Forecasting Exchange Rates Using Artificial Neural Networks

Emmanuel Erem$^{1,2,*}$

$^1$Department of Economics and Finance, Maynooth University, Ireland
$^2$Faculty of Business and Management Sciences, Mountains of the Moon University, Uganda

*Corresponding author: emmanuel.erez.2018@mmumail.ie

Received March 22, 2023; Revised April 25, 2023; Accepted May 07, 2023

Abstract This article uses the single hidden layer perceptron neural network structure to forecast daily, weekly and monthly exchange rate data on the Swiss Franc (CHF), British Pound Sterling (GBP), and United States Dollar (USD), all per Euro (EUR). The results show good accuracy of the model as evidenced by the low mean absolute error and root mean square error, especially for the daily frequency data. Furthermore, the neural network performs best in out-of-sample predictions for the CHF/EUR currency pair for daily and weekly predictions, and best for the GBP/EUR pair when it comes to monthly frequency data. The non-linear nature of the neural network goes a long way in learning and capturing complex movements in the exchange rates as shown in the in-sample and out-of-sample graphs; a clear advantage when compared to the traditional linear prediction models. The contribution of this research is that it demonstrates the applicability of machine learning techniques to financial and economic data, clearly demonstrating the data frequencies that perform best when subjected to these algorithms. These findings are very relevant to forex traders, including commercial banks, central banks, and other monetary policy authorities. It can be argued that when it comes to risk mitigation, especially with the complexity and patterns in exchange rate movements, the neural network-based models may do a much better job compared to the traditional linear models.

Keywords: predicting/forecasting, exchange rates, artificial neural networks, training period, validation period, in-sample, out-of-sample


1. Introduction

This research applied the single hidden layer Artificial Neural Network (ANN) to model and forecast the daily, weekly and monthly frequency CHF/EUR, GBP/EUR and USD/EUR exchange rate data. The results reported showed good accuracy of the model as evidenced by the low mean absolute error and root mean square error, especially for the daily frequency data.

The contribution of this research is that it demonstrates the applicability of machine learning techniques to financial and economic data, clearly demonstrating the data frequencies that perform best when subjected to these algorithms.

Exchange rate movements are of keen interest to monetary authorities, however, it is also of great importance to large firms, especially multinationals, that conduct transactions in huge amounts of foreign currency, thus, several scholars have tried to develop and apply forecasting techniques like the ARIMA, ARMA, ARCH, GARCH and VAR models (all autoregressive in nature), just to mention a few, to exchange rate data. Artificial Neural Networks, a form of artificial intelligence, still remain an area worth exploring when it comes to exchange rate forecasting. According to Huang et al. [1], an ANN is a system loosely modelled on the human brain, which can detect underlying functional relationships within a dataset and performs tasks such as pattern recognition, classification, evaluation, modelling, prediction and control. ANNs are well-suited to finding accurate solutions in an environment characterized by complex, noisy, irrelevant or partial information. A number of reasons have been put forward as advantages for the use of ANNs which include; ANNs are data-driven self-adaptive techniques in that there are few a priori assumptions about the models, ANNs can generalize, ANNs are universal functional approximators and finally, ANNs are nonlinear, and for these reasons, ANNs are very much applicable to time series data, particularly exchange rates.

According to Meese and Rogoff [2], econometric models used to forecast exchange rates based on economic fundamentals have had limited success, especially when the forecast horizon is at a 1 to 12-month period. Time series models produce plausible point estimates in exchange rate prediction but are poor at predicting the direction in which the rates move. Machine learning methods such as shallow ANNs and support vector machines may be marginally better at predicting the
direction of change, but their success depends critically on the input features used to train the models. This improvement comes at a cost; obtaining a good set of features from raw input data may require significant efforts from domain experts [3].

When it comes to the inputs used in the ANNs, there are generally two ways to approach the problem; one may use the lags of the exchange rate variable as inputs or use the economic fundamentals believed to be important in the determination of exchange rates, these fundamentals are: relative money supply, relative GDP, nominal interest rate differential, and the long-run expected inflation differential. One may also add the current account as a possible variable.

The structure of the ANN determines the nature of the output; the structure may be characterized by the number of hidden layers and the number of neurons per layer. It is important to note that if there is no hidden layer in the system, this may be similar to a simple OLS regression type model, particularly when the activation function is linear in nature. Of course, the more complex the structure of the ANN, the higher the model’s ability to capture complex relationships and key turning points. However, there may also be a problem of over-fitting if the structure is too complex, thus, it is important to strike a balance when dealing with ANNs. Another important consideration is that the output produced by an ANN changes each time the model is run despite the fact that key input parameters remain fixed; this perhaps may be one of the downsides of ANN models.

