Lancaster University Management School
Working Paper
2010/022

Modelling Deterministic Seasonality with Artificial Neural Networks for Time Series Forecasting

Nikolaos Kourentzes and Sven Crone

The Department of Management Science
Lancaster University Management School
Lancaster LA1 4YX
UK

The LUMS Working Papers series can be accessed at http://www.lums.lancs.ac.uk/publications/
LUMS home page: http://www.lums.lancs.ac.uk/

Working Paper

Modelling Deterministic Seasonality with Artificial Neural Networks for Time Series Forecasting

Nikolaos Kourentzes and Sven F. Crone, Lancaster Centre for Forecasting, Lancaster University Department of Management Science

Email: n.kourentzes@lancaster.ac.uk

This study explores both from a theoretical and empirical perspective how to model deterministic seasonality with artificial neural networks (ANN) to achieve the best forecasting accuracy. The aim of this study is to maximise the available seasonal information to the ANN while identifying the most economic form to code it; hence reducing the modelling degrees of freedom and simplifying the network's training. An empirical evaluation on simulated and real data is performed and in agreement with the theoretical analysis no deseasonalising is required. A parsimonious coding based on seasonal indices is proposed that showed the best forecasting accuracy.

Keywords: neural networks; deterministic seasonality; input variable selection

Modelling Deterministic Seasonality with Artificial Neural Networks for Time Series Forecasting

Nikolaos Kourentzes and Sven F. Crone, Lancaster Centre for Forecasting, Lancaster University Department of Management Science

1. Introduction

Artificial neural networks (ANNs) are nowadays widely recognised as a potent forecasting tool with several research and practical applications (Hippert, et al., 2005; Zhang, et al., 1998). Theoretically ANNs are universal approximators, which is desirable in forecasting (Hornik, et al., 1989). They have been shown to be able to forecast linear and nonlinear synthetic series and real time series at least as well as established benchmarks, like exponential smoothing and ARIMA models (Hill, et al., 1996; Zhang, 2001; Zhang, et al., 2001). Furthermore, ANNs are able to forecast across a wide range of data frequencies, when the appropriate input variables are provided (Crone and Kourentzes, 2009) making them a potent and flexible forecasting tool. However, they are criticised to have inconsistent performance across different applications and in empirical evaluations (Armstrong, 2006; Callen, et al., 1996; Makridakis and Hibon, 2000). The ANN literature suggests that the observed inconsistency is a product of bad modelling practices or limited understanding of the modelling process; for instance, there is no consensus on how to select a relevant set of input variables and lags (Anders and Korn, 1999; Zhang, et al., 1998). A literature survey identified that 71% (out of 105) published papers model ANNs based on trial and error approaches. This has a significant impact on the consistency of their performance and also hinders our understanding of how to model them (Adya and Collopy, 1998). It is therefore important to rigorously evaluate competing ANN modelling strategies in order to gain insight on best practices.

The ANN literature has identified a set of open questions in modelling neural networks that need to be solved before their application can become more consistent and potentially perform better (Curry, 2007; Zhang, et al., 1998). One such open research question is whether ANNs are able to model seasonal time series or if the time series need to be deseasonalised first. A standard way of performing this is through seasonal integration of the time series, which follows the same ideas of ARIMA modelling (Zhang and Kline, 2007). Hill et al. (1996) show that ANN using deseasonalised time series from the M1 competition outperformed standard statistical models, suggesting significant improvements in ANNs performance. Nelson et al. (1999) verifies that deseasonalising the M1 forecasting competition time series provided ANNs with the performance edge. They repeated the experiment without deseasonalising the time series and the forecasting performance got significantly worse, therefore arguing that deseasonalising was a necessary step. They argued that this way ANNs can focus on learning the trend and the cyclical components. To learn seasonality in addition would require larger networks, meaning a larger input vector, which may lead to overfitting. Zhang and Qi (2005) reached the same conclusion that deseasonalising helps. They suggest that deseasonalised time series do not contain long dynamic autocorrelation structures that would make the choice of the input vector more difficult, thus leading to smaller more parsimonious models. Curry (2007) examines the ability of ANN to model seasonality from a theoretical perspective. He suggests that for ANN to model seasonality they should have adequately long input vector to capture the seasonal effects. Ill selected input vector can make the ANN unable to forecast seasonality, implying that Zhang and Qi results can potentially hide input misspecification errors. Crone and Dhawan (2007) demonstrate that ANNs are able to model robustly monthly seasonal patterns using only an adequate number lags of the time series. Zhang and Kline (2007) explore the ability of ANNs to forecast quarterly time series. They again find that deseasonalising helps, however this time they also evaluated a large variety of models, including models with deterministic dummy variables. They argue that such additional variables do not help because they cannot capture dynamic and complex seasonal structures.

