FORECASTING EXCHANGE RATES USING GENERAL REGRESSION NEURAL NETWORKS

Mark T. Leung

Department of Operations and Decision Technologies
Kelley School of Business
Indiana University
Bloomington, Indiana 47405
U.S.A.
Tel: 812-855-8440
Fax: 812-856-5222
Email: mleung@indiana.edu

An-Sing Chen

Department of Finance
National Chung Cheng University
Ming Hsiung, Chia Yi
Taiwan, R.O.C. 612
Tel: 011-886-5-272-0411 ext 6320
Fax: 011-886-5-272-0818
Email: finasc@ccunix.ccu.edu.tw

Hazem Daouk

Department of Applied Economics and Management
Cornell University
Ithaca, New York 14853
U.S.A.
Tel: 607-255-6459
Fax: 607-255-9984
Email: hd35@cornell.edu

JEL classification code: G15, C45, C53.

FORECASTING EXCHANGE RATES USING GENERAL REGRESSION NEURAL NETWORKS

Scope and Purpose -- Predicting currency movements has always been a problematic task as most conventional econometric models are not able to forecast exchange rates with significantly higher accuracy than a naive random walk model. For large multinational firms which conduct substantial currency transfers in the course of business, being able to accurately forecast the movements of exchange rates can result in considerable improvement in the overall profitability of the firm. In this study, we apply the General Regression Neural Network (GRNN) to predict the monthly exchange rates of three currencies, British pound, Canadian dollar, and Japanese yen. Our empirical experiment shows that the performance of GRNN is better than other neural network and econometric techniques included in this study. The results demonstrate the predictive strength of GRNN and its potential for solving financial forecasting problems.

Keywords: General regression neural networks, currency exchange rate, forecasting.

1. INTRODUCTION

Applying quantitative methods for forecasting in financial markets and assisting investment decision making has become more indispensable in business practices than ever before. For large multinational firms which conduct substantial currency transfers in the course of business, being able to accurately forecast movements of currency exchange rates can result in substantial improvement in the overall profitability of the firm. Nevertheless, predicting exchange rate movements is still a problematic task. Most conventional econometric models are not able to forecast exchange rates with significantly higher accuracy than a naive random walk model. In recent years, there has been a growing interest to adopt the state-of-the-art artificial intelligence technologies to solve the problem. One stream of these advanced techniques focuses on the use of artificial neural networks (ANN) to analyze the historical data and provide predictions to future movements in the foreign exchange market. In this study, we apply a special class of neural networks, called General Regression Neural Network (GRNN), to forecast the monthly exchange rates for three internationally traded currencies, Canadian dollar, Japanese yen, and British pound. In order to provide a fair and robust evaluation of the GRNN, relative performance against other forecasting approaches are included in our empirical investigation. Specifically, the performance of the GRNN is compared with those of the multi-layered feedforward neural network (MLFN), the type of ANN used in many research studies, and the models based on multivariate transfer function, a general econometric forecasting tool. In addition, random walk forecasts are generated to give benchmark comparisons.

The remainder of this paper is organized as follows. A review of the literature is given in the next section. In section 3, we give an explanation of the logic and operations used by the GRNN. Then, a description of the empirical data employed in our study is provided in section 4.

This section also explains the neural network training paradigm and presents the model specifications for the forecasting approaches tested in the experiment. In section 5, a description to our statistical analyses is presented and the experimental findings are discussed. The paper is then concluded in section 6.

2. LITERATURE REVIEW

2.1 Exchange Rate Forecasting

The theory of exchange rate determination is in an unsatisfactory state. From the theoretical point of view, the debates between the traditional flow approach and the modern asset-market approach have seen the victory of the latter. However, from the empirical point of view, the forecasting ability of the theoretical models remains very poor. The results of Meese and Rogoff [1][2][3] came as a shock to the profession. According to these studies, the current structural models of exchange-rate determination failed to outperform the random walk model in out-of-sample forecasting experiments even when the actual realized values of the explanatory variables were used (ex-post forecasts). The results of Meese and Rogoff have been confirmed in many subsequent papers (e.g., Alexander and Thomas [4], Gandolfo, Padoan and Paladino [5][6], and Sarantis and Stewart [7][8]). Moreover, even with the use of sophisticated techniques, such as time-varying-coefficients, the forecasts are only marginally better that the random walk model (Schinasi and Swamy [9]).

The models considered by Meese and Rogoff were the flexible-price (Frenkel-Bilson) monetary model, the sticky-price (Dornbusch-Frankel) monetary model, and the sticky-price (Hooper-Morton) asset model. Other models for exchange rate forecasting include the portfolio balance model, and the uncovered interest parity model. Of the three models used by Meese and

Rogoff, the Hooper-Morton model is more general and nests the other two models. However, as argued in the previous paragraph, those three models have been shown to be unsatisfactory. The evidence related to the portfolio balance model was ambiguous (Sarantis and Stewart [7]). On the other hand, both Fisher et al. [10] and Sarantis and Stewart [7] found that the models based on a modified uncovered interest parity (MUIP) approach showed some promising results compared to the other conventional theoretical approaches. In a related paper, Sarantis and Stewart [8] provided evidence that exchange rate forecasting models based on the MUIP produced more accurate out-of-sample forecasts than the portfolio balance model for all forecast horizons, thus arguing in favor of the MUIP approach to exchange rate determination and forecasting.

In light of the previous literature, we will consider both univariate and multivariate approaches to exchange rate forecasting. Furthermore, given the mounting evidence favoring the MUIP approach to exchange rate modeling, we use the MUIP relationship as the theoretical basis for our multivariate specifications in our experimental investigation.

2.2 MUIP Exchange Rate Relationship

Essentially, Sarantis and Stewart [7][8] showed that the MUIP relationship can be modeled and written as follow:

$$e_t = \alpha_0 + \alpha_1(r_t^* - r_t) + \alpha_2(\pi_t^* - \pi_t) + \alpha_3(p_t^* - p_t) + \alpha_4(ca_t/ny_t) + \alpha_5(ca_t^*/ny_t^*) + \mu_t \qquad (1)$$

where e is the natural logarithm of the exchange rate, defined as the foreign currency price of domestic currency. r, π, p, and (ca/ny) represent the logarithm of nominal short-term interest rate, expected price inflation rate, the logarithm of the price level, and the ratio of current account to nominal GDP for the domestic economy, respectively. Asterisks denote the corresponding foreign variables. μ is the error term. The variables (*ca/ny*) and (*ca*/ny**) are proxies for the risk premium. Equation 1 is a modified version of the real uncovered interest parity (MUIP) relationship adjusted for risk. Sarantis and Stewart [7] provided a more detailed discussion and investigation of this relationship.

2.3 Applications of Neural Networks

In the last decade, applications associated with artificial neural network (ANN) has been gaining popularity in both the academic research and practitioner's sectors. The basic structure of and operations performed by the ANN emulate those found in a biological neural system. Basic notions of the concepts and principals of the ANN systems and their training procedures can be found in Rumelhart and McClelland [11], Lippman [12], and Medsker, Turban and Trippi [13]. Many of the countless applications of this artificial intelligence technology are related to the financial decision making. Hawley, Johnson, and Raina [14] provides an overview of the neural network models in the field of finance.