2. Literature Review

Scholars have shown interest in ANNs in recent times; others have modified the ANNs or applied them in combination with other models and most have reported the superiority of such models. Neural networks were originally developed in cognitive or biological science and were later used in engineering for pattern recognition and classification. They have also been used in the tourism industry, energy, especially renewables [4]. Adewole et al. [5] applied daily data on NGN/USD, NGN/EUR, NGN/GBP and NGN/JPY to an ANN and a hidden Markov model and found that the multi-layer perceptron ANN reported an accuracy rate of 81.2% compared to the hidden Markov model that reported a rate of 69.9%. Panda and Narasimhan [6] apply ANNs to INR/USD weekly data comparing its forecast performance to the linear AR and RW models and their results showed that the neural network has a superior in-sample performance compared to the other two models, reporting a more convincing evaluation result regardless of the evaluation criteria used in the study. Furthermore, the ANN also beats the linear autoregressive model in four out of the six evaluation criteria in their out-of-sample comparison.

Aydin and Cavdar [7] applied the Multi-Layered Feed Forward Neural Network (MLFFNN) and VAR models to monthly data on USD/TRY, gold prices and the Borsa Istanbul (BIST). On comparing the forecast results, it was evident that the ANN technique performed better compared to the VAR model. Lasheras et al. [8] compared the performance of the MLFFNN and the Elman neural network to the ARIMA using copper spot prices data and concluded that the performance of the MLFFNN and Elman Recurrent Neural Network (RNN) are better than the ARIMA when evaluated in terms of Root Mean Square Error (RMSE) values.

Koprinska et al. [9] show that Convolutional Neural Networks (CNN) and the Multi-Layered Perceptron Neural Networks performed similarly in terms of accuracy and training time, and outperformed other models used in their study; highlighting the potential of CNNs for energy time series forecasting. See also Matyjaszek et al. [10], Eskandari et al. [11] and Yang et al. [12] for similar studies in the energy sector.

Borovykh et al. [13] show that the CNN can effectively learn dependencies in and between a series without the need for long historical data. Their study subjected data on the S&P 500, volatility index, the CBOE interest rate, and many exchange rates to a CNN and VAR model.

Lai et al. [14] proposed a deep learning framework, the Long- and Short-term Time-series Network (LSTNet), that combines the methods of the CNN and RNN to extract short-term local dependency patterns among variables and to discover long-term patterns for trends; complementing the CNN and RNN with an AR model to solve the scale insensitivity problem that neural network models suffer from. The LSTNet model was applied to data on traffic, solar power production, electricity consumption and exchange rates. Their findings showed that by combining the strengths of CNN, RNN and AR models, the LSTNet significantly improved the state-of-the-art results in time series forecasting on multiple benchmark datasets.

Leung et al. [15] use the non-parametric General Regression Neural Network (GRNN) to predict the monthly exchange rate movements of the GBP, CAD and JPY. Their results revealed that the GRNN performed better than the Multi-Layered Feed Forward Neural Network, the parametric multivariate transfer function and the RW model included in their study. Their findings revealed that except for the GBP, the GRNN reported significantly lower Mean Absolute Error (MAE) and Root Mean Square Error compared to the other approaches.

Ni et al. [16] propose a Convolution Recurrent Neural Network (C-RNN) applying the model to exchange rate data of nine major currencies; findings revealed that the C-RNN model has better applicability and higher accuracy.

Alizadeh et al. [17] use an Adaptive Neural-Fuzzy Inference System (ANFIS) to forecast USD/JPY exchange rates and find that the ANFIS is superior in terms of prediction error minimization, robustness and flexibility when compared to the Sugeno-Yasukawa model, MLFFNN and multiple regression models. They further argue that the ANFIS can be used to better explain solutions when compared to the black-box neural networks. A similar argument is put forward by Sharma et al. [18] who applied ANFIS to daily CNY/USD, INR/USD and JPY/USD data and reported that ANFIS based models outperformed the ANN based models when evaluated based on Mean Absolute Percentage Error (MAPE) values.

