Neural network protocols and model performance

Teo Jasic*a,†, Douglas Woodb

a Accenture GmbH, Campus Kronberg 1, Kronberg 61476, Germany
b Manchester Business School, Booth Street West, Manchester M15 6PB, UK

Abstract

The benchmark performance of neural network models in exchange rate forecasting has been established in a number of recent papers. In this paper, we drop a number of customary features that are used in neural modeling to improve performance yet are able to predict daily returns for the four main exchange rates with accuracy above levels previously reported for comparable data sets. Significant outperformance relative to a random walk without drift and against the mean value predictor is reported, indicating that the transparent neural network used in the study is finding new and potentially profitable information.

© 2003 Elsevier B.V. All rights reserved.

Keywords: Neural networks; Exchange rates; Forecasting

1. Introduction

Ever since the seminal paper of Meese and Rogoff [14] it has been generally accepted that it is hard to beat a random walk in predicting exchange rates out-of-sample, possibly because there is no coherent underlying economic theory of short-to-medium-run movements in exchange rates [13]. Subsequent studies including [19] provide support for some of the long-run relationships implied by economic theory, but suggest further attempts to offer explanations of short-term exchange rate movements based only on macroeconomic fundamentals may not prove successful.

However, motivated by the growing use of technical analysis in currency trading [2] and recent nonlinear modeling of exchange rates using neural networks [6,12], we build univariate neural network prediction models using time-delay embedding technique to
forecast short-run exchange rate movements. In doing so, common procedures such as normalisation and out-of-sample validation were omitted to reduce problems of data snooping and only model architectures approximating accepted rules of thumb were considered. Even so, the results showed significant outperformance of accepted benchmarks and interestingly of previous comparable studies for all four exchange series.

2. Data and neural network models

Daily dollar rates for the Japanese Yen (JPY), British Pound (GBP), Deutsche Mark (DEM) and Swiss Franc (CHF) from January 2, 1986 to November 11, 1999 (3616 observations for each currency) were obtained from the Datastream/ICV International Database and converted to returns by taking the first difference of log rates. The third moment of return series distribution, skewness, is statistically significant at the 1% level for \$/JPY and \$/GBP rates indicating deviation from normal distribution. The fourth moment, kurtosis, is positive and statistically significant at the 1% level for all return series indicating fat-tail (leptokurtic) distributions. This is confirmed with the Kolmogorov–Smirnov test [16] that rejected normality at the 1% significance level for all return series. The Ljung–Box Q-statistic [11] for 10th- and 15th-order serial correlations of the raw returns revealed no significant autocorrelation structure but the corresponding Q-statistic for squared returns suggested the presence of some time-varying volatility in all series.

Neural networks are universal approximators capable of approximating a large class of functions with a high degree of accuracy without any prior assumption of the model form [5,8]. However, this power comes at a cost since neural networks are exposed to major problems of transparency, overfitting and data snooping. Given these concerns, it therefore seems desirable to use simple network architectures and minimise the number of subjective steps in estimation and selection. For this study, a single hidden layer feedforward univariate neural network as shown in Fig. 1 was used. The
input units represent a time-delay vector \((x_{t-\tau}, x_{t-2\tau}, \ldots, x_{t-(d-1)\tau}, x_{t-d\tau})\), where \(x_t\) is the observation at time \(t\), \(\tau\) is time-delay set at 1 in this study and \(d\) is the embedding dimension of the state space in which the dynamics of the underlying system is viewed. A theoretical framework for the time-delay embedding technique is provided in [17,18].

The functional form of the network with time-delay inputs for computing forecasts \(y_t\) used in this study can be written as

\[
y_t = \hat{x}_t = f(x_{t-1}, \ldots, x_{t-d}) = \phi \left( \theta_{\phi} + \sum_{i=1}^{n} \omega_i g \left( w_{ci} + \sum_{j=1}^{d} w_{ij} x_{t-j} \right) \right),
\]

where \(x_{t-j}\) denote the returns in the previous \(j\) days, with \(j\) running over an index set of sequential positive integers, \(d\) denotes the number of inputs in the model and \(n\) is the number of hidden units. \(\{w_{ij}, i=0, 1, \ldots, n, j=0, 1, \ldots, d\}\) is a matrix of the weights from input to the hidden-layer units and \(\{\omega_i, i=0, 1, \ldots, n\}\) is a vector of weights from the \(i\)th hidden-layer unit to the output unit. \(w_{ci}\) and \(\theta_{\phi}\) represent the bias terms as weights from an additional input and an extra hidden unit, respectively. Following the recommendation of Faraway and ChatBYYeld [4] for forecasting applications, we set \(g\) to logistic and \(\phi\) as an identity function. Generally, inputs are then subjected to linear transformation [10] or statistical normalization [20] to zero mean and unit variance to reflect the constraint that nonlinear activation functions, such as the logistic function, restrict the possible output from a node to a specific interval (e.g., \((0, 1)\) or \((-1, 1)\)). Other than this, previous research gives no clear indication that normalization is necessary [21]. Since we use the linear output transfer function, we do not need to normalize or scale data inputs. The results show we can avoid this element of subjectivity without any obvious handicap.

