Modelling and Trading the EUR/USD Exchange Rate: Do Neural Network Models Perform Better?

by

Christian L. Dunis* **and Mark Williams****
(Liverpool Business School and CIBEF**)***

February 2002

Abstract

This research examines and analyses the use of Neural Network Regression (NNR) models in foreign exchange (FX) forecasting and trading models. The NNR models are benchmarked against traditional forecasting techniques to ascertain their potential added value as a forecasting and quantitative trading tool.

In addition to evaluating the various models using traditional forecasting accuracy measures, such as root mean squared errors, they are also assessed using financial criteria, such as risk-adjusted measures of return.

Having constructed a synthetic EUR/USD series for the period up to 4 January 1999, the models were developed using the same in-sample data, leaving the remainder for out-of-sample forecasting, October 1994 to May 2000, and May 2000 to July 2001, respectively. The out-of-sample period results were tested in terms of forecasting accuracy, and in terms of trading performance via a simulated trading strategy. Transaction costs are also taken into account.

It is concluded that NNR models do have the ability to forecast EUR/USD returns for the period investigated, and add value as a forecasting and quantitative trading tool.

* *Christian Dunis is Girobank Professor of Banking and Finance at Liverpool Business School and Director of CIBEF (E-mail: cdunis@totalise.co.uk). The opinions expressed herein are not those of Girobank.*
** *Mark Williams is an Associate Researcher with CIBEF (E-mail: markpickfordwilliams@hotmail.com).*
*** *CIBEF – Centre for International Banking, Economics and Finance, JMU, John Foster Building, 98 Mount Pleasant, Liverpool L3 5UZ.*

1. Introduction

Since the breakdown of the Bretton Woods system of fixed exchange rates in 1971-1973 and the implementation of the floating exchange rate system, researchers have been motivated to explain the movements of exchange rates. The global FX market is massive with an estimated current daily trading volume of USD 1.5 trillion, the largest part concerning spot deals, and is considered deep and very liquid. By currency pairs, the EUR/USD is the most actively traded.

The primary factors affecting exchange rates include economic indicators, such as growth, interest rates and inflation, and political factors. Psychological factors also play a part given the large amount of speculative dealing in the market. In addition, the movement of several large FX dealers in the same direction can move the market. The interaction of these factors is complex, making FX prediction generally difficult.

There is justifiable scepticism in the ability to make money by predicting price changes in any given market. This scepticism reflects the efficient market hypothesis according to which markets fully integrate all of the available information, and prices fully adjust immediately once new information becomes available. In essence, the markets are fully efficient making prediction useless. However, in actual markets the reaction to new information is not necessarily so immediate. It is the existence of market inefficiencies that allows forecasting. However, the FX spot market is generally considered the most efficient, again making prediction difficult.

Forecasting exchange rates is vital for fund managers, borrowers, corporate treasurers, and specialised traders. However, the difficulties involved are demonstrated by that only three out of every ten spot foreign exchange dealers make a profit in any given year (Carney and Cunningham, 1996).

It is often difficult to identify a forecasting model because the underlying laws may not be clearly understood. In addition, FX time series may display signs of nonlinearity which traditional linear forecasting techniques are ill equipped to handle, often producing unsatisfactory results. Researchers confronted with problems of this nature increasingly resort to techniques that are heuristic and nonlinear. Such techniques include the use of Neural Network Regression (NNR) models.

The prediction of FX time series is one of the most challenging problems in forecasting. Our main motivation in this paper is to determine whether NNR models can extract any more from the data than traditional techniques. Over the past few years, NNR models have provided an attractive alternative tool for researchers and analysts, claiming improved performance over traditional techniques. However, they have received less attention within financial areas than in other fields.

Typically, NNR models are optimised using a mathematical criterion, and subsequently analysed using similar measures. However, statistical measures are often inappropriate for financial applications. Evaluation using financial measures may be more appropriate, such as risk-adjusted measures of return. In essence, trading driven by a model with a small forecast error may not be as profitable as a model selected using financial criteria.

The motivation for this research is to determine the added value, or otherwise, of NNR models by benchmarking their results against traditional forecasting techniques. Accordingly, financial trading models are developed for the EUR/USD exchange rate, using daily data from 17 October 1994 to 18 May 2000 for in-sample estimation, leaving the period from 19 May 2000 to 3 July 2001 for out-of-sample forecasting[1]. The trading models are evaluated in terms of forecasting accuracy *and* in terms of trading performance via a simulated trading strategy.

Our results clearly show that NNR models do indeed add value to the forecasting process.

The research is organised as follows. Section 2 presents a brief review of some of the research in FX markets. Section 3 describes the data used, addressing issues such as stationarity. Section 4 presents the benchmark models selected and our methodology. Section 5 briefly discusses NNR model theory and methodology, raising some issues surrounding the technique. Section 6 describes the out-of-sample forecasting accuracy and trading simulation results. Finally, Section 7 provides some concluding remarks.

2. Literature Review

Financial applications of NNR models began to emerge in the late Eighties. It is outside the scope of this research to provide an exhaustive survey of all FX applications. However, a brief review of some of the material is presented.

Bellgard and Goldschmidt (1999) examined the forecasting accuracy and trading performance of several traditional techniques, including random walk, exponential smoothing, and ARMA models with recurrent neural network (RNN) models[2]. The research was based on the Australian Dollar to US dollar (AUD/USD) exchange rate using half hourly data during 1996. They conclude that statistical forecasting accuracy measures do not have a direct bearing on profitability, and FX time series exhibit nonlinear patterns that are better exploited by neural network models.

Tyree and Long (1995) disagree, finding the random walk model more effective than the NNR models examined. They argue that although price changes are not strictly random, in their case the US dollar to Deutsche Mark (USD/DEM) daily price changes from 1990 to 1994, from a forecasting perspective what little structure is actually present may well be too negligible to be of any use. They acknowledge that the random walk is unlikely to be the optimal forecasting technique. However, they do not assess the performance of the models financially.

The USD/DEM daily price changes were also the focus for Refenes and Zaidi (1993). However they use the period 1984 to 1992, and take a different approach. They developed a hybrid system for managing exchange rates strategies. The idea was to use a neural network model to predict which of a portfolio of strategies is likely to perform best in the current context. The evaluation was based upon returns, and concludes that the hybrid system is superior to the traditional techniques of moving averages and mean-reverting processes.

El Shazly and El Shazly (1997) examined the one month forecasting performance of a NNR model compared with the forward rate of the British pound (GBP), German Mark (DEM), and Japanese Yen (JPY) against a common currency, although they do not state which, using weekly data from 1988 to 1994. Evaluation was based on forecasting accuracy and in terms of correctly forecasting the direction of the exchange

[1] The EUR/USD exchange rate only exists from 4 January 1999: it was retropolated from 17 October 1994 to 31 December 1998 and a synthetic EUR/USD series was created for that period using the fixed EUR/DEM conversion rate agreed in 1998, combined with the USD/DEM daily market rate.

[2] A brief discussion of RNN models is presented in Section 5.

rate. Essentially, they conclude that neural networks outperformed the forward rate both in terms of accuracy and correctness.

Similar FX rates are the focus for Gençay (1999). He examined the predictability of daily spot exchange rates using four models applied to five currencies, namely the French Franc (FRF), DEM, JPY, Swiss Franc (CHF), and GBP against a common currency from 1973 to 1992. The models include random walk, GARCH(1,1), NNR models and nearest neighbours. The models are evaluated in terms of forecasting accuracy and correctness of sign. Essentially, he concludes that non-parametric models dominate parametric ones. Of the non-parametric models, nearest neighbours dominate NNR models.

Yao *et al.* (1996) also analysed the predictability of the GBP, DEM, JPY, CHF, and AUD against the USD, from 1984 to 1995, but using weekly data. However, they take an ARMA model as a benchmark. Correctness of sign and trading performance were used to evaluate the models. They conclude that NNR models produce a higher correctness of sign, and consequently produce higher returns, than ARMA models. In addition, they state that without the use of extensive market data or knowledge, useful predictions can be made and significant paper profit can be achieved.

Yao *et al.* (1997) examine the ability to forecast the daily USD/CHF exchange rate using data from 1983 to 1995. To evaluate the performance of the NNR model, 'buy and hold' and 'trend following' strategies were used as benchmarks. Again, the performance was evaluated through correctness of sign and via a trading simulation. Essentially, compared with the two benchmarks, the NNR model performed better and produced greater paper profit.

