A Weighted Window Approach to Neural Network Time Series Forecasting

Bradley H. Morantz, Thomas Whalen, G. Peter Zhang

Department of Management

Robinson College of Business

Georgia State University

Atlanta, GA 30303

March 13, 2003

A Weighted Window Approach to Neural Network Time Series Forecasting

Abstract

In this paper, we propose a neural network based weighted window approach to time series forecasting. We compare the weighted window approach with two commonly used methods of rolling and moving windows in modeling and evaluating time series. Seven economic data sets are used to compare the performance of these three data windowing methods on observed forecast errors. We find that the proposed approach can be an effective way to improve forecasting performance.

INTRODUCTION

Forecasting is an important part of decision making. The success of decision making is dependent, to a large extent, on the quality of forecasts. There are two major approaches to quantitative forecasting. Causal methods model the relationship between a set of predictor variables and the value of the criterion variable to be forecast. Time series methods, on the other hand, use past or lagged values of the same criterion variable as a surrogate for all the underlying causal factors and model the relationship between these past observations and the future value. Each approach has its advantages and disadvantages as discussed in Bowerman and O'Connell (1993). This paper will focus on the time series forecasting methods.

A time series is a chronological sequence of observations taken at periodic points in time. Many models exist for the task of time series forecasting. Some of these models include

- Naïve or random walk – use the most recent observation as forecast.
- Moving average – use the average of a fixed number of past data points as forecast.
- Exponential smoothing – a weighted moving average approach that weights more heavily the recent values than the past values.
- Autoregression (AR) – a linear regression technique that estimates the relationship among observations in a time series.
- Auto regressive integrated moving average (ARIMA) – a versatile linear system that models the relationship among time series observations and random shocks (Box and Jenkins, 1976).
- Bilinear – a simple nonlinear model (Granger and Anderson, 1978)
- Threshold autoregressive – a specialized nonlinear AR model (Tong and Lim, 1980).
- Autoregressive conditional heteroscedastic (ARCH) – a parametric nonlinear model for nonconstant conditional variance (Engle, 1982)

- Artificial neural networks (ANNs) – adaptive models based upon biological neural systems capable of representing nonlinear relationships.

These models represent some of the most popularly used linear and nonlinear approaches to practical time series forecasting. The first five models are linear while the rest ones are nonlinear. It is important to note that most of these models are parametric in nature. That is, the model form is assumed before the model is built from data. Among other things, the assumed model form specifies how many lagged observations of the variables are used as inputs to the function that forecasts future values. Therefore, a good knowledge of both data set and the underlying data generating process is necessary for successful forecasting applications of these models.

Artificial neural networks (ANNs) are a relatively new approach that has received an increasing amount of attention among researchers and business practitioners in time series forecasting. ANNs represent a mathematical attempt to emulate part of the function of a biological neural network or brain. With its relatively large number of parameters utilized in building a model, it is capable of representing an input-output relationship of a non-linear system and it is fairly immune to noise (Haykin, 1994). The nonparametric and data driving properties of ANNs have made them valuable for general forecasting applications. Much effort has been devoted to neural networks to improve the forecasting performance. Zhang et al. (1998) provide a survey of some advances in the filed.

One of the major issues in neural network forecasting is how much data are necessary for neural networks to capture the dynamic nature of the underlying process in a time series. There are two facets to this issue:

1) how many lagged observations are should be used as inputs to the neural network (or, equivalently, how many input nodes the neural network should have)?

2) how many past observation to use in training the neural network. Each training pattern is a tuple consisting of the actual historical value plus a number of preceding values equal to the number of input nodes.

Although larger sample size, in the form of a longer time series, is usually recommended in model development, empirical results suggest that longer time series do not always yield models that provide the best forecasting performance. For example, Walczak (2001) investigates the issue of data requirement for neural networks in financial time series forecasting. He finds that using smaller sample of time series or data close in time to the out-of-sample can produce more accurate neural networks.

Determining an appropriate sample size for model building is not necessarily an easy task, especially in time series modeling when a larger sample inevitably means using older data. Theoretically speaking, if the underlying data generating process for a time series is stationary, more data should be helpful to reduce the effect of noise inherent in the data. However, if the process is not stationary or changing in structure or parameters over time, longer time series do not help and in fact can hurt the model's forecasting performance. In this situation, more recent observations should be more important in indicating the possible structural or parameter change in the data while older observations are not useful and even harmful to forecasting model building.

In this paper, we propose a weighted window approach to identify an appropriate size of time series for ANN model building. In addition, we believe that even with an appropriate training sample, each observation in the time series does not necessarily play an equal role in modeling the underlying time series structure and predicting the future. More specifically, we believe that more recent observations should be more important than older observations and therefore should receive higher weights. The idea is superficially similar to that in a weighted

moving average method where past lagged observations are weighted according to their relative position in the time series history. However, the concept of a weighted moving average concerns the actual function used to make a specific forecast, while the concept of weighted window concerns the process by which historical data are used to learn the parameters of the model.

The rest of the paper is organized as follows. In the next section, we provide a description of two traditional approaches of rolling and moving windows to training sample size selection in building neural networks. Then, the detail of the proposed weighted window approach is given. The research methodology along with the empirical evidence on seven economic time series is discussed next. Finally, we give some concluding remarks.

