

## ELECTRICITY LOAD FORECASTING WITH ARTIFICIAL NEURAL NETWORKS

**H. Borraine**

Department of Statistics  
University of Pretoria, South Africa

[hboraine@hakuna.up.ac.za](mailto:hboraine@hakuna.up.ac.za)

**V.S.S. Yadavalli**

Department of Industrial and Systems Engineering  
University of Pretoria, South Africa

[yadavalli@postino.up.ac.za](mailto:yadavalli@postino.up.ac.za)

### ABSTRACT

Artificial neural networks are powerful tools for time series forecasting. The problem addressed in this article is to do multi-step prediction of a stationary time series, and to find the associated prediction limits. Artificial neural network models for time series are non-linear. However, results that are applicable to linear models are sometimes mistakenly applied to non-linear models. One example where this is observed is in multi-step forecasting. A bootstrap method is proposed to calculate one- and multi-step predictions and prediction limits. The results are applied to an electricity load time series as well as to a pure autoregressive time series.

### OPSOMMING

Kunsmatige neurale netwerke is kragtige instrumente vir tydreeksvoorspelling. In hierdie artikel word multistap-vooruitberaming van 'n stasionêre tydreeks en die gepaardgaande vertrouinterval behandel. Resultate wat slegs geldig is vir lineêre modelle word soms verkeerdelik op neurale netwerkmodelle toegepas. 'n Voorbeeld hiervan kom in multistap-voorspelling voor. 'n Skoenlusmetode, word voorgestel waarvolgens eenstap- en multistap-voorspellings en vertrouintervalle bereken kan word. Die resultate word op 'n elektrisiteitslastydreeks en op 'n suiwer outoregressiewe tydreeks toegepas.

## 1. INTRODUCTION

Artificial neural networks are powerful tools for time series forecasting. See [10], [6], [8] and [17] for various applications. The problem addressed in this article is to do multi-step prediction of a stationary time series, and to find the associated prediction limits.

Artificial neural network models for time series are non-linear. However, results that are applicable to linear models are sometimes mistakenly applied to non-linear models. One example where this is observed is in multi-step forecasting. Let  $Z_1, \dots, Z_n$  represent a stationary time series. Suppose that the observation at time  $t$  can be described by the following model:

$$Z_t = \eta(Z_{t-1}, Z_{t-2}, \dots, Z_{t-p}; \theta) + a_t \quad (1)$$

where  $\eta(\cdot)$  is a linear or non-linear function,

$p$  is the autoregressive order of the model, in other words  $Z_t$  can be explained by  $p$  previous observations  $Z_{t-1}, Z_{t-2}, \dots, Z_{t-p}$  which will be used as inputs of the neural network,

$\theta$  is the parameter vector and

$\dots, a_{t-1}, a_t, \dots$  are uncorrelated and identically distributed random variables with mean zero and variance  $\sigma_a^2$ .

The minimum mean square error forecast of a future observation is the expected value of the future value, given the observed time series [1]. Let the vector  $\mathbf{Z}_n$  represent the  $n$  observations  $(Z_1, \dots, Z_n)$ . At time  $n$  (which is the number of terms in the time series), the  $h$ -step forecast of  $Z_{n+h}$ , given observations  $\mathbf{Z}_n$  is the conditional expected value

$$Z_n(h) = E(Z_{n+h} | \mathbf{Z}_n) \quad (2)$$

From (1) it follows that

$$Z_{n+h} = \eta(Z_{n+h-1}, Z_{n+h-2}, \dots, Z_{n+h-p}; \theta) + a_{n+h} \quad (3)$$

The notation  $h$  is used to indicate the number of time steps of the forecast. It is assumed that forecasts are made from time  $n$ , in other words, the observed time series is used up to its last value to produce forecasts.  $Z_n(1)$  is, for instance, the forecast of the future value  $Z_{n+1}$  which is unknown at time  $n$ , and likewise,  $Z_n(2)$  is the forecast of the future value  $Z_{n+2}$  at time  $n$ . The number of previous time series observations used to explain the current observation is denoted by  $p$  and it also corresponds to the number of input terms in a neural network. If this model is used for electricity load forecasting on an hourly basis,  $p$  previous hours' loads will be used to forecast the current load.