We divide the data sets in this study into a training or estimation set of the historically oldest data (January 2, 1986–December 31, 1995) containing two-thirds (2606) of the observations with the remainder from January 1, 1996 to November 11, 1999, providing an out-of-sample set for testing. Validation where the final predictive network is selected on out-of-sample data was not used so no data had to be used for this purpose.

In determining network architecture, the number of inputs as lagged observations, \(d\), is the most important parameter to be estimated because it determines the (nonlinear) autocorrelation structure of the time series [21]. However, with no theory or systematic procedure to set \(d\) we assume that an embedding dimension of between 2 and 10 lags is sufficient to capture the dynamics of exchange rate movements. In selecting the number of hidden units, we follow a usual rule of thumb in statistical modeling that for a set of \(N\) observations, the degrees of freedom in the model, which are the network weights, should not exceed \(N^{1/2}\). Given 2606 data points in our training set, an upper bound would be 51 weights, each corresponding to a degree of freedom in the neural network to be trained. Considering this upper bound value and previous choice of the length of input vectors, we define the number of hidden units as from one up to approximately half the number of the input units. The following network architectures are therefore chosen: 10–6–1 (i.e., 10 inputs, 6 hidden units and 1 output), 9–6–1, 8–5–1, 7–4–1,
Table 1
Out-of-sample results for neural network (NN) models selected on training RMSE

<table>
<thead>
<tr>
<th>Exchange rate series and NN model</th>
<th>No of Tr. Iter.</th>
<th>Training RMSE</th>
<th>Testing (out-of-sample) RMSE</th>
<th>NMSE</th>
<th>MAE</th>
<th>$S_{\text{stat}}$ (%)</th>
<th>$D_{\text{stat}}$ (%)</th>
<th>$R_{\text{RMSE(AR(1))}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>USD</td>
<td>1200</td>
<td>0.0001864</td>
<td>0.000186</td>
<td>0.00393</td>
<td>0.000183</td>
<td>93.95</td>
<td>99.60</td>
<td>0.0440</td>
</tr>
<tr>
<td>JPY</td>
<td>600</td>
<td>0.0001817</td>
<td>0.000157</td>
<td>0.00418</td>
<td>0.000126</td>
<td>92.44</td>
<td>98.71</td>
<td>0.0495</td>
</tr>
<tr>
<td>CHF</td>
<td>700</td>
<td>0.0001694</td>
<td>0.000134</td>
<td>0.00237</td>
<td>0.000104</td>
<td>94.15</td>
<td>98.90</td>
<td>0.0488</td>
</tr>
<tr>
<td>GBP</td>
<td>900</td>
<td>0.0001659</td>
<td>0.000115</td>
<td>0.00324</td>
<td>0.000089</td>
<td>93.85</td>
<td>99.31</td>
<td>0.0570</td>
</tr>
</tbody>
</table>

No. of Tr. Iter. is the number of iterations in training for the best model. $S_{\text{stat}}$ and $D_{\text{stat}}$ give accuracy of sign and direction change statistics for the neural networks predictions in percentages. $R_{\text{RMSE(AR(1))}}$ and $R_{\text{RMSE(AR(1))}}$ are the error ratios of neural network and AR(1) model predictions relative to the mean value predictor.

6–4–1, 5–3–1, 4–2–1, 3–2–1 and 2–1–1. We therefore train nine common architecture models with the number of degrees of freedom in each model determined by a specific number of input and hidden units. Two architectures, 10–6–1 and 9–6–1, have the number of weights marginally above the specified upper bound of 51 (i.e., 9–6–1 architecture has $(9 + 1) \times 6 + (6 + 1) = 67$ weights taking into account the bias nodes) but the remainder are within the upper bound limit.

The neural network models were trained using the conjugate gradient method which relies on a form of gradient descent with momentum [1]. We set values for initial weights to 0.5. The stopping criterion for training was set at multiples of 100 iterations up to a maximum of 2000 iterations with training results observed in steps of 100 iterations. Root mean squared error (RMSE) in predicting data in the estimation set was used to select the best network architecture for out-of-sample testing. Testing was of one-step-ahead predictions using actual values of return series. One-step prediction involves more variability than multi-period horizons but provides a useful mechanism to evaluate the adaptability and robustness of the prediction system.