Carney and Cunningham (1996) used four datasets over the period 1979 to 1995 to examine the single-step and multi-step prediction of the weekly GBP/USD, daily GBP/USD, weekly DEM/SEK (Swedish Krona) and daily GBP/DEM exchange rates. The neural network models were benchmarked by a naïve forecast and the evaluation was based on forecasting accuracy. The results were mixed, but concluded that neural network models are useful techniques that can make sense of complex data that defies traditional analysis.

A number of the successful forecasting claims using NNR models have been published. Unfortunately, some of the work suffers from inadequate documentation regarding methodology (El Shazly and El Shazly, 1997; Gençay, 1999). This makes it difficult to both replicate previous work and obtain an accurate assessment of just how well NNR modelling techniques perform in comparison to other forecasting techniques.

Notwithstanding, it seems pertinent to evaluate the use of NNR models as an alternative to traditional forecasting techniques, with the intention to ascertain their potential added value to this specific application, namely forecasting the EUR/USD exchange rate.

3. The Exchange Rate and Related Financial Data

The FX market is perhaps the only market that is open 24 hours a day, seven days a week. The market opens in Australasia, followed by the Far East, the Middle East and Europe, and finally America. Upon the close of America, Australasia returns to the market and begins the next 24-hour cycle. The implication to forecasting applications is that in certain circumstances, because of time-zone differences, researchers should be mindful when considering which data and which subsequent time lags to include.

In any time series analysis it is critical that the data used is clean and error free since the learning of patterns is totally data-dependent. Also significant in the study of FX time series forecasting is the rate at which data from the market is sampled. The sampling frequency depends on the objectives of the researcher and the availability of data. For example, intraday time series can be extremely noisy and “a typical off-floor trader…would most likely use daily data if designing a neural network as a component of an overall trading system” (Kastra and Boyd, 1996:220). For these reasons the time series used in this paper are all daily closing data obtained from a historical database provided by Datastream.

The investigation is based on the London daily closing prices for the EUR/USD exchange rate[3]. The obvious place to start selecting data, along with the EUR/USD, is with the other leading traded exchange rates. In addition, other related financial market data can be used, including stock market price indices, 3-month interest rates, 10-year benchmark bond yields, the price of Brent Crude oil, and the price of gold bullion. The price of commodities as represented by the CRB Index is also considered. The data obtained is presented in Table 1 along with their Datastream mnemonics.

Table 1 - Data and Datastream mnemonics

Number	Variable	Mnemonics
1	*FTSE 100 - PRICE INDEX*	FTSE100
2	*DAX 30 PERFORMANCE - PRICE INDEX*	DAXINDX
3	*S&P 500 COMPOSITE - PRICE INDEX*	S&PCOMP
4	*NIKKEI 225 STOCK AVERAGE - PRICE INDEX*	JAPDOWA
5	*FRANCE CAC 40 - PRICE INDEX*	FRCAC40
6	*MILAN MIB 30 - PRICE INDEX*	ITMIB30
7	*DJ EURO STOXX 50 - PRICE INDEX*	DJES50I
8	*US EURO-$ 3 MONTH (LDN:FT) - MIDDLE RATE*	ECUS$3M
9	*JAPAN EURO-$ 3 MONTH (LDN:FT) - MIDDLE RATE*	ECJAP3M
10	*EURO EURO-CURRENCY 3 MTH (LDN:FT) - MIDDLE RATE*	ECEUR3M
11	*GERMANY EURO-MARK 3 MTH (LDN:FT) - MIDDLE RATE*	ECWGM3M
12	*FRANCE EURO-FRANC 3 MTH (LDN:FT) - MIDDLE RATE*	ECFFR3M
13	*UK EURO-£ 3 MONTH (LDN:FT) - MIDDLE RATE*	ECUK£3M
14	*ITALY EURO-LIRE 3 MTH (LDN:FT) - MIDDLE RATE*	ECITL3M
15	*JAPAN BENCHMARK BOND -RYLD.10 YR (DS) - RED. YIELD*	JPBRYLD
16	*ECU BENCHMARK BOND 10 YR (DS)'DEAD' - RED. YIELD*	ECBRYLD
17	*GERMANY BENCHMARK BOND 10 YR (DS) - RED. YIELD*	BDBRYLD
18	*FRANCE BENCHMARK BOND 10 YR (DS) - RED. YIELD*	FRBRYLD
19	*UK BENCHMARK BOND 10 YR (DS) - RED. YIELD*	UKMBRYD
20	*US TREAS.BENCHMARK BOND 10 YR (DS) - RED. YIELD*	USBD10Y
21	*ITALY BENCHMARK BOND 10 YR (DS) - RED. YIELD*	ITBRYLD
22	*JAPANESE YEN TO US $ (WMR) - EXCHANGE RATE*	JAPAYE$
23	*US $ TO UK £ (WMR) - EXCHANGE RATE*	USDOLLR
24	*US $ TO EURO (WMR) - EXCHANGE RATE*	USEURSP
25	*Brent Crude-Current Month,fob U$/BBL*	OILBREN
26	*GOLD BULLION $/TROY OUNCE*	GOLDBLN
27	*Bridge/CRB Commodity Futures Index - PRICE INDEX*	NYFECRB

All the series span the period from 17 October 1994 to 3 July 2001, totalling 1749 trading days. The data is divided into two periods: the first period runs from 17 October 1994 to 18 May 2000 (1459 observations) used for model estimation and is classified

[3] EUR/USD is quoted as the number of USD per Euro: for example, a value of 1.2657 is USD1.2657 per Euro. The EUR/USD series for the period 1994-1998 was constructed as indicated in footnote 1.

in-sample, while the second period from 19 May 2000 to 3 July 2001 (290 observations) is reserved for out-of-sample forecasting and evaluation. The division amounts to approximately 17% being retained for out-of-sample purposes.

Over the review period there has been an overall appreciation of the USD against the Euro, as presented in Figure 1. The summary statistics of the EUR/USD for the examined period are presented in Table 2, highlighting slight skewness and low kurtosis. The indication is that the series requires some type of transformation. The use of data in levels in the FX market has many problems, "FX price movements are generally non-stationary and quite random in nature, and therefore not very suitable for learning purposes… Therefore for most neural network studies and analysis concerned with the FX market, price inputs are not a desirable set" (Mehta, 1995:191).

Figure 1 - EUR/USD London daily closing prices (17 October 1994 to 3 July 2001)

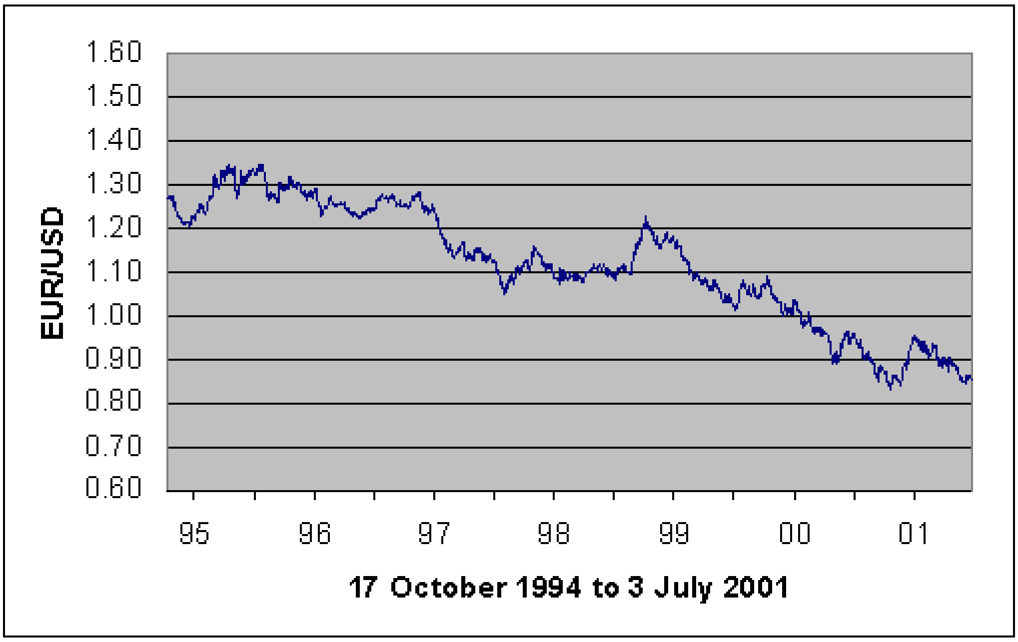


Table 2 - EUR/USD summary statistics (17 October 1994 to 3 July 2001)

Minimum	Mean	Maximum	Std. Dev.	Skewness	Kurtosis
0.8287	1.117697	1.3470	0.136898	-0.329711	2.080124

To overcome these problems, the EUR/USD series is transformed into rates of return. Given the price level P_1, P_2,…, P_t, the rate of return at time t is formed by:

$$R_t = \left(\frac{P_t}{P_{t-1}}\right) - 1$$

An advantage of using a returns series is that it helps in making the time series stationary, a useful statistical property.

