Forecasting Nonlinear Time Series with Feed-Forward Neural Networks: A Case Study of Canadian Lynx Data

Yoshio Kajitani[1], Keith W.Hipel [2,3], and A Ian.McLeod [4,2]

1. Central Research Institute of Electric Power Industry, 1646 Abiko, Abiko City, Chiba, 611-0011, Japan (email:y-kaji@criepi.denken.or.jp)
2. Department of Systems Design Engineering, University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada (email:kwhipel@uwaterloo.ca; tel: (519) 888 4567 ext.2830; fax:519 746 4791)
3. Department of Statistics and Actuarial Science, University of Waterloo
4. Department of Statistical and Actuarial Sciences, The University of Western Ontario, London, Ontario N6A 5B7, Canada

Abstract:

The forecasting capabilities of Feed-Forward Neural Network (FFNN) models are compared to those of other competing time series models by carrying out forecasting experiments. As demonstrated by the detailed forecasting results for the Canadian lynx data set, FFNN models perform very well, especially when the series contains nonlinear and non-Gaussian characteristics. To compare the forecasting accuracy of a FFNN model with an alternative model, Pitman's test is employed to ascertain if one model forecasts significantly better than another when generating one-step ahead forecasts. Moreover, the residual-fit spread plot is utilized in a novel fashion in this paper to compare visually out-of-sample forecasts of two alternative forecasting models. Finally, forecasting findings on the lynx data, are used to explain under what conditions one would expect FFNN models to furnish reliable and accurate forecasts.

Keywords: Feed-Forward Neural Networks, Lynx Data, Pitman's test, SETAR Model, Visualization

Revised Paper

August, 2003

1 Introduction

In recent years, neural networks or neural nets have been applied to many areas of statistics, such as regression analysis (De Veux et al., 1998), classification and pattern recognition (Ripley, 1996), and time series analysis. General discussions of employment of neural networks in statistics are presented by Warner and Misra (1996) and Chen and Titterington (1994). Within the statistical literature, the theory and application of neural networks have been advanced and in certain situations neural networks have been found to work as well or better than rival statistical models. For an account of the historical development of neural computation, one can refer to books by authors such as Anderson and Rosenfeld (1998) and Johnson and Brown (1988). Well-written textbooks on neural networks include contributions by Kasabov (1998), Mehrotra et al. (1997), Freeman (1994) and Hertz et al. (1991). Neural networks have been featured in mass-circulation popular magazines as such as *Maclean's* magazine in Canada (Kurzweil, 1999). Dyson (1997) provides an entertaining and speculative look at the future of neural computation and its impact on the *World-Wide-Web*.

In spite of the diverse applicability of neural networks in many different areas, much controversy surrounds their employment for tackling problems that can also be studied using well established statistical models. One such controversial domain is time series forecasting. Accordingly, the main objective of this paper is to use forecasting experiments to explain under what conditions FFNN(Feed-Forward Neural Network) models forecast well when compared to competing statistical models.

Following a description of FFNN models in the next section, an overview is given about the use of neural networks in time series prediction. In Section 3, model calibration methods for FFNN models and techniques for comparing forecasts from competing models are described. As one of the comparison methods, Pitman's test is introduced because it is utilized in the subsequent forecasting experiments to determine if one model forecasts significantly better than another. In addition, the residual-fit plot of Cleveland (1993) is put forward as an insightful visual means for comparing the forecasting abilities of two models. In Section 4, forecasting experiments with Lynx data are presented based on the analytical framework explained in Section 3. By making comparisons with a statistical model suggested by Tong (1983), many advantages of FFNN models are shown. Overall, FFNN models work well for forecasting certain types of "messy" data that may, for example, be nonlinear and not follow a Gaussian distribution.

2 Feed-Forward Neural Network Models in Time Series Prediction

A variety of neural net architectures has been examined for addressing the problem of time series prediction. These architectures include: *Multilayer Perceptron (MLP)* (Lisi and Schiavo, 1999; Faraway and Chatfield, 1998; Stern, 1996; Hill et al., 1996; Lachtermacher and Fuller, 1995; Jayawardena and Fernando, 1995); *Recurrent Networks* (Freeman, 1994, §6.2); *Radial Basis Functions (RBF)* (Jayawardena and Fernando, 1998; Hutchinson, 1994); *comparison of MLP and RBF* (Fernando and Jayawardena, 1998; Jayawardena et al., 1996).

There is substantial motivation for using FFNN for predicting time series data. Hill et al. (1996), for example, mention the following drawbacks of statistical time series models which neural network models might solve:

- Without expertise, it is possible to misspecify the functional form relating the independent and dependent variables, and fail to make necessary data transformations.
- Outliers can lead to biased estimates of model parameters.
- Time series models are often linear and thus may not capture nonlinear behavior.

2.1 Nonlinear Time Series

Lisi and Schiavo (1999) used a FFNN model for predicting European exchange rates. The FFNN model was found to perform as well as the best model which was a chaos model. Chaos or dynamical nonlinear systems provide another new approach to time series forecasting which have had some success (Casdagli, 1989). Both the FFNN and chaos models outperformed the classical random walk model for one-step ahead forecasting of daily exchange rate data. According to Lisi and Schiavo (1999), based on a statistical test, there was no significant difference between FFNN and the chaos models but both of these models performed significantly better than the traditional random walk model, which is usually the best model for such data.

