DEPARTAMENTO DE ECONOMIA
PUC-RIO

TEXTO PARA DISCUSSÃO
Nº. 445

STATISTICAL METHODS FOR MODELLING NEURAL NETWORKS

MARCELO C. MEDEIROS
TIMO TERÄSVIRTA

SETEMBRO 2001

Statistical Methods for Modelling Neural Networks *

Marcelo C. Medeiros
Dept. of Economics, Catholic University of Rio de Janeiro

Timo Teräsvirta
Dept. of Economic Statistics, Stockholm School of Economics.

September 17, 2001

Abstract

In this paper modelling time series by single hidden layer feedforward neural network models is considered. A coherent modelling strategy based on statistical inference is discussed. The problems of selecting the variables and the number of hidden units are solved by using statistical model selection criteria and tests. Misspecification tests for evaluating an estimated neural network model are considered. Forecasting with neural network models is discussed and an application to a real time series is presented.

Keywords: Model misspecification, neural computing, nonlinear forecasting, nonlinear time series, smooth transition autoregression, sunspot series, threshold autoregression

JEL Classification Codes: C22, C51, C52, C61, G12

Acknowledgments: This research has been supported by the Tore Browaldh's Foundation. The research of the first author has been partially supported by CNPq. A part of the work was carried out during the visits of the first author to the Department of Economic Statistics, Stockholm School of Economics and the second author to the Department of Economics, PUC-Rio. The hospitality of these departments is gratefully acknowledged. Material from this paper has been presented at the Fifth Brazilian Conference on Neural Networks, Rio de Janeiro, April 2001. We wish to thank participants at the conference as well as Dick van Dijk and Birgit Strikholm for useful remarks. The responsibility for any errors or shortcomings in the paper remains ours.

*Address for correspondence: Timo Teräsvirta, Department of Economic Statistics, Stockholm School of Economics, Box 6501, SE-113 83, Stockholm, Sweden. E-mail: timo.terasvirta@hhs.se.

1 Introduction

Artificial neural network (ANN) models form an important class of nonlinear models that has attracted considerable attention in many fields of application. The use of these models in applied work is generally motivated by a mathematical result stating that under mild regularity conditions, a relatively simple ANN model is capable of approximating any Borel-measurable function to any given degree of accuracy; see, for example, Funahashi (1989), Cybenko (1989), Hornik, Stinchombe, and White (1989,1990), White (1990), or Gallant and White (1992). How to specify such a model, that is, how to find the right combination of parameters and variables, is a central topic in the ANN literature and has been considered in a large number of books such as Bishop (1995), Ripley (1996), Fine (1999), Haykin (1999), and Reed and Marks II (1999), and articles. Many popular specification techniques are "general-to-specific" or "top-down" procedures: the investigator begins with a large model and applies appropriate algorithms to reduce the number of parameters using a predetermined stopping-rule. Such algorithms usually do not rely on statistical inference.

In this paper, we propose a coherent modelling strategy for simple single hidden-layer feedforward ANN time series models. These models discussed here are univariate, but adding exogenous regressors to them does not pose problems. The results presented here are a summary of some of the findings described in detail in Medeiros, Teräsvirta and Rech (2001). The difference between our strategy and the general-to-specific approaches is that ours works in the opposite direction, from specific to general. We begin with a small model and expand that according to a set of predetermined rules. This way we hope to avoid the estimation of excessive large models and keep the computational effort relatively limited. The general idea is to make use statistical methods in building the model (choosing the neural network architecture) and evaluating the estimated model. More recently, Anders and Korn (1999) presented a strategy that shares certain features with our procedure. Swanson and White (1995,1997a,b) also discussed and applied a specific-to-general strategy that deserves mention here.

The plan of the paper is as follows. Section 2 describes the model and Section 3 considers identification. A model specification strategy, consisting of specification, estimation, and evaluation of the model is described in Section 4. Section 5 discusses the issues related to forecasting from

neural network models. An empirical application is presented in Section 6. Section 7 contains concluding remarks.

2 The Autoregressive Neural Network Model

The AutoRegressive Neural Network (AR-NN) model with a single hidden layer and no feedback is defined as

$$y_t = G(\mathbf{x}_t; \boldsymbol{\psi}) + \varepsilon_t = \boldsymbol{\alpha}'\tilde{\mathbf{x}}_t + \sum_{i=1}^{h} \lambda_i F(\tilde{\boldsymbol{\omega}}_i'\mathbf{x}_t - \beta_i) + \varepsilon_t \tag{1}$$

where $G(\mathbf{x}_t; \boldsymbol{\psi})$ is a nonlinear function of the variables $\mathbf{x}_t$ with parameter vector $\boldsymbol{\psi} \in \mathbb{R}^{(q+2)h+q+1}$ defined as $\boldsymbol{\psi} = [\boldsymbol{\alpha}', \lambda_1, \ldots, \lambda_h, \tilde{\boldsymbol{\omega}}_1', \ldots, \tilde{\boldsymbol{\omega}}_h', \beta_1, \ldots, \beta_h]'$. The vector $\tilde{\mathbf{x}}_t \in \mathbb{R}^{q+1}$ is defined as $\tilde{\mathbf{x}}_t = [1, \mathbf{x}_t']'$, where $\mathbf{x}_t \in \mathbb{R}^q$ is a vector of lagged values of y_t and/or some exogenous variables. The function, or sigmoid, $F(\tilde{\boldsymbol{\omega}}_i'\mathbf{x}_t - \beta_i)$ is the logistic function

$$F(\tilde{\boldsymbol{\omega}}_i'\mathbf{x}_t - \beta_i) = (1 + e^{-(\tilde{\boldsymbol{\omega}}_i'\mathbf{x}_t - \beta_i)})^{-1} \tag{2}$$

where $\tilde{\boldsymbol{\omega}}_i = [\tilde{\omega}_{1i}, \ldots, \tilde{\omega}_{qi}]' \in \mathbb{R}^q$ and $\beta_i \in \mathbb{R}$, and the linear combination of these functions or hidden units in (1) forms the hidden layer. Instead of viewing (1) as an approximation to an unknown function representing the data-generator, as is customary in the neural network literature, we follow the tradition in mathematical statistics and assume that model (1) is the true data-generating process. Function $G(\mathbf{x}_t; \boldsymbol{\psi})$ is thus the conditional mean of the process. Furthermore, $\{\varepsilon_t\}$ is a sequence of independently normally distributed random variables with zero mean and variance σ^2. The normality assumption enables us to define the log-likelihood function, which is required for statistical inference, although this assumption can be relaxed.

It should be pointed out that the assumption of (1) being a true model is a technical assumption required in model building. As we do not know the truth, it is clear that the AR-NN model represents just one attempt to approximate the process that generates the observations. As our modelling approach is sequential, statistical inference is at every stage conditional on previous decisions made in the sequence. Thus, for example, it is clear that the standard deviation estimates reported in the empirical example of Section 6 underestimate the uncertainty of the parameter

estimates.