Formal confirmation that the EUR/USD returns series is stationary is confirmed at the 5% significance level by both the ADF and Phillips-Peron test statistics.

Transformation into returns often creates a noisy time series. Formal confirmation through testing the significance of the autocorrelation coefficients reveals that the series is white noise at the 95% confidence level. For such series the best predictor of a future value is zero. In addition, very noisy data often makes forecasting difficult.

The EUR/USD returns summary statistics for the examined period are presented in Table 3. They reveal a slight skewness and high kurtosis, which “is common in high frequency financial time series data” (Gençay, 1999:94).

Table 3 - EUR/USD returns summary statistics (17 October 1994 to 3 July 2001)

Minimum	Mean	Maximum	Std. Dev.	Skewness	Kurtosis
-0.024898	-0.000214	0.033767	0.005735	0.434503	5.009624

A further transformation includes the creation of interest rates yield curve series, generated by:

$$YC = 10 \text{ year benchmark bond yields - } 3 \text{ month interest rates}$$

In addition, all of the time series are transformed into returns series in the manner described above to account for their non-stationarity.

4. Benchmark Models: Theory and Methodology

The premise of this research is to examine the use of NNR models in EUR/USD forecasting and trading models. Their performance is compared with other traditional forecasting techniques to ascertain their potential added value as a forecasting tool. Such methods include ARMA modelling, logit estimation, moving average convergence/divergence (MACD) technical models and a naïve strategy. Except for the straightforward naïve strategy, all benchmark models were estimated on our in -sample period. As all of these methods are well documented in the literature, they are as a result simply outlined below.

4.1 Naïve Strategy

The naïve strategy assumes that the most recent period change is the best predictor of the future. The simplest model is defined by:

$$\hat{Y}_{t+1} = Y_t$$

where Y_t is the actual rate of return at period *t*

$\hat{Y}_{t+1}$ is the forecast rate of return for the next period

The performance of the strategy is evaluated in terms of forecasting accuracy and in terms of trading performance via a simulated trading strategy.

4.2 MACD Strategy

Moving average methods are considered quick and inexpensive and as a result are routinely used in financial markets. The techniques use a weighted average of past observations to smooth short-term fluctuations. In essence, “a moving average is obtained by finding the mean for a specified set of values and then using it to forecast the next period” (Hanke and Reitsch, 1998:143).

The MACD model is defined as:

$$M_t = \hat{Y}_{t+1} = \frac{(Y_t + Y_{t-1} + Y_{t-2} + \ldots + Y_{t-n+1})}{n}$$

where M_t is the moving average at time *t*

n is the number of terms in the moving average

Y_t is the actual level at period t[4]

$\hat{Y}_{t+1}$ is the level forecast for the next period

The MACD strategy used is quite simple. Two moving average series are created with different moving average lengths. The decision rule for taking positions in the market is straightforward. Positions are taken if the moving averages intersect. If the short-term moving average intersects the long-term moving average from below a 'long' position is taken. Conversely, if the long-term moving average is intersected from above a 'short' position is taken[5].

The forecaster must use judgement when determining the number of periods *n* on which to base the moving averages. The combination that performed best over the in-sample period was retained for out-of-sample evaluation. The model selected was a combination of the EUR/USD and its 35-day moving average, namely *n* = 1 and 35 respectively or a (1,35) combination. The performance of this strategy is evaluated solely in terms of trading performance.

Several other 'adequate' models were produced and their performance evaluated. The trading performance of some of these combinations, such as the (1,40) and (1,60) combinations, and the (1,35) combination results were only marginally different. For example, the Sharpe ratio differs only by 0.02, and the average gain/loss ratio by 0.15. However, the (1,35) combination has the lowest maximum drawdown at –12.43% and lowest probability of a 10% loss at 4.95%[6]. On balance, the (1,35) combination was considered 'best' and therefore retained.

4.3 ARMA Methodology

ARMA models are particularly useful when information is limited to a single stationary series[7], or when economic theory is not useful. They are a "highly refined curve-fitting device that uses current and past values of the dependent variable to produce accurate short-term forecasts" (Hanke and Reitsch, 1998:407).

The ARMA methodology does not assume any particular pattern in a time-series, but uses an iterative approach to identify a possible model from a general class of models. Once a tentative model has been selected, it is subjected to tests of adequacy. If the specified model is not satisfactory, the process is repeated using other models until a satisfactory model is found. Sometimes, it is possible that two or more models may approximate the series equally well, in this case the most parsimonious model should prevail. For a full discussion on the procedure refer to Box *et al.* (1994), Gouriéroux and Monfort (1995), Pindyck and Rubinfeld (1998).

The ARMA model takes the form:

$$Y_t = f_0 + f_1 Y_{t-1} + f_2 Y_{t-2} + ... + f_p Y_{t-p} + e_t - w_1 e_{t-1} - w_2 e_{t-2} - ... - w_q e_{t-q}$$

---

[4] In this strategy the EUR/USD levels series is used as opposed to the returns series.

[5] A 'long' EUR/USD position means buying Euros at the current price, while a 'short' position means selling Euros at the current price.

[6] A discussion of the statistical and trading performance measures used to evaluate the strategies is presented in Section 6.

[7] The general class of ARMA models is for stationary time-series. If the series is not stationary an appropriate transformation is required.

where Y_t is the dependent variable at time *t*

Y_{t-1}, Y_{t-2}, and Y_{t-p} are the lagged dependent variable

f_0, f_1, f_2, and f_p are regression coefficients

e_t is the residual term

e_{t-1}, e_{t-2}, and e_{t-p} are previous values of the residual

w_1, w_2, and w_q are weights

Several ARMA specifications were tried out. In particular, an ARMA(4,4) model was estimated but was unsatisfactory as several coefficients were not significant at the 95% confidence level. However, once its non-significant AR(1) and MA(1) terms are removed all of the coefficients become significant at the 95% confidence level. Examination of the autocorrelation function of the error terms reveals that the residuals are random at the 95% confidence level and a further confirmation is given by the serial correlation LM test.

The selected ARMA model takes the form:

$$Y_t = -0.0002 + 1.1510 Y_{t-2} - 0.3620 Y_{t-3} - 0.7489 Y_{t-4} - 1.1686 e_{t-2} + 0.3519 e_{t-3} + 0.7569 e_{t-4}$$

The model selected was retained for out-of-sample estimation. The performance of the strategy is evaluated in terms of traditional forecasting accuracy and in terms of trading performance. Several other adequate models were produced and their performance evaluated. For example, ARMA(5,5), and ARMA(10,10) models were produced to check for any 'weekly' effect. None performed better, consequently the model selected was retained. Ultimately, we picked the model with the best in-sample trading performance and that satisfied the usual statistical tests.

4.4 Logit Estimation

The logit model belongs to a group of models termed classification models. They are a multivariate statistical technique used to estimate the probability of an upward or downward movement in a variable. As a result they are well suited to rates of return applications where a recommendation for trading is required. For a full discussion of the procedure refer to Thomas (1997), Pesaran and Pesaran (1997) or Maddala (2001).