ROLLING AND MOVING WINDOW APPROACHES

In neural network time series forecasting, almost all studies adopt a fixed window approach in building and evaluating neural network performance. That is, all available data are divided into a training sample, a validation sample, and a test sample. The model is estimated and developed using the first two samples and then the established model is evaluated with the last sample. The division of the data is usually quite arbitrary and the rule of thumb is to assign more data to the training sample and relatively less data to other samples. Once the data splitting is done, the size of each portion of the data is fixed in terms of model building, selection, and evaluation.

However, the above approach is effective only if the data characteristics in each portion are about the same or the underlying data generating process as well as the parameters characterizing the process do not change from sample to sample. If the data are not stationary,

i.e. changing in characteristics over time, then the static model built from a fixed section of historical data will produce poorer forecasts over time. Even if the time series under study is relatively stationary, there still is a disadvantage of using the static model to evaluate recent or future forecasts because of the possible changes in noise level or model parameters. Therefore, as time goes by, it is often necessary and beneficial to include new observations to update model or model parameters.

There are essentially two approaches in updating forecasting models over time: the rolling and moving window approaches. The rolling window approach uses all available data to train neural networks while the moving window approach uses a set of most recent observations to estimate model. Each time a new observation is received, the rolling window approach adds one training example to its database, consisting of the new observation as the new criterion variable and the next most recent k observations (where k is the number of input nodes) as predictor variables. The moving window approach differs in that as it adds the new training example, it also drops the oldest observations from the training sample to update the model. Thus, with rolling window, the sample size increases over time, while with moving window, the sample size is fixed. Figure 1(a) (b) shows the ideas in rolling and moving approaches.

Figure 1 is about here

The rolling window has a constant starting point in training neural networks. The main advantage of using this approach is that with increasing sample size, model parameters can be estimated more accurately because larger sample sizes will usually have smaller error variance. However, as indicated earlier, the disadvantage of this approach is that it will not work well when the time series process is changing over time. On the other hand, the moving

window approach has changing starting point in the training sample and is better positioned to reflect changes occurred in the underlying process of a time series. The disadvantage of the moving window approach is that an appropriate sample size is not easy to determine and a fixed sample size is not necessarily effective for all subsequent or future model updating and forecasting. In addition, relatively smaller sample size with moving window may also limit the power to accurately estimate the model parameters.

The rolling and moving window approaches have been evaluated by Hu et al. (1999) in the context of foreign exchange rate forecasting. Their purpose was to evaluate the robustness of neural networks with respect to sampling variation. They found that neural networks are not very sensitive to the sampling variation and both rolling and moving approaches perform about the same in out of sample forecasting. Because of the relatively stable condition for the specific exchange rate time series they studied as well as the efficient market theory dominated for exchange rate movement, these results are not unexpected. In addition, their window size incremental was 12 months. That is, the model is updated not after every time period, but rather after observing 12 periods of data.

WEIGHTED WINDOW APPROACH

In this study, we propose a weighed window approach that is similar to the moving window idea discussed above. However, unlike in moving window approach where past observations or prediction errors are treated equally, we consider a weighted objective function to train neural networks. That is, forecast errors are weighted differently with most recent errors receiving higher weight and older errors carrying less weight.

The parameters of the model are selected to minimize the weighted sum of squared deviations between the computed and actual forecast variable for the training examples in the

window. For training examples in the most recent part of the window, the "core," the squared error is weighted 100%. For the remainder of the window, the squared error is weighted by a linear function that is equal to 1 for the oldest training example in the core and zero for the most recent training example that is not in the window at al. In effect, the weights define a fuzzy set of recent data whose membership function follows a trapezoidal form whose core consists of the "completely recent" data and whose support consists of the entire window.

Different weighting schemes can be used to define weights for observations in a window of time series. A simple but natural choice is the linear weighting approach with which the size of weight decreases in a linear fashion from the most recent observation to the oldest observation in the window. Figure 2 shows a general linear weighting scheme that is defined by a support set and a core set. Outside the core, all observations have weights of less than 1 and within the core, all observations receive the full weights of 1. The core is defined as the most recent time interval and thus any older data will be discounted. On the other hand, support is the interval beyond which older data points will have zero weights. In other words, all observations that contribute to model building are at least as recent as the starting point of support. The importance of each data points in the support but outside the core attenuates in a linear descending pattern as shown in Figure 2. More specifically, we define weight for time period t as

$$W_t = \begin{cases} 1 & \text{if } t \geq c \\ (t-s)/(c-s) & \text{if } s < t < c \\ 0 & \text{if } t < s \end{cases}$$

where s is the starting point of support and c is the starting point of core.

Figure 2 is about here

The weighted window approach does not modify the input data. Rather, our approach is to modify the objective function in training neural networks. Suppose we have N observations $y_1, y_2, \ldots, y_N$ in the training sample. Then using a network with k input nodes and one output node, we have $N - k$ training patterns. The first training pattern is composed of $y_1, y_2, \ldots, y_k$ as the inputs and y_k+1 as the target. The second training pattern contains $y_2, y_3, \ldots, y_{k+1}$ for the inputs and y_{k+2} for the target. Finally, the last training pattern is $y_{N-k}, y_{N-k+1}, \ldots, y_{N-1}$ for the inputs and y_N for the target. The training objective is to find the arc weights such that a weighted overall error measure such as the weighted sum of squared errors (WSSE) is minimized. For this network structure, WSSE can be generally written as:

$$WSSE = \sum_{i=k+1}^{N} W_i (y_i - \hat{y}_i)^2$$

where $\hat{y}_i$ is the ANN forecast value for ith observation.