Therefore, using (2) and (3) the forecast of  $Z_{n+h}$ , given observations  $\mathbf{Z}_n$  is

$$\begin{aligned} Z_n(h) &= E[\eta(Z_{n+h-1}, Z_{n+h-2}, \dots, Z_{n+h-p}; \theta) + a_{n+h} | \mathbf{Z}_n] \\ &= E[\eta(Z_{n+h-1}, Z_{n+h-2}, \dots, Z_{n+h-p}; \theta) | \mathbf{Z}_n] \text{ since } E[a_{n+h} | \mathbf{Z}_n] = 0 \text{ by assumption.} \end{aligned} \quad (4)$$

In the case of a *linear* model, forecasts can be calculated recursively, by using forecasts calculated in previous steps, since the expected value of a linear function is the linear function of the expected values:

$$\begin{aligned} &E[\eta(Z_{n+h-1}, Z_{n+h-2}, \dots, Z_{n+h-p}; \theta) + a_{n+h} | \mathbf{Z}_n] \\ &= \eta(E[Z_{n+h-1} | \mathbf{Z}_n], E[Z_{n+h-2} | \mathbf{Z}_n], \dots, E[Z_{n+h-p} | \mathbf{Z}_n]) + E[a_{n+h} | \mathbf{Z}_n] \\ &= \eta(E[Z_{n+h-1} | \mathbf{Z}_n], E[Z_{n+h-2} | \mathbf{Z}_n], \dots, E[Z_{n+h-p} | \mathbf{Z}_n]) \text{ since } E[a_{n+h} | \mathbf{Z}_n] = 0 \end{aligned} \quad (5)$$

It follows from (2) that

$$\begin{aligned} E[Z_{n+h-1} | \mathbf{Z}_n] &= Z_n(h-1), \\ E[Z_{n+h-2} | \mathbf{Z}_n] &= Z_n(h-2), \text{ etc.} \end{aligned} \quad (6)$$

Therefore, from equations (4), (5) and (6), the  $h$ -step forecast in the linear case can be written as

$$\begin{aligned} Z_n(h) &= E[\eta(Z_{n+h-1}, Z_{n+h-2}, \dots, Z_{n+h-p}; \theta) + a_{n+h} | \mathbf{Z}_n] \\ &= \eta(Z_n(h-1), Z_n(h-2), \dots, Z_n(h-p)) \end{aligned} \quad (7)$$

It can be seen that the forecast  $Z_n(h)$  depends on forecasts of the preceding time points,  $Z_n(h-1), Z_n(h-2), \dots, Z_n(h-p)$ . In neural network terminology, outputs from the neural network are fed back as inputs to produce subsequent forecasts.

However, if the function defined by the neural network architecture,  $\eta(Z_{t-1}, Z_{t-2}, \dots, Z_{t-p}; \theta)$ , is non-linear, equation (5) is not true. Even so, it is often seen in neural network literature that forecasts produced by the network are used as inputs for subsequent forecasts [8]. A number of different approaches to obtain one-step and multi-step forecasts for non-linear models are discussed by [11]. One of the methods, namely the bootstrap, is used in this article to produce multi-step forecasts.

Since no forecast is complete without standard error and prediction limits, it is also shown how these quantities can be calculated with bootstrap methods. In [13] confidence intervals for *linear models* based on bootstrap methodology are proposed. The results are extended to non-linear models in this article. Confidence intervals are derived for a time series that is generated by an autoregressive process. The results are also applied to find predictions and confidence limits for an electricity load time series.

## 2. FORECASTING WITH AUTOREGRESSIVE MODELS

In the case of an autoregressive model of order  $p$  or  $AR(p)$  model (see [1]), which is linear,

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + a_t, \quad (8)$$

the  $h$ -step forecast (see equation (7)) is

$$\begin{aligned} Z_n(h) &= E(Z_{n+h} | \mathbf{Z}_n) \\ &= \phi_1 Z_n(h-1) + \phi_2 Z_n(h-2) + \dots + \phi_p Z_n(h-p), \quad h = 1, 2, \dots \end{aligned} \quad (9)$$

Note that it is possible that  $h-j \leq 0$  for  $j = 2, \dots, p$ . In that case, the observed time series value is used. Using the property of conditional expectation, it follows that

$$\begin{aligned} Z_n(0) &= E(Z_n | \mathbf{Z}_n) = Z_n, \\ Z_n(-1) &= E(Z_{n-1} | \mathbf{Z}_n) = Z_{n-1}, \\ \text{and in general } Z_n(j) &= Z_{n-j} \text{ if } j < 0. \end{aligned}$$

The forecasts are calculated recursively, and converge to the mean of the time series for large values of  $h$ .

Assuming that the distribution of  $\dots, a_{t-1}, a_t, \dots$  is normal, the distribution of  $Z_{n+h} | \mathbf{Z}_n$  is normal with mean  $Z_n(h)$  and variance  $\sigma_a^2 (1 + \sum_{j=1}^{h-1} \theta_j^2)$  where the  $\theta_j$ 's are the weights in the moving average representation of  $Z_t$  (see reference [1]):

$$Z_t = \theta_0 + a_t + \theta_1 a_{t-1} + \theta_2 a_{t-2} + \dots \quad (10)$$

Approximate  $1 - \alpha$  probability limits for  $Z_{n+h} | \mathbf{Z}_n$  are given by

$$Z_n(h) \pm z_{1-\frac{\alpha}{2}} \{ \sigma_a^2 (1 + \sum_{j=1}^{h-1} \theta_j^2) \}^{\frac{1}{2}} \quad (11)$$

where  $z_{1-\frac{\alpha}{2}}$  is the  $100(1 - \frac{\alpha}{2})$ -th percentile of the standard normal distribution.