The approach assumes the following regression model:

$$Y_t^* = b_0 + b_1 X_{1,t} + b_2 X_{2,t} + \ldots + b_p X_{p,t} + e_t$$

where Y_t^* is the dependent variable at time *t*

$X_{1,t}$, $X_{2,t}$, and $X_{p,t}$ are the explanatory variables at time *t*

b_0, b_1, b_2, and b_p are the regression coefficients

e_t is the residual term

However, Y_t^* is not directly observed; what is observed is a dummy variable Y_t defined by

$$Y_t = \begin{cases} 1 & \text{if } Y_t^* > 0 \\ 0 & \text{otherwise} \end{cases}$$

Therefore, the model requires a transformation of the explanatory variable, namely the EUR/USD returns series into a binary series. The procedure is quite simple: a binary variable equal to one is produced if the return is positive, and a zero otherwise. The same transformation for the explanatory variables, although not necessary, was performed for homogeneity reasons.

A basic regression technique is used to produce the logit model. The idea is to start with a model containing several variables, including lagged dependent terms, then through a series of tests the model is modified.

The selected logit model takes the form:

$$Y_t^* = 0.2492 - 0.3613X_{1t} - 0.2872X_{2t} + 0.2862X_{3t} + 0.2525X_{4t} - 0.3692X_{5t} - 0.3937X_{6t} + e_t$$

where $X_{1t},\ldots,X_{6t}$ are the JP_yc(-2), UK_yc(-9), JAPDOWA(-1), ITMIB30(-19), JAPAYE$(-10), and OILBREN(-1) binary explanatory variables, respectively (Datastream mnemonics as mentioned in Table 1, yield curves, and lags in brackets are used to save space).

All of the coefficients in the model are significant at the 95% confidence level. The overall significance of the model is tested using the likelihood ratio (LR) test. The null hypothesis that all the coefficients except the constant are not significantly different from zero is rejected at the 95% confidence level.

To justify the use of Japanese variables, which seems difficult from an economic perspective, the joint overall significance of this subset of variables is tested using the LR test for redundant variables. The null hypothesis that these coefficients, except the constant, are not jointly significantly different from zero is rejected at the 95% confidence level. In addition, a model that did not include the Japanese variables, but otherwise identical, was produced and the trading performance evaluated. The Sharpe ratio, average gain/loss ratio and correct directional change were 1.33, 1.01, and 54.31% respectively. The corresponding values for the selected model were 2.27, 1.01, and 58.19%.

The model selected was retained for out-of-sample estimation. The forecasts produced range between zero and one, requiring transformation into a binary series. Again, the procedure is quite simple: a binary variable equal to one is produced if the forecast is greater than 0.5, and a zero otherwise.

The performance of the strategy is evaluated solely in terms of trading performance. Several other adequate models were produced and their performance evaluated. None performed better in-sample, therefore the above model was retained.

5. Neural Network Models: Theory and Methodology

Neural networks require few *a priori* assumptions about the model under study, as a result they are well suited to problems where economic theory is of little use. In addition, neural networks are universal approximators capable of approximating any continuous function (Hornik *et al.*, 1989).

Many researchers are confronted with problems where important nonlinearities exist between the independent variables and the dependent variable. Often, in such circumstances, traditional forecasting methods lack explanatory power. Recently, nonlinear models have attempted to cover this shortfall. In particular, NNR models have been applied with increasing success to financial markets, which often contain nonlineraties (Dunis and Jalilov, 2001).

Theoretically, the advantage of NNR models over traditional forecasting methods is because, as is often the case, the model best adapted to a particular problem cannot be identified. It is then better to resort to method that is a generalisation of many models, than to rely on an *a priori* models (Dunis and Huang, 2001).

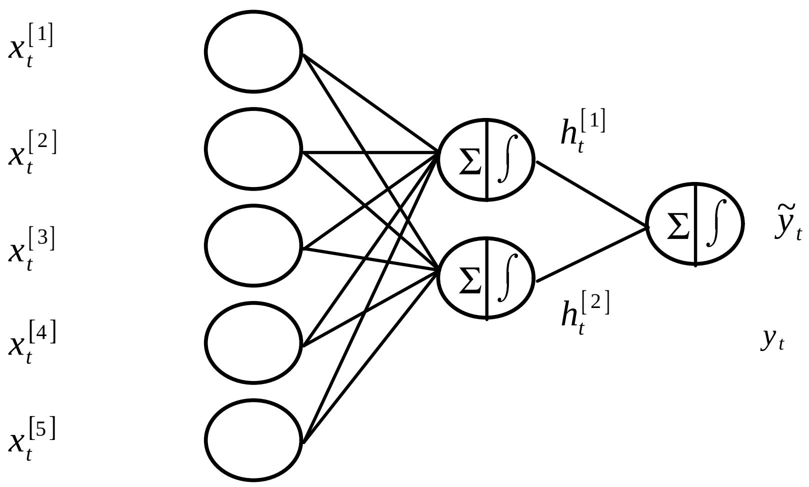
However, NNR models have been criticized and their widespread success has been hindered because of their black-box nature, excessive training times, danger of overfitting, and the large number of ‘parameters’ required for training. As a result, deciding on the appropriate network involves much trial and error.

For a full discussion on neural networks, please refer to Haykin (1999), Kaastra and Boyd (1996), Kingdon (1997), and Zhang *et al.* (1998). Notwithstanding, we give below a brief description of NNR models and procedures.

5.1 Neural Network Models

A neural network is typically organised into several layers of nodes. The first layer is the input layer, the number of nodes corresponding to the number of variables, and the last layer is the output layer, the number of nodes corresponding to the forecasting horizon for a forecasting problem[8]. The input and output layer can be separated by one or more hidden layers[9]. The nodes in adjacent layers are fully connected. Each neuron receives information from the preceding layer and transmits to the following layer only[10]. The neuron performs a weighted summation of its inputs; if the sum passes a threshold the neuron transmits, otherwise it remains inactive. In addition, a bias neuron may be connected to each neuron in the hidden and output layers. The bias has a value of positive one and is analogous to the intercept in regression models. An example of a fully connected NNR model with one hidden layer and two nodes is presented in Figure 2.

Figure 2 - A single output fully connected NNR model



[8] Linear regression models may be viewed analogous to neural networks with no hidden layers (Kaastra and Boyd, 1996).
[9] Networks with hidden layers are multilayer networks; a multilayer perceptron network is used in this research.
[10] If the flow of information through the network is from the input to the output, it is known as ‘feedforward'.

where $x_t^{[i]}$ $(i = 1,2,\cdots,5)$ are the NNR model inputs at time t

$h_t^{[j]}$ $(j = 1,2)$ are the hidden nodes outputs

y_t and $\tilde{y}_t$ are the actual value and NNR model output, respectively

The vector $A = (x^{[1]}, x^{[2]}, \cdots, x^{[n]})$ represents the input to the NNR model where $x_t^{[i]}$ is the level of activity of the i^{th} input. Associated with the input vector is a series of weight vectors $W_j = (w_{1j}, w_{2j}, \cdots, w_{nj})$ so that w_{ij} represents the strength of the connection between the input $x_t^{[i]}$ and the processing unit b_j. There may also be the input bias φ_j modulated with the weight w_{0j} associated with the inputs. The total input of the node b_j is the dot product between vectors A and W_j, less the weighted bias. It is then passed through a nonlinear activation function to produce the output value of processing unit b_j:

$$b_j = f\left(\sum_{i=1}^{n} x^{[i]} w_{ij} - w_{0j}\varphi_j\right) = f\left(X_j\right)$$

Typically, the activation function takes the form of the logistic function, which introduces a degree of nonlinearity to the model and prevents outputs from reaching very large values that can 'paralyse' NNR models and inhibit training (Kaastra and Boyd, 1996; Zhang *et al.*, 1998). This research uses the logistic function:

$$f\left(X_j\right) = \frac{1}{1 + e^{-Xj}}$$

The modelling process begins by assigning random values to the weights. The output value of the processing unit is passed on to the output layer. If the output is optimal, the process is halted, if not, the weights are adjusted and the process continues until an optimal solution is found. The output error, namely the difference between the actual value and the NNR model output, is the optimisation criterion. Commonly, the criterion is the root mean squared error (RMSE). The RMSE is systematically minimised through the adjustment of the weights. Basically, training is the process of determining the optimal solutions network weights, as they represent the knowledge learned by the network. Since inadequacies in the output are fed back through the network to adjust the network weights, the NNR model is trained by backpropagation[11] (Shapiro, 2000).