Lachtermacher and Fuller (1995) mention that Lapedes and Farber (1987, 1988) generated two deterministic nonlinear time series, which look chaotic, and found neural networks performed excellent in generating forecasts. They think that neural networks have a key role to play in time series forecasting.

Jayawardena and Fernando (1995) applied FFNN models to daily discharge data at a streamflow gauging station in Hong Kong. They found that the FFNN approach is better than the traditional tank model method for forecasting in terms of root mean square error (RMSE) out-of-sample forecasting.

Jayawardena et al. (1996) applied the RBF method, which is similar to FFNN, to runoff forecasting. The RBF approach has the advantage that it does not require a long calculation time and does not suffer with the overtraining problem. In their study, they found that the RBF method performs the same as FFNN in terms of RMSE out-of-sample forecasting for mean water levels.

2.2 Linear Time Series

Faraway and Chatfield (1998) compared FFNN models with a SARIMA (seasonal autoregressive integrated moving average) model on their accuracy for forecasting airline data. In their paper, they discovered that FFNN models also give smaller mean square errors (MSEs) of out-of-sample prediction, but they mention that one has to be cautious when applying FFNN models to time series. For choosing an appropriate FFNN architecture, they recommend using the Baysian information criterion (BIC) (Akaike, 1977; Schwarz, 1978). However, the FFNN procedure is not a probabilistic type of neural network which assumes random errors, and therefore it is strange to use the BIC which is based on a likelihood obtained by random errors. In fact, Chen and Titterington (1994) mention that the traditional neural network approach proposes an optimality criterion without any mention of random errors and probability models. Another interesting result from Faraway and Chatfield (1998) is that their log transformation for the airline data did not improve the forecasting accuracy.

On the contrary, Lachtermacher and Fuller (1995) suggest using the Box-Cox transformation recommended by Hipel and McLeod (1994) in their modeling framework. They employ the Box-Jenkins method approach to build a suitable neural network structure by identifying the lag components of the time series. Moreover, they demonstrate the usefulness of their hybrid methodology by applying it to four stationary time series (annual river flows) and four non-stationary time series (annual electricity consumption).

Stern (1996) applied FFNN models to several data sets generated by autoregressive models of order 2 (abbreviated as AR(2) models) with different signal to noise ratios. He concluded that if the signal to noise ratio is small, FFNN models cannot produce good predictions. However, his FFNN architecture is chosen without regard to sound theoretical reasons.

Hill et al. (1996) mention that the length of training data (number of historical data) influences the forecasting accuracy. Overall, many issues have been discussed by researchers with respect to time series forecasting using FFNN models.

Based on these previous forecasting results, FFNN models seem to be suitable for time series forecasting with small signal to noise ratios if we have enough data and use appropriate data transformation techniques. Therefore, FFNN models should be more widely applied to this type of data not only for forecasting purposes but also for other reasons such as checking the performance of developed statistical models or producing combinations of forecasts as is done by Lachtermacher and Fuller (1995). Especially when a time series is nonlinear or messy and statistical modeling is difficult, FFNN models can be advantageous in providing quick and accurate forecasts of the series. Accordingly, more forecasting experiments should be carried out to compare the performance of FFNN models with other types of models not only for experimentally generated data but also for actual time series. The Lynx data studied Section 5 in our paper constitute a typical nonlinear time series for which FFNN models outperform other statistical models.

Comparison methods are another important issue for the FFNN application. Since FFNN models are not probabilistic, residuals do not usually follow a probability distribution. Therefore, we adopt a methodology to compare the forecasts considering the nonprobabilistic feature of FFNN models. Specifically, Pitman's test constitutes an appropriate statistical test for comparing forecasting accuracy between FFNN models and other statistical models. In addition, a visualization method called residual-fit spread (RFS) plot is introduced to compare two different forecasting methods.

3 Forecasting Procedures for FFNN Models

3.1 Model Calibration Method

Software for fitting neural networks is widely available. The research in this paper uses the SPSS (1998) package called *Neural Connection*. This software supports a variety of node activation functions, weight functions and learning algorithms. We examine six types of

FFNN models combining three kinds of node activation functions and two types of weight functions shown in Table 1. Two types of learning algorithms can be selected, but in our case we examine only the default setting, which is the conjugate gradient method. For the time series data, we have to construct a input data with several lags. The way to select a number for the lag can be arbitrary, but a reasonable idea is to select a lag for which the autocorrelation between the original data and lagged data becomes large. Several dirrerent of lags are examined in our case study, but the lag is chosen considering the autocorrelation. Following Faraway and Chatfields' (1998) notation, in time series modeling using the FFNN, let the model be denoted by $\mathrm{NN}(\mathcal{L}; h)$, where $\mathcal{L}$ is the set of lags and h is the number of hidden nodes.

Typically the data are divided into three portions, $N = N_1 + N_2 + N_3$. N_1 of the data are used for training and N_2 are used as validation data to check the prediction accuracy for the model selection. N_3 of the data are employed for the out-of-sample predictions by the calibrated model. The smallest number of RMSEs for the validation data becomes a desirable model, but N_2 can be arbitrarily chosen and the forecasting accuracy by the calibrated model depends on the numbers, N_1 and N_2. Therefore, we examine several numbers for N_1 and N_2. The model which produces the smallest RMSE (root-mean-square errors) for both the training and validation data is desirable. An overtrained model can produce a small RMSE for the training data set, but tends to produce a large RMSE for the validation data set. On the other hand, a large RMSE for the training data set and a small RMSE for the validation data set may indicate that the forecasts for the validation happen to fit the data well. Therefore, we recommend to use the number of validations, which makes smallest the larger of the RMSE for the training and validation data sets.