METHODOLOGY

Data

Seven financial data sets were obtained from Economagic (http://www.economagic.com). All data sets were terminated at February 1, 2001 and in the case of daily data, we convert them to monthly by using the first recorded value in each month, to maintain consistency and to allow the same point of prediction for all forecasts. These time series are

(1) New one family houses sold: 458 monthly observations starting from January 1963.

(2) Federal funds rate: 560 monthly observations starting from July 1954.

(3) One month CD rates (secondary market): 423 monthly observations starting from December 1965.

(4) One month Eurodollar deposits: 362 monthly observations starting from January 1971.

(5) US Treasury bill: 439 monthly observations starting September from 1963.

(6) French Franc to US Dollar exchange rate: 364 monthly observations starting from January 1, 1971.

(7) German Mark to US Dollar exchange rate: 364 monthly observations starting from January 1971.

Figure 3 (a)-(g) plots all the time series, which shows various different patterns in these time series. For model building and evaluation purposes, each time series is split first by holding out the last 31 data points for final testing. The remaining data is then split by an 80/20 method. That is, the 80% of the earliest observations (training set) is used to estimate the parameters of neural network models and the remaining 20%, the validation set, is used to test these models in order to select the best model. The precise sizes of the training and validation sets are found by integer truncation.

Figure 3 is about here

Neural Network Model Building

The standard three layer feedforward neural network structure is exclusively used in this study. One output node is employed for one-step ahead forecasting. We use linear activation function for the output node and the sigmoid or logistic activation function for hidden nodes. Biases are used for both hidden and output nodes. The numbers of input and hidden nodes are usually not possible to determine in advance and therefore are selected via experimentation with the training and validation sample. This is the commonly used cross-

validation approach. That is, parameters are estimated for each of candidate architectures using the training sample and then the final architecture is selected based on the validation sample.

For time series forecasting problems, the number of input nodes corresponds to the number of lagged observations used to determine the autocorrelation structure of a time series. This number was varied from 2 to 9 in the experiment. On the other hand, we vary number of hidden nodes from 1 to 10. For each ANN architecture experimented, the model parameters are estimated with the training sample while the best model is selected with the validation sample.

For moving and weighed window approaches, the architecture specification also includes the amount of observations used for training, which is also difficult to specify exactly in advance and is generally data dependent. To find the best size of the window, we vary it from 50 to the highest integer multiple of 50 that was possible within the training set. The choice of 50 is somewhat arbitrary but follows general recommendation in the time series forecasting literature that at least 50 observations are needed in order to build a successful forecasting model (Box and Jenkins, 1976).

For each moving window model, the amount of data in the support set that receives partial weights is also determined. In this study, we consider three levels of the size: 0, 50, and 100. The zero value means that all data points receive the same weight of one, which reflects the fact that a moving window is a special case of a weighted window. Otherwise, the oldest 50 or 100 observations in the support set are given fractional weights in proportion to their age. It is likely that some other models that incorporate WW might provide greater accuracy with different amounts of weighted data other than 50 or 100, but for parsimonious reasons this study was limited to these two values.

Error calculation for a given architecture was accomplished by doing a one-month-ahead forecast for each observation in the validation set. For an ANN model with k inputs, the last k months of the training set are input to the model to make a prediction for the first month of the validation sample. The prediction is compared to the actual, and the error was computed. Then the last k training months and the first validation month are input to the same model (without re-estimating the parameters) to make a prediction for the second month of the validation sample and calculate its error, and so on. This is done iteratively to evaluate the model for the entire validation period as well as the testing period.

In the model building phase, our goal is to find the most robust model that predicts well for the validation sample. Therefore, the best model is the one that gives the most accurate performance in the validation sample. The selected model is then used to examine the true out-of-sample forecasting performance in the testing sample that is not used in the model building phase.

Although there are many performance measures that can be use to judge the forecasting performance of neural networks, this study elects to choose the mean absolute percentage error (MAPE) as the overall performance measure because of its robustness and usefulness for model comparison across different time series.

Neural network training is the process of assimilating historical data and learning the input to output relationship. The process entails nonlinear optimization to estimate the model parameters. In this study, we use Generalized Reduced Gradient (GRG2) (Lasdon et al., 1978) to train the neural network models. GRG2 uses a generalized reduced gradient algorithm for general nonlinear optimization. As such, it is capable of solving a system of nonlinear optimization problems. Since a global optimum is not guaranteed in a nonlinear

optimization training of a neural network, each network is trained 50 times with different initial random weights.

RESULTS

We focus on out-of-sample forecasting evaluation with the three different methods. In this final testing phase, we have three sets of 31 error values for each data set, corresponding to three methods: rolling, moving and weighted window (WW). For direct comparison, we use absolute percentage error (APE) as measure of forecasting error for each period in the hold-out sample and the mean absolute percentage error (MAPE) as the summary performance measure across all periods in the sample.

Table 1 gives the summary results of MAPE for all seven data sets across three different approaches. It shows that the neural network model based on the weighted window approach is the most accurate predictor for five of the seven data sets. These five data sets are one month CD rate, Eurodollar deposit, federal fund rate, French Franc exchange rate and new one family houses sold. In some of these cases, the improvement of using WW approach over the rolling and moving approaches is substantial. For example, in the one-month CD rate forecasting case, the WW approach has more than 50% reduction in the overall error measured by MAPE over the rolling and moving approach. In forecasting new house sales, the WW approach reduces MAPE from rolling and moving approaches by approximately 45%. However, the WW approach is not as effective as rolling and moving approaches in two of the seven data sets: German Mark exchange rate and T-bill, although the difference between WW and rolling or moving is not considerable.