According to Hawley et. al., there is a diversity of financial applications for neural networks in the areas of corporate finance, investment analysis, and banking. A review on recent literature also reveals financial studies on a wide variety of subjects such as bankruptcy prediction (Tsukuda and Baba [15], Tam and Kiang [16]), prediction of savings and loan association failures (Salchenberger, Cinar, and Lash [17]), credit evaluation (Jensen [18], Collins, Ghosh and Scofield [19]), analysis of financial statements (Kryzanowski and Galler [20]), corporate distress diagnosis (Altman, Marco, and Varetto [21]), bond rating (Dutta and

Shekhar [22]), initial stock pricing (Jain and Nag [23]), currency exchange rate forecasting (Refenes [24]), stock market analysis (Hutchinson, Lo, and Poggio [25], Kamijo and Tanigawa [26], Kimoto, Asakawa, Yoda, and Takeoka [27]), and forecasting in futures market (Kaastra and Boyd [28], Trippi and DeSieno [29]).

2.4 Studies Using General Regression Neural Networks

GRNN is a class of neural networks which is capable of performing kernel regression and other non-parametric functional approximations. Although GRNN has been shown to be a competent tool for solving many scientific and engineering problems (e.g., signal processing (Kendrick, Acton, and Duncan [30]), chemical processing (Mukesh [31]), structural analysis (Williams and Gucunski [32]), assessment of high power systems (Wehenkel [33]), the technique has not been widely applied to the field of finance and investment. The only one financial study primarily based on GRNN is Wittkemper and Steiner [34] in which the regression network is used to verify the predictability of the systematic risk of stocks.

3. GENERAL REGRESSION NEURAL NETWORK

The General Regression Neural Network (GRNN) was originally proposed and developed by Specht [35]. This class of network paradigm has the distinctive features of learning swiftly, working with simple and straightforward training algorithm, and being discriminative against infrequent outliers and erroneous observations. As its name implies, GRNN is capable of approximating any arbitrary function from historical data. The foundation of GRNN operation is essentially based on the theory of nonlinear (kernel) regression --- estimation of the expected value of output given a set of inputs. Although GRNN can provide a multivariate vector of outputs, without any loss of generality, our description of the GRNN operating logic is simplified to the case of univariate output. Equation 2 summarizes the GRNN logic in an equivalent nonlinear regression formula:

$$E[y \mid X] = \frac{\int_{-\infty}^{\infty} y f(X, y) dy}{\int_{-\infty}^{\infty} f(X, y) dy} \tag{2}$$

where y = the output predicted by GRNN,

X = the input vector ($x_1, x_2, \ldots, x_n$) which consists of n predictor variables,

$E[y \mid X]$ = the expected value of the output y given an input vector X, and

$f(X, y)$ = the joint probability density function of X and y.

3.1 Topology of GRNN

The topology of GRNN developed by Specht [35] primarily consists of four layers of processing units (i.e., neurons). Each layer of processing units is assigned with a specific computational function when nonlinear regression is performed. The first layer of processing units, termed input neurons, are responsible for reception of information. There is a unique input neuron for each predictor variable in the input vector X. No processing of data is conducted at the input neurons. The input neurons then present the data to the second layer of processing units called pattern neurons. A pattern neuron is used to combine and process the data in a systematic fashion such that the relationship between the input and the proper response is "memorized." Hence, the number of pattern neurons is equal to the number of cases in the training set. A typical pattern neuron *i* obtains the data from the input neurons and computes an output 2_i using the transfer function of:

$$\theta_i = e^{\frac{-(X-U_i)'(X-U_i)}{2\sigma^2}} \qquad \textbf{(3)}$$

where X is the input vector of predictor variables to GRNN,

U_i is the specific training vector represented by pattern neuron *i*, and

F is the smoothing parameter.

Equation 3 is the multivariate Gaussian function extended by Cacoullos [36] and adopted by Specht in his GRNN design.

The outputs of the pattern neurons are then forwarded to the third layer of processing units, summation neurons, where the outputs from all pattern neurons are augmented. Technically, there are two types of summations, simple arithmetic summations and weighted summations, performed in the summation neurons. In GRNN topology, there are separate processing units to carry out the simple arithmetic summations and the weighted summations. Equations 4a and 4b express the mathematical operations performed by the 'simple' summation neuron and the 'weighted' summation neuron, respectively.

$$S_s = \sum_i \theta_i \qquad (4a)$$

$$S_w = \sum_i w_i \theta_i \qquad (4b)$$

The sums calculated by the summation neurons are subsequently sent to the fourth layer of processing unit, the output neuron. The output neuron then performs the following division to obtain the GRNN regression output *y*:

$$y = \frac{S_w}{S_s} \tag{5}$$

Figure 1 illustrates the schematic sketch of the GRNN design and its operational logic described above. It should be noted that the depicted network construct is valid for any model with multivariate outputs. The specific design for exchange rate forecasting in our study has only one (univariate) output neuron.

4. EXCHANGE RATE FORECASTING

4.1 Data

The sample data applied to this study are collected from the AREMOS data base maintained by the Department of Education of Taiwan. The entire data set covers 259 monthly periods running from January 1974 through July 1995. To serve different purposes, the data are divided into three sets: the training set (129 months from January 1974 through September 1984), the model selection/determination set (65 months from October 1984 through February 1990), and the out-of-sample testing set (65 months from March 1990 through July 1995). This breakdown uses one half of all data for model training and estimation and the remaining half for model selection and holdout testing. Once the selected forecasting models are appropriately specified, they are retrained using data from January 1974 through February 1990 to generate the out-of-sample forecasts for the hold out testing period.

Several transformations are applied to the original data. Following the postulations from previous work on MUIP models, we take the natural logarithm of all variables in the data set. Then lagged differences are computed for each period such that

$$\Delta x_t = x_t - x_{t-1} \tag{6}$$

where x_t is any variable (either predictor or independent) observed at period t. The exact specification for each forecasting model will be presented in the following section 4.3.

4.2 Training of GRNN and Model Selection

GRNN requires supervised training. The network performs learning by examining the relationship between each pair of input vector *X* and the observed corresponding output *y* and finally deduces the underlying function by summarizing all of these relationships encountered in the training set. Let p denote the number of lags (autoregressive terms) used in the forecasting model. For the training in this experiment, (129-p) cases from the training set, each corresponding to an individual training vector of inputs and its matching output, are sequentially presented to the network. The network then learns by creating a pattern neuron for each training case. This procedure is repeated until all cases in the training set are gone through. Figure 2 illustrates the GRNN construct after training with (129-p) cases in the training set.

Prediction of the exchange rate forecasts for the model selection/determination period begins after the network is trained. The 65 in-sample forecasts are then compared with the actual observations and the network parameters are adjusted. Afterward, the best three model specifications in terms of root mean square error (RMSE) are chosen. The network is then trained using the cases in both training and model selection sets. Estimation of the exchange rates in the holdout testing period is then performed and performance statistics for these 65 out-of-sample forecasts are computed. The entire experimental procedure is repeated for different currencies.

