. 30(2), pp. 381–399, 2017.

[28] Meade N.: A comparison of the accuracy of short term foreign exchange forecasting methods, *International Journal of Forecasting*, vol. 18(1), pp. 67–83, 2002.

[29] Nassirtoussi A.K., Aghabozorgi S., Wah T.Y., Ngo D.C.L.: Text mining for market prediction: A systematic review, *Expert Systems with Applications*, vol. 41(16), pp. 7653–7670, 2014.

[30] Nayak S.C., Misra B.B., Behera H.S.: Artificial chemical reaction optimization of neural networks for efficient prediction of stock market indices, *Ain Shams Engineering Journal*, vol. 8(3), pp. 371–390, 2017.

[31] Nayak S.C., Misra B.B., Behera H.S.: Exploration and incorporation of virtual data positions for efficient forecasting of financial time series, *International Journal of Industrial and Systems Engineering*, vol. 26(1), pp. 42–62, 2017.

[32] Nayak S.C., Misra B.B., Behera H.S.: ACFLN: artificial chemical functional link network for prediction of stock market index, *Evolving Systems*, vol. 10(4), pp. 567–592, 2019.

[33] Nayak S.C., Misra B.B., Behera H.S.: Efficient forecasting of financial time-series data with virtual adaptive neuro-fuzzy inference system, *International Journal of Business Forecasting and Marketing Intelligence*, vol. 2(4), pp. 379–402, 2016.

[34] Nayak S.C., Misra B.B., Behera H.S.: Efficient financial time series prediction with evolutionary virtual data position exploration, *Neural Computing and Applications*, vol. 31(2), pp. 1053–1074, 2019.

[35] Neely C.J., Weller P., Dittmar R.: Is Technical Analysis in the Foreign Exchange Market Profitable? a Genetic Programming Approach, *Journal of Financial and Quantitative Analysis*, vol. 32(4), pp. 405–426, 1997.

[36] Pao Y.: The Functional Link Net: Basis for an Integrated Neural-Net Computing Environment. In: *Adaptive Pattern Recognition and Neural Networks*, pp. 197–222, Addisson-Wesley, Reading, MA, 1989.

[37] Pincheira P.M., Neumann F.: *Can we beat the Random Walk? The case of survey-based exchange rate forecasts in Chile*, MPRA Paper 90432, University Library of Munich, Germany, 2018.

[38] Sahu K.K., Biswal G.R., Sahu P.K, Sahu S.R, Behera H.S.: A CRO Based FLANN for Forecasting Foreign Exchange Rates Using FLANN. In: Jain L., Behera H., Mandal J., Mohapatra D. (eds.), *Computational Intelligence in Data Mining – Volume* 1. *Smart Innovation, Systems and Technologies*, vol 31, pp. 647–664, Springer, New Delhi, 2015.

[39] Sahu K.K., Panigrahi S., Behera H.S.: A novel chemical reaction optimization algorithm for higher order neural network training, *Journal of Theoretical & Applied Information Technology*, vol. 53(3), pp. 402–409, 2013.

[40] Sahu K.K., Nayak S.C., Behera H.S.: Towards Designing and Performance Analysis of Evolving Higher Order Neural Networks for Modeling and Forecasting Exchange Rate Time Series Data. In: Singh P., Panigrahi B., Suryadevara N., Sharma S., Singh A. (eds.), *Proceedings of ICETIT 2019*. Lecture Notes in Electrical Engineering, vol. 605, pp. 258–268, Springer, Cham, 2019.

[41] Sahu K.K., Sahu S.R., Biswal G.R., Sahu P.K., Behera H.S.: Chemical reaction optimisation: a hybrid technique applied to functional link artificial neural networks with least mean square learning for foreign exchange rates forecasting, *International Journal of Swarm Intelligence*, vol. 2(2–4), pp. 254–282, 2016.

[42] Sahu K.K., Sahu S.R., Nayak S.C., Behera H.S.: Forecasting foreign exchange rates using CRO based different variants of FLANN and performance analysis, *International Journal of Computational Systems Engineering*, vol. 2(4), pp. 190–208, 2016.

[43] Sermpinis G., Theofilatos K., Karathanasopoulos A., Georgopoulos E.F., Dunis C.: Forecasting foreign exchange rates with adaptive neural networks using radial-basis functions and Particle Swarm Optimization, *European Journal of Operational Research*, vol. 225(3), pp. 528–540, 2013.

[44] Shen W., Guo X., Wu C., Wu D.: Forecasting stock indices using radial basis function neural networks optimized by artificial fish swarm algorithm, *Knowledge--Based Systems*, vol. 24(3), pp. 378–385, 2011.

[45] Sun W., Xu Y.: Using a back propagation neural network based on improved particle swarm optimization to study the influential factors of carbon dioxide emissions in Hebei Province, China, *Journal of Cleaner Production*, vol. 112, pp. 1282–1291, 2016.

[46] Suresh A., Harish K.V., Radhika N.: Particle Swarm Optimization over Back Propagation Neural Network for Length of Stay Prediction, *Procedia Computer Science*, vol. 46, pp. 268–275, 2015.

[47] Tan Y., Zhu Y.: Fireworks Algorithm for Optimization. In: Tan Y., Shi Y., Tan K.C. (eds.), *Advances in Swarm Intelligence. ICSI 2010*, Lecture Notes in Computer Science, vol. 6145, pp. 355–364, Springer, Berlin–Heidelberg, 2010.

[48] Yao J., Tan C.L.: A case study on using neural networks to perform technical forecasting of forex, *Neurocomputing*, vol. 34(1–4), pp. 79–98, 2000.

Affiliations

Kishore Kumar Sahu
Veer Surendra Sai University of Technology, Department of Information Technology, Burla, Sambalpur, India, itkishore2000@gmail.com,
http://vssut.ac.in/faculty-profile.php?furl=kishore-kumar-sahu,
ORCID ID: https://orcid.org/0000-0002-1067-0855

Sarat Chandra Nayak
CMR College of Engineering & Technology, Department of Computer Science and Engineering, Hyderabad – 501401, India, saratnayak234@gmail.com,
ORCID ID: https://orcid.org/0000-0003-0804-1264

Himansu Sekhar Behera
Veer Surendra Sai University of Technology, Department of Information Technology, Burla, Sambalpur, India, hsbehera_india@yahoo.com,
http://www.vssut.ac.in/faculty-profile.php?furl=h.s.behera,
ORCID ID: https://orcid.org/0000-0002-8952-8383

Received: 21.10.2019
Revised: 11.03.2020
Accepted: 21.03.2020