Table 1 is about here

Table 1 shows the comparative results based on overall measure of MAPE. In order to examine differences among the three approaches in detail, we perform separate ANOVA analysis for each data set. A complete blocking design with the blocking factor of time period in the test set is used for the analysis to highlight the difference in modeling and evaluation approaches. See Neter et al. (1996) for details on block design. The 31 time periods are served as 31 levels of the blocking factor to better isolate the specific effects due to different approaches in neural network model updating. Table 2 summarizes the ANOVA results for all seven data sets. In the ANOVA tables, "Time" is the blocking factor while "Method" is the main factor to be investigated.

Table 2 is about here

From Table 2, we find that the blocking effect is significant for all data sets, suggesting the usefulness of blocks in providing additional detection power of the model for the factor of main concern. In addition, the main factor "Method" is highly significant for three data sets. They are CD rate, federal fund, and house sale. For data sets of Euro Dollar deposit and France Franc exchange rate, there is no significant difference between the three approaches although the weighted window approach outperforms the other two approaches judged by the average performance measure. Similarly, while the WW approach is worse than both rolling and moving approaches from the overall measure of MAPE, the difference is not significant.

To further identify the significant differences among the three approaches, we employ the Tukey's honestly significant test (HST) for three significant ANOVA data sets. At the 5% significance level, we find that in all three cases, there are significant differences existing between WW and moving and between WW and rolling. In addition, there is no significant

difference between the rolling and the moving approach, which is in line with the finding in Hu et al. (1999). Therefore, one may conclude that by employing weighting scheme that emphasizes recent data can be very helpful for some economic time series.

SUMMARY AND CONCLUSION

In this paper, we propose to use a neural network based weighted window approach in modeling and evaluating neural networks for time series forecasting. We believe that some time series in business especially in economics and finance may exhibit changing dynamics in the underlying data generating process. Therefore, the one-model-fit-all approach to building and evaluating the performance of a forecasting model is not possibly the best for this situation. By weighting the past observations differently and more specifically by having higher weights tied with more recent forecast errors, our weighted modeling approach is able to capture the dynamic nature of the process more quickly and therefore provide better forecasts for the near future.

Based on our empirical investigation, we find that the WW approach is fairly effective for economic time series forecasting. Of the seven data sets examined, we find that the WW approach is better than the traditional rolling or moving approach in five data sets from the perspective of overall prediction accuracy and it is significantly better than the rolling and moving approaches in three data sets. Although in two data sets, the WW approach does not provide superior forecasts over rolling and moving judging from the overall MAPE measure, the differences are not significant.

Future research should address the question: under what condition the proposed weighted window approach is more effective? Theoretically, if the structure or parameters of the underlying process governing the time series changes over time, the weighted window

approach should be more useful. However, it may be practically difficult to judge whether such situations have occurred. Practical approaches to identifying this changing environment are needed. Furthermore, in this study, we only consider a very simple way to assign weights, i.e., the linear weighting scheme. Better ways to assign weights should be developed that tailor different time series behaviors. Finally, the WW approach examined in this study may update the model too frequently with each and every new observation, which may cause the model unstable in terms of capturing the changing noises rather than the underlying pattern. Thus, the length and/or frequency to update model is an interesting research question to address.

REFERENCES

Bowerman , B & O'Connell, R; Forecasting And Time series: An Applied Approach, Duxbury, Pacific Grove, CA: 1993.

Box, G; & Jenkins, G; Time Series Analysis: Forecasting and Control, 2d edition, Holden-Day, San Francisco: 1976.

Engle, R; Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of UK Inflation, Econometrica, vol 50 Nbr. 4, pg. 987-1008, 1982.

Granger, C; & Anderson, A; An Introduction to Bilinear Time Series Models, Vandenhoeck and Ruprecht, Gottingen: 1978.

Haykin, S; Neural Networks A comprehensive Foundation, IEEE Press, Macmillan, New York, 1994.

Hu, M.Y.; Zhang, G.P.; Jiang, C.X.; & Patuwo, B.E.; A Cross-Validation Analysis of Neural Network Out-of Sample Performance in Exchange rate Forecasting, Decision Sciences, Vol 30, No. 1, Winter 1999.

Lasdon, L; Waren, A; Jain, A; & Ratner, M; Design and testing of a Generalized Reduced Gradient Code for Nonlinear Programming, ACM Transactions on Mathematical Software, vol. 4 no. 1 March 1978, 34-50.

Neter, J.; Kutner, M.H.; Nachtsheim, C.J.; & Wasserman, W; Applied Linear Statistical Models, fourth Edition, Irwin, Chicago, 1996.

Tong, H; & Lim, K; Threshold Autoregression, Limit cycles and Cyclical Data, Journal of Royal Statistical Society Vol. 42, B, 245-292, 1980.

Walczak S. An empirical analysis of data requirements for financial forecasting with neural networks. Journal of management information systems 17(4): 203-222, 2001.

Zhang, G; Patuwo, B; & Hu, M; Forecasting with Artificial Neural Networks: The State of the Art, International Journal of Forecasting, vol. 14, 35-62, 1998.