3 Identifiability of the AR-NN model

Our aim is to apply statistical inference to building AR-NN models. This being the case, the question of identifiability of model (1) arises. This model is, in principle, neither globally nor locally identified, which means that there exist two or more sets of parameters $\boldsymbol{\psi}$ (two or more different models) that correspond to the same distribution of $(y, \mathbf{x})$. In other words, there exist models that cannot be distinguished from each other on the basis of data. A consequence of this is that we cannot estimate the parameters of an unidentified model consistently: the estimates cannot be expected to converge to their true values in probability as the number of observations approaches infinity. As a result we do not have asymptotic distribution results for our parameter estimators, which precludes a statistical approach to neural network modelling. The following three features of model (1) imply non-identifiability. The first one is the exchangeability property of the AR-NN model. The value in the likelihood function of the model remains unchanged if we permute the hidden units. Permutation results in $h!$ different models that are indistinguishable from each other, and the log-likelihood function has $h!$ equal local maxima. The second feature is that $F(x) = 1 - F(-x)$ in (2). This yields two observationally equivalent parametrizations for each hidden unit. The final problem is the potential presence of irrelevant hidden units in the model. If model (1) has hidden units such that $\lambda_i = 0$ for at least one i, the parameters $\tilde{\boldsymbol{\omega}}_i$ and β_i remain unidentified. Conversely, and this is the other side of the same coin, if $\tilde{\boldsymbol{\omega}}_i = \mathbf{0}$ then any values of λ_i and β_i lead to the same maximum of the likelihood function.

The first problem is solved by imposing, say, the restrictions $\beta_1 \leq \cdots \leq \beta_h$ or $\lambda_1 \geq \cdots \geq \lambda_h$. The second source of unidentification can be circumvented, for example, by imposing the restrictions $\tilde{\omega}_{1i} > 0$, $i = 1, \ldots, h$. To remedy the third problem it is necessary to ensure that the model contains no irrelevant hidden units. This difficulty is dealt with by applying statistical inference in model specification; see Section 4. For further discussion of the identifiability of ANN models see, for example, Sussman (1992), Kurková and Kainen (1994), Hwang and Ding (1997), and Anders and Korn (1999).

4 Strategy for Building AR-NN Models

4.1 Three Stages of Model Building

As mentioned in the Introduction, our aim is to construct a coherent strategy for building AR-NN models using statistical inference. The structure or architecture of an AR-NN model has to be determined from the data, a problem that has received plenty of attention in the neural network literature. It involves selecting the lags or variables to be included in the model and determining the number of hidden units. Finding the correct number of hidden units is particularly important because, as indicated above, selecting too many neurons yields an unidentified model. In this work, the lag structure or the variables included in the model are determined using well-known variable selection techniques. This *specification* stage of AR-NN modelling also requires *estimation* because we suggest choosing the hidden units sequentially. After estimating a model with h hidden units we shall test it against the one with $h+1$ hidden units (without at this stage estimating the latter) and continue until the first acceptance of a null hypothesis. What follows thereafter is *evaluation* of the final estimated model. Neural network models are typically evaluated out-of-sample, but our statistical approach allows us to derive in-sample misspecification tests for the purpose. These tests do not replace out-of-sample evaluation, in particular as neural network models for time series are a forecasting tool. They are rather complements to out-of-sample evaluation techniques.

We begin the discussion of our modelling strategy by considering variable selection. After dealing with that problem we turn to parameter estimation. Finally, after discussing statistical inference for selecting the hidden units and after briefly discussing our in-sample model evaluation tools we put the pieces together and present the whole modelling strategy.

4.2 Variable Selection

In this paper the idea is to first select the variables and find an appropriate number of hidden units thereafter, conditionally on the variables selected. In pruning neural network models these two selection problems are typically solved simultaneously using an appropriate decision rule. Here variable selection is carried out by linearizing the model and applying well-known techniques of linear variable selection to the linearized version. This keeps computational cost to a minimum.

We follow a simple procedure proposed in Rech, Teräsvirta and Tschernig (2001). The first step consists of approximating function $G(\mathbf{x}_t;\boldsymbol{\psi})$ in (1) by a general k-th order polynomial. By the Stone-Weierstrass theorem, the approximation can be made arbitrarily accurate if some mild conditions, such as the parameter space $\boldsymbol{\Psi}$ being compact, are imposed on function $G(\mathbf{x}_t;\boldsymbol{\psi})$. Thus the AR-NN model, itself a universal approximator, is approximated by another function. We have

$$\begin{aligned} G(\mathbf{x}_t;\boldsymbol{\psi}) = \boldsymbol{\pi}'\tilde{\mathbf{x}}_t + \sum_{j_1=1}^{q}\sum_{j_2=j_1}^{q}\theta_{j_1 j_2}x_{j_1,t}x_{j_2,t} \\ + \cdots + \sum_{j_1=1}^{q}\cdots\sum_{j_k=j_{k-1}}^{q}\theta_{j_1\ldots j_k}x_{j_1,t}\cdots x_{j_k,t} + R(\mathbf{x}_t;\boldsymbol{\psi}), \end{aligned} \tag{3}$$

where $R(\mathbf{x}_t;\boldsymbol{\psi})$ is the approximation error that can be made negligible by choosing k sufficiently high. The $\theta' s$ are parameters, and $\boldsymbol{\pi} \in \mathbb{R}^{q+1}$ is a vector of parameters. The right-hand side of (3) is linear in parameters, and its form is independent of the number of hidden units in (1).

In equation (3), every product of variables involving at least one redundant variable has the regression coefficient zero because the approximation to the true model does not contain any of those variables. The redundant variables can be found by using this property of (3). In order to do that, we first regress y_t on all variables in (3) and compute the value of a model selection criterion (MSC), AIC (Akaike 1974) or SBIC (Schwarz 1978, Rissanen 1978) for example. After doing that, we remove one variable from the original model and regress y_t on all the remaining terms in the corresponding polynomial and again compute the value of the MSC. This procedure is repeated by omitting each variable in turn. We continue by simultaneously omitting two regressors of the original model and proceed in that way until the polynomial is of a function of a single regressor and, finally, just a constant. Having done that, we choose the combination of variables that yields the lowest value of the MSC. This amounts to estimating $\sum_{i=1}^{q}\frac{q!}{i!(p-i)!}+1$ linear models by ordinary least squares (OLS). This technique has the favourable property that it can be successfully applied even in large samples when available nonparametric model selection techniques, another alternative in this situation, become computationally infeasible.