## 3. MULTI-STEP FORECASTS AND CONFIDENCE INTERVALS FOR NEURAL NETWORK MODELS

Neural network models are non-linear, and the expected value of a non-linear function is generally not of a simple form. In the case where the model in equation (1) is non-linear, the observation at time  $n+1$  is

$$Z_{n+1} = \eta(Z_n, Z_{n-1}, \dots, Z_{n+1-p}; \theta) + a_{n+1}, \quad (12)$$

and the minimum MSE single-step ( $h=1$ ) forecast (from equation (4)) is

$$\begin{aligned} Z_n(1) &= E[\eta(Z_n, Z_{n-1}, \dots, Z_{n+1-p}; \theta) | \mathbf{Z}_n] \\ &= \eta(Z_n, Z_{n-1}, \dots, Z_{n+1-p}; \theta) \end{aligned} \tag{13}$$

since the expected value of a function of known values (in this case  $Z_n, Z_{n-1}, \dots, Z_{n+1-p}$ ) is equal to the function of the known values. The maximum likelihood estimator of  $Z_n(1)$  is

$$\hat{Z}_n(1) = \eta(Z_n, Z_{n-1}, \dots, Z_{n+1-p}; \hat{\theta}) \tag{14}$$

where  $\hat{\theta}$  is an estimator of  $\theta$  which is obtained by maximising the likelihood function of the observations, or in neural network terminology, the vector  $\hat{\theta}$  represents the weights of the trained network.

The 2-, 3-, ... step forecasts are not of such a simple form, since they depend on the future time series observations that are unknown. Consider

$$Z_{n+2} = \eta(Z_{n+1}, Z_n, \dots, Z_{n+2-p}; \theta) + a_{n+2} \tag{15}$$

with minimum MSE forecast

$$Z_n(2) = E[\eta(Z_{n+1}, \dots, Z_{n+2-p}; \theta) | \mathbf{Z}_n] \tag{16}$$

which is the expected value of a non-linear function. Instead of calculating this expected value, which may be a complex integral, bootstrap methodology can be used to estimate the minimum MSE forecasts.

Bootstrapping is a resampling method introduced by Efron [4]. Different methods have been proposed for bootstrapping in the context of time series [2],[7],[9],[14],[15]. In this article, the method based on residuals is used ([3],[9] and [5]). The residuals of the neural network are the difference between observed time series values and the corresponding values predicted by the neural network. The idea is to generate a large number of time series from the same population as the observed time series, called bootstrap time series. For each bootstrap time series, some statistic of interest is calculated. In this way, a large sample of possible realizations of the statistic is obtained, called the empirical distribution of the statistic. The empirical distribution of the statistic is an estimator of the true sampling distribution. The variance of the statistic is for instance estimated by using the variance of the empirical distribution.

In this application, the statistic is a future value of the time series. The residuals of the neural network and the trained network are used to produce many different realizations of the time series from the point  $n+1$  onwards. It is possible to generate a whole distribution of possible values at each time point. The mean of the empirical distribution is an estimate of the forecast and the percentiles are used as confidence limits.

The two-step forecast in equation (16) depends on  $Z_{n+1}, Z_n, \dots, Z_{n+2-p}$ . Since  $Z_{n+1}$  is unknown, it has to be estimated.

$$Z_{n+1} = \eta(Z_n, Z_{n-1}, \dots, Z_{n+1-p}; \theta) + a_{n+1}$$

A number,  $m$ , of possible realizations of  $Z_{n+1}$ , denoted by  $Z_{n+1}^{*(1)}, Z_{n+1}^{*(2)}, \dots, Z_{n+1}^{*(m)}$ , can be obtained by using the one-step forecast and adding a typical error term to it. Recall from (12) and (13) that

$$\begin{aligned} Z_{n+1} &= \eta(Z_n, Z_{n-1}, \dots, Z_{n+1-p}; \theta) + a_{n+1} \\ &= Z_n(1) + a_{n+1} \end{aligned}$$

A possible realization of  $Z_{n+1}$  is given by:

$$\begin{aligned} Z_{n+1}^{*(j)} &= \hat{Z}_n(1) + a_{n+1}^{*(j)}, \quad j = 1, 2, \dots, m \\ &= \eta(Z_n, Z_{n-1}, \dots, Z_{n+1-p}; \hat{\theta}) + a_{n+1}^{*(j)} \end{aligned} \tag{17}$$

The error terms, denoted by  $a_{n+1}^{*(j)}, j=1, 2, \dots, m$  are observations (drawn randomly with replacement) from the residuals of the model,  $\hat{a}_{p+1}, \dots, \hat{a}_n$ , with

$$\hat{a}_t = Z_t - \eta(Z_{t-1}, \dots, Z_{t-p}; \hat{\theta}), \quad t = p+1, \dots, n \tag{18}$$