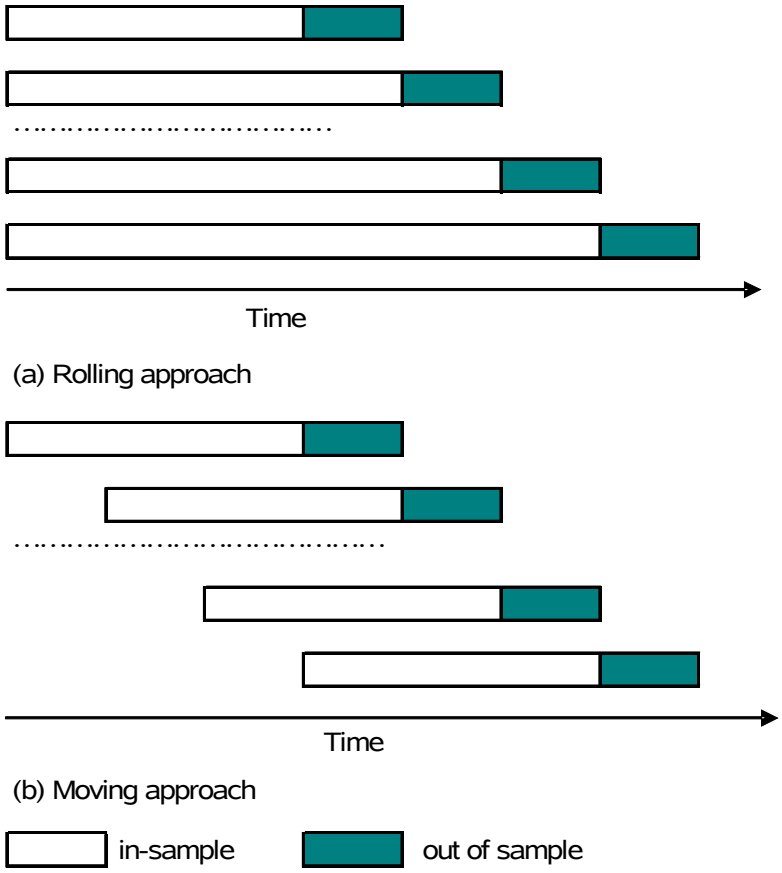


Figure 1: The rolling and moving window approaches