4.3 Parameter Estimation

4.3.1 Maximum Likelihood Estimation

Selecting the number of hidden units requires estimation of neural network models, and we now turn to this problem. We estimate the parameters of our AR-NN model by maximum likelihood making use of the normality assumption of ε_t. It may be argued that maximum likelihood estimation of neural network models is most likely to lead to convergence problems, and that penalizing the log-likelihood function is a necessary precondition for satisfactory results. Many estimation methods applied in neural network models are based on penalizing the likelihood one way or the other; see, for example, Fine (1999, Chapter 6).

Two things can be said in favour of maximum likelihood here. First, model building proceeds from small to large models, so that estimation of unidentified or nearly unidentified models which is a major reason for the need to penalize the log-likelihood, is avoided. Second, the initial values of the parameter estimates are chosen carefully, and we discuss the details of this in Section 4.3.2.

For estimation purposes it is useful to reparametrize the logistic function (2) as

$$F\left(\gamma_i\left(\boldsymbol{\omega}_i'\mathbf{x}_t - c_i\right)\right) = \left(1 + e^{-\gamma_i(\boldsymbol{\omega}_i'\mathbf{x}_t - c_i)}\right)^{-1} \tag{4}$$

where $\gamma_i > 0$, $i = 1, \ldots, h$, and $\|\boldsymbol{\omega}_i\| = 1$ with

$$\omega_{i1} = \sqrt{1 - \sum_{j=2}^{q} \omega_{ij}^2} > 0, i = 1, \ldots, h. \tag{5}$$

The parameter vector $\boldsymbol{\psi}$ of model (1) becomes

$$\boldsymbol{\psi} = [\boldsymbol{\alpha}', \lambda_1, \ldots, \lambda_h, \gamma_1, \ldots, \gamma_h, \omega_{12}, \ldots, \omega_{1q}, \ldots, \omega_{h2}, \ldots, \omega_{hq}, c_1, \ldots, c_h]'.$$

In this case the first two identifying restrictions discussed in Section 3 can be defined as follows. First, $c_1 \leq \cdots \leq c_h$ or $\lambda_1 \geq \cdots \geq \lambda_h$ and, second, $\gamma_i > 0, i = 1, \ldots, h$.

The AR-NN model is similar to many linear or nonlinear time series models in that the information matrix of the logarithmic likelihood function is block-diagonal in such a way that we

can concentrate the likelihood and first estimate the parameters of the conditional mean. Thus conditional maximum likelihood is equivalent to nonlinear least squares. Under mild regularity conditions that include the requirement that the AR-NN process is weakly stationary, it is possible to prove consistency and asymptotic normality of the maximum likelihood estimator $\boldsymbol{\psi}$; see Medeiros et al. (2001) for details. In the estimation, the use of algorithms such as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) or the Levenberg-Marquardt algorithms is strongly recommended. See, for example, Bertsekas (1995) for details of optimization algorithms or Fine (1999, Chapter 5) for ones especially applied to the estimation of NN models.

4.3.2 Starting-values

Many iterative optimization algorithms are sensitive to the choice of starting-values, and this is certainly so in the estimation of AR-NN models. Besides, an AR-NN model with h hidden units contains h parameters, γ_i, $i = 1, ..., h$, that are not scale-free. Our first task is thus to rescale the input variables in such a way that they have the standard deviation equal to unity. In the univariate AR-NN case, this simply means normalizing y_t. This, together with the fact that $\|\omega_h\| = 1$, gives us a basis for discussing the choice of starting-values of $\gamma_i, i = 1, ..., h$. Another advantage of this is that normalization generally facilitates numerical optimization. When all variables have the same standard deviation the corresponding parameters are more likely to be of the same order of magnitude. This, however, does not apply to γ_i, $i = 1, ..., h$, as these parameters control the "slope" of the logistic functions.

Suppose now that we have estimated an AR-NN model with $h - 1$ hidden units and want to estimate one with h units. A natural choice of initial values for the estimation of parameters in the model with h neurons is to use the final estimates for the parameters in the first $h - 1$ hidden neurons and the linear unit. The starting-values for the parameters in the hth hidden unit can be found by making use of the fact that model (1) is linear in parameters $\boldsymbol{\alpha}$ and λ_i, $i = 1, ..., h$. The idea is to construct a grid of parameter values $\gamma_h, \theta_h, \boldsymbol{\omega}_h$ and c_h, estimate $\boldsymbol{\alpha}$ and λ_h conditionally on the other parameters and, finally, choose the set of parameter values that minimizes the residual sum of squares. For details, see Medeiros et al. (2001).

4.4 Determining the Number of Hidden Units

The number of hidden units included in a neural network model is usually determined from the data. In this work the hidden units are selected using a sequence of statistical tests. How to carry out the testing is not completely straightforward due to the identification problem already mentioned in Section 3. The idea is to circumvent the identification problem in a way that enables us to control the significance level of the tests in the sequence and thus also the overall significance level of the procedure. Following Teräsvirta and Lin (1993) we propose a test that is repeated until the first acceptance of the null hypothesis. Assume now that our AR-NN model (4) contains $h+1$ hidden units and write it as follows

$$y_t = \boldsymbol{\alpha}'\tilde{\mathbf{x}}_t + \sum_{i=1}^{h} \lambda_i F(\gamma_i(\boldsymbol{\omega}_i'\mathbf{x}_t - c_i)) + \lambda_{h+1} F(\gamma_{h+1}(\boldsymbol{\omega}_{h+1}'\mathbf{x}_t - c_{h+1})) + \varepsilon_t. \tag{6}$$

Suppose now that we have accepted the hypothesis of model (6) containing h hidden units and want to test for the $(h+1)^{\text{th}}$ hidden unit. The appropriate null hypothesis is

$$\mathrm{H}_0 : \gamma_{h+1} = 0 \tag{7}$$

because this choice makes $F(\gamma_i(\boldsymbol{\omega}_i'\mathbf{x}_t - c_i)) \equiv$ constant $(=1/2)$ so that the $(h+1)^{\text{th}}$ hidden unit vanishes. The alternative hypothesis $\mathrm{H}_1 : \gamma_{h+1} > 0$.