A common practice is to divide the time-series into three sets called the training, test and validation (out-of-sample) sets, and to partition them roughly $^2/_3$, $^1/_6$, and $^1/_6$ respectively. The testing set is used to evaluate the generalisation ability of the network. The technique consists of tracking the error on the training and test sets. Typically, the error on the training set continually decreases, however the test set error starts by decreasing and then begins to increase. From this point the network has stopped learning the similarities between the training and test sets, and has started to learn meaningless differences, namely the noise within the training data. For good generalisation ability, training should stop when the test set error reaches its lowest

[11] Backpropagation networks are the most common multilayer network and are the most used type in financial time series forecasting (Kaastra and Boyd, 1996). We exclusively use them in this research.

point. The stopping rule reduces the likelihood of overfitting, i.e. that the network will become overtrained (Mehta, 1995; Dunis and Huang, 2001).

An evaluation of the performance of the trained network is made on new examples not used in network selection, namely the validation set. Crucially, the validation set should never be used to discriminate between networks, as any set that is used to choose the best network is, by definition, a test set. In addition, good generalisation ability requires that the training and test sets are representative of the population, inappropriate selection will affect the network generalisation ability and forecast performance (Kaastra and Boyd, 1996; Zhang *et al.*, 1998).

5.2 Issues in Neural Network Modelling

Despite the satisfactory features of NNR models, the process of building them should not be taken lightly. There are many issues that can affect the networks performance and should be considered carefully.

The issue of finding the most parsimonious model is always a problem for statistical methods and particularly important for NNR models because of the problem of overfitting. Parsimonious models not only have the recognition ability but the more important generalisation ability. Overfitting and generalisation are always going to be a problem for real-world situations, this is particularly true for financial applications where time-series may well be quasi-random, or at least contain noise.

One of the most commonly used heuristics to ensure good generalisation is the application of some form of Occam's Razor. The principle states, "unnecessary complex models *should not* be preferred to simpler ones. However…more complex models always fit the data better" (Kingdon, 1997:49). The two objectives are, of course, contradictory. The solution is to find a model with the smallest possible complexity, and yet can still describe the data set (Kingdon, 1997; Haykin, 1999).

A reasonable strategy in designing NNR models is to start with one layer containing a few hidden nodes, and increase the complexity while monitoring the generalisation ability. The issue of determining the optimal number of layers and hidden nodes is a crucial factor for good network design, as the hidden nodes provide the ability to generalise. However, in most situations there is no way to determine the best number of hidden nodes without training several networks. Several rules of thumb have been proposed to aid the process, however none work well for all applications. Notwithstanding, simplicity must be the aim (Mehta, 1995).

Since NNR models are pattern matchers, the representation of data is critical for a successful network design. The raw data for the input and output variables are rarely fed into the network, they are generally scaled between the upper and lower bounds of the activation function. For the logistic function the range is [0, 1], avoiding the functions saturation zones. Practically, as in this research, a normalisation [0.2, 0.8] is often used with the logistic function, as its limits are only reached for infinite input values (Zhang *et al.*, 1998).

Crucial for backpropagation learning is the learning rate of the network as it determines the size of the weight changes. Smaller learning rates slow the learning process, while larger rates cause the error function to change wildly without continuously improving. To improve the process a momentum parameter is used which allows for larger learning rates. The parameter determines how past weight changes affect current weight changes, by making the next weight change in approximately the same direction as the previous one[12] (Kaastra and Boyd, 1996; Zhang *et al.*, 1998).

5.3 Neural Network Modelling Procedure

Conforming to standard heuristics, the training, test and validation sets were partitioned approximately $^2/_3$, $^1/_6$, and $^1/_6$ respectively. The training set runs from 17 October 1994 to 8 April 1999 (1169 observations), the test set runs from 9 April 1999 to 18 May 2000 (290 observations) and the validation set runs from 19 May 2000 to 3 July 2001 (290 observations), reserved for out-of-sample forecasting and evaluation, identical to the out-of-sample period for the benchmark models.

To start, traditional linear cross-correlation analysis helped establish the existence of a relationship between EUR/USD returns and potential explanatory variables. Although NNR models attempt to map nonlinearities, linear cross-correlation analysis can give some indication of which variables to include in a model, or at least a starting point to the analysis (Diekmann and Gutjahr, 1998, Dunis and Huang, 2001).

The analysis was performed for all potential explanatory variables. Lagged terms that were most significant as determined via the cross-correlation analysis are presented in Table 4.

Table 4 - Most significant lag of each potential explanatory variable (in returns)

Variable	Best Lag
DAXINDX	10
DJES50I	10
DMARKE$	16
FRCAC40	10
FTSE100	5
GOLDBLN	19
ITMIB	9
JAPAYE$	10
OILBREN	1
SPCOMP	1
USDOLLR	12
BD_yc	19
EC_yc	2
FR_yc	9
IT_yc	2
JP_yc	6
UK_yc	19
US_yc	1
NYFECRB	20

The lagged terms SPCOMP(-1) and US_yc(-1) could not be used because of time-zone differences between London and the US, as discussed at the beginning of Section 3. As an initial substitute SPCOMP(-2) and US_yc(-2) were used. In addition, various lagged terms of the EUR/USD returns were included as explanatory variables.

Variable selection was achieved via a forward stepwise NNR procedure, namely potential explanatory variables were progressively added to the network. If adding a new variable improved the level of explained variance over the previous 'best' network,

[12] The problem of convergence did not occur within this research, as a result a learning rate of 0.1 and momentum of zero were exclusively used.

the pool of explanatory variables was updated. Since the aim of the model building procedure is to build a model with good generalisation ability, a model that has a higher level of explained variance has a better ability. In addition, a good measure of this ability is to compare the level of explained variance of the test and validation sets: if the test set and validation set levels are similar the model has been built to generalise well.

The decision to use explained variance is because EUR/USD returns is a stationary series and stationarity remains important if NNR models are assessed on the level of explained variance (Dunis and Huang, 2001). The level of explained variance for the training, test and validation sets of the selected model are presented in Table 5.

Table 5 - NNR model explained variance for the training, test, and validation sets

Training Set	Test Set	Validation Set
3.4%	2.3%	2.2%

If after several attempts there was failure to improve on the previous 'best' model, variables in the model were alternated in an attempt to find a better combination. This procedure recognises the likelihood that some variables may only be relevant predictors when in combination with certain other variables.

Once a tentative model is selected, post-training weights analysis helps establish the importance of the explanatory variables. The idea is to find a measure of the contribution a given weight has to the overall output of the network, in essence allowing detection of insignificant variables. Such analysis includes an examination of the weight matrix within the network. The principle is to include in the network variables that are strongly significant. In addition, a small bias weight is preferred. The weight matrix of the selected model suggests that the explanatory variables are strongly significant. The input to hidden layer weight matrix of the final model is presented in Appendix 1.

The selected model contained the returns of the explanatory variables presented in Table 6, having one hidden layer containing five hidden nodes.

Table 6 – NNR model explanatory variables (in returns)

Variable	Lag
GOLDBLN	19
JAPAYE$	10
JAPDOWA	15
OILBREN	1
USDOLLR	12
FR_yc	2
IT_yc	6
JP_yc	9
JAPAYE$	1
JAPDOWA	1

Here again, to justify the use of the Japanese variables a further model that did not include these variables, but otherwise identical, was produced and the performance evaluated. The levels of explained variance for the training and test sets of this further model were 1.4 and 0.6 respectively, which are much lower than the selected model.

The model selected was retained for out-of-sample estimation. The performance of the strategy is evaluated in terms of traditional forecasting accuracy and in terms of trading performance.

Several other adequate models were produced and their performance evaluated, including recurrent neural network (RNN) models[13]. In essence, the only difference from NNR models is the addition of a loop back from a hidden or the output layer, to the input layer. The loop back is then used as an input in the next period. There is no theoretical or empirical answer to whether the hidden layer or the output should be looped back. However, the looping back of either allows RNN models to keep the memory of the past[14], a useful property in forecasting applications. However, this feature comes at a cost, as RNN models require more connections, raising the issue of complexity. Since simplicity is the aim, a less complex model that can still describe the data set is preferred.

The statistical forecasting accuracy results of the NNR model and the RNN model were only marginally different, namely the mean absolute percentage error (MAPE) differs by 0.06%, and the Theil's Inequality Coefficient by 0.0002. However, the results in terms of trading performance were identical.

The decision to retain the NNR model over its RNN counterpart is because the RNN model is more complex and yet does not possess any decisive added value over the simpler model.