1 The combination of these variables form the monetary or macroeconomic type models. When applied to ANNs, then we have a non-linear monetary model.
Galeshchuk and Mukherjee [3] argue that time series models and shallow neural networks provide acceptable point estimates for future rates but are poor at predicting the direction of change. They advocate for the use of deep networks that may have the ability to learn abstract features in the data. In their study, they investigate the ability of Deep Convolution Neural Networks (DCNN) to predict the direction of change in EUR/USD, GBP/USD and JPY/USD, and they state that trained deep networks produce satisfactory out-of-sample accuracy. They further point out that the Absolute Percentage Error rate for forecasts in the ARIMA, Exponential Smoothing (ETS) and ANN models were less than 2.4% in all instances, which are generally acceptable error rates that imply the point estimates are acceptable and satisfactory.

Shen et al. [19] in their study, while modifying a Deep Belief Network (DBN), applied weekly exchange rate data on GBP/USD, BRL/USD and INR/USD to a DBN, MLFFNN, RW and ARMA models. The findings in the study reported that the DBN outperformed the MLFFNN and the traditional forecasting techniques by all evaluation criteria used in the study.

Henriquez and Kristjanpoller [20] propose a hybrid model that uses Independent Component Analysis (ICA) as a deconstruction model and then employs neural networks to predict the future values of the deconstructed series. The hybrid model was applied to five daily frequency currencies with respect to the USD; EUR, GBP, JPY, CHF and CAD. Their results revealed a significant performance improvement in the Mean Square Error (MSE) and MAPE when compared to the RW model and the econometric models of the ARMA and GARCH family.

Markova [4] presents a Nonlinear Autoregressive with Exogenous Input (NARX) neural network using three different training algorithms applying the model to EUR/USD. Results reported were convincing and the study concluded that ANNs are an effective method of forecasting exchange rates; there was a close relationship between the outputs and the targets after.

There are many other hybrid adaptations and modifications to the neural network structures using a number of functions; see for example Sermpinis et al. [21,22,23], Dunis et al. [24] and Stasinakis et al. [25].

### 3. The Model and Data

#### 3.1. The Model

<table>
<thead>
<tr>
<th>Neural Networks form the base of Deep Learning, a subfield of Machine Learning where the algorithms are inspired by the structure of the human brain. Neural Networks take in data, train themselves to recognize the patterns in this data, and then predict the outputs for a new set of similar data. Neural Networks are made up of layers of neurons. These neurons are the core processing units of the network.</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Input Layer; Receives the input.</td>
</tr>
<tr>
<td>• Output Layer; Predicts the final output.</td>
</tr>
<tr>
<td>• Hidden Layer; Performs most of the computations required by the network.</td>
</tr>
</tbody>
</table>

Neurons of one layer are connected to neurons of the other layer through channels. Each channel is assigned a numerical value known as a weight. Each data point is fed as input to each neuron of the first layer. The inputs are multiplied by the corresponding weights and their sum is sent as input to the neurons in the hidden layer. Each of the neurons is associated with a numerical value called the bias which is then added to the input sum. This value is then passed through a threshold function called the activation function. The result of the activation function determines whether the particular neuron will get activated or not. An activated neuron transmits data to the neurons of the next layer over the channels. In this manner, the data is propagated through the network. This is called Forward Propagation. In the output layer, the neuron with the highest value fires and determines the output. However, this output can be wrong, thus the need for Training.

During the Training Period, along with the input, the network also has the output fed to it. The predicted output is then compared to the actual output to realize the error in the prediction. The magnitude of the error indicates how wrong the algorithm is and the sign suggests if the predicted values are higher or lower than expected. Therefore, the objective of the algorithm is to minimize the Loss Function, which in this case is the RMSE.

During the process, the algorithm gives the direction and magnitude of change required to reduce the error. This information is then transferred backward through the network. This is known as Backward Propagation. Based on this information, the weights are adjusted. This cycle of Forward and Backward Propagation is iteratively performed with multiple inputs. This process continues until the weights are assigned such that the network can predict the output correctly in most cases. This then brings the training to an end. Training may take minutes, hours, months or weeks.

Source: [26].