The outputs of the neural network corresponding with the  $m$  input sets

$$\begin{aligned} &Z_{n+1}^{*(1)}, Z_n, \dots, Z_{n+2-p} \\ &Z_{n+1}^{*(2)}, Z_n, \dots, Z_{n+2-p} \\ &\vdots \\ &Z_{n+1}^{*(m)}, Z_n, \dots, Z_{n+2-p} \end{aligned}$$

are averaged to get the two-step forecast:

$$\hat{Z}_n(2) = \frac{1}{m} \sum_{j=1}^m \eta(Z_{n+1}^{*(j)}, Z_n, \dots, Z_{n+2-p}; \hat{\theta}) \tag{19}$$

This approach can be extended to three steps, four steps, and so on. Suppose one-step, two-step, up to  $h$ -step forecasts are required. Following the above procedure, the  $j$ -th bootstrap ( $j=1, 2, \dots, m$ ) time series is generated as follows (see (17)):

$$\begin{aligned}
 Z_{n+1}^{*(j)} &= \eta(Z_n, Z_{n-1}, \dots, Z_{n+1-p}; \hat{\theta}) + a_{n+1}^{*(j)} \\
 Z_{n+2}^{*(j)} &= \eta(Z_{n+1}^{*(j)}, Z_n, Z_{n-1}, \dots, Z_{n+2-p}; \hat{\theta}) + a_{n+2}^{*(j)} \\
 Z_{n+3}^{*(j)} &= \eta(Z_{n+2}^{*(j)}, Z_{n+1}^{*(j)}, Z_n, Z_{n-1}, \dots, Z_{n+3-p}; \hat{\theta}) + a_{n+3}^{*(j)} \\
 &\vdots \\
 Z_{n+h}^{*(j)} &= \eta(Z_{n+h-1}^{*(j)}, Z_{n+h-2}^{*(j)}, \dots, Z_{n+h-p}^{*(j)}; \hat{\theta}) + a_{n+h}^{*(j)}
 \end{aligned} \tag{20}$$

A total of  $m$  bootstrap series is required. In general, the forecast of a future observation  $Z_{n+h}$  is the average network output associated with the  $p$  most recent observations which may be only bootstrap time series values (if  $h > p$ ): or a combination of bootstrap time series and actual observations (if  $h \leq p$ ) across the  $m$  bootstrap time series:

$$\hat{Z}_n(h) = \frac{1}{m} \sum_{j=1}^m \eta(Z_{n+h-1}^{*(j)}, Z_{n+h-2}^{*(j)}, \dots, Z_{n+h-p}^{*(j)}; \hat{\theta}) \tag{21}$$

where

$$Z_{n+k}^{*(j)} = \eta(Z_{n+k-1}^{*(j)}, \dots, Z_{n+k-p}^{*(j)}; \hat{\theta}) + a_{n+k}^{*(j)} \tag{22}$$

and

$$Z_{n+h-p}^{*(j)} = Z_{n+h-p} \text{ if } h - p \leq 0. \tag{23}$$

Various methods have been proposed for the calculation of bootstrap confidence intervals. See [16] for a review. The method based on percentiles is used in this application. The  $100(1 - \alpha)\%$  confidence limits for  $Z_{n+h}$  are the  $100(\frac{\alpha}{2})^{\text{th}}$  and  $100(1 - \frac{\alpha}{2})^{\text{th}}$  percentiles of the empirical distribution of  $Z_{n+h}$ .

The procedure can be summarized in a number of steps:

- Train the neural network on the time series, in other words, fit the model (1) to the time series  $Z_1, \dots, Z_n$ .
- Calculate estimates of the residual terms (using (18)):  $\hat{a}_t = Z_t - \eta(Z_{t-1}, \dots, Z_{t-p}; \hat{\theta})$ ,  $t = p + 1, \dots, n$ .
- Calculate  $Z_{n+1}^*, \dots, Z_{n+h}^*$  conditional on  $Z_1, \dots, Z_n$ :  $Z_{n+k}^* = \eta(Z_{n+k-1}^*, \dots, Z_{n+k-p}^*; \hat{\theta}) + a_{n+k}^*$  where  $Z_{n+h-p}^* = Z_{n+h-p}$  if  $h - p \leq 0$  and  $a_{n+k}^*$  is an observation (drawn randomly with replacement) from  $\hat{a}_{p+1}, \dots, \hat{a}_n$ .