The papers above do not distinguish between different forms of seasonality. Deterministic seasonality and seasonal unit root theoretically require a different modelling approach (Ghysels and Osborn, 2001; Matas-Mir and Osborn, 2004; Osborn, et al., 1999), which has been largely ignored in the ANN literature and the respective debate on how to model seasonality. In this analysis, we discuss that this distinction implies a different modelling procedure from a theoretical perspective. Modelling deterministic seasonality is impaired by deseasonalising the time series and different modelling practises should be followed. An empirical evaluation of competing methods to model seasonality is performed on simulated and real time series. We found that using a set of dummy variables can improve forecasting accuracy over the standard ANN modelling practise. Removing seasonality does not perform well for the case of deterministic seasonality. Finally, a parsimonious coding based on seasonal indices is proposed, which outperforms other candidate models while keeping the modelling degrees of freedom to a minimum.

The paper is organised as follows: section 2 discusses the different types of seasonality from a theoretical perspective. Section 3 introduces the methods that will be used to model deterministic seasonality. Section 4 provides information on the experimental design for the empirical evaluation on synthetic data, followed by section 5 where the results are discussed. In section 6 the empirical evaluation on real time series from the T-competition is presented and analysed. Conclusions and limitations of this study are discussed together with further research objectives in section 7.

2. Seasonal Time Series

2.1. Deterministic Seasonality

A time series is said to have deterministic seasonality when its unconditional mean varies with the season and can be represented using seasonal dummy variables,

$$y_t = \mu + \sum_{s=1}^{S} m_s \delta_{st} + z_t \, , \tag{1}$$

where y_t is the value of the time series at time t, μ is the level of the time series, m_s is the seasonal level shift due to the deterministic seasonality for season s, δ_{st} is the seasonal dummy variable for season s at time t, z_t is a weak stationary stochastic process with zero mean and S is the length of the seasonality. Furthermore, the level of the time series μ can be generalised to include trend. Note that the seasonality is defined as a series of seasonal level shifts m_s, which describe the seasonal profile and are constant across time, i.e. $m_s = m_{st}$. Also note that the $\sum m_s = 0$ over a full season. This implies that with the appropriate transformations of μ and m_s a set of S-1 or S seasonal dummies can be used to code seasonality. Furthermore, due to z_t each value of the time series deviates over its respective seasonal mean with a constant variance over both s and t, which means that the deterministic seasonal process forces the observations to remain close to their underlying mean (Ghysels and Osborn, 2001). Modelling (1) with S seasonal dummies and $\mu \neq 0$ using a linear model, like linear regression, introduces the problem of multicollinearity, therefore S-1 dummies should be used in this case (Kvanli, et al., 2002).

An alternative way to code deterministic seasonality is through its trigonometric representation. In respect to (1) seasonality can be expressed as

$$y_t = \mu + \sum_{k=1}^{S/2} \left[\alpha_k \cos\left(\frac{2\pi kt}{S} \right) + \beta_\kappa \sin\left(\frac{2\pi kt}{S} \right) \right] + z_t \, , \tag{2}$$

where α_k and β_k create linear combinations of S/2 sines and cosines of different frequencies following the idea of spectral analysis of seasonality. Equations (1) and (2) have μ and z_t expressed as individual components in both cases, allowing separate modelling of seasonality and the remaining time series components (Ghysels and Osborn, 2001). Note that if less than S/2 linear combinations of sines and cosines are used, the representation of seasonality is imperfect and it is approximated with some error, the size of which is related to the number of combinations used.

2.2. Seasonal Unit Root

Seasonality can also be the result of an autoregressive integrated moving average (ARIMA) process,

$$\phi(L)\Delta_S y_t = \gamma + \theta(L)\varepsilon_\tau \, , \tag{3}$$

where L is the lag operator, Δ_S is the seasonal difference operator, **ϕ** and **θ** are the coefficients of the autoregressive and moving average process respectively, γ is a drift, and ε_t i.i.d. $N(0,\sigma^2)$. The variance of y_t under the case of deterministic seasonality is constant over t and the seasonal period s, which is not true here. This stochastic seasonal process can be viewed as a seasonal unit root process, i.e. for each s there is a unit root, which in turn requires seasonal differencing. More details about the seasonal unit root process can be found in (Ghysels and Osborn, 2001; Matas-Mir and Osborn, 2004; Osborn, et al., 1999).