A reasonable method to determine the numbers of hidden nodes and layers is still controversial in the FFNN application field. (e.g. Moshiri and Cameron (2000), Amilon (2003)). In this paper, the numbers of hidden nodes and hidden layers are experimentally determined as either 1 or 2. If the best model is obtained, then we try the 3 hidden nodes case. This approach is designed to minimize the time required to obtain the forecasts and minimize the forecast errors.

We focus primarily on one-step ahead forecasting. One reason for this is that forecasting methods for a stationary series predict the mean as the long-run forecast. Another justification is that one-step ahead forecasts are most useful in practice, since one should automatically recalibrate a model and generate new forecasts as more data become available. Moreover, for many models, one step ahead forecast errors are independent of one another. Hence, one-step forecasts are best for discriminating among competing models. Additionally, in most previous forecasting experiments with time series, one-step forecasts have been used. A simple comparison of the performance of the methods is given by simply comparing the MSE (mean-square error) or RMSE of the forecasts. Median absolute error, mean absolute percentage error and other criteria have been used in previous forecasting studies (Hipel and McLeod,1994, §15) and there are many other possibilities.

3.2 Comparing Forecasts from Competing Models

Methods for rigorously comparing forecasts between FFNN models and statistical models are required for properly carrying out forecasting experiments. In this paper, we introduce two approaches – a statistical test and a visualization method.

Since the one-step forecast errors under a valid model are theoretically uncorrelated, we can use Pitman's test for testing the equality of variances in the paired sample case (Hipel and McLeod, 1994, §8.3.2). This test can indicate if the forecasting performance as measured by MSE is significantly different between two models.

Visualization and graphical methods often provide insights that complement significance tests by indicating the magnitude of differences and often revealing unexpected outliers and other features of the data. Cleveland (1993) stresses the use of visualization methods as diagnostic methods to check the statistical assumptions behind fitting and also for exploratory data analysis (Tukey, 1977) where no explicit probability model is entertained. We adapt the RFS plot discussed by Cleveland (1993) to compare the out-of-sample forecasting performance.

First, we look at time series plots showing the observed values and forecasts. This is an obvious plot to provide a quick examination of the forecasting capability of a model. However, the time series plot is not very good at revealing differences between competing models.

The RFS plot provides a visual summary of the amount of variability accounted for by a model and is readily adapted to out-of-sample forecast comparisons. The RFS plot is comprised of two panels. The left panel shows the quantile plot of the targets minus their mean. The right panel of the RFS plots is a quantile plot of the forecast errors. For comparing the forecasting performance of two models, we keep the scales identical on both parts of the RFS plots.

4 Forecasting Experiments with Lynx Data

4.1 Previous Time Series Analysis Studies

The lynx data consist of the set of annual numbers of lynx trappings in the Mackenzie River District of North-West Canada for the period from 1821 to 1934. It is assumed by ecologists that these data indicate the relative magnitude of the lynx population and are, therefore, of great interest to ecological researchers. The data are plotted in Figure 1. Ecologists believe that cycles in population, such as those observed with the lynx series, occur when there is a strong predator-prey relationship. The predator causes the prey to decline and this is followed by a decline in the predator population. The prey population recovers and this causes an increase in the predator population and so the cycle continues.

This lynx data are one of the most frequently used time series. It is actually part of a much larger collection of time series derived originally from Hudson Bay Company archives and first published by Elton (1927) and analyzed by Elton and Nicholson (1942). The first modern time series analysis was carried out by Moran (1953) who fit an AR(2) model to the logged data. Almost every type of nonlinear time series models has been shown to produce a better fit than the AR(2) model. In an extensive review, Tong (1990, §7.2) discusses the numerous nonlinear models that have been tried. Tong (1990, §7.2) favors the self-exciting threshold autoregression (SETAR) model. Tong (1983) used the SETAR model for some out-of-sample forecast comparisons with other nonlinear models. Lin and Pourahmadi (1998) analyzed this data by additive and projection pursuit regression (PPR) and concluded that again Tong's SETAR model performed much better than the PPR and other models. Nonetheless, in this paper, it is shown that the FFNN model is just as good or better than SETAR models for one-step out-of-sample forecasting of the lynx data. The FFNN model is also compared to the AR(2) model to further justify this claim.

4.2 Forecasting Results

We divide the lynx data into three data sets, consisting of training, validation and test data. The number of test data is set to 14 and its predicted values are compared with the predicted values generated employing the model reported by Tong (1983). Because Tong (1983) discussed his results for the lynx data in the transformed domain, Pitman's test and RFS plots are presented here for the transformed domain.

The histograms of the lynx data were skewed to the right and, hence, it is recommended to use the transformation $\ln(x + a)$, where a is a constant (SPSS, 1998). In fact, Tong (1983) used log with base = 10, which makes the lynx data more symmetrical looking.