4.3 Forecasting Models

In addition to GRNN, a number of different specifications (models) based on MLFN (a.k.a backpropagation network), and multivariate transfer function are examined in our experiment. MLFN, a type of neural network architecture widely used in research studies, should provide a parallel comparison to GRNN in the area of non-parametric neural network forecasting. On the other hand, multivariate transfer function represents the parametric counterpart of the more conventional econometric forecasting. Interested readers should refer to Wasserman [37] for a detailed description of MLFN. Also, Markradis, Wheelwright, and McGee [38] provide a good discussion of the fundamentals of multivariate transfer function and a brief summary of this forecasting approach can be found in the Appendix.

In order to obtain a fair and more robust comparison of GRNN’s performance with the others, we pick up the best three specifications from each forecasting approach. Like the one for GRNN, the selection criterion is root mean square error (RMSE); however, an alternative criterion of mean absolute error (MAE) also leads to the same selection. In addition, a random walk model is included for benchmark comparisons. After the specifications are chosen and the parameters are determined, the out-of-sample forecasts from each of these forecasting specifications are generated and subject to statistical evaluations. Hence, our final statistical testings involve a total of 10 different forecasting models for each currency.

Let F(C) and G(C) denote the implied functional relationships estimated by GRNN and MLFN, respectively. Tables 1a, 1b, and 1c show the selected specifications for Canadian dollar, Japanese yen, and British pound forecasting, respectively. These selected specifications are chosen from the best three models from each of the GRNN, MLFN, and multivariate transfer function approaches. The Appendix contains a detailed description of the notation for representing the transfer function model.

5. PERFORMANCE EVALUATION

5.1 Primary Statistical Evaluation

Five primary statistical measures are computed from the out-of-sample forecasts made by each selected specification. As described in the previous section, each series of forecasts is checked against with the 65 observed cases in the holdout testing set. The five evaluation criteria are: mean absolute error (MAE), root mean square error (RMSE), U statistic, and the bias (*a*) and regression proportion coefficient (*b*) from the Theil's Decomposition Test [39]. The U statistic is the ratio of the RMSE of a model's forecasts to the RMSE of the random walk forecasts of no change in the dependent variable. Since the random walk forecast of next month's exchange rate is equal to the current month's rate, a U statistic of less than 1 implies that the tested model outperforms the random walk model during the holdout period. Likewise, a U statistic in excess of 1 implies that the model performs worse than the random walk model. The U statistic has a major advantage over the RMSE in the comparison of forecasting models because it is a unit-free measurement. Thus, it is easier to calibrate the relative performances by U statistic than by the unit-bound RMSE.

The Theil's Decomposition Test [39] is conducted by regressing the actual observed exchange rates on a constant and the predicted exchange rates $\hat{e}_t$ estimated by a particular model.

$$e_t = a + b\hat{e}_t \tag{7}$$

The constant *a* (bias coefficient) should be insignificantly different from zero and the coefficient *b* for predicted exchange rate variable (regression proportion coefficient) should be insignificantly different from one for the forecast to be acceptable.

Table 2 summarizes the performance statistics based on the 65 out-of-sample forecasts (from March 1990 through July 1995) generated by the GRNN, MLFN, multivariate transfer function and random walk models. Results in Table 2 illustrate that the GRNN models generally outperform the parametric multivariate transfer function and the random walk models. In addition, the GRNN models generates more accurate forecasts than MLFN, its neural network counterpart. For all currencies, GRNN models yield lower U statistics than the other forecasting models. On the other hand, it is also encouraging to find out both MLFN and transfer function models do improve the forecasting accuracy than the random walk model, which does not make use of additional information in forecasting.

It is interesting to observe that the forecasts improve by a reasonable degree when the same input information of a parametric model is provided to the GRNN. This improvement in the forecast can be viewed in several directions -- drop in forecast error, gain in explanatory power, and increase in correlation to the actual observations. For all currencies examined in this study, all parametric models based on multivariate transfer function and random walk pattern exhibit an insignificant correlation to the actual observations. This notion is supported by the significance of the regression proportion coefficients computed by the Theil's Decomposition tests. Although the forecasting bias for the two types of neural network models, GRNN and

MLFN, are both insignificant, improvements in forecast error (MAE and RMSE) as well as proportion coefficient is apparent when we move from the MLFN architecture to the GRNN paradigm. The superior performance of the GRNN models is also supported by a comparison between the transfer function model (ARIMA(1,0,0)-MUIP(1,0)) with the corresponding GRNN model (AR(1)-MUIP(1)). From Table 2, the U statistic and the results of Theil's Decomposition Test indicate that the parametric models perform closely to the random walk model. However, when the same set of historical information is applied to the GRNN models, not only the good performances in bias (*a*) and correlation to actual observation (*b*) are retained but also the forecast errors (MAE, RMSE, and U statistic) are reduced drastically. It is believed that the GRNN is capable of deducing the complex nonlinear relationships that most linear models cannot and that this advantage may contribute to the overall superior performance of the regression network.

5.2 Auxiliary Statistical Comparison

Both researchers and practitioners are interested not only in the forecasting accuracy of the various approaches but also in their performances relative to that of the GRNN. In other words, it is important to find out whether GRNN forecasting significantly outperforms the others. Given this notion, we perform a series of t tests to determine the significance of the differences of means with respect to the MAE and RMSE measures.

Table 3 reports the results of the conducted pairwise t tests. For Canadian dollar and Japanese yen, all comparisons are found to be significant at $\alpha = 0.05$ level, suggesting that GRNN forecasts are statistically better than the forecasts made by MLFN, multivariate transfer function, and random walk models. These results are also consistent with the findings in the previous section. However, the conclusions are different in the case of British pound forecasting. As indicated in Table 3, the GRNN forecasts yield lower forecast errors than the random walk model at " = 0.05 level. Nevertheless, the significance in the difference is reduced to " = 0.10 level when GRNN forecasts are compared with those estimated by the transfer function models. For the comparison with MLFN, the quality of the two neural network approaches are statistically indifferent, showing that the more conventional MLFN is still a valuable tool in forecasting research.

The disparity in the ranking of the comparison tests here suggests some interesting points.

First, although the forecasting accuracy in terms of MAE and RMSE are quite similar between MLFN and GRNN, a comparison of their U statistics, which is a relative improvement measure to the random walk model, indicates the existence of a gap in their performances. This belief is further supported by the (regression proportion coefficients) results of Theil’s Decomposition test in that the GRNN forecasts closely follow the actual exchange rate movements but the MLFN prediction has some difficulty to match with the reality. The implication to the researchers is that a more comprehensive framework of evaluation has to be developed and misleading conclusions may be drawn if the study relies upon only a couple of conventional statistics (e.g., RMSE, R-square).

Second, as we have expected, the experiment illustrates a varying degree of predictability of different currencies. For example, based on the values of RMSE and U statistics, we can observe that, to a certain extent, the movement of Canadian dollar is more predictable than the other two currencies. Japanese yen is the worst among the three examined currencies. This may be explained by the more volatile economy in Asia than in North America. A closer examination of the MAE and RMSE across different currencies substantiates the notion that the Japanese and British currencies have undergone wider fluctuations in their history. The closer performance of GRNN, MLFN, and multivariate transfer function in British pound suggests the possibility of developing a hybrid forecasting approach in these hard-to-predict environments. The hybrid framework may utilize different neural network architectures or even a mix of learning network and parametric technique. In this case, it is hoped that the simultaneous use of neural network and parametric technique may add predictive strength to the combined network.