---

2 Levenberg-Marquardt, Bayesian regularisation and Scaled Conjugate Gradient.
The input variables in this model, that is the $x$ variables, are the lags of the exchange rate series. The output of any neuron $j$ in the hidden layer is given by;

$$h_j = \sigma \left( b_j + \sum_{i=1}^{N} w_{ij} x_i \right)$$  \hspace{1cm} (1)$$

Where $\sigma$ is the sigmoid logistic activation function which has the important property of being non-linear in nature, $b_j$ is the bias term specific to neuron $j$, that is to say, every neuron already has a bias term. This bias, sometimes referred to as the threshold term, is the value required for the neuron to have a meaningful performance. The bias can be compared to the intercept term in a regression model. $w_{ij}$ is the weight of the synapse from neuron $i$ to neuron $j$, it may also be looked at as the contribution of neuron $i$ to the output of neuron $j$. $x_i$ is the input into a neuron in the input layer and $N$ the number of neurons in the input layer.

$$\hat{y} = \sigma \left( b_j + \sum_{j=1}^{K} w_{3j} h_j \right)$$ \hspace{1cm} (2)$$

The error, which in this case is the Sum Squared Error (SSE) for the training iteration $t$ and training vector $p$ is given by;

$$E^p(t) = \frac{1}{2} \left( \hat{y}^p(t) - y^p(t) \right)^2$$ \hspace{1cm} (3)$$

Where $\hat{y}^p(t)$ is the output value and $y^p(t)$ is the target value.

The total error is therefore computed as;

$$E(t) = \sum_{p=1}^{P} E^p(t)$$ \hspace{1cm} (4)$$

$^3$ There has been a movement towards the use of the Rectified Linear Unit (ReLU) activation function. The argument is that this type of function enables the algorithm to detect and learn patterns faster.

The relationship between the weight, $w_{ij}$, bias, $b_j$, during each training iteration and the error function is given by;

$$\Delta w_{ij}(t) = -\tau \frac{\partial E^p(t)}{\partial w_{ij}(t)}$$ \hspace{1cm} (5)$$

$$\Delta b_j(t) = -\tau \frac{\partial E^p(t)}{\partial b_j(t)}$$ \hspace{1cm} (6)$$

Where $\tau$ is the learning rate and $\frac{\partial E^p(t)}{\partial w_{ij}(t)}$ and $\frac{\partial E^p(t)}{\partial b_j(t)}$ are the gradient terms of the error function with respect to the weights and bias terms at iteration $t$ and training vector $p$. The model is trained using a gradient descent algorithm which is designed to allow the model to adjust the parameters (the weights and biases) of the ANN in a way that best minimises the loss function, and consequently the output deviation. The gradient of the loss function is computed by the backpropagation algorithm using the chain rule, one layer at a time, iterating backwards right from the output layer.

The errors reported are the Mean Absolute Error and Root Mean Square Error as defined below.

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$ \hspace{1cm} (7)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} = \sqrt{MSE}$$ \hspace{1cm} (8)$$

$^4$ The learning rate has to be appropriate; it should not be too high or too low. For instance, if it is too high, the model may not reach the local minimum and may just keep bouncing back and forth between the convex function.

$^5$ This algorithm is generally used in training machine learning models; it tweaks the parameters iteratively to minimise a loss function to its local minimum.
Well-behaved activation functions in this case need to be non-linear, continuous, differentiable, monotonic and bounded. Some of these functions are:

- The logistic function, as shown in Figure 2;
  \[ f(x) = \frac{1}{1 + e^{-x}} \]
- The hyperbolic tangent; \[ f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \]
- Gaussian; \[ f(x) = e^{-x^2/2} \]
- Sine and Cosine; \[ f(x) = \sin(x), f(x) = \cos(x) \]

3.2. Data

The exchange rate data on CHF/EUR, GBP/EUR and USD/EUR covers three frequencies; daily, weekly and monthly and is all downloaded from www.global-view.com/forex-trading-tools/forex-history/index.html. The daily data runs from 02/01/2020 to 03/12/2020, that is, 242 data points. The weekly data runs from the week of 18/04/2016 to 03/12/2020, that is, 242 data points and the monthly data runs from January 2000 to December 2020, that is 252 observations.

The data is then divided into two parts; the training and validation data sets. The daily frequency training data for the CHF/EUR runs from 30/01/2020 to 16/09/2020 (166 observations), GBP/EUR runs from 23/01/2020 to 16/09/2020 (171 observations) and USD/EUR runs from 09/01/2020 to 16/09/2020 (181 observations). The validation data runs from 17/09/2020 to 03/12/2020 (56 observations) for all three currencies.

The weekly frequency training data for the CHF/EUR runs from the week of 15/08/2016 to 08/11/2019 (169 observations), GBP/EUR runs from the week of 05/09/2016 to 08/11/2019 (166 observations) and USD/EUR runs from the week of 05/09/2016 to 08/11/2019 (166 observations). The validation data runs from the week of 11/11/2019 to 03/12/2020 (56 observations) for all three currencies.