The results of the forecasting experiments are summarized in Tables 2, and 3. Table 2 shows the models which produce the smallest RMSEs for the test data. SETAR(2;6,3) means that there are two switching regimes, one of which has 6 lagged variables and the other has 3 lagged variables. Among these models, the $NN(1, 2; 2)$ model has the smallest RMSEs for the test data. Because the RMSEs for the validation data are also smallest, there is a higher chance to select a $NN(1, 2; 2)$ structure compared with other FFNN structures. However, there is no guarantee on selecting the numbers of training and validation data sets. Table 3 shows the results of the models with the recommended number of validation data. The number of validation data is selected such that the larger RMSE between training and validation becomes smallest. We can see that the $NN(1, 2; 2)$ model performs best again in both validation and test, and it outperforms SETAR(2,6,3) in terms of the RMSEs of forecasts.

Figure 2 shows that the RMSEs of $NN(1, 2; 2)$ models are mostly smaller than the SETAR model even if the number of validation data is changed. Overall, the forecasting findings support the employment of the best FFNN model, NN$(1, 2; 2)$ with a logarithmic transformation, to obtain smaller RMSE values than those from the SETAR model.

Using RFS plots, we can visually compare the out-of-sample forecasts with the observed data. The plots offer more information than just reporting σ_p, the RMSE of the out-of-sample predictions. The RFS plots in Figure 4 show that the NN$(1, 2; 2)$ model draws our attention to only one positive outlier, while the SETAR(2,6,3) model is suffering from not only one positive outlier but also one negative outlier. This indicates that there is a pattern of the series that the NN$(1, 2; 2)$ model can detect, but the SETAR(2,6,3) cannot.

Then, we examined Pitman's test for 14 predicted values from SETAR(2;6,3) and NN(1,2:2) models. The difference between the SETAR(2;6,3) and NN(1,2:2) forecast errors is not statistically significant in the transformed domain for which, Pitman's test $r = 0.122$, $\hat{t} = 1.292$, indicates a significance level of $2Pr(t > \hat{t}) = 0.220$. In terms of the Pitman statistical test, the FFNN model does not significantly outperform the SETAR in out-of-sample forecasting for the lynx data at the 5% level of significance. However, the SETAR model building explained in Tong (1983) shows that it takes a long time to find the appropriate SETAR model. On the other hand, FFNN models do not take much time to design the most suitable model.

Overall, when one takes into account the results of RMSE performance, RFS plots, and model building speed for the Lynx data, the FFNN model performs well when compared to its competitors. Kajitani (1999) presents the results of forecasting experiments when various types of FFNN models are compared to other time series models for actual seasonal time series and nonlinear data composed of deterministic trends. Once again, FFNN models perform well.

5 Conclusions

This study examines the forecasting performance of FFNN models compared to other competing statistical models. From the previous studies on time series forecasting with FFNN models, it is shown that FFNN models work well even when the signal to noise ratio is small. Therefore, we selected the lynx data, which looks nonlinear and less noisy, as suitable data for testing and showing the usefulness of the FFNN modeling approach in time series forecasting. Because of the complex nonlinearity of the lynx data, considerable effort has been expended by many statisticians and ecologists to model this interesting time series.

In terms of modeling speed, FFNN models work well. There are sensible approaches to construct FFNN models but, in fact, there are not many restrictions for building FFNN models because of their black box characteristics. One can obtain a comparatively reasonable model more quickly than when one builds a competitive statistical model. In this paper, the structure of a FFNN model is experimentally found by considering its autocorrelation, and we simply set a rule to select the number of validation data such that the larger RMSE between training and validation becomes smallest (see Section 3). Even this simple rule is enough to show the high forecasting performance of FFNN models and it is demonstrated that FFNN models outperform an AR(2) model and a SETAR(2,6,3) model in terms of RMSEs with respect to forecasting the lynx data. Previously, the SETAR(2,6,3) model of Tong (1983) was the most recommended statistical model for applying to the lynx data.

Comparison of RMSEs is one useful approach to determine forecasting accuracy between competing models. However, it is recommended to use appropriate statistical tests or other complementary approaches when carrying out a thorough forecasting study. In this paper, RFS plots and Pitman's statistical test are proposed as useful comparison techniques which do not have to satisfy any random error assumptions for the residuals. As a result of Pitman's test, it is not clearly shown that the FFNN model statistically outperforms the SETAR model. However, RFS plots indicate that the FFNN model possesses only one outlier, while the SETAR model has two outliers. For these two outliers produced by the SETAR model, one is recognized as a deterministic pattern by the FFNN model. Since outliers tend to influence statistical test results, it is important to complement the test by this type of visualization techniques.

Overall, our research findings demonstrate that the FFNN can perform as well as or even outperform competing statistical models, especially in nonlinear and non-Gaussian situations. Moreover, the simple rules for constructing FFNN models and visual comparison techniques are helpful for allowing FFNN models to be conveniently employed in forecasting studies.

6 References

Akaike H. 1977. On entropy maximization principle. In *Applications of Statistics* , Krishnaiah, PR (ed); North-Holland: Amsterdam; 27–41.

Amilon, H. 2003. A neural network versus Black-Scholes: A comparison of pricing and hedging performances, *Journal of Forecasting*, **22**, 317–335.

Anderson JA, Rosenfeld E. 1998. *Talking Nets: An Oral History of Neural Networks*, MIT Press: Cambridge, Massachusetts.

Box GEP, Jenkins GM. 1973. Comments on "Box-Jenkins seasonal forecasting: Problems in a case-study" by C. Chatfield and D.L. Prothero, *Journal of the Royal Statistical Society* **A136**: 295–308.