It is interesting to examine what happens if deterministic seasonality is misspecified as a seasonal unit root process. Considering seasonal differences (1) becomes

$$\Delta_S y_t = \Delta_S z_t \, . \tag{4}$$

Essentially in (4) seasonality has been removed, i.e. a deseasonalised form of y_t is modelled. Comparing (1) and (4) we can deduce that it is now impossible to estimate m_s and furthermore $\Delta_S z_t$ is overdifferenced (Ghysels and Osborn, 2001). Therefore, it is preferable to keep deterministic seasonality and model it appropriately.

3. Forecasting with artificial neural networks

3.1. Multilayer Perceptrons for Time Series Prediction

We use the common multilayer perceptron (MLP), which represents the most widely employed ANN architecture (Zhang, et al., 1998). MLPs are well researched and have proven abilities in time series prediction to approximate and generalise any linear or nonlinear functional relationship to any degree of accuracy (Hornik, 1991) without any prior assumptions about the underlying data generating process (Qi and Zhang, 2001), providing a potentially powerful forecasting method for linear or non-linear, non-parametric, data driven modelling. In univariate forecasting MLPs are used similarly to autoregressive models, capable of using as inputs a set of lagged observations of the time series and explanatory variables to predict its next value (Anders and Korn, 1999). Data are presented to the network as a sliding window over the time series history. The ANN tries to learn the underlying data generation process during training so that valid forecasts are made when new input values are provided (Lachtermacher and Fuller, 1995). In this analysis single hidden layer ANN are used, based on the proof of universal approximation (Hornik, 1991). The general function of single hidden layer networks is

$$f(X,w) = \beta_0 + \sum_{h=1}^{H} \beta_h g\left(\gamma_{0i} + \sum_{i=0}^{I} \gamma_{hi} x_i \right). \tag{5}$$

X = [x_0, x_1, …, x_n] is the vector of the lagged observations (inputs) of the time series. **X** can also contain observations of explanatory variables. The network weights are **w** = (**β**, **γ**), **β** = [β_1, β_2…,β_h] and **γ** = [γ_{11}, γ_{12}…, γ_{hi}]. The β_0 and γ_{0i} are the biases of each respective neuron. I and H are the number of input and hidden units in the network and g(·) is a non-linear transfer function (Anders, et al., 1998). In this analysis the hyperbolic tangent transfer function is used. For computational reasons this can be approximated as

$$\tanh(x) = \frac{2}{\left(1+e^{-2x}\right)-1}, \tag{6}$$

which is frequently used for modelling ANNs (Vogl, et al., 1988).

3.2. Coding Deterministic Seasonality

It is easy to include seasonal information in ANNs. Seasonal dummy variables can be included as explanatory variables. As noted in section 2.1 if S dummy variables are included in linear models the problem of multicollinearity appears, so only S-1 dummies should be used. For ANNs this is more complicated. Assuming only linear transfer functions and H>1 multicollinearity can exist even for S-1 dummies, since they are inputted in several hidden nodes. This hinders inference from a ANN, but does not necessarily harm its predictive power, which is also true for the nonlinear transfer function case (Zhang, et al., 1998). Based on this observation both S-1 and S number of seasonal dummies make sense for ANN models. Deterministic seasonality as expressed in (2) can be modelled easily through the use of dummy variables. Note that an alternative is to approximate (2) using fewer frequencies by increasing the number of hidden nodes H in a network (Hornik, et al., 1989). Following the same procedure, based on the increase of H, ANN are able to approximate seasonal patterns by combining seasonal dummies in a single integer dummy defined as δ = [1, 2...S] (Crone and Kourentzes, 2007). Alternatively m_s can be combined to form a series of seasonal indices that can be used as an explanatory variable for the ANN. The problem that arises in this alternative is how to estimate the unknown m_s. It is also possible to model seasonality as a misspecified stochastic seasonal unit root process, with the problems discussed in section 2.2. Another alternative is to use seasonal integration to remove seasonality and another alternative would be to use an adequate AR structure to model the seasonality as discussed in (Curry, 2007). Note that much of the debate in literature, as mentioned in section 1, regarding deseasonalising time series or not falls in the latter two alternatives which in theory are not advisable for deterministic seasonality. However, for practical applications with small samples it can be shown that it is difficult to distinguish between deterministic and stochastic seasonality (Ghysels and Osborn, 2001), therefore these alternatives are still viable options.