The monthly frequency data for the CHF/EUR runs from October 2000 to April 2016 (187 observations), GBP/EUR runs from December 2000 to April 2016 (185 observations) and USD/EUR runs from January 2001 to April 2016 (184 observations). The validation data runs from May 2016 to December 2020 for all three currencies.

![Figure 2. Sigmoid logistic activation function](image)

4. Discussion of Results

4.1. Descriptive statistics

| Table 1. Moment summary statistics of daily exchange rates |
|-----------------|-----------------|-----------------|-----------------|-----------------|
| Mean  | Std. Dev | Min  | Max  | Skewness | Kurtosis |
| CHF/EUR | 1.069 | 0.010 | 1.051 | 1.086 | -0.335 | 1.980 |
| GBP/EUR | 0.888 | 0.024 | 0.831 | 0.938 | -0.806 | 2.729 |
| USD/EUR | 1.135 | 0.040 | 1.064 | 1.214 | 0.096 | 1.502 |

| Table 2. Moment summary statistics of weekly exchange rates |
|-----------------|-----------------|-----------------|-----------------|-----------------|
| Mean  | Std. Dev | Min  | Max  | Skewness | Kurtosis |
| CHF/EUR | 1.108 | 0.037 | 1.051 | 1.197 | 0.465 | 2.072 |
| GBP/EUR | 0.875 | 0.028 | 0.761 | 0.929 | -1.314 | 5.697 |
| USD/EUR | 1.136 | 0.045 | 1.043 | 1.245 | 0.320 | 2.654 |
Table 1, Table 2 and Table 3 show the key moment summary statistics of the exchange rate data at levels for the daily, weekly and monthly frequencies respectively. For instance, from Table 1, it is observable that the CHF/EUR has an average rate of 1.069 with a standard deviation of 0.010, reaching a minimum rate of 1.051 and a maximum rate of 1.086. The tail behaviour, described by the skewness and kurtosis values indicates that the data is negatively skewed. The kurtosis on the other hand is less than 3, implying that the data is platykurtic. All the data have a platykurtic distribution except for the GBP/EUR weekly frequency that has a kurtosis greater than 3, making it leptokurtic.

4.2. Architecture of the Neural Network Models

Table 4 shows the structure of the neural networks by number of neurons per layer.

<table>
<thead>
<tr>
<th>Input</th>
<th>Hidden</th>
<th>Input</th>
<th>Hidden</th>
<th>Input</th>
<th>Hidden</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHF/EUR</td>
<td>20</td>
<td>10</td>
<td>17</td>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td>GBP/EUR</td>
<td>15</td>
<td>15</td>
<td>20</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>USD/EUR</td>
<td>5</td>
<td>10</td>
<td>20</td>
<td>5</td>
<td>12</td>
</tr>
</tbody>
</table>

Table 4 shows the structure of the neural networks by number of neurons per layer. There is no specific formula that gives the optimal number of neurons that may be used by a layer, but the bigger the number of neurons, the more complex the relationships being captured by the model as noted earlier. The model uses a single hidden layer with a single output neuron as illustrated in Figure 1.

4.3. Error/Accuracy Measure and Performance of the Models

In-sample predictions are associated with the training period while the out-of-sample predictions are associated with the validation period. The validation period is an unbiased period that typically is an evaluation of the model’s performance.

Table 5 and Table 6 show the Mean Absolute Error and Root Mean Square Error for the training and validation periods of the 3 data frequencies. It is observable that there is a lower error (regardless of the measure) reported during the training period compared to the validation period for all currency pairs and frequencies. For example, looking at Table 5 and Table 6, GBP/EUR weekly data; the training period reports a MAE and RMSE of 0.00016 and 0.00025 respectively while the validation period reports higher MAE and RMSE of 0.01844 and 0.02168 receptively. This implies that the model performs better for in-sample predictions compared to out-of-sample predictions. It is also important to note that the error reported for daily frequency data is lower than that for both the weekly and monthly frequency data for each of the currency pairs during the validation period. For example, taking the USD/EUR pair; daily, weekly and monthly MAE are 0.00882, 0.02334 and 0.07140 respectively; the model performs best for high frequency data during the validation period. This assertion may not apply to the training period; comparing GBP/EUR daily and weekly frequencies during the training period, it is observable that the weekly data reports a lower MAE and RMSE compared to the daily frequency data.