6. CONCLUSIONS

This exploratory research examines the potential of using GRNN to predict the foreign currency exchange rates. Empirical results suggest that this nonparametric regression neural network may provide better forecasts than the other forecasting approaches tested in this study. This is illustrated by a comparison of the forecasting quality of the GRNN with that of the MLFN, a neural network architecture widely used in the literature, and multivariate transfer function, also another general parametric technique in econometric analyses. The comparative evaluation is based on a variety of statistics which measure different forms of forecast errors. For all currencies included in our empirical investigation, GRNN models outperform the models associated with MLFN, multivariate transfer function, and random walk. Moreover, our analysis finds that, except the case of British pound, the means of MAE and RMSE for the GRNN forecasts are significantly lower than those of other approaches. In the future, it would be interesting and beneficial to verify the predictive strength of GRNN on other financial assets, especially on the relative basis of MLFN.

Acknowledgments

The authors would like to thank M. Nimalendran and the participants at the 1999 Financial Management Association Meeting.

Appendix

Multivariate Transfer Function

A multivariate transfer function model is essentially an ARIMA with added exogenous variables. The general model can be stated as:

$$y_t = a_0 + a_1 y_{t-1} + ... + a_p y_{t-p} + \varepsilon_t + b_1 \varepsilon_{t-1} + ... + b_q \varepsilon_{t-q}$$
$$+ \frac{(\omega_{10} + \omega_{11} L + ... + \omega_{1n} L^n)}{(1 - \delta_{11} L - .... - \delta_{1m} L^m)} x_{1,t-b} + \frac{(\omega_{20} + \omega_{21} L + ... + \omega_{2r} L^r)}{(1 - \delta_{21} L - .... - \delta_{2s} L^s)} x_{2,t-b} + ... \tag{8}$$

where $x_{1,t-b}$, $x_{2,t-b}$, ... are exogenous variables lagged for b periods, L is the lag operator, y_t is the dependent variable, and ε_t is the residual. The addition of exogenous variables can improve forecasting if these variables help explaining the variation of the dependent variable. Interested readers should refer to Makridakis, Wheelwright, and McGee (1983) for a detailed explanation of the technique.

Notation

In this paper, we adopt the notation used in RATS (Regression Analysis of Time Series), a statistical package found in the computer systems of most academic institutes. The specification of the transfer function is composed of two parts, one for the univariate (ARIMA) terms and the other for the exogenous variables. Thus, a model is denoted by ARIMA(p,i,q)-MUIP(n/d), where p and q are the numbers of lags in the univariate ARIMA, i is the number of differencing, and MUIP(n/d) means MUIP with n numerator lags and d denominator lags. This corresponds to the format of Equation 8 stated above. Below are some representative examples using this notation.

ARIMA(1,0,1)-MUIP(0/0)

$$\Delta e_t = a_0 + a_1 y_{t-1} + \varepsilon_t + b_1 \varepsilon_{t-1}$$
$$+ (\omega_{10})\ \Delta(r^* - r)_{t-1}, + (\omega_{20})\ \Delta(\pi^* - \pi)_{t-1} + \ldots\ldots..$$

ARIMA(1,0,1)-MUIP(0/1)

$$\Delta e_t = a_0 + a_1 y_{t-1} + a_2 y_{t-2} + \varepsilon_t + b_1 \varepsilon_{t-1}$$
$$+ \frac{(\omega_{10})}{(1 - \delta_{11} L)}\ \Delta(r^* - r)_{t-1} + \frac{(\omega_{20})}{(1 - \delta_{21} L)}\ \Delta(\pi * - \pi)_{t-1} + \ldots\ldots..$$

ARIMA(1,0,1)-MUIP(1/0)

$$+ (\omega_{10} + \omega_{11} L)\ \Delta(r^* - r)_{t-1}, + (\omega_{20} + \omega_{21} L)\ \Delta(\pi^* - \pi)_{t-1} + \ldots\ldots..$$

$$\Delta e_t = a_0 + a_1 y_{t-1} + \varepsilon_t + b_1 \varepsilon_{t-1}$$

References

[1] Meese R, Rogoff K. Exchange rate models of the seventies: Do they fit out of sample? *Journal of International Economics* 1983;14:3-24.

[2] Meese R, Rogoff K. The out-of-sample failure of empirical exchange rate models: Sampling error or misspecification? In: Frenkel J, editor. *Exchange Rates and International Macroeconomics,* Chicago: University of Chicago Press, 1983.

[3] Meese R, Rogoff K. Was it real? The exchange rate-interest differential relation in 1973-1974. *International finance discussion paper no. 268*, Federal Reserve Board, Washington, DC. 1985.

[4] Alexander D, Thomas LR. Monetary/Asset models of exchange rate determination: How well have they performed in the 1980's? *International Journal of Forecasting* 1987;3:53-64.

[5] Gandolfo G, Padoan PC, Paladino G. Structural models versus random walk: The case of the lira/$ exchange rate. *Eastern Economic Journal* 1990;16:101-113.

[6] Gandolfo G, Padoan PC, Paladino G. Exchange rate determination: Single equation or economy-wide models? A test against the random walk. *Journal of Banking and Finance* 1990;14:965-992.

[7] Sarantis N, Stewart C. Monetary and asset market models for sterling exchange rates: A cointegration approach. *Journal of Economic Integration* 1995;10:335-371.

[8] Sarantis N, Stewart C. Structural, VAR and BVAR models of exchange rate determination: A comparison of their forecasting performance. *Journal of Forecasting* 1995;14:201-215.

[9] Schinasi GJ, Swamy PA. The out-of-sample forecasting performance of exchange rate models when coefficients are allowed to change. *Journal of International Money and Finance* 1989;8:373-390.

[10] Fisher PG, Tunna SK, Turner DS, Wallis KF, Whitley JD. Econometric evaluation of the exchange rate models of the U.K. economy. *Economic Journal* 1990;100:1230-1244.

[11] Rumelhart D, McClelland J. *Parallel Distributed Processing*, Cambridge, Mass.: MIT Press, 1986.

[12] Lippman R. An introduction to computing with neural nets. In: *IEEE ASSP Magazine,* April 1987: 4-22.

[13] Medsker L, Turban E, Trippi R. Neural network fundamentals for financial analysts. In: Trippi R, Turban E. editors. *Neural Networks in Finance and Investing*, Chicago: Probus Publishing, 1993.

[14] Hawley D, Johnson J, Raina D. Artificial neural systems: A new tool for financial decision-making. *Financial Analysts Journal* 1990:63-72.

[15] Tsukuda J, Baba S. Predicting Japanese corporate bankruptcy in terms of financial data using a neural network. *Computers and Industrial Engineering* 1994;*27*:445-448.

[16] Tam KY, Kiang MY. Managerial applications of neural networks: The case of bank failures. Management Science 1992;38:926-947.

[17] Salchenberger LM, Cinar EM, Lash NA. Neural networks: A new tool for predicting thrift failures. Decision Sciences 1992;23:899-916.

[18] Jensen H. Using neural networks for credit scoring. *Managerial Finance* 1992;18:15-26.

[19] Collins E, Ghosh S, Scofield C. An application of a multiple neural network learning system to emulation of mortgage underwriting judgments. *Proceedings of the IEEE International Conference on Neural Networks* 1988;2:459-466.