We assume that under (7) maximum likelihood estimators of the parameters are asymptotically normal (the null hypothesis being valid implies that we only estimate the model with h hidden units). Model (6) is only identified under the alternative so that, as discussed above, the standard asymptotic inference is not available. This problem is circumvented as in Luukkonen, Saikkonen and Teräsvirta (1988) by expanding the $(h+1)^{\text{th}}$ hidden unit into a Taylor series around the $\gamma_{h+1} = 0$. Using a third-order Taylor expansion, rearranging and merging terms results in the following local approximation to (6):

$$y_t = \boldsymbol{\pi}'\tilde{\mathbf{x}}_t + \sum_{i=1}^{h} \lambda_i F(\gamma_i(\boldsymbol{\omega}_i'\mathbf{x}_t - c_i)) + \sum_{i=1}^{q}\sum_{j=i}^{q} \theta_{ij} x_{i,t} x_{j,t} + \sum_{i=1}^{q}\sum_{j=i}^{q}\sum_{k=j}^{q} \theta_{ijk} x_{i,t} x_{j,t} x_{k,t} + \varepsilon_t^* \tag{8}$$

where $\varepsilon_t^* = \varepsilon_t + \lambda_{h+1}R(\mathbf{x}_t)$; $R(\mathbf{x}_t)$ is the remainder. It can be shown that $\theta_{ij} = \gamma_{h+1}^2\tilde{\theta}_{ij}$, $\tilde{\theta}_{ij} \neq 0$, $i = 1, \ldots, q$; $j = i, \ldots, q$, and $\theta_{ijk} = \gamma_{h+1}^3\tilde{\theta}_{ijk}$, $\tilde{\theta}_{ijk} \neq 0$, $i = 1, \ldots, q$; $j = i, \ldots, q$, $k = j, \ldots, q$. Thus the null hypothesis $\mathrm{H}_0' : \theta_{ij} = 0$, $i = 1, \ldots, q$; $j = i, \ldots, q$, $\theta_{ijk} = 0$, $i = 1, \ldots, q$; $j = i, \ldots, q$; $k = j, \ldots, q$. Note that under $\mathrm{H}_0' : \varepsilon_t^* = \varepsilon_t$, so that the properties of the error process are not affected by the remainder when the null hypothesis holds. Assuming $E|x_{i,t}|^\delta < \infty$, $i = 1, \ldots, q$, for some $\delta > 6$, it is possible to derive a score or Lagrange multiplier type statistic for testing H_0'. By trading off information about the structure of the $(h+1)$th hidden unit we in return obtain a simple test. It can be carried out in stages as follows:

1. Estimate model (4) with h hidden units. If the sample size is small and the model thus difficult to estimate, numerical problems in applying the maximum likelihood algorithm may lead to a solution such that the residual vector is not precisely orthogonal to the gradient matrix of $G(\mathbf{x}_t; \hat{\boldsymbol{\psi}})$. This has an adverse effect on the empirical size of the test. To circumvent this problem, we regress the residuals $\hat{\varepsilon}_t$ on $\hat{\mathbf{h}}_t$, the elements of the score vector of the model with h hidden units, and compute the sum of squared residuals $SSR_0 = \sum_{t=1}^T \tilde{\varepsilon}_t^2$. The new residuals $\tilde{\varepsilon}_t$ are orthogonal to $\hat{\mathbf{h}}_t$.

2. Regress $\tilde{\varepsilon}_t$ on $\hat{\mathbf{h}}_t$ and $\hat{\boldsymbol{\nu}}_t$, the vector containing as its elements all combinations $x_{i,t}x_{j,t}$ and $x_{i,t}x_{j,t}x_{k,t}$. Compute the sum of squared residuals $SSR_1 = \sum_{t=1}^T \hat{v}_t^2$.

3. Compute the χ^2 statistic

$$LM_{\chi^2}^{hn} = T\frac{SSR_0 - SSR_1}{SSR_0}, \tag{9}$$

   or the F version of the test

$$LM_F^{hn} = \frac{(SSR_0 - SSR_1)/m}{SSR_1/(T-n-m)}. \tag{10}$$

Under H_0, $LM_{\chi^2}^{hn}$ has an asymptotic χ^2 distribution with m degrees of freedom and LM_F^{hn} is approximately F-distributed with m and $T-n-m$ degrees of freedom, where $n = (q+2)h + p + 1$. The F-version is recommended in small samples where the χ^2 test can be severely size-distorted (the empirical significance level exceeds the nominal one). See Medeiros et al. (2001) for details and more discussion.

4.5 Evaluation of the Estimated Model

Although the AR-NN model is a very flexible nonlinear model and motivated by as being a universal approximator, there exist situations in practice where the model cannot adequately capture the nonlinear dynamic structure it is expected to approximate. Evaluating the estimated model is thus important. We propose two in-sample misspecification tests for this purpose. The first one tests for the instability of the parameters. Under the alternative the parameters are assumed to change smoothly and deterministically over time. The change is parameterized much in the same way as the hidden units are. We consider a model with time-varying parameters defined as

$$y_t = \tilde{G}(\mathbf{x}_t; \boldsymbol{\psi}, \tilde{\boldsymbol{\psi}}) + \varepsilon_t = \tilde{\boldsymbol{\alpha}}'(t)\tilde{\mathbf{x}}_t + \sum_{i=1}^{h} \left\{ \tilde{\lambda}_i(t) F(\gamma_i(\boldsymbol{\omega}_i'\mathbf{x}_t - c_i)) \right\} + \varepsilon_t, \tag{11}$$

where

$$\tilde{\boldsymbol{\alpha}}(t) = \boldsymbol{\alpha} + \breve{\boldsymbol{\alpha}} F\left(\zeta(t-\eta)\right), \tag{12}$$

and

$$\tilde{\lambda}_i(t) = \lambda_i + \breve{\lambda}_i F\left(\zeta(t-\eta)\right), i = 1, \ldots, h. \tag{13}$$

The function F in (12) and (13) is defined as in (2), and $\zeta > 0$. The parameter vector $\boldsymbol{\psi}$ is defined as before, and $\tilde{\boldsymbol{\psi}} = [\breve{\boldsymbol{\alpha}}, \breve{\lambda}_1, \ldots, \breve{\lambda}_h, \zeta, \eta]'$. The parameter ζ controls the smoothness of the monotonic change in the autoregressive parameters. When $\zeta \to \infty$, equations (11)–(13) represent a model with a single structural break at $t = \eta$. The null hypothesis is $\zeta = 0$, and it is seen from (12) and (13) and the model is not identified under the null hypothesis. Deriving the test is again based on expanding $F\left(\zeta(t-\eta)\right)$ into a Taylor series around the null point expansion of $\zeta = 0$. For details of derivation of the test, see Medeiros et al. (2001). When the model is assumed not to contain any hidden units: $\lambda_i(t) \equiv 0$, $i = 1, \ldots, h$, the test collapses into the parameter constancy test in Lin and Teräsvirta (1994).

The test of no serial correlation in the errors is an application of the results in Eitrheim and Teräsvirta (1996) and Godfrey (1988, pp. 112–121). The alternative hypothesis is that the error process

$$\varepsilon_t = \boldsymbol{\pi}'\boldsymbol{\nu}_t + u_t, \tag{14}$$

where $\boldsymbol{\pi}' = [\pi_1, \ldots, \pi_r]$ is a parameter vector, $\boldsymbol{\nu}_t' = [\varepsilon_{t-1}, \ldots, \varepsilon_{t-r}]$, and $u_t \sim \text{NID}(0, \sigma^2)$. The null hypothesis $\text{H}_0 : \boldsymbol{\pi} = \mathbf{0}$. In this case there is no identification problem, and the test is a standard Lagrange multiplier test. It can be carried out in stages like the test for the number of hidden units; see Medeiros et al. (2001) for details. It may be pointed out that the Ljung-Box test or its asymptotically equivalent counterpart, the Box-Pierce test, both recommended for use in connection with neural networks models by Zapranis and Refenes (1999), are not available. Their asymptotic null distribution is unknown when the estimated model is an AR-NN model.