For instance,

$$Z_{n+1}^{*(j)} = \eta(Z_n, Z_{n-1}, \dots, Z_{n+1-p}; \hat{\theta}) + a_{n+1}^{*(j)},$$

$$Z_{n+2}^{*(j)} = \eta(Z_{n+1}^{*(j)}, Z_n, \dots, Z_{n+2-p}; \hat{\theta}) + a_{n+2}^{*(j)}$$

$$Z_{n+3}^{*(j)} = \eta(Z_{n+2}^{*(j)}, Z_{n+1}^{*(j)}, Z_n, \dots, Z_{n+3-p}; \hat{\theta}) + a_{n+3}^{*(j)}, \text{ etc.}$$

- Repeat the previous step  $m$  times, where the minimum value for  $m$  is 1000. A minimum of 1000 observations is required for the calculation of the percentiles. The 2.5<sup>th</sup> and 97.5<sup>th</sup> percentiles are on the left and right extremes of the distribution. The generated bootstrap time series values will tend to be close to the mean, with only a few extreme values. A relatively large sample is therefore required to estimate values in the extremes of the distribution more accurately. There are therefore  $m$  bootstrap time series with possible values for  $Z_{n+1}, Z_{n+2}, \dots, Z_{n+h} | Z_n$ .
- The  $h$ -step forecast is given by (21).
- The 2.5<sup>th</sup> and 97.5<sup>th</sup> percentiles of  $Z_{n+h}^{*(1)}, Z_{n+h}^{*(2)}, \dots, Z_{n+h}^{*(m)}$  for  $h = 1, 2, \dots$  are the 95% confidence limits associated with the  $h$ -step forecast.