[20] Kryzanowski L, Galler M. Analysis of small-business financial statements using neural nets: Professional adaptation. *Journal of Accounting, Auditing, and Finance* 1995;10:147-172.

[21] Altman E, Marco G, Varetto F. Corporate distress diagnosis: Comparisons using linear discriminant analysis and neural networks. *Journal of Banking and Finance* 1994;18: 505-529.

[22] Dutta S, Shekhar S. Bond rating: A non-conservative application of neural networks. *Proceedings of the IEEE International Conference on Neural Networks* 1988;2:443-450.

[23] Jain B, Nag B. Artificial neural network models for pricing initial public offerings. *Decision Sciences* 1996;26:283-302.

[24] Refenes A. Constructive learning and its application to currency exchange rate forecasting. In: Trippi R, Turban E. editors. *Neural Networks in Finance and Investing,* Chicago: Probus Publishing 1993.

[25] Hutchinson J, Lo A, Poggio T. A nonparametric approach to pricing and hedging derivative securities via learning networks. *Journal of Finance* 1994;49:851-889.

[26] Kamijo K, Tanigawa T. Stock price pattern recognition: A recurrent neural network

approach. *Proceedings of the International Joint Conference on Neural Networks* 1992;1:1-6.

[27] Kimoto T, Asakawa K, Yoda M, Takeoka M. Stock market prediction system with modular neural networks. *Proceedings of the International Joint Conference on Neural Networks* 1992;1:215-221.

[28] Kaastra I, Boyd M. Forecasting futures trading volume using neural networks. *Journal of Futures Markets* 1995;15:953-970.

[29] Trippi R, DeSieno D. Trading equity index futures with a neural network. *Journal of Portfolio Management* 1992;19:27-34.

[30] Kendrick R, Acton D, Duncan A. Phase-diversity wave-front sensor for imaging systems. *Applied Optics* 1994;33:6533-6547.

[31] Mukesh D. Introduction to neural networks: Application of neural computing for process chemists, part 1. *Journal of Chemical Education* 1996;73:431-434.

[32] Williams T, Gucunski N. Neural networks for backcalculation of modula from SASW test. *Journal of Computing in Civil Engineering* 1995;9:1-9.

[33] Wehenkel L. Contingency severity assessment for voltage security using non-parametric regression techniques. *IEEE Transactions on Power Systems* 1996;11:101-111.

[34] Wittkemper H, Steiner M. Using neural networks to forecast the systematic risk of stocks. *European Journal of Operational Research* 1996;90:577-589.

[35] Specht D. A general regression neural network. *IEEE Transactions on Neural Networks* 1991;2:568-576.

[36] Cacoullos T. Estimation of a multivariate density. *Annals of the Institute of Statistical Mathematics* 1966;18:179-189.

[37] Wasserman PD. Advanced Methods in Neural Computing, New York:Van Nostrand Reinhold, 1993:97-118.

[38] Makridakis S, Wheelwright SC, McGee VE. *Forecasting: Methods and Applications,* 2nd Ed, New York: John Wiley and Sons, 1983.

[39] Theil H. Applied Economic Forecasting, Amsterdam: North Holland, 1966:26-36.

Table 1a **The Selected Model Specifications for Canadian Dollar Forecasting in the Holdout Testing Period**

The three model specifications yielding the best RMSE in the model selection/determination period are chosen. These models will be evaluated using the data in the holdout testing set (from March 1990 through July 1995). A brief interpretation of the transfer function modeling and its notation can be found in the Appendix and Markradis, Wheelwright, and McGee [38] provide a further discussion of this forecasting technique.

GRNN Specification 1: AR(1)

$$\Delta e_t = F(\ \Delta e_{t-1}\ )$$

GRNN Specification 2: AR(12)

$$\Delta e_t = F(\ \Delta e_{t-1},\ldots,\ \Delta e_{t-12}\ )$$

GRNN Specification 3: AR(1)-MUIP(1)

$$\Delta e_t = F(\ \Delta e_{t-1},\ \Delta(r^* - r)_{t-1},\ \Delta(\pi^* - \pi)_{t-1},\ \Delta(p^* - p)_{t-1},\ \Delta(ca/ny)_{t-1},\ \Delta(ca^*/ny^*)_{t-1}\ )$$

MLFN Specification 1: AR(4)

$$\Delta e_t = G(\ \Delta e_{t-1},\ldots,\ \Delta e_{t-4}\ )$$

MLFN Specification 2: AR(5)

$$\Delta e_t = G(\ \Delta e_{t-1},\ldots,\ \Delta e_{t-5}\ )$$

MLFN Specification 3: AR(1)-MUIP(1)

$$\Delta e_t = G(\ \Delta e_{t-1},\ \Delta(r^* - r)_{t-1},\ \Delta(\pi^* - \pi)_{t-1},\ \Delta(p^* - p)_{t-1},\ \Delta(ca/ny)_{t-1},\ \Delta(ca^*/ny^*)_{t-1}\ )$$

(Continued)

Transfer Function Specification 1: MUIP(0/1)

$$\Delta e_t \;=\; \frac{\omega_{10}}{(1-\delta_{11}L)}\;\Delta(r^*-r)_{t-1} \;+\; \frac{\omega_{20}}{(1-\delta_{21}L)}\;\Delta(\pi^*-\pi)_{t-1} \;+\; ....$$

Transfer Function Specification 2: ARIMA(1,0,0)-MUIP(1/1)

$$\Delta e_t \;=\; a_0 \;+\; a_1\Delta e_{t-1}$$

$$+\;\frac{(\omega_{10}+\omega_{11}L)}{(1-\delta_{11}L)}\;\Delta(r^*-r)_{t-1} \;+\; \frac{(\omega_{20}+\omega_{21}L)}{(1-\delta_{21}L)}\;\Delta(\pi^*-\pi)_{t-1} \;+\; ....$$

Transfer Function Specification 3: ARIMA(0,0,2)-MUIP(0/1)

$$\Delta e_t \;=\; \varepsilon_t \;+\; b_1\varepsilon_{t-1} \;+\; b_2\varepsilon_{t-2}$$

$$+\;\frac{\omega_{10}}{(1-\delta_{11}L)}\;\Delta(r^*-r)_{t-1}, \;+\; \frac{\omega_{20}}{(1-\delta_{21}L)}\;\Delta(\pi^*-\pi)_{t-1} \;+\; ....$$

Table 1b The Selected Model Specifications for Japanese Yen Forecasting in the Holdout Testing Period

The three model specifications yielding the best RMSE in the model selection/determination period are chosen. These models will be evaluated using the data in the holdout testing set (from March 1990 through July 1995). A brief interpretation of the transfer function modeling and its notation can be found in the Appendix and Markradis, Wheelwright, and McGee [38] provide a further discussion of this forecasting technique.