4.6 Modelling strategy

At this point we are ready to combine the above statistical ingredients into a coherent modelling strategy. We first define the potential variables (lags) and select a subset of them applying the variable selection technique considered in Section 4.2. After selecting the variables we select the number of hidden units sequentially. We begin testing linearity against a single hidden unit as described in Section 4.4 at significance level α. The model under the null hypothesis is simply a linear AR(p) model. If the null hypothesis is not rejected, the AR model is accepted. In case of a rejection, an AR-NN model with a single unit is estimated and tested against a model with two hidden units at the significance level $\alpha\varrho$, $0 < \varrho < 1$. Another rejection leads to estimating a model with two hidden units and testing it against a model with three hidden neurons at the significance level $\alpha\varrho^2$. The sequence is terminated at the first acceptance of the null hypothesis. By letting the significance level of the tests to converge to zero when the number of steps in the sequence approaches infinity we avoid excessively large models that are difficult to estimate and may contain redundant units. The parsimony of the model is controlled by the parameters α and ϱ. Setting, for example, $\alpha = 0.1$ and $\rho = 0.5$ is often a reasonable choice. In the empirical example of Section 6 the results are quite robust to the choice of the original significance level and ρ.

In following the above path we have assumed that all hidden neurons contain the variables that are originally selected to the AR-NN model. We may also augment the strategy by separately choosing a subset of variables for each hidden unit from the set originally selected as discussed in 4.2. This can be done without re-estimating the model; for details see Medeiros et al. (2001).

4.7 Discussion and comparisons

There exist other bottom-up approaches in the literature. Swanson and White (1995,1997a,b) apply SBIC model selection criterion as follows. They start with a linear model, adding potential variables to it until SBIC indicates that the model cannot be further improved. Then they estimate models with a single hidden unit and select regressors sequentially to it one by one unless SBIC shows no further improvement. Next Swanson and White add another hidden unit and proceed by adding variables to it. The selection process is terminated when SBIC indicates that no more hidden units should be added or when a predetermined maximum number of hidden units has been reached. This modelling strategy can be termed fully sequential. A problem of this technique is the use SBIC. How it should be applied when choosing between an identified and an unidentified model, a situation that occurs frequently and was discussed in Section 4.4, is not clear. There is also a strong possibility of estimating at least one unidentified or at least nearly unidentified model before terminating the sequence.

Anders and Korn (1999) essentially adopt the procedure of Teräsvirta and Lin (1993) described in Section 4.4 for selecting the number of hidden units. After estimating the largest model they suggest proceeding from general-to-specific by sequentially removing those variables from hidden units whose parameter estimates have the lowest (t-test) p-values. Note that this presupposes parameterizing the hidden units as in (2), not as in (4) and (5).

The strategy Swanson and White applied is computationally the most intensive one, as the number of steps involving an estimation of a neural networks model is large. Our procedure is in this respect considerably less demanding. The difference between our scheme and the Anders and Korn one is that in our strategy, variable selection does not require estimation of neural networks models because it is wholly based on LM type tests (the model is only estimated under the null hypothesis). Furthermore, there is a possibility of omitting certain potential variables before even estimating neural network models.

Like ours, the Swanson and White strategy is truly sequential: the modeller proceeds by considering nested models. The difference lies in how to compare two nested models in the sequence. Swanson and White apply SBIC, which, as we indicated, is not unproblematic, whereas Anders and Korn and we use LM type tests.

5 Forecasting

Forecasting with nonlinear models is numerically more involved than carrying a similar exercise with linear models, see Tong (1990, Chapter 6) and Granger and Teräsvirta (1993, Section 8.1) for general reviews. We briefly illustrate the forecasting issues in the neural network context. Consider the AR-NN model (1) with sigmoid activation functions (2). Suppose we want to forecast y_{T+k}, $k \geq 1$, at time T. Assuming that the loss function of the forecaster is quadratic, the optimal forecast is the conditional mean

$$\hat{y}_{T+k|T} = E(y_{T+k}|\mathcal{F}_T), \tag{15}$$

where $\mathcal{F}_T$ contains all the available information at time T. In this case the information can be expressed using q lags of y_T: $\mathcal{F}_T = \mathbf{x}_T = [y_{T-1}, y_{T-2}, \ldots, y_{T-q}]'$, in (15). Forecasting one period ahead poses no problem, but nonlinearity of the AR-NN model affects forecasting for $k \geq 2$, because $\hat{y}_{T+k|T}$ is a nonlinear function of $\varepsilon_{T+1}, \varepsilon_{T+2}, \ldots, \varepsilon_{T+k-1}$.

Consider the case $k = 2$. We have

$$\hat{y}_{T+2|T} = \boldsymbol{\alpha}' E(\tilde{\mathbf{x}}_{T+2|T}) + \sum_{i=1}^{h} \lambda_i E\{(1 + e^{-\gamma_i(\boldsymbol{\omega}_i' \mathbf{x}_{t+2} - c_i)})\}, \tag{16}$$

where $E(\tilde{\mathbf{x}}_{T+2|T}) = [1, E(\mathbf{x}_{T+2}|\mathcal{F}_T)']'$, with $\mathbf{x}_{T+2} = [\hat{y}_{T+1|T} + \varepsilon_{T+1}, y_T, \ldots, y_{T-q+1}]'$, and $E(\mathbf{x}_{T+2}) = [\hat{y}_{T+1|T}, y_T, \ldots, y_{T-q+}]'$. In order to obtain (16), we have to compute the h conditional expectations

$$E\left\{\left(1 + e^{-\gamma_i(\boldsymbol{\omega}_i' \mathbf{x}_{T+2} - c_i)}\right)^{-1} |\mathcal{F}_t\right\} = \int_{-\infty}^{\infty} \left(1 + e^{-\gamma_i(\boldsymbol{\omega}_i' \mathbf{x}_{T+2} - c_i)}\right)^{-1} f(\varepsilon_{T+1}) d\varepsilon_{T+1}, \ i = 1, \ldots, h, \tag{17}$$

where $f(z)$ is the density of z. Expectations (17) may be calculated using numerical integration. However, for $k > 2$, multiple integrals have to be evaluated, so that numerical integration becomes tedious. It is easier to use Monte Carlo simulation and approximate

$$E\{F(\gamma_i(\boldsymbol{\omega}_i' \mathbf{x}_{T+2} - c_i))|\mathcal{F}_t\} \approx \frac{1}{M} \sum_{m=1}^{M} F(\gamma_i(\boldsymbol{\omega}_i' \mathbf{x}_{T+2}^{(m)} - c_i)), \tag{18}$$

where $\mathbf{x}_{T+2}^{(m)} = [\hat{y}_{T+1|T} + \varepsilon_{T+1}^{(m)}, y_T, \ldots, y_{T-q+1}]'$ and $\varepsilon_{T+1}^{(m)}$ is a random draw from the distribution of ε_t. Alternatively, if one does not want to assume a specific error distribution, it is possible to use

a bootstrap, that is, draw the values $\varepsilon_{T+1}^{(m)}$ randomly, with replacement, from the set of residuals $\{\hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_T\}$ of the estimated AR-NN model. The forecasts $y_{T+k|T}$, $k > 2$, are obtained in the same way.