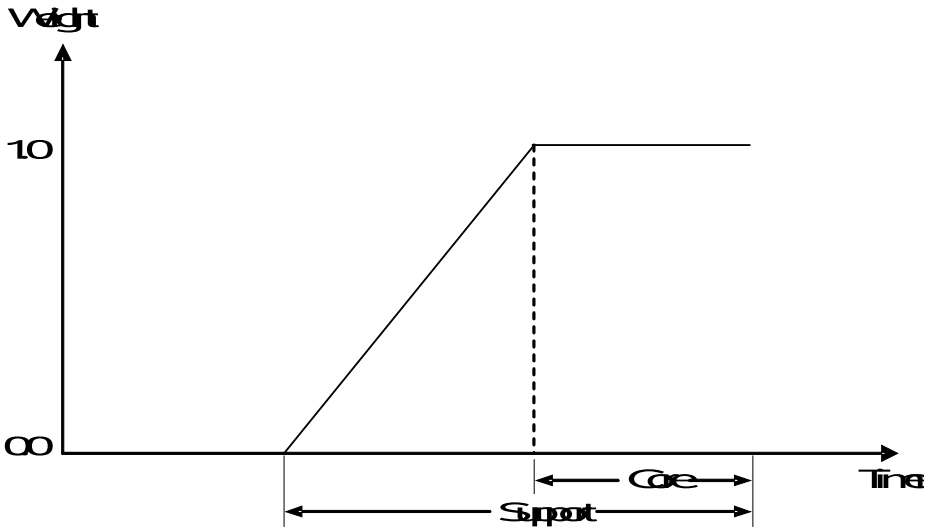
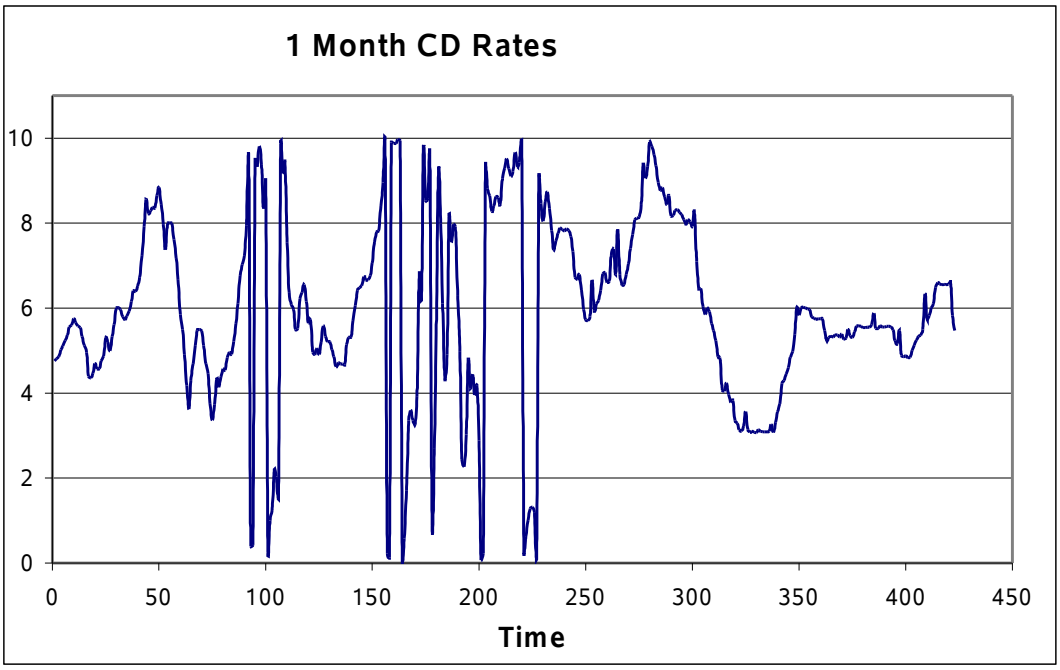
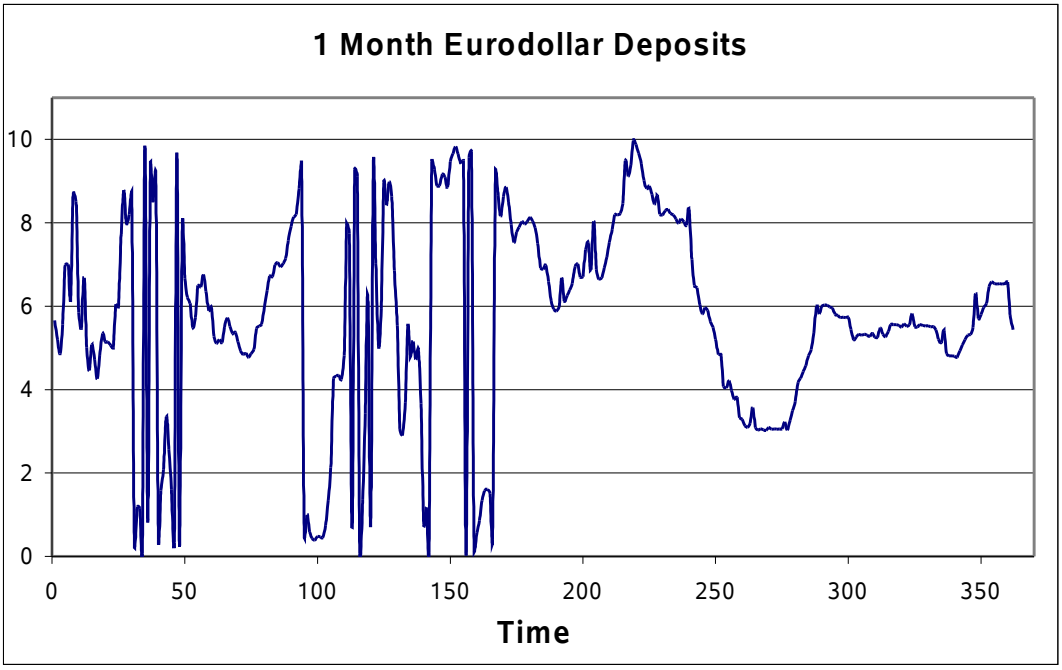