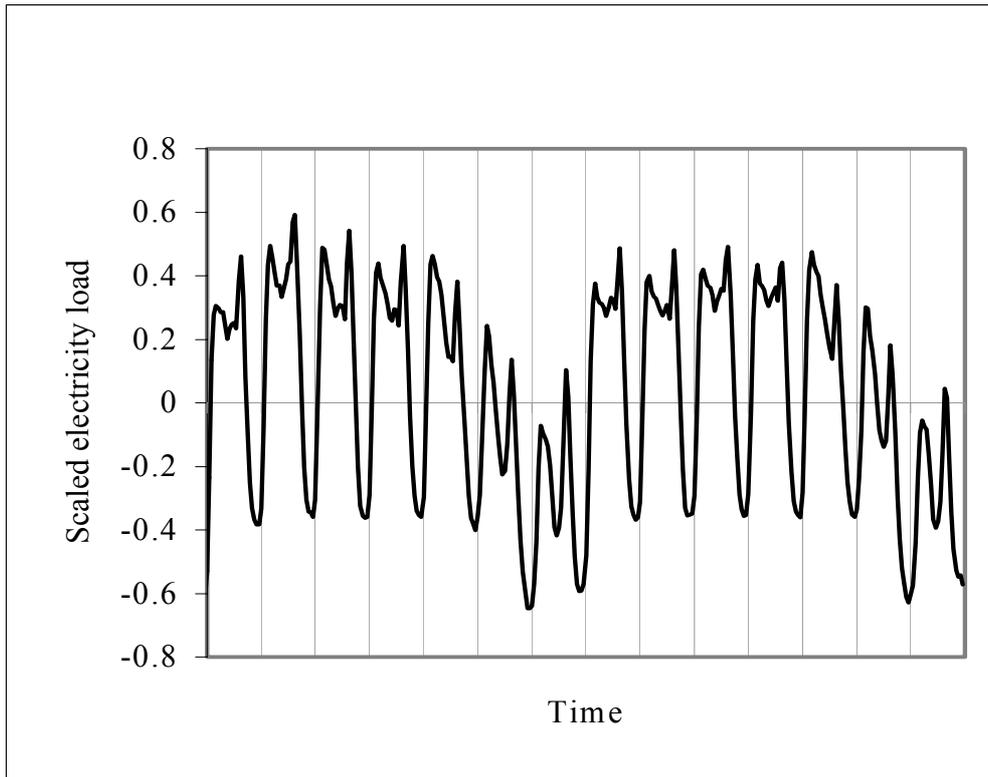
#### 4. EMPIRICAL RESULTS

##### Example 1

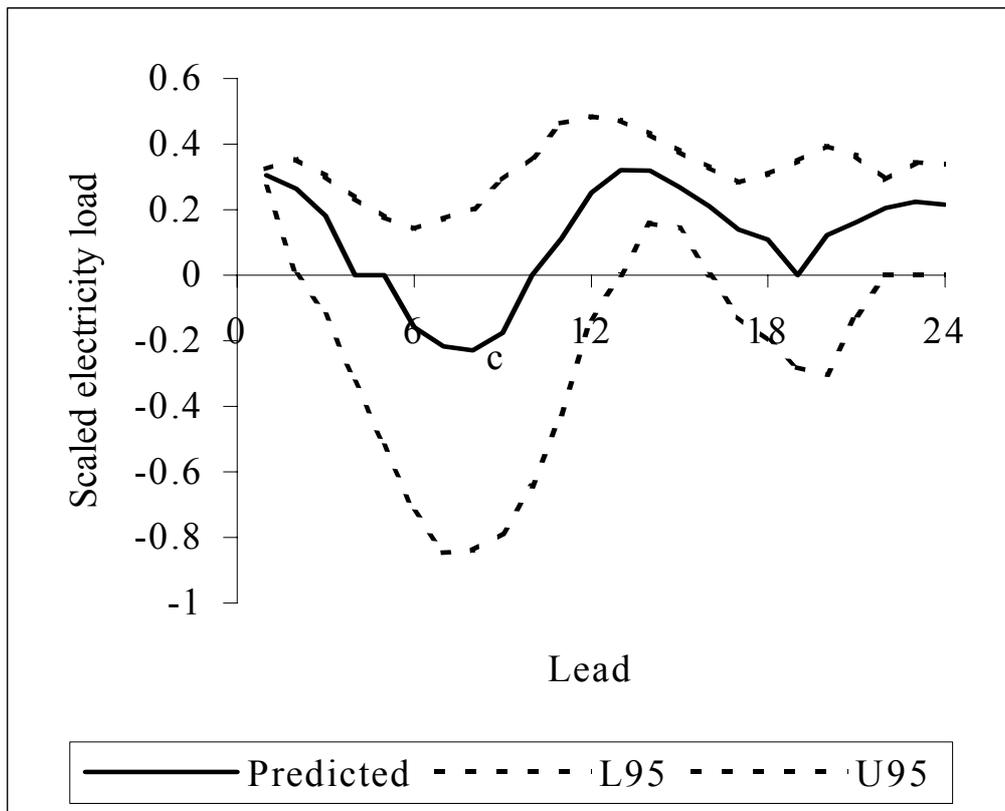
The time series considered is total electricity load demand measured in kilowatt hours accumulated hourly by Eskom (the largest electricity supplier in South Africa). The data were scaled between  $-1.0$  and  $1.0$ . A time plot of two weeks' data is given in Figure 1. Four weeks' data, a total of  $n=672$  observations, were used to train the network. Input variables included periodic terms to take care of the 168-hour, 24-hour and 12-hour consumption patterns present in the data, as well as electricity load values of previous time points. A feed forward neural network with one hidden layer and a sigmoidal activation function on the hidden layer was trained to predict the next load value, given the set of inputs. Bootstrap methodology was used to predict the electricity loads for 1 to 24 hours ahead, together with their standard errors and a 95% confidence interval. The results are presented in Figure 2. The notation L95 and U95 on the graph indicates the lower and upper 95% confidence limits respectively.

A linear regression model was also used to predict the electricity load. The same set of explanatory variables (inputs) was used. In this case, the prediction limits are based on the assumption that the prediction errors follow a normal distribution. The predictions and 95% prediction limits for the linear model are given in Figure 3. It can be seen that the forecasts obtained by the neural network and linear model compares well.

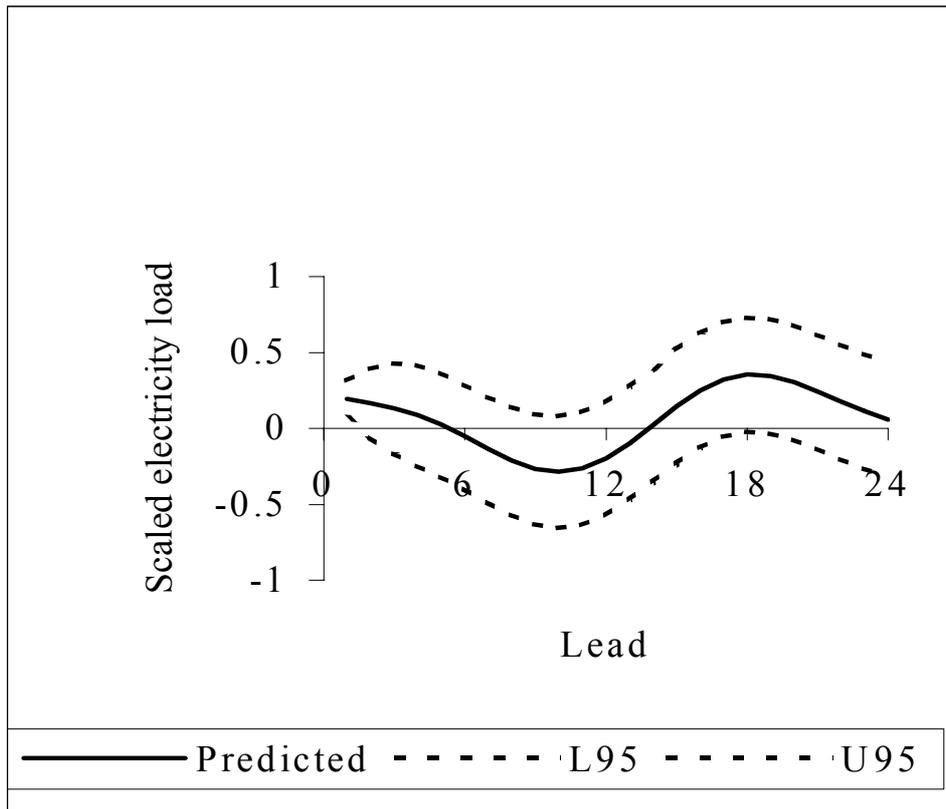
It can be seen that the predictions are quite smooth and that the width of the interval gradually increases with the forecasting period in the case of the linear model. The same is not true for the confidence interval obtained by the bootstrap method, especially towards the end of the forecasting period. This is due to the fact that the bootstrap method is essentially data driven and does not rely on any assumptions of the probability distribution of the data.