The numerical approach yields M forecasts, and it is even more informative to report the whole forecast density. A good way of summing it up is to use highest density regions (HDR); see Hyndman (1995,1996). An HDR is defined as

$$HDR_\alpha = \{z : g(z) \geq g_\alpha\}, \tag{19}$$

where $g(z)$ is the density of z and g_α is such that $P(z \in HDR_\alpha) = 1 - \alpha$. In other words, HDR_α is a subset of the support of z such that the value of the density at every point of the subset is at least equal to g_α. As example of the use of HDRs follows in Section 6. In our case, HDRs are straightforward to compute when the Monte Carlo or the bootstrap methods are used to compute the point forecasts.

Forecasting with AR-NN models also requires human control because the parameters of the model have to be estimated. The estimation algorithm may sometimes converge to a local maximum such that the estimated model yields unrealistic forecasts. Swanson and White (1995) apply an "insanity filter": if the forecast exceeds the maximum value or lies below the minimum value hitherto observed (they are concerned with sequential forecasting) in the stationary series to be predicted, it is replaced by the sample mean of the series. As the authors put it, ignorance is substituted for craziness.

6 Empirical Example: Annual Sunspot Numbers, 1700–2000

In this example we build an AR-ANN model for the annual sunspot numbers over the period 1700–1979 and forecast with the estimated model up until the year 2000. The series, consisting of the years 1700–2000, was obtained from the National Geophysical Data Center web page. [1] The sunspot numbers are a heavily modelled nonlinear time series: for a neural network example see Weigend, Huberman and Rumelhart (1992). In this work we adopt the square-root transformation

[1] http://www.ngdc.noaa.gov/stp/SOLAR/SSN/ssn.html

of Ghaddar and Tong (1981) and Tong (1990, p. 420). The transformed observations have the form $y_t = 2\left[\sqrt{(1+N_t)} - 1\right]$, $t = 1, \ldots, T$, where N_t is the original number of sunspots in the year t. The graph of the transformed series appears in Figure 1. Most of the published examples of fitting neural networks models to sunspot series deal with the original and not the square-root transformed series.

We begin the AR-NN modelling of the series by selecting the relevant lags using the variable selection procedure described in Section 4.2. We use a third-order polynomial approximation to the true model. The use of SBIC leads to selecting lags 1,2, and 7. However, the residuals of the estimated linear AR model are strongly autocorrelated. This serial correlation is removed by also including y_{t-3} in the set of selected variables. When building the AR-NN model we select the input variables for each hidden unit separately using the specification test described in Section 4.4. Linearity is rejected at any reasonable significance level and the p-value of the linearity test minimized with lags 1 2, and 7 as input variables. The process of adding hidden units is discontinued after including the second hidden unit, see Table 1, and the final estimated model is

$$\begin{aligned} y_t = & \underset{(0.83)}{-0.17} + \underset{(0.09)}{0.85}\, y_{t-1} + \underset{(0.12)}{0.14} y_{t-2} - \underset{(0.06)}{0.31}\, y_{t-3} + \underset{(0.05)}{0.08}\, y_{t-7} \\ & + \underset{(7.18)}{12.80} \times F\left[\underset{(0.23)}{0.46}\left(\underset{(-)}{0.29} y_{t-1} - \underset{(0.83)}{0.87}\, y_{t-2} + \underset{(0.09)}{0.40}\, y_{t-7} - \underset{(0.05)}{6.68}\right)\right] \\ & + \underset{(0.48)}{2.44} \times F\left[\underset{(8.45\times 10^3)}{1.17\times 10^3}\left(\underset{(-)}{0.83} y_{t-1} - \underset{(0.12)}{0.53}\, y_{t-2} - \underset{(0.08)}{0.18}\, y_{t-7} + \underset{(7.18)}{0.38}\right)\right] + \hat{\varepsilon}_t. \end{aligned} \tag{20}$$

$$\hat{\sigma} = 1.89 \quad \hat{\sigma}/\hat{\sigma}_L = 0.70 \quad R^2 = 0.89 \quad pLJB = 1.8 \times 10^{-7}$$

$$pARCH(1) = 0.94 \quad pARCH(2) = 0.75 \quad pARCH(3) = 0.90 \quad pARCH(4) = 0.44,$$

where $\hat{\sigma}$ is the residual standard deviation, $\hat{\sigma}_L$ is the residual standard deviation of the linear AR model, R^2 is the determination coefficient, $pLJB$ is the p-value of the Lomnicki-Jarque-Bera test of normality, and $pARCH(j)$, $j = 1, \ldots, 4$, is the p-value of the LM test of no ARCH against ARCH of order j. The estimated correlation matrix of the linear term and the output of the hidden units

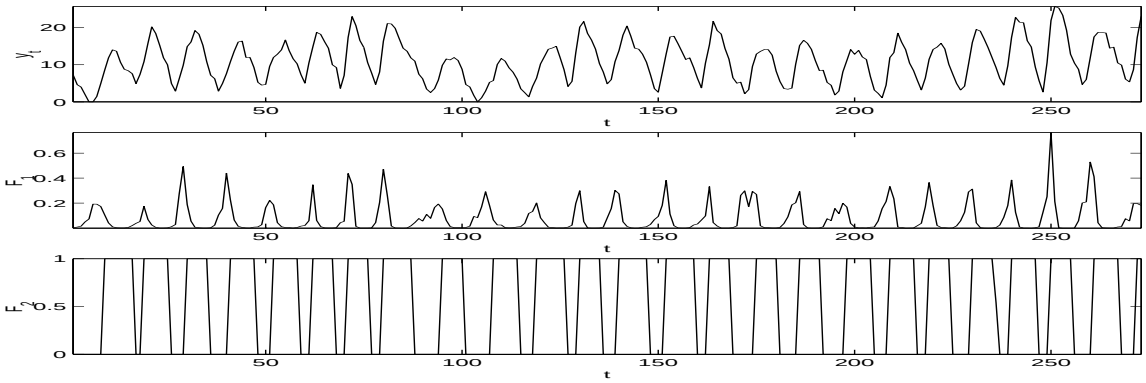


Figure 1: Panel (a): Transformed sunspot time series. Panel (b): Output of the first hidden unit. Panel (c): Output of the second hidden unit.