6. Forecasting Accuracy and Trading Simulation

To compare the performance of the strategies, it is necessary to evaluate them on previously unseen data. This situation is likely to be the closest to a true forecasting or trading situation. To achieve this, all models retained an identical out-of-sample period allowing a direct comparison of their forecasting accuracy and trading performance.

6.1 Out-of-Sample Forecasting Accuracy Measures

Several criteria are used to make comparisons between the forecasting ability of the benchmark and NNR models, including mean absolute error (MAE), root mean squared error (RMSE)[15], mean absolute percentage error (MAPE) and Theil's inequality coefficient (Theil-U)[16]. For a full discussion on these measures, refer to Hanke and Reitsch (1998), and Pindyck and Rubinfeld (1998). We also include correct directional change (CDC) which measures the capacity of a model to correctly predict the subsequent actual change of a forecast variable. The statistical performance measures used to analyse the forecasting techniques are presented in Appendix 2.

6.2 Out-of-sample Trading Performance Measures

Statistical performance measures are often inappropriate for financial applications. Typically, modelling techniques are optimised using a mathematical criterion, but ultimately the results are analysed on a financial criterion upon which it is not optimised. In other words, the forecast error may have been minimised during model

---

[13] For a discussion on recurrent neural network models refer to Dunis and Huang (2001).

[14] The looping back of the output layer is an error feedback mechanism, implying the use of a nonlinear error-correction model (Dunis and Huang, 2001).

[15] The MAE and RMSE statistics are scale-dependent measures but allow a comparison between the actual and forecasts values, the lower the values the better the forecasting accuracy.

[16] When it is more important to evaluate the forecast errors independently of the scale of the variables, the MAPE and Theil-U are used. They are constructed to lie within [0,1], zero indicating a perfect fit.

estimation, but the evaluation of the true merit should be based on the performance of a trading strategy. Without actual trading, the best means of evaluating performance is via a simulated trading strategy. The procedure to create the buy and sell signals is quite simple: a EUR/USD buy signal is produced if the forecast is positive, and a sell otherwise[17].

For many traders and analysts market direction is more important than the value of the forecast itself, as in financial markets money can be made simply by knowing the direction the series will move. In essence, "low forecast errors and trading profits are not synonymous since a single large trade forecasted incorrectly … could have accounted for most of the trading system's profits" (Kaastra and Boyd, 1996:229).

The trading performance measures used to analyse the forecasting techniques are presented in Appendix 3. Some of the more important measures include the Sharpe ratio, maximum drawdown and average gain/loss ratio. The Sharpe ratio is a risk-adjusted measure of return, with higher ratios preferred to those that are lower, the maximum drawdown is a measure of downside risk and the average gain/loss ratio is a measure of overall gain, a value above one being preferred (Fernandez-Rodriguez *et al.*, 2000; Dunis and Jalilov, 2001).

The application of these measures may be a better standard for determining the quality of the forecasts. After all, the financial gain from a given strategy depends on trading performance, not on forecast accuracy.

6.3 Out-of-Sample Forecasting Accuracy Results

The forecasting accuracy statistics do not provide very conclusive results, unless one includes the CDC measure. Each of the models evaluated are nominated 'best' at least once. Interestingly, the naïve model has the lowest Theil -U statistic at 0.69; if this model is believed to be the 'best' model there is likely to be no added value using more complicated forecasting techniques. The ARMA model has the lowest MAPE statistic at 99.80%. The NNR model has the lowest MAE and RMSE statistics, however the values are only marginally less than the ARMA model. It is really the CDC measure that singles out the NNR model as 'best' performer, predicting most accurately 57.24% of the time. A majority decision rule would therefore select the NNR model as the overall 'best' model. A comparison of the forecasting accuracy results is presented in Table 7.

Table 7 - Forecasting accuracy results[18]

	Naïve	MACD	ARMA	Logit	NNR
Mean Absolute Error	0.0080	-	0.0057	-	**0.0056**
Mean Absolute Percentage Error	315.67%	-	**99.80%**	-	107.38%
Root Mean Squared Error	0.0102	-	0.0074	-	**0.0073**
Theil's Inequality Coefficient	**0.6900**	-	0.9452	-	0.8788
Correct Directional Change	55.86%	28.57%	52.76%	53.79%	**57.24%**

6.4 Out-of-Sample Trading Performance Results

A comparison of the trading performance results is presented in Table 8. The results of the NNR model are quite impressive. It generally outperforms the benchmark

[17] A buy signal is to buy Euros at the current price or continue holding Euros, while a sell signal is to sell Euros at the current price or continue holding US dollars.
[18] As the MACD model is not based on forecasting the next period and binary variables are used in the logit model, statistical accuracy comparisons with these models were not always possible.

strategies, both in terms of overall profitability with annualised return of 29.68%, and in terms of risk-adjusted performance with a Sharpe ratio 2.57. The downside risk as measured by the probability of a 10% loss is the lowest at 0.09%; however the logit model has the lowest downside risk as measured by maximum drawdown at -5.79%.

Table 8 - Trading performance results

	Naïve	**MACD**	**ARMA**	**Logit**	**NNR**
Annualised Return	21.34%	15.25%	4.99%	21.05%	**29.68%**
Cumulative Return	24.56%	17.55%	5.74%	24.22%	**34.16%**
Annualised Volatility	11.64%	11.70%	11.71%	11.64%	**11.56%**
Sharpe Ratio	1.83	1.30	0.43	1.81	**2.57**
Maximum Daily Profit	3.38%	1.84%	3.38%	1.88%	**3.38%**
Maximum Daily Loss	-2.10%	-3.23%	-2.10%	-3.38%	**-1.82%**
Maximum Drawdown	-9.06%	-6.12%	-10.66%	**-5.79%**	-9.12%
% Winning Trades	55.86%	28.57%	52.76%	53.79%	**57.24%**
% Losing Trades	44.14%	71.43%	47.24%	46.21%	**42.76%**
Number of Up Periods	162	4	153	156	**166**
Number of Down Periods	126	**10**	135	132	122
Number of Transactions	127	**15**	53	141	136
Total Trading Days	290	290	290	290	290
Avg Gain in Up Periods	0.58%	**6.31%**	0.56%	0.61%	0.60%
Avg Loss in Down Periods	-0.56%	-0.77%	-0.59%	**-0.53%**	-0.54%
Avg Gain/Loss Ratio	1.05	**8.19**	0.95	1.14	1.12
Probability of 10% Loss	0.70%	10.81%	38.39%	0.76%	**0.09%**
Profits T-statistics	76.50	54.39	7.25	30.79	43.71
Number of Periods Daily returns Rise	128	128	128	128	128
Number of Periods Daily returns Fall	162	162	162	162	162
Number of Winning up Periods	**65**	-	40	49	52
Number of Winning down Periods	97	-	113	106	**114**
% Winning up Periods	**50.78%**	-	31.25%	38.28%	40.63%
% Winning down Periods	59.88%	-	69.75%	65.43%	**70.37%**

The NNR model predicted the highest number of winning down periods at 114. The naïve model forecast the highest number of winning up periods at 65, however the NNR model was 'second best' for this measure. Interestingly, all models were more successful at forecasting a fall in the EUR/USD returns series, as indicated by a greater percentage of winning down periods to winning up periods.

The NNR model has the highest number of transactions at 136, while the MACD strategy has the lowest at 15. In essence, the MACD strategy has longer 'holding' periods compared to the other models, suggesting that the MACD strategy is not compared 'like with like' to the other models. In addition, the MACD strategy has the highest average gain/loss ratio at 8.19, but again this value cannot be compared 'like with like' to the other models.

As with statistical performance measures, financial criteria clearly single out the NNR model as the one with the most consistent performance: it is therefore considered the 'best' model for this particular application.

6.5 Transaction Costs

So far, our results have been presented without accounting for transaction costs during the trading simulation. However, it is not realistic to account for the success or otherwise of a trading system unless transactions costs are taken into account. Between market makers, a cost of 3 pips (0.0003 EUR/USD) per trade (one way) for a tradable amount, typically USD 5-10 million, would be normal. The NNR model had the highest number of transactions at 136. The procedure to approximate the transaction costs for the NNR model is quite simple. A cost of 3 pips per trade and an average out-of-sample EUR/USD 0.8971 value produce an average cost of 0.033% per trade. Since the EUR/USD time series is a series of bid rates, the approximate out-of-sample transactions costs for the NNR model trading strategy is about 2.27%, namely 0.033%*(136/2). Therefore, even accounting for transaction costs, the extra returns achieved with the NNR model still make this strategy the most attractive one despite its relatively high trading frequency.