Figure 2. The weighting scheme

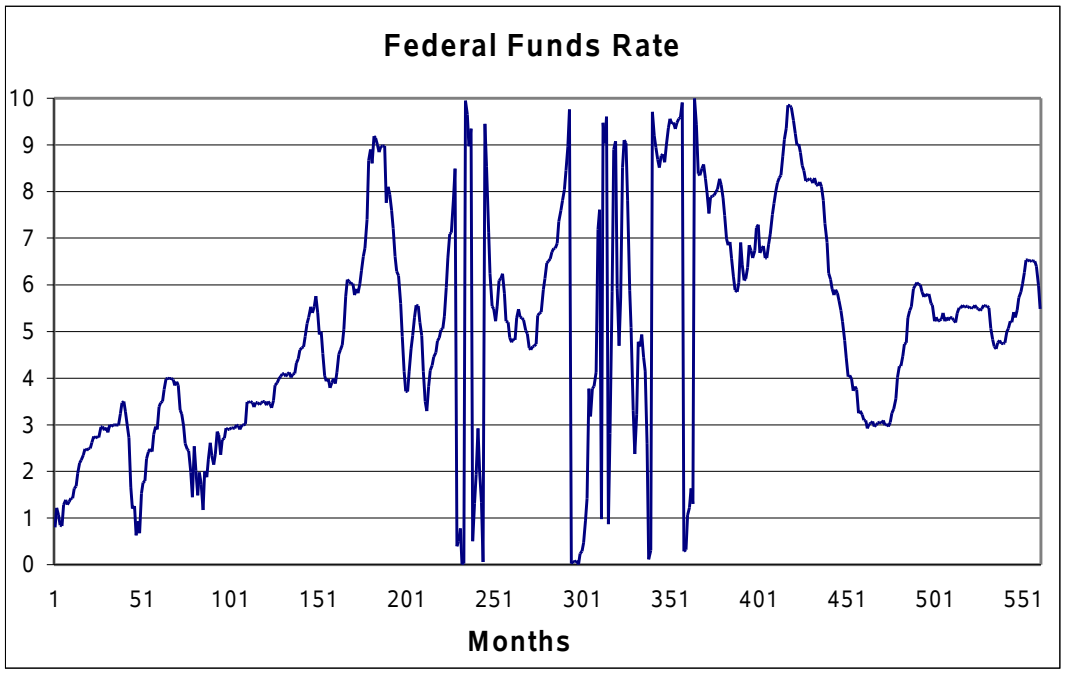


(a). One-month CD rates

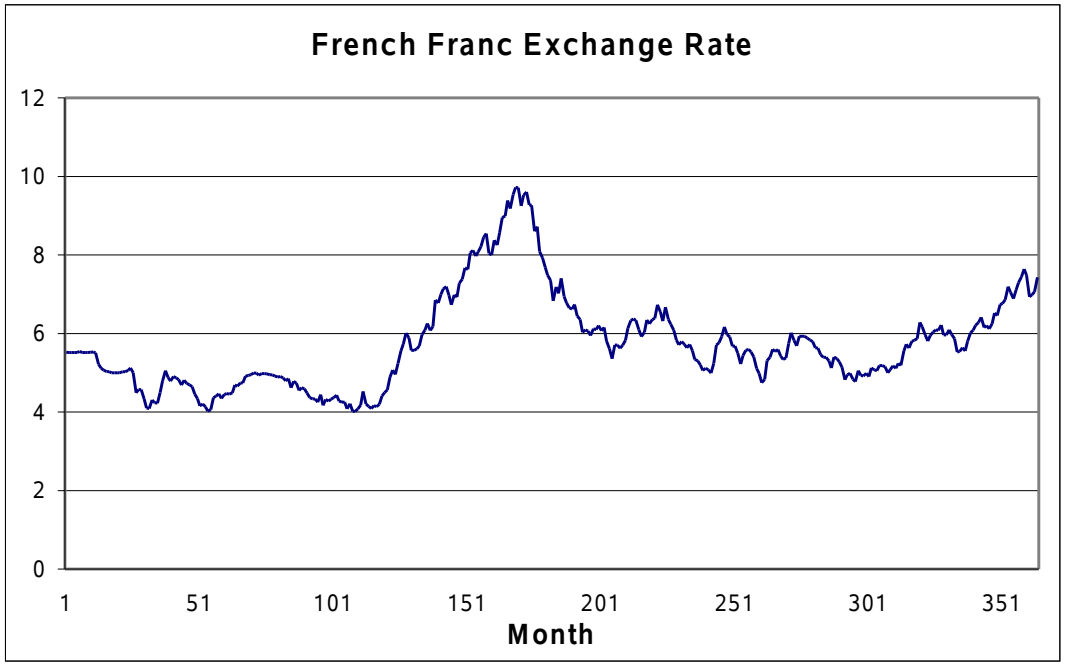


(b). One month Eurodollar deposits

Figure 3. Seven economic time series

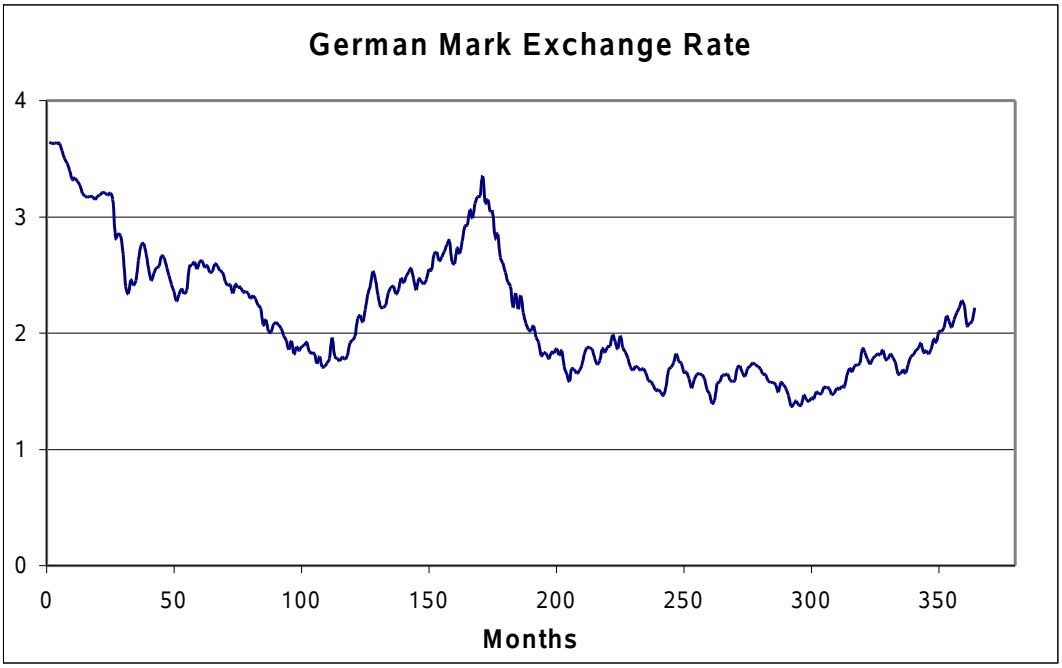


(c) Federal funds rates

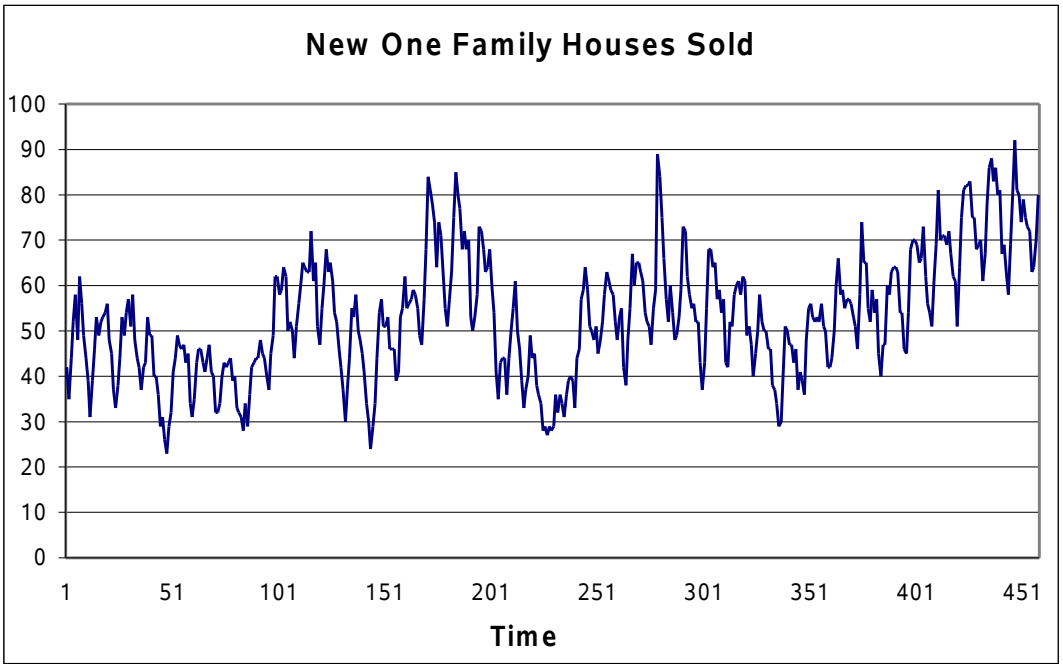


(d). Frech Franc exchange rate

Figure 3. Seven economic time series

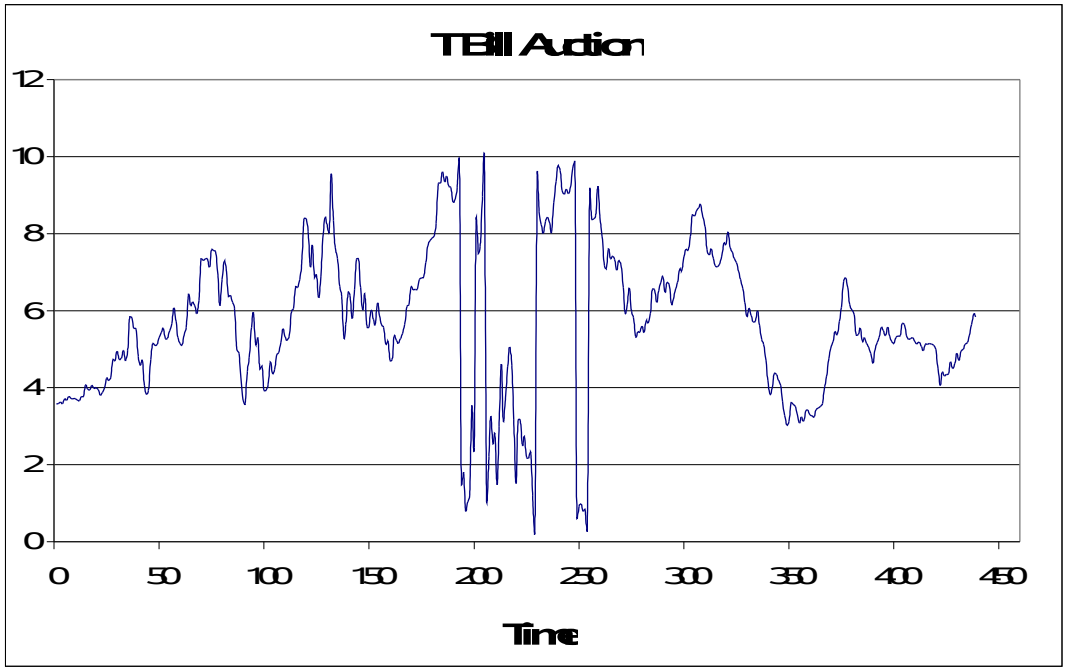


(e). German Mark exchange rate



(f). One family hoursing sales

Figure 3. Seven economic time series



(g). T-bill auction

Figure 3. Seven economic time series

Table 1. Summy results of out-of-sample forecasting

	Method		
Data Set	Rolling	Moving	Weighted Window
CD Rate	12.20	11.40	5.36
EuroDollar	4.98	5.25	4.13
Fed Fund	4.41	4.45	2.50
France Franc	3.26	3.29	2.78
German Mark	2.57	2.46	2.74
Houses	20.21	20.12	11.19
T Bill	5.04	5.27	6.18

Table 2. ANOVA results