**Figure 1: Hourly electricity load demand time series**



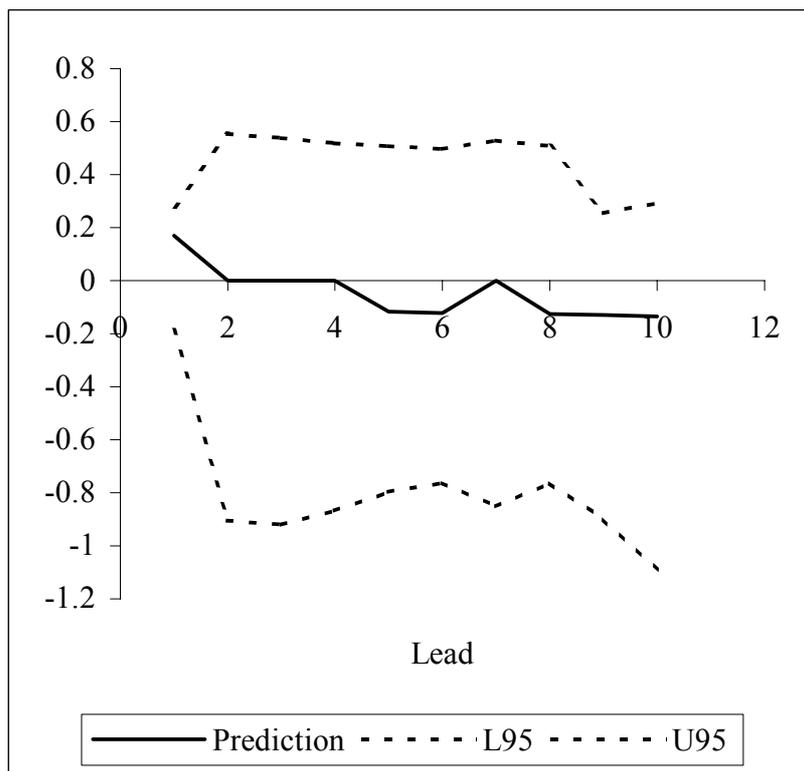
**Figure 2: Predicted hourly electricity load  
(Bootstrap approach applied to neural network)**



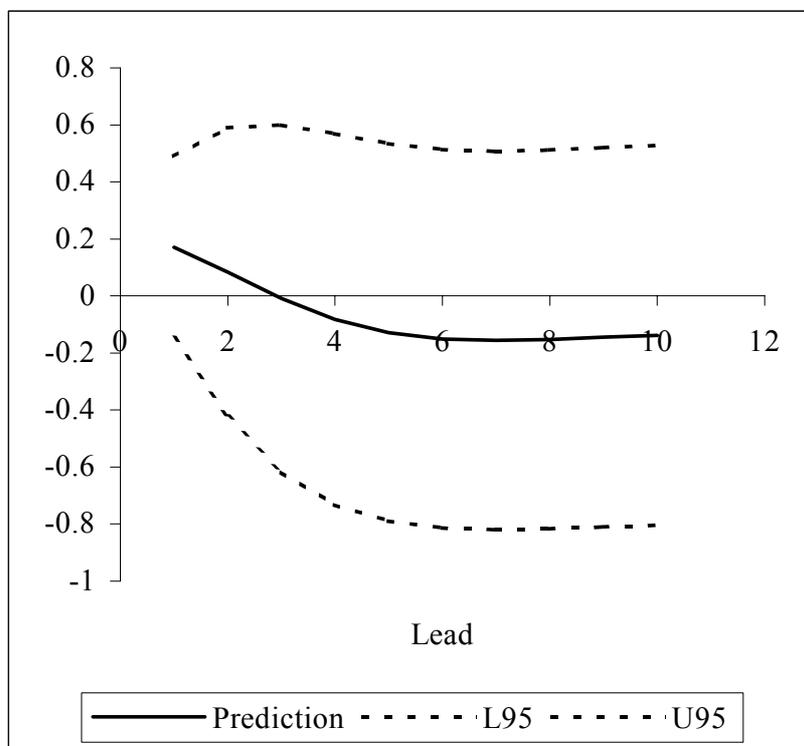
**Figure 3: Predicted hourly electricity load (linear regression model)**