4. Synthetic Data Simulations Setup

4.1. Time Series Data

Eight synthetic time series are used to evaluate the competing ways discussed in section 3.2 to model deterministic seasonality using ANN. The time series are constructed using as a data generating process the dummy variable representation of deterministic seasonality (1). Two different sets of m_s are modelled, reflecting two different seasonal patterns (A & B). The first seasonal pattern resembles retail data that peak during Christmas sales, whereas pattern B approximates sales of products that sell more during the summer months. The parameter μ is set to 240 units and z_t ~ i.i.d. $N(0, \sigma_j^2)$. Four different levels of noise are simulated through σ_j^2. For no noise $\sigma = 0$, reflecting a zero error for all t. For low, medium and high noise levels σ is 1, 5 and 10 respectively. Note that these synthetic time series are constructed in a stricter way than that required by (1). This is done in order to create time series in which only the effect of the deterministic seasonal pattern needs to be modelled, simplifying the modelling of the input vector of the ANN and allowing to focus solely on the effects of the different seasonal coding schemes. All time series have S=12, i.e. simulate monthly data, and are 480 observations long. For the purpose of this experiment the time series are divided in three equal training, validation and test subsets, to facilitate the ANN training. The first 72 observations of each time series are plotted in figure 1 to provide a visual representation of the two seasonal patterns and the different noise levels.

4.2. Experimental setup

The forecast horizon for all competing models is 12 months. Rolling origin evaluation is used to assess the error 1 to 12 months in the future. This evaluation scheme is preferred because it provides a reliable estimation of the out of sample error (Tashman, 2000). Two error measures are used. Firstly, the mean absolute error (MAE) that allows a direct comparison of the predictive accuracy and the known noise level. For given actuals X_t and forecasts F_t for all periods t in the sample $MAE = \frac{1}{n}\sum_{t=1}^{n}\left|X_t - F_t\right|$. (7)

The symmetric mean absolute percent error (sMAPE) is also used to measure accuracy. This measure is scale independent and allows comparing accuracy across time series. It can be calculated as

$$sMAPE = \frac{1}{n}\sum_{t=1}^{n}\left(\frac{\left|X_t - F_t\right|}{\left(\left|X_t\right| + \left|F_t\right|\right)/2}\right). \qquad (8)$$

Note that the formula used here is the corrected form of sMAPE as in (Chen and Yang, 2004). Both validation and test sets contain 160 observations (1/3 of the total sample each). The accuracy of the competing ANN models is evaluated for statistically significant differences using the nonparametric Friedman test and the Nemenyi test, to facilitate an evaluation of nonparametric models without the need to relax assumptions of ANOVA or similar parametric tests (Demšar, 2006). To compare the models against the benchmark the best ANN initialisation is selected by minimum validation set error.

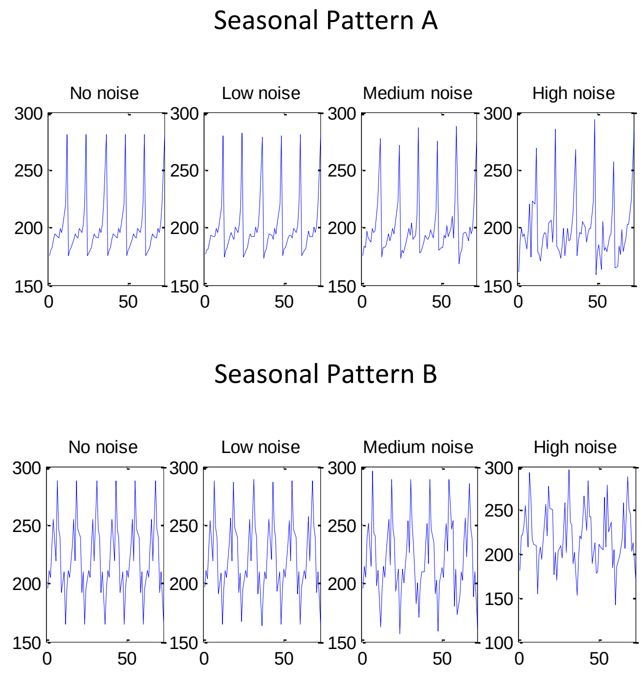

Fig. 1: Plot of the first 72 observations of each synthetic time series.