Data Set	Source of Variation	SS	df	F	P-value
CD rate	Time	5773.04	30	6.04	0.0000
	Method	867.19	2	13.6	0.0000
	Error	1912.97	60		
Eurodollar	Time	1813.91	30	4.05	0.0000
	Method	21.37	2	0.71	0.4952
	Error	901.71	60		
Federal fund	Time	897.85	30	8.99	0.0000
	Method	77.09	2	11.59	0.0000
	Error	199.55	60		
French Franc	Time	483.54	30	10.01	0.0000
	Method	4.91	2	1.53	0.2257
	Error	96.59	60		
German Mark	Time	233.95	30	14.67	0.0000
	Method	1.93	2	1.82	0.1716
	Error	31.89	60		
House sale	Time	5236.44	30	6.24	0.0000
	Method	1664.84	2	29.78	0.0000
	Error	1677.04	60		
T-bill	Time	2099.77	30	8.06	0.0000
	Method	13.57	2	0.78	0.4623
	Error	520.82	60		

Table 3. Pairwise comparison results

Pairwise	CD rate		Federal fund		House sale	
Comparison	Difference	Significant	Difference	Significant	Difference	Significant
WW to moving	-6.04	yes	-1.95	yes	-9.02	yes
WW to rolling	-6.84	yes	-1.91	yes	-8.93	yes
Moving to rolling	-0.80	no	-0.03	no	-0.09	no