Casdagli M. 1989. Nonlinear prediction of chaotic time series, *Physica* **D35**: 335–356.

Chen B, Titterington DM. 1994. Neural networks: A review from a statistical perspective, *Statistical Science* **9**: 2–54.

Chatfield C, Prothero DL. 1973. Box-Jenkins seasonal forecasting: Problems in a case-study, *Journal of the Royal Statistical Society* **A136**: 295–308.

Cleveland WS. 1993. *Visualizing Data*, Hobart Press: Summit, New Jersey.

De Veux RD, Schumi J, Schweinsberg J, Ungar LH. 1998. Prediction intervals for neural networks via nonlinear regression, *Technometrics* **40**: 273–282.

Dyson G. 1997. *Darwin Among the Machines: The Evolution of Global Intelligence*, Addison-Wesley: Reading, Massachusetts.

Elton C. 1927. *Animal Ecology*, MacMillan: New York.

Elton C, Nicholson M. 1942. The ten-year cycle in numbers of the lynx in Canada, *Journal of Animal Ecology* **11**: 215–244.

Faraway J. 1998. http://www.stat.lsa.umich.edu/∼ faraway/ [1 December 1998]

Faraway J, Chatfield C. 1998. Time series forecasting with neural networks: a comparative study using the airline data, *Applied Statistics* **47**: 231–250.

Fernando DAK, Jayawardena AW. 1998. Runoff forecasting using RBF networks with OLS algorithm. *Journal of Hydrologic Engineering* **3 No.3**: 203–209.

Freeman, JA. 1994. *Simulating Neural Networks with Mathematica*, Addison-Wesley: Reading, Massachusetts.

Hertz J, Krogh A, Palmer RG. 1991. *Introduction to the Theory of Neural Computation*, Addison-Wesley: Reading, Massachusetts.

Hill T, O'Conner M, Remus, W. 1996. Neural network models for time series forecasts, *Management Science* **42, No.7**: 1082–1092.

Hipel KW, Mcleod AI. 1994. *Time Series Modelling of Water Resources and Environmental Systems.* Elsevier: Amsterdam.

Hutchinson JM. 1994. *A Radial Basis Function Approach to Financial Time Series Analysis*, Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge, Massachusetts.

Jayawardena AW, Fernando DAK. 1995. Artificial neural networks In hydro-meteorological modeling, *Developments in Neural Networks and Evolutionary Computing for Civil and Structural Engineering*, Topping, BHV(ed). Civil-Comp: Edinburgh; 115–120.

Jayawardena AW, Fernando DAK. 1998. Use of radial basis function type artificial neural networks for runoff simulation. *Computer-Aided Civil and Infrastructure Engineering* **13 No.2**, 91–99.

Jayawardena AW, Fernando DAK, Zhou MC. 1996. Comparison of multilayer perceptron and radial basis function networks as tools for flood forecasting. *Destructive Water: Water-Caused Natural Disaster, their Abatement and Control*, International Association of Hydrological Sciences Press: Oxfordshire, UK; 173–182.

Johnson RC, Brown C. 1988. *Cognizers: Machines that Think*, Wiley: New York.

Kajitani Y. 1999. *Forecasting Time Series with Neural Nets*, Master's Thesis, Department of Statistical and Actuarial Sciences, The University of Western Ontario, London, Ontario, Canada.

Kasabov NK. 1998. *Foundations of Neural Networks, Fuzzy Systems and Knowledge Engineering*, MIT Press: Cambridge, Massachusetts.

Kurzweil R. 1999. When machines think, *Maclean's*, March 1, 54–57.

Lachtermacher G, Fuller JD. 1995. Backpropagation in time-series forecasting, *Journal of Forecasting* **14**: 381–393.

Lapedes A, Farber R. 1987. *Nonlinear signal processing using neural networks: prediction and system modelling.* Technical Report LA-UR-87-2662, Los Alamos National Laboratory, New Mexico.

Lapedes A, Farber R. 1988. How neural nets works. In *Neural Information Processing Systems*, Anderson DZ (ed). American Institute of Physics: New York; 442–456.

Lin TC, Pourahmadi M. 1998. Nonparametric and nonlinear models and data mining in time series: a case-study on the Canadian lynx data. *Applied Statistics* **47, Vol.2**: 187–201.

Lisi F, Schiavo RA. 1999. A comparison between neural networks and chaotic models for exchange rate prediction, *Computational Statistics and Data Analysis* **30**:87–102.

Mcleod AI. 1993, Parsimony, Model Adequacy and Periodic Correlation in Forecasting Time Series, *International Statistical Review* 61, 387–393.

Mcleod AI, Hipel KW. 1999. *MHTS PC Package*, http://www.stats.uwo.ca/mcleod/epub/mhts [1 Jan 1999]

Mehrotra K, Mohan CK, Ranka S. 1997. *Elements of Artificial Neural Nets*, MIT Press: Cambridge, Massachusetts.

Moran PAP. 1953. The statistical analysis of the Canadian lynx cycle, *Australian Journal of Zoology* **1**: 163–173.

Moshiri S, Cameron N. 2000. Neural network versus econometric models in forecasting inflation, *Journal of Forecasting*, **19**: 201–217.

Ripley BD. 1996. *Pattern Recognition and Neural Networks*, Cambridge University Press: Cambridge, Massachusetts.

Ripley, BD. 1999. http://www.stats.ox.ac.uk/pub/MASS2/ [1 Jan 1999].