4.3. Neural Network Models

MLP models that code the deterministic seasonality with the seven alternative ways described in section 3.2 are compared. To model seasonality as stochastic, we use an adequate univariate MLP model which employs lags t-1 and t-12, which is named *AR*. To model seasonality as a seasonal unit root process the time series is used after seasonal differencing. No lags are used and the correct level is estimated by the MLP by assigning the correct weights to the bias terms in the different nodes. This is named *SRoot* and essentially covers the case where seasonality is removed before inputting the time series to the MLP. The common deterministic seasonality coding through seasonal dummy variables is implemented in models *Bin11* and *Bin12* which use 11 and 12 seasonal binary dummy variables respectively to model each month. No past lags of the time series are used for these models. The integer dummy variable representation uses only an integer dummy that repeats values from 1 to 12, which is implemented in model *Int*. The trigonometric representation is modelled through the use of two additional variables, one for sin(2πt/12) and one for cos(2πτ/12) and is named *SinCos*. Finally, seasonal indices for the time series are identified by calculating the average value for each period of the season in the training set. This is an adequate estimation since the time series exhibit no trend or irregularities. The seasonal indices are repeated to create an explanatory variable which is then used as the only input to the MLP model *SIndex.* An overview of the inputs for each model is provided in table I.

Table I: Summary of MLP Inputs

Model	Lags*	Explanatory variables**	No of inputs
AR	1, 12	-	2
Bin11	-	11 Seasonal Dummies	11
Bin12	-	12 Seasonal Dummies	12
Int	-	Integer Dummy [1,2...12]	1
SinCos	-	sin(2πt/12), cos(2πt/12)	2
Sindex	-	Seasonal Indices	1
SRoot	-***	-	0

*The Lags specify the time lagged realisations *t-n* used as inputs; **For all explanatory variables only the contemporary lag is used; ***Time series is modelled after seasonal integration, i.e. $\Delta_S y_t$.

The remaining parameters of the MLP are constant for all models. This allows attributing any differences in the performance of the models solely to the differences in modelling seasonality. All use a single hidden layer with six hidden nodes. The topology of the *AR* model can be seen in figure 2. The networks are trained using the Levenberg-Marquardt algorithm, which requires setting the μ_{LM} and its increase and decrease steps. Here $\mu_{LM}=10^{-3}$, with an increase step of $\mu_{inc}=10$ and a decrease step of $\mu_{dec}=10^{-1}$. For a detailed description of the algorithm and the parameters see (Hagan, et al., 1996). The maximum training epochs are set to 1000. The training can stop earlier if μ_{LM} becomes equal of greater than $\mu_{max}=10^{10}$ or the validation error increases for more than 50 epochs. This is done to avoid over-fitting. When training is stopped the network weights that gave the lowest validation error during training are used. Each MLP is initialised 50 times with randomised starting weights to accommodate the nonlinear optimisation and to provide an adequate sample to estimate the distribution of the forecast errors in order to conduct the statistical tests. The MLP initialisation with the lowest error for each time series on the validation dataset is selected to predict all values of the test set. Lastly, the time series and all explanatory variables that are not binary are linearly scaled between [-0.5, 0.5].

4.4. Statistical Benchmark

Any empirical evaluation of time series methods requires the comparison of their accuracy with established statistical benchmark methods, in order to assess the increase in accuracy and its contribution to forecasting research. This is often overlooked in ANN experiments (Adya and Collopy, 1998). In this analysis seasonal exponential smoothing models (*EXSM*) are used. The seasonality is coded as additive seasonality, which is appropriate for the deterministic seasonality in the simulated dataset. The smoothing parameters are identified by optimising the one step ahead in-sample mean squared error. This model is selected as a benchmark due to its proven track record in univariate time series forecasting (Makridakis and Hibon, 2000). For more details on exponential smoothing models and the guidelines that were used to implement them in this analysis see (Gardner, 2006a).