### Example 2

The data used for this example were 200 computer-generated values of a stationary autoregressive model of order 2 ( $AR(2)$ ) process as given by equation (8). The generating process is linear, and it can therefore not be expected that a neural network model would produce better prediction results than a linear model. A feed forward neural network with two input nodes for two past values of the time series and one hidden layer with a sigmoidal activation function was trained. Bootstrap methodology was used to calculate predictions and prediction limits. The prediction results are given in Figures 4 and 5. As in the previous example, the results for the neural network, where bootstrap methods were used, and the linear model correspond more or less.



**Figure 4: Autoregressive time series: predictions and 95% prediction limits (neural network)**



**Figure 5: Autoregressive time series: predictions and 95% prediction limits (linear regression model)**

## 5. CONCLUSIONS

It is shown that bootstrap methodology can be used to calculate multi-step predictions of neural networks and their associated prediction limits.

Effective algorithms are essential for network training, since the bootstrap approach is very computer intensive. In order to derive prediction limits, a minimum of 1000 time series should be resampled from the observed time series. Considering the ever increasing speed of computers and the development of more effective algorithms for network training, this is not viewed as a real problem.

Two examples are given. In both cases, neural networks, as well as linear regression models, are used to model the time series, and predictions and prediction limits are calculated. The results compare reasonably well. Further research is required to establish whether the bootstrap results can be improved by, for instance, increasing the number of bootstrap replications.

## 6. REFERENCES

- [1] **Box, G.E.P., Jenkins, G.M. and Reinsel, G.C.** (1994) *Time Series Analysis: Forecasting and Control*. Third Edition. Prentice-Hall, New Jersey.
- [2] **Carlstein, E.** (1986). The use of subseries values for estimating the variance of a general statistic from a stationary sequence. *Annals of Statistics*, Vol. 14, 1171-1194.
- [3] **Chatterjee, S.** (1986) Bootstrapping ARMA Models: Some Simulations. *IEEE Transactions on Systems, Man and Cybernetics*. SMC-16, 294-299.
- [4] **Efron, B.** (1979) Bootstrap Methods: Another look at the jackknife. *Annals of Statistics*. Vol. 7, 1-26.
- [5] **Efron, B. and Tibshirani, R.J.** (1986). Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical Science*. Vol. 1, 54-77.
- [6] **Faraway, J. and Chatfield, C.** (1998). Time series forecasting with neural networks: a comparative study using the airline data. *Appl. Statist.* Vol. 47, Part 2, 231-250.
- [7] **Hall, P., Horowitz, J.L. and Jing, B.** (1995). On blocking rules for the bootstrap with dependent data. *Biometrika*, Vol. 82, 561-574.
- [8] **Hill, T., O'Connor, M. and Remus, W.** (1996). Neural Network Models for Time Series Forecasts. *Management Science*, Vol. 42, No. 7, 1082-1092.
- [9] **Holbert, D. and Son, M.S.** (1986). Bootstrapping a time series model: Some empirical results. *Communications in Statistics*. Vol. 15, 3669-3691.
- [10] **Kuan, C. and Liu, T.** (1995). Forecasting exchange rates using feedforward and recurrent neural networks. *Journal of Applied Econometrics*, Vol. 10, 347-364.
- [11] **Lin, J.L. and Granger, C.W.J.** (1994). Forecasting from Non-linear Models in Practice. *Journal of Forecasting*, Vol. 13, 1-9.
- [12] **Liu, R.Y.** (1988). Bootstrap procedures under some non-i.i.d. models. *Annals of Statistics*, Vol. 16, 1697-1708.
- [13] **Masarotto G.** (1990) Bootstrap prediction intervals for autoregressions. *International Journal of Forecasting*, Vol 6, 229-239.
- [14] **Politis, D.N. Romano, J.P. and Wolf, M.** (1997). Subsampling for heteroskedastic time series. *Journal of Econometrics*. Vol. 81, 281-317.
- [15] **Souza, R.C. and Neto, A.C.** (1996) A Bootstrap Simulation Study in ARMA(p,q)

- Structures. *Journal of Forecasting*. Vol. 15, 343-353.
- [16] **Swanepoel, J.W.H.** (1990). A Review of Bootstrap Methods. *South African Statistical Journal*. Vol.24, 1-34.
- [17] **Zhang, G., Patuwo, B.E. and Hu, M.Y.** (1998). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, Vol. 14, 35-62.

## 7. ACKNOWLEDGEMENT

The authors wish to thank Dr Gerard van Harmelen, Electricity Demand, Marketing, Eskom for supplying the electricity data.

Thanks is also due to the National Research Foundation for financial support during the preparation of this article.