Schwarz G. 1978. Estimating the dimension of a model. *Ann. Statist.* **6**: 1310-17.

SPSS Inc. 1998. *Neural Connection, Version 2.0*, SPSS Inc./ Recognition System Inc., Chicago.

Stern HS. 1996. Neural networks in applied statistics, *Technometrics* **38**: 205–220.

Tong H. 1983. *Threshold Models in Nonlinear Time Series Analysis*, Springer-Verlag: New York.

Tong H. 1990. *Nonlinear Time Series* Oxford University Press: Oxford.

Tukey, JW. 1977. *Exploratory Data Analysis.* Addison-Wesley: Reading, Massachusetts.

Venables WN, Ripley BD. 1997. *Modern Applied Statistics with S-plus*, 2nd edn. Springer-Verlag: New York.

Warner B, Misra M. 1996. Understanding neural networks as statistical tools. *The American Statistician* **50**: 284–293.

Wilson GT. 1973. Comments on "Box-Jenkins seasonal forecasting: Problems in a case-study" by C. Chatfield and D.L. Prothero, *Journal of the Royal Statistical Society* **A 136**: 295–308.

Model	Node Activation Function	Weight Function
A	Sigmoid	Gaussian
B	Sigmoid	Uniform
C	Tanh	Gaussian
D	Tanh	Uniform
E	Linear	Gaussian
F	Linear	Uniform

Table 1: *Examined FFNN models featured by Node activation functions and weight functions.*

Model	Type	N_V	σ_t	σ_v	σ_p
NN(1:1)	F	80	0.342	0.361	0.221
NN(1:2)	A	80	0.343	0.359	0.218
NN(1,2:1)	C	35	0.209	0.289	0.130
	D	35	0.209	0.289	0.130
NN(1,2:2)	A	50	0.199	0.240	0.095
NN(1,2:3)	A	5	0.209	0.399	0.091
NN(1,2,3:1)	F	10	0.229	0.312	0.139
NN(1,2,3:2)	B	15	0.201	0.325	0.123
NN(1,10:1)	F	5	0.321	0.522	0.214
NN(1,10:2)	A	15	0.327	0.313	0.208
NN(1,2,10:1)	E	15	0.229	0.299	0.151
NN(1,2,10:2)	C	50	0.173	0.253	0.118
AR(2)			0.310		0.190
SETAR(2;6,3)			0.220		0.122

Table 2: *Comparison of the RMSEs among AR(2) model, SETAR(2,6,3) and several FFNN models for the logged lynx data with base* = 10. *FFNN models, which produce the smallest RMSEs for out-of-sample forecasts, are selected. The number of test data is set as 14. The sum of the training and validation is set as 100. The N_V is the number of the validation data, the σ_t is the RMSE of the training data, σ_v is the RMSE of the validation data and the σ_p is the RMSE of the test data.*

	Model Type	N_V	σ_t	σ_v	σ_p
NN(1:1)	B	70	0.352	0.353	0.240
NN(1:2)	C	70	0.344	0.348	0.245
NN(1,2:1)	F	70	0.240	0.241	0.138
NN(1,2:2)	B	65	0.196	0.229	0.114
NN(1,2:3)	D	65	0.209	0.235	0.129
NN(1,2,3:1)	A	65	0.210	0.249	0.161
NN(1,2,3:2)	D	65	0.201	0.244	0.159
NN(1,10:1)	F	55	0.302	0.356	0.250
NN(1,10:2)	C	55	0.291	0.347	0.280
NN(1,2,10:1)	F	60	0.215	0.260	0.220
NN(1,2,10:2)	D	55	0.166	0.246	0.138
AR(2)			0.310		0.190
SETAR(2;6,3)			0.220		0.122

Table 3: *Comparison of RMSEs among AR(2) model, SETAR model and several FFNN models for the logged lynx data with base* $= 10$*. The number of validation data is selected such that the larger RMSE between training and validation becomes smallest. The* N_V *is the number of the validation data, the* σ_t *is the RMSE of the training data,* σ_v *is the RMSE of the validation data and the* σ_p *is the RMSE of the test data.*