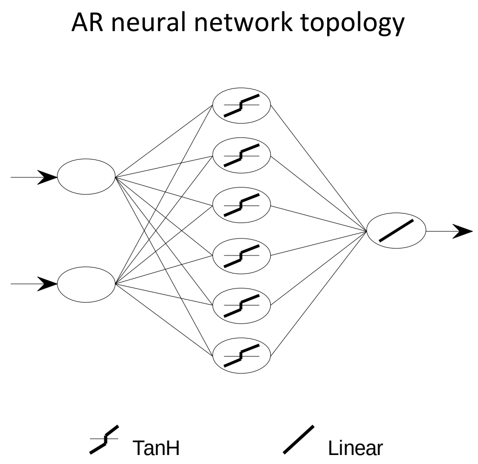

Fig. 2: Plot of the AR neural network model, showing the transfer functions of each layer. All other ANN models have similar topology other than the different number of inputs.

5. Simulation Results

5.1. Nonparametric MLP Comparisons

The competing MLP are tested for statistically significant differences using the Friedman and the post-hoc Nemenyi tests. Both use the mean rank of the errors. In this analysis MAE and sMAPE provided the same ranking, so there is no difference which error is used for these tests. The results of the MLP comparisons are provided in table II.

The Friedman test indicates that across all time series, across different noise levels and for all time series separately there are statistically significant differences among the MLP models. Inspecting the results of the Nemenyi tests in table II a more detailed view on the ranking of each individual model is revealed, along with statistically significant differences among them. It can be observed that across all different noise levels and across all time series at 5% significance level the *SIndex* outperforms all other models with a statistically significant difference from the second best model. *Bin11* and *Bin12* perform equally with no statistically significant differences both ranking second after *SIndex* in all cases apart from the high noise case. At 1% significance level *BIn11* and *Bin12* have no significant differences in all cases. This means that for ANN models there is no essential difference between using S-1 or S binary dummies. When only the no, low and medium noise time series are considered, the *SinCos* has no statistically significant differences with the seasonal binary dummies *Bin11* and *Bin12* models. For the case of high noise time series the *SinCos* ranks third after the *SIndex* and seasonal binary dummy variables models. This demonstrates that although the *SinCos* model is not equivalent to the trigonometrical representation of deterministic seasonality as expressed in (2) it is able to approximate it and in many cases with no statistically significant differences from the equivalent seasonal dummy coding. Furthermore, this representation is S/4 times more economical in inputs compared to (2). Compared to (1) or *Bin11* and *Bin12* this coding is S-2 and S-1 inputs more economical respectively. For the low, medium and high noise the *Int* model follows in ranking. Although this model performs worse than the previous seasonality encodings it still outperforms the misspecified seasonal models *AR* and *SRoot*. This is not true for the no noise time series, which also affects the overall ranking across time series as well. The *AR* model follows second to the last in all cases. This demonstrates that it is better to code the deterministic seasonality through explanatory dummy variables, than as an autoregressive process, as it would be fitting for stochastic seasonality. Furthermore, in agreement to the discussion in section 2.2, removing the seasonality through seasonal integration, as in *SRoot*, performs poorly and ranks last in most cases. The reason for this is that the ANNs are not able to estimate directly the m_s and $\Delta_S y_t$ is overdifferenced. Note that in the case of no noise all models with the exception of *Int* are able to capture the seasonality perfectly with no error.

Table II: Summary of MLP nonparametric comparisons

Time series	All	No noise	Low noise	Medium noise	High noise
Friedman p-value	0.000	0.000	0.000	0.000	0.000
			Mean Model Rank		
AR	240.59	**<u>165.25</u>**	260.01	261.01	276.10
Bin11	**<u>140.38</u>**	**<u>165.25</u>**	**<u>140.43</u>**	**<u>129.43</u>**	126.41
Bin12	**<u>142.08</u>**	**<u>165.25</u>**	**<u>136.90</u>**	**<u>132.96</u>**	133.20
Int	201.85	237.00	212.43	198.76	159.21
SinCos	146.22	**<u>165.25</u>**	**<u>139.22</u>**	**<u>137.40</u>**	143.03
Sindex	85.01	**<u>165.25</u>**	42.53	57.45	74.81
SRoot	272.38	**<u>165.25</u>**	297.00	311.50	315.75
			Ranking		
AR	5	**<u>1</u>**	4	4	6
Bin11	**<u>2</u>**	**<u>1</u>**	**<u>2</u>**	**<u>2</u>**	2
Bin12	**<u>2</u>**	**<u>1</u>**	**<u>2</u>**	**<u>2</u>**	3
Int	4	2	3	3	5
SinCos	3	**<u>1</u>**	**<u>2</u>**	**<u>2</u>**	4
Sindex	1	**<u>1</u>**	1	1	1
SRoot	6	**<u>1</u>**	5	5	7