Table 1: Test of no additional hidden units: minimum p-value of the set of tests against each null model.

	Number of hidden units under null hypothesis		
	0	1	2
p-value	3×10^{-14}	2×10^{-9}	0.019

is

$$\hat{\mathbf{\Sigma}} = \begin{pmatrix} 1 & -0.30 & 0.74 \\ -0.30 & 1 & -0.19 \\ 0.74 & -0.19 & 1 \end{pmatrix}. \tag{21}$$

It is seen from (21) that there are no irrelevant neurons in the model as none of the correlations is close to unity in absolute value. Figure 1 illuminates the contributions of the two hidden units to the explanation of y_t. The linear unit can only represent a symmetric cycle, so that the hidden units must handle the nonlinear part of the cyclical variation in the series. It is seen from Figure 1 that the first hidden unit is activated at the beginning of every upswing, and its values return to zero before the peak. The unit thus helps explain the very rapid recovery of the series following each trough. The second hidden unit is activated roughly when the series is obtaining values higher than its mean. It contributes to characterizing the general asymmetry of the sunspot cycle in which the peaks and the troughs have distinctly different shapes. The switches in the value of the hidden unit from zero to unity and back again are quite rapid (γ_2 large), which is the cause of the large standard deviation of the estimate of γ_2, see the discussion in Section 4.3.

Table 2: p-values of tests of no error autocorrelation and parameter constancy for model (20).

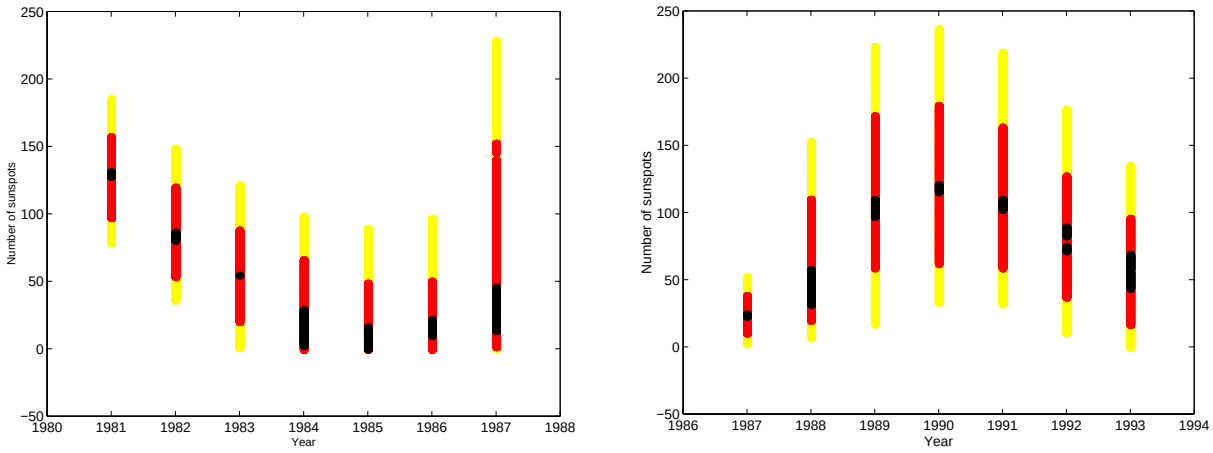
	LM Test for q-th order serial correlation						LM test for parameter constancy			
	Lags						K			
	1	2	3	4	8	12	1	2	3	4
p-value	0.55	0.61	0.34	0.49	0.47	0.22	0.98	0.95	0.93	0.88

The results of the misspecification tests of model (20) in Table 2 indicate no model misspecification. In order to assess the out-of-sample performance of the estimated model we compare our forecasting results with the ones obtained from the two SETAR models, the one reported in Tong (1990, p. 420) and the other in Chen (1995), an artificial neural network (ANN) model with 10 hidden neurons and the first 9 lags as input variables, estimated with Bayesian regularization (MacKay 1992a,b), and a linear autoregressive model with lags selected using SBIC. The SETAR model estimated by Chen (1995) is one in which the threshold variable is a nonlinear function of lagged values of the time series whereas it is a single lag in Tong´s model. Starting from thirteen consecutive years $1979, \ldots, 1992$, we computed the out-of-sample forecasts, $\hat{y}_{T+k|T}$, $k = 1, \ldots, 8$, from each model, and the associated forecast errors. The models were not re-estimated when the starting-point was moved. The forecasts of the nonlinear models were computed using Monte Carlo simulation with 4000 replications. the root mean square error (RMSE) and the mean absolute deviation (MAE) were used as summary.

The results can be found in Table 3. We leave a statistical comparison of the forecast accuracy aside and only make a couple of brief remarks. The AR-NN model appears to yield most accurate forecasts at short horizons. It is clearly better than the neural network model obtained by applying Bayesian regularization. The TAR models do not seem superior to the neural network models. It should be noted that the linear AR model is quite robust in that when the forecast horizon increases, the forecasts from it become very competitive. For short forecast horizons, the linear approximation to the data-generating process is not sufficiently good for forecasting purposes.

Table 3: Multi-step ahead forecasts, their root mean square errors, and mean absolute errors for the annual number of sunspots from a set of time series models, for the period 1980-2000.

Horizon	AR-NN		ANN model		SETAR model (Tong 1990)		SETAR model (Chen 1995)		AR	
(k)	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
1	12.9	10.4	15.0	12.7	21.0	14.9	19.4	17.2	18.9	15.1
2	18.4	14.6	20.7	16.7	31.6	21.0	27.2	21.6	26.5	18.8
3	21.6	14.6	24.3	19.3	38.4	25.2	33.6	24.8	28.2	19.9
4	22.2	15.6	27.3	21.6	42.2	26.4	31.8	23.6	27.8	20.2
5	22.4	14.0	32.4	23.2	42.2	27.0	30.6	21.6	26.9	19.1
6	20.6	14.0	36.5	25.3	41.6	26.4	31.9	23.0	26.8	19.7
7	27.5	18.4	42.2	30.2	43.3	30.3	34.0	25.0	27.5	19.8
8	25.1	20.0	39.6	30.1	45.2	35.0	33.8	26.0	26.7	19.6



(a) (b)

Figure 2: 50% (darkest), 90% and 99% (lightest) highest density regions for the multi-step forecasts made at (a) $t = 1979$ and (b) $t = 1985$.

In order to illustrate forecast densities we graphed HDRs for the ones obtained from model (20) in 1979 for the years 1981-1987 and the ones from the same model in 1985 for the years 1987-1993. It is seen from Figure 2 that the forecast uncertainty increases rather slowly with the forecast horizon. Comparing the two panels it seems that values around troughs are easier to predict than the ones around peaks. Uncertainty in forecasting peak observations is remarkably large. An interesting detail is a bimodal density that is apparent in the Panel (b) for the year 1992. Also note that the forecast densities for the trough values of the sunspot cycle are strongly skewed as the sunspot numbers cannot be negative. The figure also indicates that it is difficult to predict the strength of the upswing from afar: this is seen from the very long right-hand tail of the density for 1987.