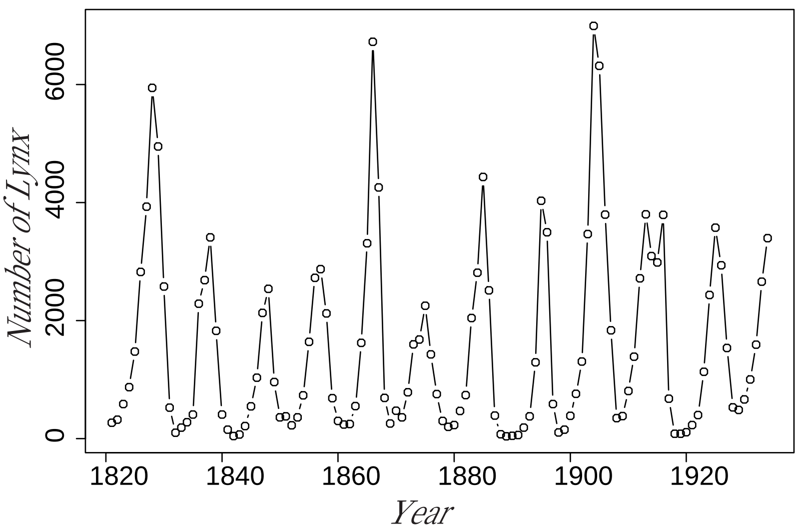

Figure 1: *Lynx data.*

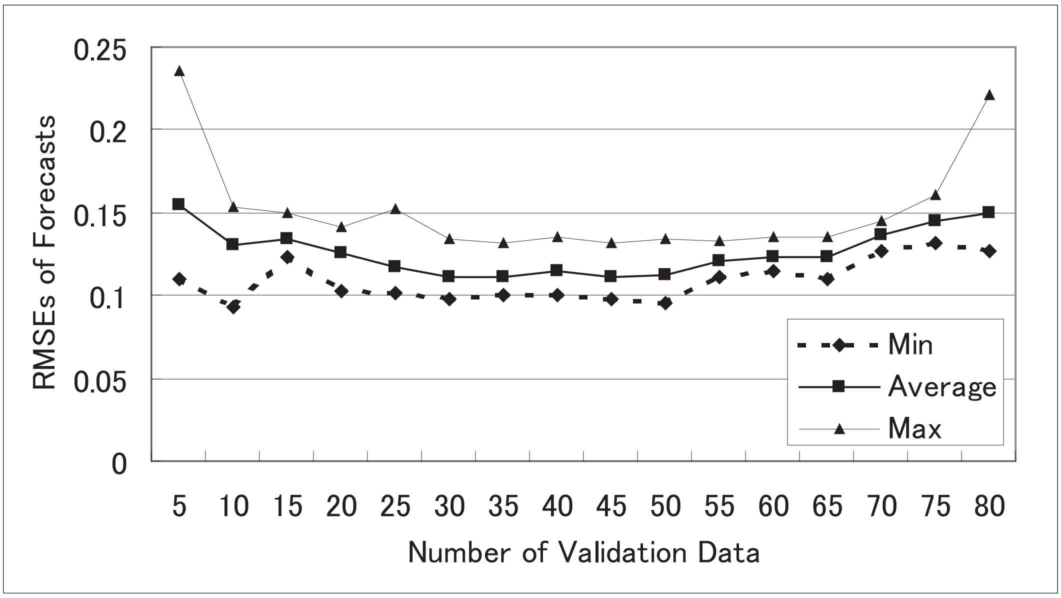

Figure 2: *RMSEs of the test data for $NN(1,2;2)$ with the various numbers of validation data sets.*