* In each column MLP with no statistically significant differences under the Nemenyi test at 5% significance are **<u>underlined</u>**; the critical distance for the Nemenyi test for all time series at 1% significance level is 3.73, at 5% significance level is 3.18 and at 10% significance level is 2.91. The critical distance for any noise category at 1% significance level is 7.46, at 5% significance level is 6.37 and at 10% significance level is 5.82.

It is apparent that the best method to model the deterministic seasonality is to use the seasonal indices as an explanatory input variable for the MLP. Not only does this method perform best, but also it is very parsimonious, requiring a single input to model the deterministic seasonality, as shown in table I.

5.2. Comparisons against Benchmarks and Noise Level

Taking advantage of the synthetic nature of the time series the error of each forecasting model with the artificially introduced error level can be compared directly and derive how close each model is to an ideal accuracy. The ideal accuracy is when the model's error is exactly equal to the noise, since that would mean that the model has captured perfectly the data generating process and ignores completely the randomness. On the other hand, a lower error than the noise level would imply possible overfitting to randomness. The comparison is done in MAE for each time series individually. The results are presented in figure 3. Moreover the benchmark accuracy in MAE for each time series is provided in the same figure.

In figure 3 it is clear than when there is no noise, for both seasonal patterns, all MLP models and the benchmark forecast the time series perfectly with zero error. Comparing the MLP models to the benchmark the misspecified *AR* and *SRoot* models perform worse than *EXSM*, with the *SRoot* model ranking consistently last. This demonstrates that for the case of deterministic seasonality deseasonalising the time series, here through seasonal integration, hinders the ANN to forecast the time series accurately. For both seasonal patterns for the low noise time series 2 and 6 all MLP perform worse than the benchmark. The opposite is true for the *Bin11, Bin12, Int, SinCos* and *SIndex* MLP models for the higher noise level time series. This implies that ANN perform better than the statistical benchmark in high noise time series, being able to capture the true data generating process better.

When comparing the models' accuracy with the known error due to noise all the MLP models, with the exception of the misspecified *AR* and *SRoot*, for all time series are very close to the ideal accuracy, i.e. having error only due to randomness. Note that for the validation set, on which the best performing initialisation for each of the ANN models was chosen, their error is practically only due to noise. The benchmark error consistently increases as the noise level increases. For the case of low noise time series *EXSM* manages to forecast the time series with the error being solely due to randomness, implying a very good fit to the data generating process, however this is not true for higher noise levels. The results are consistent across both seasonal patterns.

Evaluating the performance of all models across the three training, validation and test subsets the models perform consistently, with no evidence of overfitting to the training set and all models are able to generalise well on the test set.

Due to the fact that it is impossible to aggregate results across different time series using MAE, only figures for sMAPE are reported, which is scale independent. Summary accuracy sMAPE figures for all time series are provided in table III.

Table III: Summary sMAPE across all synthetic time series

Model	Training subset	Validation subset	Test subset
AR	<u>1.90%</u>	<u>1.94%</u>	<u>1.72%</u>
Bin11	1.60%	1.59%	1.45%
Bin12	**1.58%**	**1.58%**	1.46%
Int	1.62%	1.61%	1.49%
SinCos	1.59%	1.59%	1.47%
Sindex	1.60%	**1.58%**	**1.44%**
SRoot	<u>2.36%</u>	<u>2.21%</u>	<u>1.91%</u>
EXSM	1.86%	1.68%	1.52%

The best performing model in each set is marked with bold numbers. The models that are outperformed by the EXSM benchmark are underlined