7. Concluding Remarks

This research has evaluated the use of NNR models in forecasting and trading the EUR/USD exchange rate. The performance was measured statistically and financially via a trading simulation taking into account the impact of transaction costs on models with higher trading frequencies. The logic behind the trading simulation is, if profit from a trading simulation is compared solely on the basis of statistical measures, the optimum model from a financial perspective would rarely be chosen.

The NNR model was benchmarked against traditional forecasting techniques to determine any added value to the forecasting process. Having constructed a synthetic EUR/USD series for the period up to 4 January 1999, the models were developed using the same in-sample data, 17 October 1994 to 18 May 2000, leaving the remaining period, 19 May 2000 to 3 July 2001, for out-of-sample forecasting.

Forecasting techniques rely on the weaknesses of the efficient market hypothesis, acknowledging the existence of market inefficiencies, with markets displaying even weak signs of predictability. However, FX markets are relatively efficient, reducing the scope of a profitable strategy. Consequently, the FX managed futures industry average Sharpe ratio is only 0.8, although a percentage of winning trades greater than 60% is often required to run a profitable FX trading desk (Grabbe, 1996). In this respect, it is worth noting that all our models failed to reach a 60% accuracy of winning trades, the highest of which was the NNR model at 57.24%. Nevertheless, all but one of the models examined in this research achieved an out-of-sample Sharpe ratio higher than 0.8, the highest of which was again the NNR model at 2.57. This seems to confirm that the use of quantitative trading is more appropriate in a fund management than in a treasury type of context.

Forecasting techniques are dependent on the quality and nature of the data used. If the solution to a problem is not within the data, then no technique can extract it. In addition, sufficient information should be contained within the in-sample period to be representative of all cases within the out-of-sample period. For example, a downward trending series typically has more falls represented in the data than rises. The EUR/USD is such a series within the in-sample period. Consequently, the forecasting techniques used are estimated using more negative values than positive values. The probable implication is that the models are more likely to successfully forecast a fall in the EUR/USD, as indicated by our results, with all models forecasting a higher percentage of winning down periods than winning up periods. However, the naïve model does not learn to generalise *per se*, and as a result has the smallest difference between the number of winning up to winning down periods.

Overall our results confirm the credibility and potential of NNR models as a forecasting technique. However, while NNR models offer a promising alternative to traditional techniques, they suffer from a number of limitations. One of the major disadvantages is the inability to explain their reasoning. In addition, statistical inference techniques such as significance testing cannot always be applied, resulting in a reliance on a heuristic approach. The complexity of NNR models suggests that they are capable of superior forecasts, as shown in this research, however this is not always the case. They are essentially nonlinear techniques and may be less capable in linear applications than traditional forecasting techniques (Campbell *et al.*, 1997; Balkin and Ord, 2000; Lisboa and Vellido, 2000).

Further investigation into RNN models is possible, or into combining forecasts. Many researchers agree that individual forecasting methods are misspecified in some manner, suggesting that combining multiple forecasts leads to increased forecast accuracy (Dunis and Huang, 2001). However, initial investigations proved unsuccessful, with the NNR model remaining the 'best' model. Two simple model combinations were examined, a simple averaging of the ARMA, naïve and NNR model forecasts, and a regression-type combined forecast using the ARMA, logit and NNR model. (For a full discussion on the procedures, refer to Clemen (1989), Granger and Ramanathan (1984) and Hashem (1997)). The lack of success using the combination models was undoubtedly because the performance of the benchmark models was so much weaker than that of the NNR model: it is unlikely that combining relatively 'poor' models with an otherwise 'good' one will outperform the 'good' model alone.

Overall, despite the limitations and potential improvements mentioned above, our results strongly suggest that NNR models can add value to the forecasting process, and that, for the EUR/USD exchange rate and the period considered, NNR models outperform the more traditional modelling techniques analysed in this paper.

Appendix 1 - The input to hidden layer weight matrix

	GOLD BLN (-19)	JAPAY E$ (-10)	JAP DOWA (-15)	OIL BREN (-1)	US DOLLR (-12)	FR_yc (-2)	IT_yc (-6)	JP_yc (-9)	JAPAY E$ (-1)	JAP DOWA (-1)	Bias
C[1,0]	0.2316	-0.2120	-0.4336	-0.4579	-0.2621	-0.3911	0.2408	0.4295	0.4067	0.4403	-0.0824
C[1,1]	0.4016	-0.1752	-0.3589	-0.5474	-0.3663	-0.4623	0.2438	0.2786	0.2757	0.4831	-0.0225
C[1,2]	0.2490	-0.3037	-0.4462	-0.5139	-0.2506	-0.3491	0.2900	0.3634	0.2737	0.4132	-0.0088
C[1,3]	0.3382	-0.3588	-0.4089	-0.5446	-0.2730	-0.4531	0.2555	0.4661	0.4153	0.5245	0.0373
C[1,4]	0.3338	-0.3283	-0.4086	-0.6108	-0.2362	-0.4828	0.3088	0.4192	0.4254	0.4779	-0.0447

Appendix 2 - Statistical performance measures

Performance Measure	Description
Mean Absolute Error (MAE)	$MAE = \frac{1}{T}\sum_{t=1}^{T}\lvert\tilde{y}_t - y_t\rvert$
Mean Absolute Percentage Error (MAPE)	$MAPE = \frac{100}{T}\sum_{t=1}^{T}\left\lvert\frac{\tilde{y}_t - y_t}{y_t}\right\rvert$
Root Mean Squared Error (RMSE)	$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(\tilde{y}_t - y_t)^2}$
Theil's Inequality Coefficient (Theil-U)	$U = \frac{\sqrt{\frac{1}{T}\sum_{t=1}^{T}(\tilde{y}_t - y_t)^2}}{\sqrt{\frac{1}{T}\sum_{t=1}^{T}(\tilde{y}_t)^2} + \sqrt{\frac{1}{T}\sum_{t=1}^{T}(y_t)^2}}$
Correct Directional Change	$CDC = \frac{100}{N}\sum_{t=1}^{N}D_t$ *where,* $D_t = 1$ *if* $y_t * \tilde{y}_t > 0$ *else* $D_t = 0$

where y_t is the actual value at time *t*

$\tilde{y}_t$ is forecast value

with $t = 1$ to $t = T$ for the forecast period

Appendix 3 - Trading simulation performance measures

Performance Measure	Description
Number of Periods Daily returns Rise	$NPR = \sum_{t=1}^{N}Q_t$ *where,* $Q_t = 1$ *if* $y_t > 0$ *else* $Q_t = 0$
Number of Periods Daily returns Fall	$NPF = \sum_{t=1}^{N}S_t$ *where,* $S_t = 1$ *if* $y_t < 0$ *else* $S_t = 0$
Number of Winning up Periods	$NWU = \sum_{t=1}^{N}B_t$ *where,* $B_t = 1$ *if* $R_t > 0$ *and* $y_t > 0$ *else* $B_t = 0$
Number of Winning down Periods	$NWD = \sum_{t=1}^{N}E_t$ *where,* $E_t = 1$ *if* $R_t > 0$ *and* $y_t < 0$ *else* $E_t = 0$
Winning up Periods (%)	*WUT=100*(NWU/NPR)*
Winning down Periods (%)	*WDT=100*(NWD/NPF)*

Appendix 3 - Trading simulation performance measures (continued)