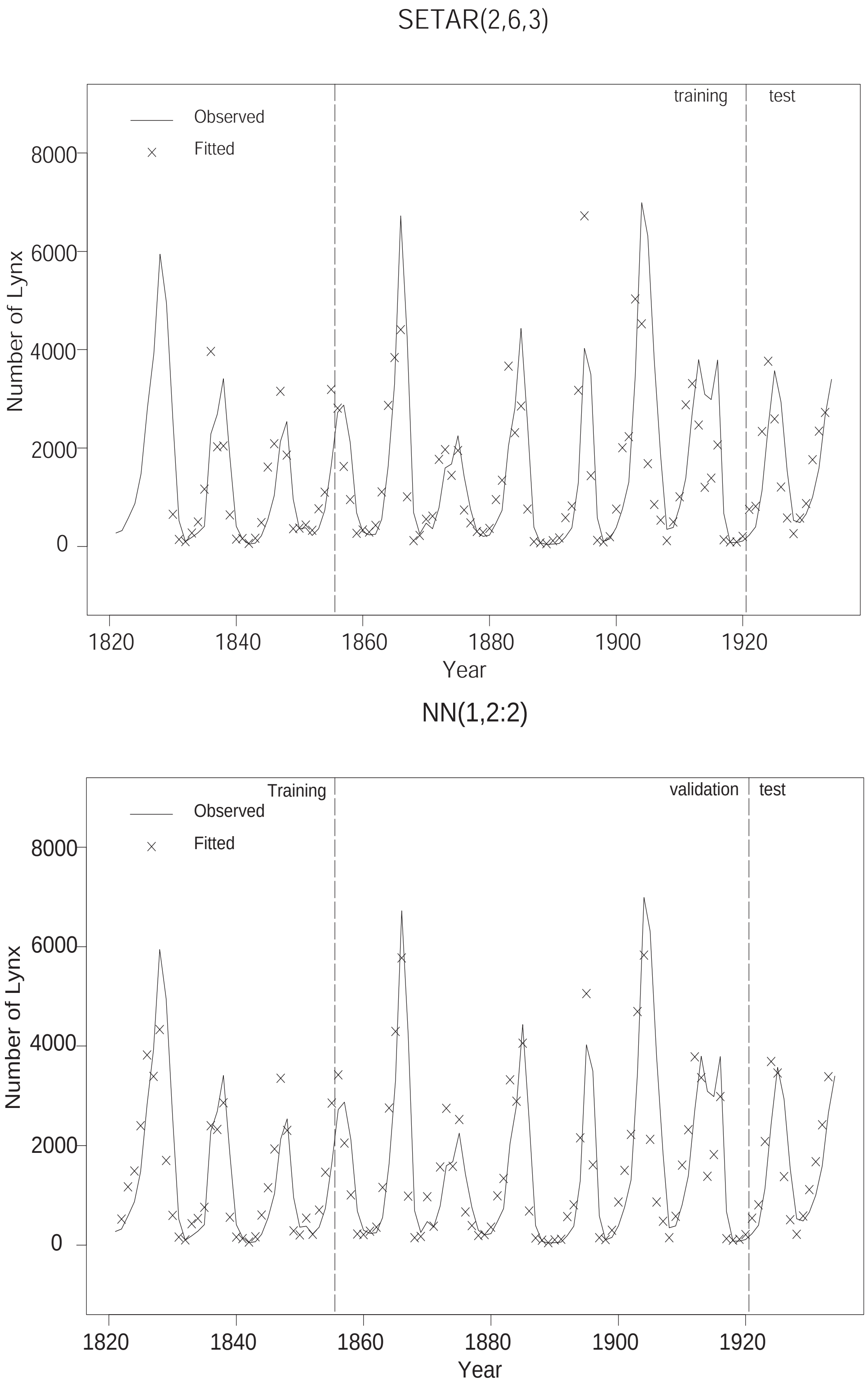

Figure 3: *Fitted values for the training data, validation data and test data of the log transformed lynx data for a* $SETAR(2;6,3)$ *model and a* $NN(1,2;2)$ *model. Fitted values are plotted after transforming back to the original domain. The observations are plotted with a solid line while the fitted values are denoted by X's. The number of training data is 35, the number of validation data is 65 and the number of test data is 14*

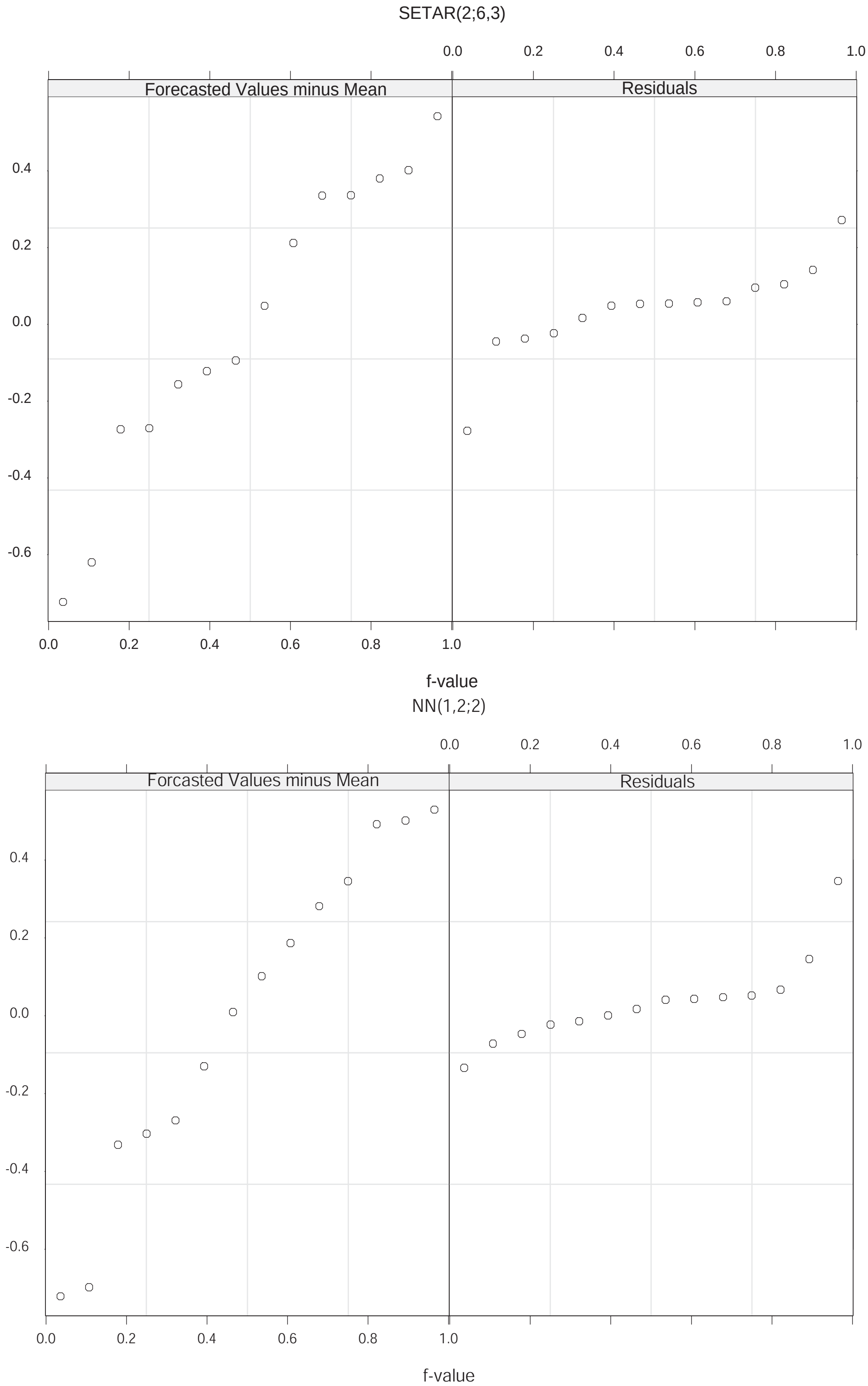

Figure 4: *RFS plots of the 14 lynx test data for the* $NN(1,2;2)$ *and SETAR(2;6,3) models. The left panel in each display shows the quantile plot of the forecasts minus the mean. The right panels are the quantile plots of the residuals.*