GRNN Specification 1: AR(1)

$$\Delta e_t = F(\Delta e_{t-1})$$

GRNN Specification 2: AR(2)

$$\Delta e_t = F(\Delta e_{t-1}, \Delta e_{t-2})$$

GRNN Specification 3: AR(12)

$$\Delta e_t = F(\Delta e_{t-1}, \ldots., \Delta e_{t-12})$$

MLFN Specification 1: AR(3)

$$\Delta e_t = G(\Delta e_{t-1}, \Delta e_{t-2}, \Delta e_{t-3})$$

MLFN Specification 2: AR(4)

$$\Delta e_t = G(\Delta e_{t-1}, \ldots., \Delta e_{t-4})$$

MLFN Specification 3: AR(1)-MUIP(1)

$$\Delta e_t = G(\Delta e_{t-1}, \Delta(r^* - r)_{t-1}, \Delta(\pi^* - \pi)_{t-1}, \Delta(p^* - p)_{t-1}, \Delta(ca/ny)_{t-1}, \Delta(ca^*/ny^*)_{t-1})$$

(Continued)

Transfer Function Specification 1: ARIMA(1,0,0)-MUIP(1/0)

$$\Delta e_t \;=\; a_0 \;+\; a_1 \Delta e_{t-1}$$
$$+\;(\omega_{10}+\omega_{11}L)\Delta(r^*-r)_{t-1} \;+\; (\omega_{20}+\omega_{21}L)\Delta(\pi^*-\pi)_{t-1} \;+\; \ldots\ldots..$$

Transfer Function Specification 2: ARIMA(0,0,2)-MUIP(0/1)

$$\Delta e_t \;=\; \varepsilon_t \;+\; b_1 \varepsilon_{t-1} \;+\; b_2 \varepsilon_{t-2}$$
$$+\;\frac{(\omega_{10})}{(1-\delta_{11}L)}\;\Delta(r^*-r)_{t-1},\;+\;\frac{(\omega_{20})}{(1-\delta_{21}L)}\;\Delta(\pi^*-\pi)_{t-1}\;+\;\ldots\ldots..$$

Transfer Function Specification 3: ARIMA(0,0,2)-MUIP(1/0)

$$\Delta e_t \;=\; \varepsilon_t \;+\; b_1 \varepsilon_{t-1} \;+\; b_2 \varepsilon_{t-2}$$
$$+\;(\omega_{10}+\omega_{11}L)\Delta(r^*-r)_{t-1} \;+\; (\omega_{20}+\omega_{21}L)\Delta(\pi^*-\pi)_{t-1} \;+\; \ldots\ldots..$$

Table 1c **The Selected Model Specifications for British Pound Forecasting in the Holdout Testing Period**

The three model specifications yielding the best RMSE in the model selection/determination period are chosen. These models will be evaluated using the data in the holdout testing set (from March 1990 through July 1995). A brief interpretation of the transfer function modeling and its notation can be found in the Appendix and Markradis, Wheelwright, and McGee [38] provide a further discussion of this forecasting technique.

GRNN Specification 1: AR(1)

$$\Delta e_t = F(\ \Delta e_{t-1}\ )$$

GRNN Specification 2: AR(2)

$$\Delta e_t = F(\ \Delta e_{t-1},\ \Delta e_{t-2}\ )$$

GRNN Specification 3: AR(3)-MUIP(1)

$$= F(\ \Delta e_{t-1},\ \Delta e_{t-2},\ \Delta e_{t-3},\ \Delta(r^* - r)_{t-1},\ \Delta(\pi^* - \pi)_{t-1},\ \Delta(p^* - p)_{t-1},\ \Delta(ca/ny)_{t-1},\ \Delta(ca^*/ny^*)_t$$

MLFN Specification 1: AR(2)

$$\Delta e_t = G(\ \Delta e_{t-1},\ \Delta e_{t-2}\ )$$

MLFN Specification 2: AR(3)

$$\Delta e_t = G(\ \Delta e_{t-1},\ \Delta e_{t-2},\ \Delta e_{t-3}\ )$$

MLFN Specification 3: AR(1)-MUIP(1)

$$\Delta e_t = G(\ \Delta e_{t-1},\ \Delta(r^* - r)_{t-1},\ \Delta(\pi^* - \pi)_{t-1},\ \Delta(p^* - p)_{t-1},\ \Delta(ca/ny)_{t-1},\ \Delta(ca^*/ny^*)_{t-1}\ )$$

(Continued)

Transfer Function Specification 1: ARIMA(1,0,2)-MUIP(0/1)

$$\Delta e_t = a_0 + a_1 y_{t-1} + \varepsilon_t + b_1 \varepsilon_{t-1} + b_2 \varepsilon_{t-2}$$

$$+ \frac{(\omega_{10})}{(1-\delta_{11}L)} \Delta(r^* - r)_{t-1}, + \frac{(\omega_{20})}{(1-\delta_{21}L)} \Delta(\pi^* - \pi)_{t-1} + \text{........}$$

Transfer Function Specification 2: ARIMA(2,0,1)-MUIP(0/1)

$$\Delta e_t = a_0 + a_1 y_{t-1} + a_2 y_{t-2} + \varepsilon_t + b_1 \varepsilon_{t-1}$$

$$+ \frac{(\omega_{10})}{(1-\delta_{11}L)} \Delta(r^* - r)_{t-1} + \frac{(\omega_{20})}{(1-\delta_{21}L)} \Delta(\pi^* - \pi)_{t-1} + \text{........}$$

Transfer Function Specification 3: ARIMA(1,0,1)-MUIP(0/1)

$$\Delta e_t = a_0 + a_1 y_{t-1} + \varepsilon_t + b_1 \varepsilon_{t-1}$$

$$+ \frac{(\omega_{10})}{(1-\delta_{11}L)} \Delta(r^* - r)_{t-1}, + \frac{(\omega_{20})}{(1-\delta_{21}L)} \Delta(\pi^* - \pi)_{t-1} + \text{........}$$

Table 2 Performance Statistics for the Out-of-Sample Forecasts in the Holdout Testing Period from March 1990 to July 1995 (65 observations)

The 65 out-of-sample forecasts are estimated by the best three specifications of the GRNN, MLFN, and multivariate transfer function models. Performance statistics for the random walk model are also computed for benchmark comparison. The performance of each forecasting model are measured by five criteria -- mean absolute error (MAE), root mean square error (RMSE), U statistic, and the bias coefficient (a) and regression proportion coefficient (b) of Theil's Decomposition Test. The bias coefficient (a) and the regression proportion coefficient (b) represent the constant and the coefficient, respectively, from regressing the actual exchange rate on a constant and the predicted value. The respective t statistic is in parenthesis. * indicates significance at 5% level for H_0: a=0 and H_0: b=1.