7 Conclusions

In this paper we have demonstrated how statistical methods can be applied in building neural network models. The idea is to specify parsimonious models and keep the computational cost small. An advantage with the modelling strategy discussed here that the modelling procedure is not a black box. Every step in model building is clearly documented and motivated. On the other hand, using this strategy requires active participation of the model builder and willingness

to make decisions. Choosing the model selection criterion for variable selection and determining significance levels for the test sequence for selecting the number of hidden units are not automated, and different choices may often produce different models. Combining them in forecasting could be an interesting topic that, however, lies beyond the scope of this paper. Nevertheless, the method shows promise, and research is being carried out in order to learn more about its properties in modelling and forecasting stationary time series.

References

Akaike, H.: 1974, A new look at the statistical model identification, *IEEE Transactions on Automatic Control* **19**, 716–723.

Anders, U. and Korn, O.: 1999, Model selection in neural networks, *Neural Networks* **12**, 309–323.

Bertsekas, D. P.: 1995, *Nonlinear Programming*, Athena Scientific, Belmont, MA.

Bishop, C. M.: 1995, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford.

Chen, R.: 1995, Threshold variable selection in open-loop threshold autoregressive models, *Journal of Time Series Analysis* **16**, 461–481.

Cybenko, G.: 1989, Approximation by superposition of sigmoidal functions, *Mathematics of Control, Signals, and Systems* **2**, 303–314.

Eitrheim, Ø. and Teräsvirta, T.: 1996, Testing the adequacy of smooth transition autoregressive models, *Journal of Econometrics* **74**, 59–75.

Fine, T. L.: 1999, *Feedforward Neural Network Methodology*, Springer, New York.

Funahashi, K.: 1989, On the approximate realization of continuous mappings by neural networks, *Neural Networks* **2**, 183–192.

Gallant, A. R. and White, H.: 1992, On learning the derivatives of an unknown mapping with multilayer feedforward networks, *Neural Networks* **5**, 129–138.

Ghaddar, D. K. and Tong, H.: 1981, Data transformations and self-exciting threshold autoregression, *Journal of the Royal Statistical Society, Series C* **30**, 238–248.

Godfrey, L. G.: 1988, *Misspecification Tests in Econometrics*, Vol. 16 of *Econometric Society Monographs*, second edn, Cambridge University Press, New York, NY.

Granger, C. W. J. and Teräsvirta, T.: 1993, *Modelling Nonlinear Economic Relationships*, Oxford University Press, Oxford.

Haykin, H.: 1999, *Neural Networks: A Comprehensive Foundation*, second edn, Prentice-Hall, Oxford.

Hornik, K., Stinchombe, M., and White, H.: 1990, Universal approximation of an unknown mapping and its derivatives using multi-layer feedforward networks, *Neural Networks* **3**, 551–560.

Hornik, K., Stinchombe, M. and White, H.: 1989, Multi-layer Feedforward networks are universal approximators, *Neural Networks* **2**, 359–366.

Hwang, J. T. G. and Ding, A. A.: 1997, Prediction intervals for artificial neural networks, *Journal of the American Statistical Association* **92**, 109–125.

Hyndman, R. J.: 1995, Highest-density forecast regions for non-linear and non-normal time series models, *Journal of Forecasting* **14**, 431–441.

Hyndman, R. J.: 1996, Computing and graphing highest density regions, *The American Statistician* **50**, 120–126.

Kurková, V. and Kainen, P. C.: 1994, Functionally equivalent feedforward neural networks, *Neural Computation* **6**, 543–558.

Lin, C. F. J. and Teräsvirta, T.: 1994, Testing the constancy of regression parameters against continuous structural change, *Journal of Econometrics* **62**, 211–228.

Luukkonen, R., Saikkonen, P. and Teräsvirta, T.: 1988, Testing linearity in univariate time series models, *Scandinavian Journal of Statistics* **15**, 161–175.

MacKay, D. J. C.: 1992a, Bayesian interpolation, *Neural Computation* **4**, 415–447.

MacKay, D. J. C.: 1992b, A practical bayesian framework for backpropagation networks, *Neural Computation* **4**, 448–472.

Medeiros, M. C., Teräsvirta, T. and Rech, G.: 2001, Modelling neural networks using statistical methods, *Paper in progress*.

Rech, G., Teräsvirta, T. and Tschernig, R.: 2001, A simple variable selection technique for nonlinear models, *Communications in Statistics, Theory and Methods* **30**.

Reed, R. D. and Marks II, R. J.: 1999, *Neural Smithing*, MIT Press, Cambridge, MA.

Ripley, B. D.: 1996, *Pattern Recognition and Neural Networks*, Cambridge University Press, Cambridge.

Rissanen, J.: 1978, Modeling by shortest data description, *Automatica* **14**, 465–471.

Schwarz, G.: 1978, Estimating the dimension of a model, *Annals of Statistics* **6**, 461–464.

Sussman, H. J.: 1992, Uniqueness of the weights for minimal feedforward nets with a given input-output map, *Neural Networks* **5**, 589–593.

Swanson, N. R. and White, H.: 1995, A model selection approach to assesssing the information in the term structure using linear models and artificial neural networks, *Journal of Business and Economic Statistics* **13**, 265–275.

Swanson, N. R. and White, H.: 1997a, Forecasting economic time series using flexible versus fixed specification and linear versus nonlinear econometric models, *International Journal of Forecasting* **13**, 439–461.

Swanson, N. R. and White, H.: 1997b, A model selection approach to real-time macroeconomic forecasting using linear models and artificial neural networks, *Review of Economic and Statistics* **79**, 540–550.

Teräsvirta, T. and Lin, C.-F. J.: 1993, Determining the number of hidden units in a single hidden-layer neural network model, *Research Report 1993/7*, Bank of Norway.

Tong, H.: 1990, *Non-linear Time Series: A Dynamical Systems Approach*, Oxford University Press, Oxford.

Weigend, A., Huberman, B. and Rumelhart, D.: 1992, Predicting sunspots and exchange rates with connectionist networks, *in* M. Casdagli and S. Eubank (eds), *Nonlinear Modeling and Forecasting*, Addison-Wesley.

White, H.: 1990, Connectionist nonparametric regression: Multilayer feedforward networks can learn arbitrary mappings, *Neural Networks* **3**, 535–550.

Zapranis, A. and Refenes, A.-P.: 1999, *Principles of Neural Model Identification, Selection and Adequacy: With Applications to Financial Econometrics*, Springer-Verlag, Berlin.