In-sample daily predictions indicate that the model performed best for the CHF/EUR pair, reporting the lowest MAE and RMSE of 0.00010 and 0.00018 respectively. Weekly estimates show that the model performed best for the GBP/EUR pair, reporting a MAE and RMSE of 0.00016 and 0.00025 respectively. The GBP/EUR currency pair again performed best when it came to monthly frequency, reporting a MAE and RMSE of 0.00673 and 0.00878 respectively.

Out-of-sample daily predictions indicate that the model performed best for the CHF/EUR currency pair, reporting the lowest MAE and RMSE of 0.00377 and 0.00473 respectively. When it came to weekly estimates, the model performed best for the CHF/EUR currency pair too, reporting a MAE and RMSE of 0.00783 and 0.00983 respectively. The GBP/EUR currency pair performed best when it came to monthly frequency, reporting a MAE and RMSE of 0.00673 and 0.00878 respectively.

Out-of-sample daily predictions indicate that the model performed best for the CHF/EUR currency pair, reporting the lowest MAE and RMSE of 0.00010 and 0.00018 respectively. Weekly estimates show that the model performed best for the GBP/EUR pair, reporting a MAE and RMSE of 0.00016 and 0.00025 respectively. The GBP/EUR currency pair again performed best when it came to monthly frequency, reporting a MAE and RMSE of 0.00673 and 0.00878 respectively. The ANN models did not perform well when it came to the USD/EUR pair, especially during the validation period, where the currency pair reported the highest MAE and RMSE regardless of the data frequency. The performance of the ANN models for the currency pairs and frequencies can be observed graphically in Figure 3, Figure 4 and Figure 5 for daily data; Figure 6, Figure 7 and Figure 8 for weekly data; Figure 9, Figure 10 and Figure 11 for monthly data.
Table 6. Validation period error measures

<table>
<thead>
<tr>
<th></th>
<th>CHF/EUR</th>
<th>GBP/EUR</th>
<th>USD/EUR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MAE</td>
<td>RMSE</td>
<td>MAE</td>
</tr>
<tr>
<td>Daily</td>
<td>0.00377</td>
<td>0.00473</td>
<td>0.00711</td>
</tr>
<tr>
<td>Weekly</td>
<td>0.00783</td>
<td>0.00983</td>
<td>0.01844</td>
</tr>
<tr>
<td>Monthly</td>
<td>0.04024</td>
<td>0.05313</td>
<td>0.03649</td>
</tr>
</tbody>
</table>

Figure 3. CHF/EUR daily step exchange rate forecasts

Figure 4. GBP/EUR daily step exchange rate forecasts

Figure 5. USD/EUR daily step exchange rate forecasts
Figure 6. CHF/EUR weekly step exchange rate forecasts

Figure 7. GBP/EUR weekly step exchange rate forecasts

Figure 8. USD/EUR weekly step exchange rate forecasts

Figure 9. CHF/EUR monthly step exchange rate forecasts
5. Conclusion

This study applied the single hidden layer neural network to predict daily, weekly and monthly frequency exchange rates of the CHF/EUR, GBP/EUR and USD/EUR. The results show good accuracy of the model as evidenced by the low MAE and RMSE, especially for the daily frequency data. Furthermore, the neural network performed best in out-of-sample predictions for the CHF/EUR currency pair for daily and weekly predictions, and performed best for the GBP/EUR pair when it came to monthly frequency. The USD/EUR pair proved more difficult to model, performing worst, especially in the validation period. The non-linear nature of the neural network went a long way in learning and capturing complex movements in the exchange rates as shown in the in-sample and out-of-sample graphs; a clear advantage when compared to the traditional linear prediction models like the ARMA and ARIMA. These findings are very relevant to forex traders, including commercial banks, central banks and other monetary policy authorities. The results clearly show the applicability of machine learning techniques to financial and economic data, thus, improving planning. It can be argued that when it comes to risk mitigation, especially with the complexity and patterns in exchange rate movements, neural networks may do a much better job than the traditional models.

Further research could include comparing the performance of this non-linear machine learning technique to the more traditional linear techniques. Furthermore, the predictors could be altered too. In this case, the predictors were the lags of the exchange rate. The monetary model, that uses economic fundamentals as predictors, can also be adopted for future studies to compare performance.

Funding

This research was funded and supported by Science Foundation Ireland (SFI) under Grant Number 16/SPP/3347.

References