Canadian Dollar					
Specification	**MAE**	**RMSE**	**U Statistic**	**Bias Coefficient (a)**	**Proportion Coefficient (b)**
GRNN AR(1)	0.00866	0.01065	0.77468	0.00067 (0.4260)	0.77729 (-0.4482)
GRNN AR(12)	0.00870	0.01056	0.76807	-0.00203 (-0.9218)	1.3264 (1.2873)
GRNN AR(1)-MUIP(1)	0.00868	0.01073	0.78054	-0.00062 (-0.3311)	1.14151 (1.3200)
MLFN AR(4)	0.00930	0.01152	0.83779	0.001562 (1.1295)	0.29306* (-3.1575)
MLFN AR(5)	0.00984	0.01178	0.85646	0.001689 (1.2172)	0.21559* (-3.5058)
MLFN AR(1)-MUIP(1)	0.00944	0.01149	0.83582	0.001419 (1.0143)	0.30875* (-3.1834)
Transfer Fcn ARIMA(1,0,0)-MUIP(1/1)	0.00974	0.01199	0.87215	0.00095 (0.7350)	0.39291* (-5.3441)
Transfer Fcn ARIMA(0,0,2)-MUIP(0/1)	0.01073	0.01332	0.96869	0.00177 (1.3137)	0.20367* (-5.9292)
Transfer Fcn MUIP(0/1)	0.01046	0.01316	0.95717	0.001968 (1.4469)	0.10925* (-5.3490)
Random Walk	0.01068	0.01375		0.00147 (1.0750)	0.20147* (-6.5332)

(Continued on the next page)

Table 2 (Continued)

Japanese Yen					
Specification	**MAE**	**RMSE**	**U Statistic**	**Bias Coefficient (a)**	**Proportion Coefficient (b)**
GRNN AR(1)	0.02119	0.02716	0.79915	-0.00253 (-0.5682)	1.15638 (0.2514)
GRNN AR(2)	0.02091	0.02687	0.79060	-0.00115 (-0.2599)	1.45347 (0.7263)
GRNN AR(12)	0.02152	0.02755	0.81067	-0.00092 (-0.0954)	0.88056 (-0.1053)
MLFN AR(3)	0.02503	0.03159	0.92952	-0.00736* (-2.0735)	0.13842* (-4.2633)
MLFN AR(4)	0.02480	0.03129	0.92063	-0.00596 (-1.6240)	0.24263* (-4.4870)
MLFN AR(1)-MUIP(1)	0.02585	0.03264	0.96043	-0.00728* (-2.0811)	0.17606* (-4.9401)
Transfer Fcn ARIMA(1,0,0)-MUIP(1/0)	0.02305	0.02925	0.86076	-0.00651 (-1.4983)	0.15776* (-2.7612)
Transfer Fcn ARIMA(0,0,2)-MUIP(0/1)	0.02476	0.03071	0.90366	-0.00684 (-1.8520)	0.17317* (-3.8799)
Transfer Fcn ARIMA(0,0,2)-MUIP(1/0)	0.02488	0.03093	0.91024	-0.00736 (-1.8033)	0.05686* (-3.9585)
Random Walk	0.02667	0.03398		-0.00588 (-1.6584)	0.23853* (-6.1236)

The respective t statistic is in parenthesis. * indicates significance at 5% level for H_0: a=0 and H_0: b=1.

(Continued on the next page)

Table 2 (Continued)

British Pound					
Specification	**MAE**	**RMSE**	**U Statistic**	**Bias Coefficient (a)**	**Proportion Coefficient (b)**
GRNN AR(1)	0.02212	0.02865	0.83143	-0.00023 (-0.0645)	1.37134 (0.8022)
GRNN AR(2)	0.02202	0.02855	0.82852	-0.00014 (-0.0398)	1.38586 (0.8526)
GRNN AR(3)-MUIP(1)	0.02161	0.02995	0.86893	0.00105 (0.2788)	1.73073 (1.6130)
MLFN AR(2)	0.02309	0.02996	0.86932	0.00092 (0.2474)	0.59779 (-1.3206)
MLFN AR(3)	0.02468	0.03119	0.90504	0.00167 (0.4422)	0.41215* (-2.3484)
MLFN AR(1)-MUIP(1)	0.02216	0.02990	0.86757	0.00437 (1.1354)	0.65691 (-1.4107)
Transfer Fcn ARIMA(1,0,2)-MUIP(0/1)	0.02478	0.03200	0.92855	-0.00024 (-0.0659)	0.40577* (-3.6312)
Transfer Fcn ARIMA(2,0,1)-MUIP(0/1)	0.02499	0.03174	0.92094	-0.00147 (-0.4137)	0.45679* (-4.1636)
Transfer Fcn ARIMA(1,0,1)-MUIP(0/1)	0.02305	0.03141	0.91136	-0.00044 (-0.1187)	0.43981* (-3.3089)
Random Walk	0.02706	0.03446		0.00075 (0.21139)	0.36029* (-5.4774)

The respective t statistic is in parenthesis. * indicates significance at 5% level for H_0: a=0 and H_0: b=1.

Table 3 Pairwise t Tests for the Difference in Forecasting Accuracy Between GRNN and Other Forecasting Approaches

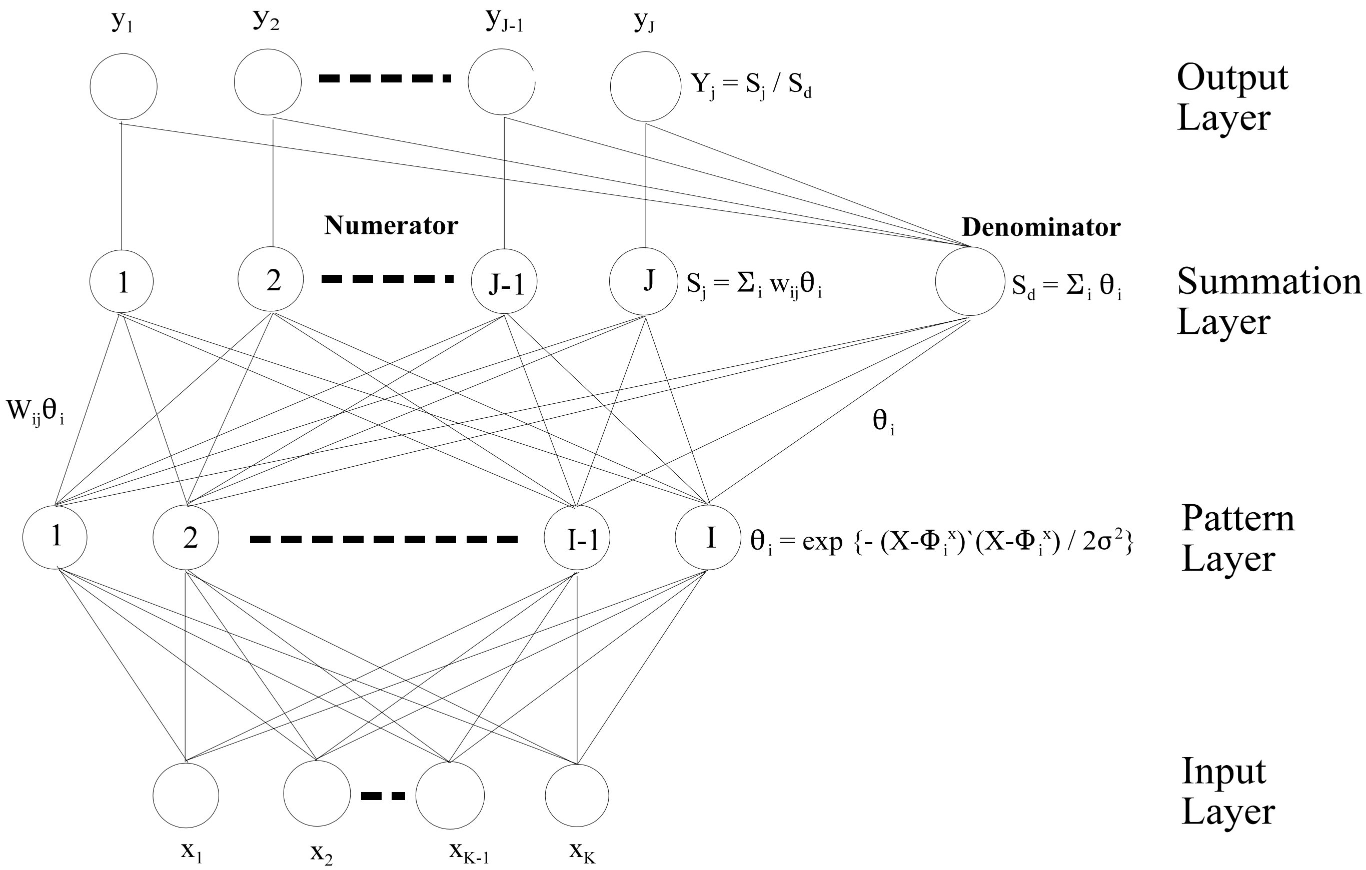
The t statistics are based on comparisons of the MAE and RMSE of the GRNN forecasts in the holdout testing period (from March 1990 to July 1995) with those of the MLFN, multivariate transfer function, and random walk models. The best three specifications from each forecasting approach provide a total of 3x65=195 matching observations for each t test. The null hypothesis is no difference in the forecasting accuracy between GRNN and the tested approach.
* and ** indicate significance at the 5% and 10% levels, respectively.