3. Results

Normalized mean squared error (NMSE), RMSE and mean absolute error (MAE) along with out-of-sample sign and direction change statistic for network predictions and error ratios relative to the mean value predictor are presented in Table 1. In financial markets, profits may depend more on accuracy in predicting direction of movements than error in predicting and this is captured by the sign of return statistic

$$S_{\text{stat}} = \frac{1}{N} \sum_{t=1}^{N} a_t,$$

(2)
where for \( N \) pairs of predictions \( y_t \) and outcomes \( x_t \), \( a_t = 1 \) if \( x_t y_t > 0 \) or \( x_t = y_t = 0 \) and \( a_t = 0 \) otherwise. Similarly, the direction change statistic that measures the direction of change in the return (gradient of return) can be defined as

\[
D_{\text{stat}} = \frac{1}{N} \sum_{t=1}^{N} b_t,
\]

where \( b_t = 1 \) if \( (x_{t+1} - x_t)(y_{t+1} - y_t) \geq 0 \) and \( b_t = 0 \) otherwise. We also compare neural network out-of-sample predictions with the mean value predictor using the ratio statistic

\[
R_{\text{RMSE}} = \frac{\left[ \sum_{i \in P} (y_i - x_i)^2 \right]^{1/2}}{\left[ \sum_{i \in P} (\hat{y}_i - x_i)^2 \right]^{1/2}}
\]

where \( P \) represents the out-of-sample prediction set comprising \( i \) pairs of the actual values (or targets, \( x_i \)) and predicted values (\( \hat{y}_i \)), and \( R_{\text{RMSE}} \) is the ratio of the RMSEs from neural network predictions compared with the RMSE using the mean value predictor. \( R_{\text{RMSE}} \) takes a value of 0 if predictions are perfect while \( R_{\text{RMSE}} \geq 1 \) indicates that the performance is no better than the mean value predictor. The sign and direction change statistic are consistently above 90% for each exchange rate which is confirmed with the \( R_{\text{RMSE}(\text{NN})} \) values for neural networks close to 0. The results of the neural network predictions are compared to a linear autoregressive model of order 1 AR(1) to demonstrate significant improvements over predictions based on linear structure. The AR(1) model provides a convenient representation of the white-noise process with short-term memory that mimics a random walk with the higher-order autocorrelations carrying little predictive power. The \( R_{\text{RMSE}(\text{AR}(1))} \) above 1 for the benchmark linear AR(1) model shows the performance of the neural network models does not depend on linear autocorrelation patterns that would, as noted by Kantz and Schreiber [9], be captured by a linear predictor.

The significance of predicted signs of returns is tested using the Pesaran and Timmermann test [15], and the significance of differences between alternative predictions (i.e., the predictions by neural networks and the mean value predictor) using a modified Diebold and Mariano test statistic [3,7]. The Pesaran–Timmermann (P–T) and Diebold–Mariano (D–M) test standardized statistic values for out-of-sample set and neural network models in Table 2 reject the independence between predictions and outcomes and show significant difference from the mean value predictor for all exchange rate series whereas none of the values for the AR(1) model are significant.

We noted that the training and testing results are similar across different common architectures and observed iteration steps, suggesting that performance was not sensitive to the choice of network architecture and stopping criterion. For example, a 3–2–1 architecture and 300 training iterations produced almost as good results for the $/GBP rate as the 7–4–1 model used. It is also interesting that the best models for each exchange rate series have similar embedding dimensions (6 and 7) and corresponding network size. We can conclude that the exchange rate series examined are not purely random systems and that they can be modeled by a low-dimensional system. However, whether the series behavior is dominated by a deterministic or a stochastic system cannot be precisely answered. Neural networks with the number of degrees of freedom
Table 2

<table>
<thead>
<tr>
<th>Exchange rate series</th>
<th>Neural network</th>
<th>Linear model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Model</td>
<td>P–T test statistic</td>
</tr>
<tr>
<td>$/DEM</td>
<td>6–4–1</td>
<td>28.13*</td>
</tr>
<tr>
<td>$/JPY</td>
<td>7–4–1</td>
<td>27.22*</td>
</tr>
<tr>
<td>$/CHF</td>
<td>6–4–1</td>
<td>28.01*</td>
</tr>
<tr>
<td>$/GBP</td>
<td>7–4–1</td>
<td>27.85*</td>
</tr>
</tbody>
</table>

*Indicates that the P–T test statistic and modified D–M test statistic are statistically significant at the 1% level.

Less than the established upper bound though provide a robust platform for effective prediction in apparently random and non-linear exchange rate data.

4. Concluding remarks

Contrary to earlier findings, our one-step-ahead forecasts of daily exchange rate returns significantly outperform the random walk and mean value predictor benchmark forecasts and the results of comparable previous research. This performance is achieved using robust models which avoid some of the problems of overfitting and information losses through data preprocessing inherent in conventional approaches. In any event the use of parsimonious procedures appears to improve transparency without any great penalty in performance. Similar training and testing results across different common architectures suggest that optimization procedures are not critical. It might be interesting to further explore whether the selection in training based on sign and direction of change indicators, as being the most influential determinants of financial performance, may lead to even simpler architectures.

Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful comments.

References