Performance Measure	Description
Annualised Return	$R^{A} = 252 * \frac{1}{N}\sum_{t=1}^{N} R_t$
Cumulative Return	$R^{C} = \sum_{t=1}^{N} \hat{R}_T$
Annualised Volatility	$s^{A} = \sqrt{252} * \sqrt{\frac{1}{N-1} * \sum_{t=1}^{N} \left(R_t - \overline{R}\right)^2}$
Sharpe Ratio	$SR = \frac{R^A}{s^A}$
Maximum Daily Profit	*Maximum value of* R_t *over the period*
Maximum Daily Loss	*Minimum value of* R_t *over the period*
Maximum Drawdown	*Maximum negative value of* $\sum\left(\hat{R}_T\right)$ *over the period* $MD = \underset{i=1,\cdots,t;t=1,\cdots,N}{Min}\left(\sum_{j=i}^{t} X_j\right)$
% Winning Trades	*WT=100*(Number of* R_t *>0)/Total number of trades*
% Losing Trades	*LT=100*(Number of* R_t *<0)/Total number of trades*
Number of Up Periods	*Nup= Number of* R_t *>0*
Number of Down Periods	*Ndown= Number of* R_t *<0*
Number of Transactions	$NT = \sum_{t=1}^{N} L_t$ *where,* $L_t = 1$ *if* $\tilde{y}_t * \tilde{y}_{t-1} < 0$ *else* $L_t = 0$
Total Trading Days	*Number of all* R_t *s*
Avg Gain in Up Periods	*AG=(Sum of all* R_t *>0)/Nup*
Avg Loss in Down Periods	*AL=(Sum of all* R_t *<0)/Ndown*
Avg Gain/Loss Ratio	*GL=AG/AL*
Probability of 10% Loss	$PoL = \left[\frac{(1-P)}{P}\right]^{\left(\frac{MaxRisk}{A}\right)}$ *where,* $P = 0.5 * \left(1 + \left(\frac{\langle (WT * AG) + (LT * AL) \rangle}{\sqrt{\left[(WT * AG^2) + (LT * AL^2)\right]}}\right)\right)$ *MaxRisk is the risk level defined by the user; this research, 10%*
Profits T-statistics	*T-statistics=* $\sqrt{N} * \frac{R^A}{s^A}$

(Dunis and Jalilov, 2001)

References

Balkin, S. D. and Ord, J. K. (2000), 'Automatic Neural Network Modelling for Univariate Time Series', *International Journal of Forecasting*, **16**, 509-515.

Bellgard, C. and Goldschmidt, P. (1999), 'Forecasting Across Frequencies: Linearity and Non-Linearity', *University of Western Australia Research Paper, Proceedings of the International Conference on Advanced Technology*, Australia, www.imm.ecel.uwa.edu.au/~cbellgar/

Box, G. E. P., Jenkins, G. M. and Gregory, G. C. (1994), *Time series Analysis: Forecasting and Control*, Prentice-Hall, New Jersey.

Campbell, I. Y., Lo, A. W. and MacKinley, A. C. (1997), 'Nonlinearities in Financial Data', 512-524, in *The Econometrics of Financial Markets,* Princeton University Press, Princeton.

Carney, J.C. and Cunningham, P. (1996), 'Neural Networks and Currency Exchange Rate Prediction', *Trinity College Working Paper, Foresight Business Journal Web page*, www.maths.tcd.ie/pub/fbj/forex4.html

Clemen, R. T. (1989), 'Combining Forecasts: A Review and Annotated Bibliography', *International Journal of Forecasting*, **5**, 559-583.

Diekmann, A. and Gutjahr, S. (1998), 'Prediction of the Euro-Dollar Future Using Neural Networks – A Case Study for Financial Time Series Prediction', *University of Karlsruhe Working Paper*, *Proceedings of the International Symposium on Intelligent Data Engineering and Learning (IDEAL'98)*, Hong Kong, http://citeseer.nj.nec.com/diekmann98prediction.html

Dunis, C. and Huang, X. (2001), 'Forecasting and Trading Currency Volatility: An Application of Recurrent Neural Regression and Model Combination', *Liverpool Business School Working Paper,* www.cibef.com , forthcoming in *The Journal of Forecasting*.

Dunis, C. and Jalilov, J. (2001), 'Neural Network Regression and Alternative Forecasting Techniques for Predicting Financial Variables', *Liverpool Business School Working Paper,* www.cibef.com

El-Shazly, M. R. and El-Shazly, H. E. (1997), 'Comparing the Forecasting Performance of Neural Networks and Forward Exchange Rates', *Journal of Multinational Financial Management*, **7**, 345-356.

Fernandez-Rodriguez, F., Gonzalez-Martel, C. and Sosvilla-Rivero, S. (2000), 'On the Profitability of Technical Trading Rules Based on Artificial Neural Networks: Evidence from the Madrid Stock Market', *Economics Letters*, **69**, 89-94.

Gençay, R. (1999), 'Linear, Non-linear and Essential Foreign Exchange Rate Prediction with Simple Technical Trading Rules', *Journal of International Economics*, **47**, 91-107.

Gouriéroux, C. and Monfort, A. (1995), *Time Series and Dynamic Models*, translated and edited by G. Gallo, Cambridge University Press, Cambridge.

Grabbe, J. O. (1996), *International Financial Markets*, 3rd edition, Prentice-Hall, New Jersey.

Granger, C. W. J. and Ramanathan, R. (1984), 'Improved Methods of Combining Forecasts', *Journal of Forecasting*, **3**, 197-204.

Hanke, J. E. and Reitsch, A. G. (1998), *Business Forecasting*, 6th edition, Prentice-Hall, New Jersey.

Haykin, S. (1999), *Neural Networks: A Comprehensive Foundation*, 2nd edition, Prentice-Hall, New Jersey.

Hashem, S. (1997), 'Optimal Linear Combinations of Neural Networks', *Neural Networks*, **10**, 4, 599-614, www.emsl.pnl.gov:2080/people/bionames/hashem_s.html

Hornik, K., Stinchcombe M. and White, H. (1989), 'Multilayer Feedforward Networks Are Universal Approximators', *Neural Networks*, **2**, 359-366.

Kaastra, I. and Boyd, M. (1996), 'Designing a Neural Network for Forecasting Financial and Economic Time Series', *Neurocomputing,* 10, 215-236.

Kingdon, J. (1997), *Intelligent Systems and Financial Forecasting,* Springer, London.

Lisboa, P. J. G. and Vellido, A. (2000), 'Business Applications of Neural Networks', vii-xxii, in P. J. G. Lisboa, B. Edisbury and A. Vellido [eds.] *Business Applications of Neural Networks: The State-of-the-Art of Real-World Applications,* World Scientific, Singapore.

Maddala, G. S. (2001), *Introduction to Econometrics*, 3rd edition, Prentice-Hall, New Jersey.

Mehta, M. (1995), 'Foreign Exchange Markets', 176-198, in A. N. Refenes [ed.], *Neural Networks in the Capital Markets*, John Wiley, Chichester.

Pesaran, M. H. and Pesaran, B. (1997), 'Lessons in Logit and Probit Estimation', 263-275 in *Interactive Econometric Analysis Working with Microfit 4*, Oxford University Press, Oxford.

Pindyck, R. S. and Rubinfeld, D. L. (1998), *Econometric Models and Economic Forecasts*, 4th edition, McGraw-Hill, New York.

Refenes, A. N. and Zaidi, A. (1993), 'Managing Exchange Rate Prediction Strategies with Neural Networks', 109-116, in P. J. G. Lisboa and M. J. Taylor [eds.], *Techniques and Applications of Neural Networks,* Ellis Horwood, Hemel Hampstead.

Shapiro, A. F. (2000), 'A Hitchhiker's Guide to the Techniques of Adaptive Nonlinear Models', *Insurance, Mathematics and Economics*, **26**, 119-132.

Thomas, R. L. (1997), *Modern Econometrics. An Introduction*, Addison-Wesley, Harlow.

Tyree, E. W. and Long, J. A. (1995), 'Forecasting Currency Exchange Rates: Neural Networks and the Random Walk Model', *City University Working Paper, Proceedings of the Third International Conference on Artificial Intelligence Applications*, New York, http://citeseer.nj.nec.com/131893.html

Yao, J., Poh, H. and Jasic, T. (1996), 'Foreign Exchange Rates Forecasting with Neural Networks', *National University of Singapore Working Paper, Proceedings of the International Conference on Neural Information Processing,* Hong Kong, http://citeseer.nj.com/yao96foreign.html

Yao, J., Li, Y. and Tan, C. L. (1997), 'Forecasting the Exchange Rates of CHF vs USD Using Neural Networks', *Journal of Computational Intelligence in Finance*, **15**, 2, 7-13.

Zhang, G., Patuwo, B. E. and Hu, M. Y. (1998), 'Forecasting with Artificial Neural Networks: The State of The Art', *International Journal of Forecasting,* **14**, 35-62.