Canadian Dollar				
	MAE		**RMSE**	
	t Statistic	**p Value**	**t Statistic**	**p Value**
MLFN	2.3543*	0.0195	2.3178*	0.0215
Transfer Function	2.8192*	0.0053	3.0259*	0.0028
Random Walk	3.7160*	0.0003	5.0664*	7.87×10^{-7}

Japanese Yen				
	MAE		**RMSE**	
	t Statistic	**p Value**	**t Statistic**	**p Value**
MLFN	3.8208*	0.0002	4.0384*	7.77×10^{-5}
Transfer Function	3.7393*	0.0002	3.3474*	0.0010
Random Walk	3.6447*	0.0003	3.5303*	0.0005

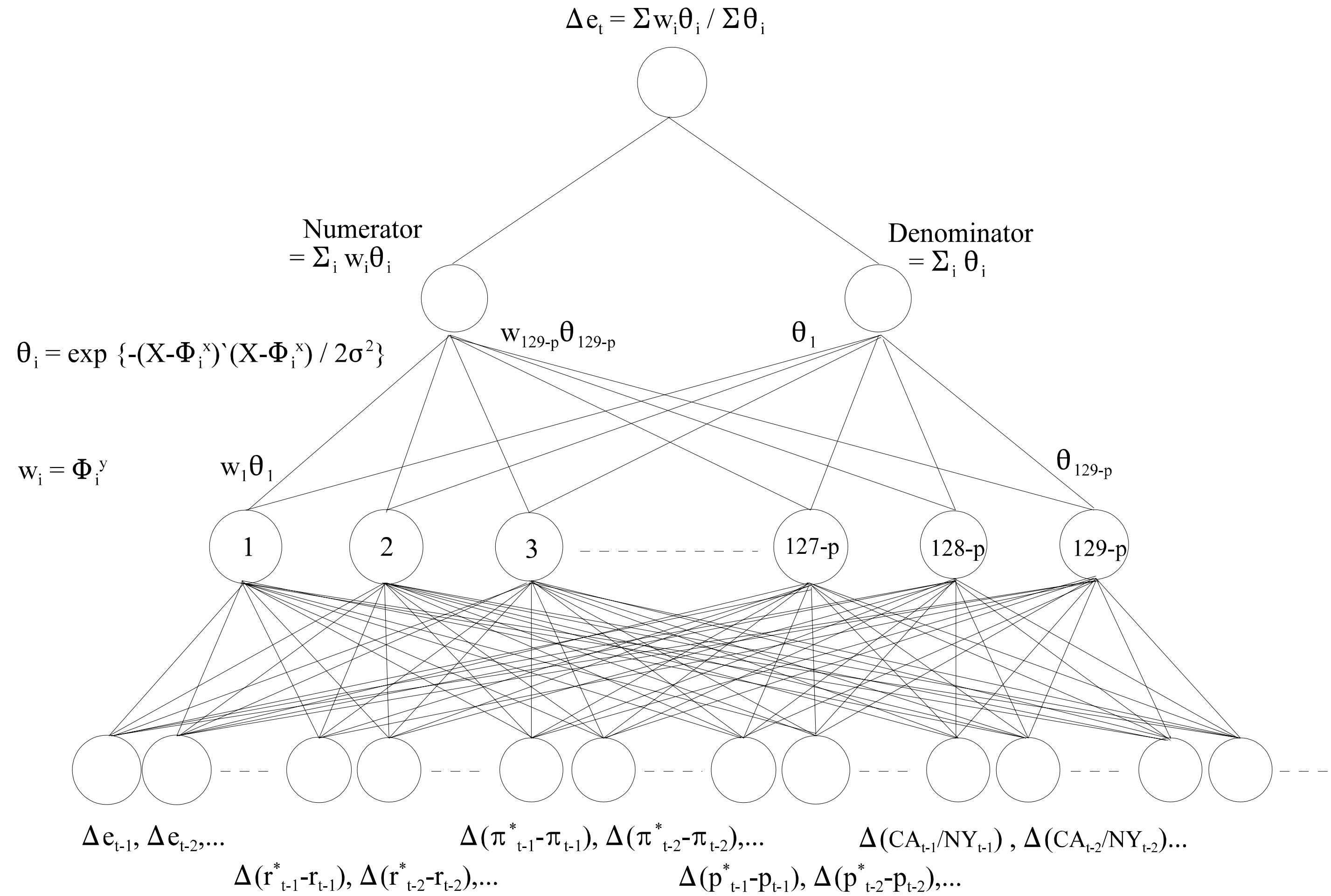
British Pound				
	MAE		**RMSE**	
	t Statistic	**p Value**	**t Statistic**	**p Value**
MLFN	1.6192	0.1071	1.4350	0.1529
Transfer Function	1.7231**	0.0865	1.7344**	0.0845
Random Walk	3.6214*	0.0004	2.6995*	0.0076

Figure 1 General GRNN Construct



A typical GRNN construct consists of four layers -of processing units - input, pattern, summation, and output neurons Input layer receives the input vector X and distributes the data to the pattern layer. Each neuron in the pattern layer then generates an outputθ and presents the result to the summation layer. The numerator and denominator neurons in the subsequent summation layer compute the weighted and simple arithmetic sums based on the values of θ and w_{ij} learned in the supervised training. The actual mathematical operations performed by these neurons in the hidden layers are illustrated in the diagram. The neurons in the output layer then carries out the division of the sums computed by the neurons in the summation layer.

Figure 2 GRNN Construct for Exchange Rate Forecasting



The design of the illustrated GRNN construct is based on the Modified Uncovered Interest Parity (MUIP) theorem . The depicted construct shows tn input vector with six groups of input variables. Each group of variables contains a finite number of lagged terms in which the total number of lags differs from group to group and from one currency to another. There are (129-p) neurons in the pattern layer, representing the training cases in the training data set. The value of p is equal to the number of lagged terms in the model specification. After the supervised training is completed, the regression network will compute the exchange rate forecast of period t based on the values of the predictor variables.