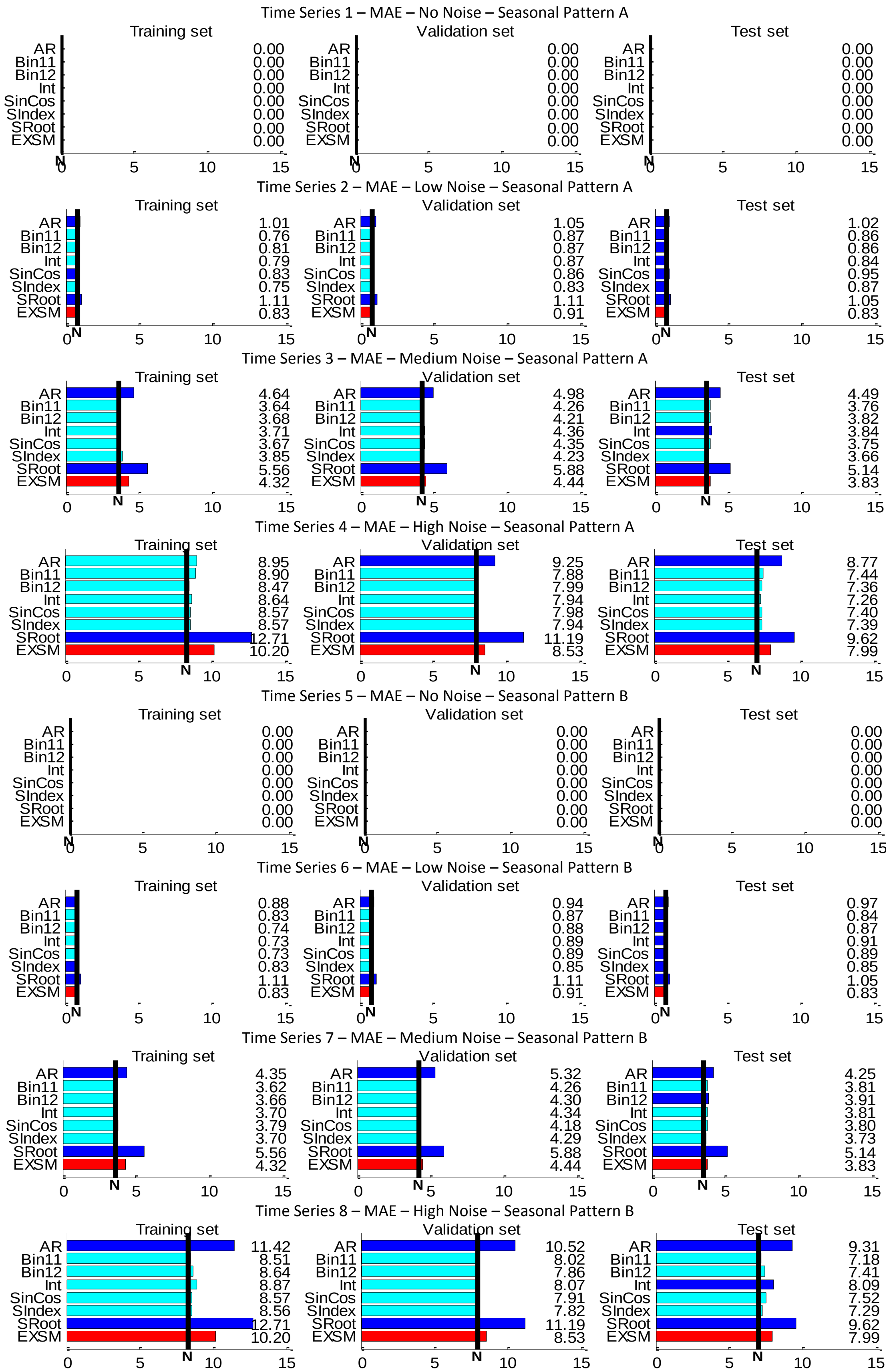

Fig. 3: MAE for each time series for each subset for all models. The noise level is marked by a thick black vertical line. Light coloured bars are models which are better than the benchmark (EXSM). The value of each error is provided at the right side.

The results are in accordance with figure 3. The *AR* and *SRoot* models are outperformed by the benchmark, which is turn is outperformed by all other MLP models. In agreement with the results in table II the *SIndex* model is overall the most accurate, followed by the *Bin12* and *Bin11*. Note that the small sMAPE figures imply that all the models managed to capture the seasonal profile in all the time series and a visual inspection of the forecasts would reveal very small if no differences at all. Finally, the overall error level seems to be different between the three subsets. This is due to the random noise. Although each set contains 160 observations, which simulates in total 40 years of data, longer sample was required to ensure equal noise distribution across all subsets.

6. Transportation Data Experiments

6.1. The Dataset

A dataset of 60 time series from the T-competition (Hibon, et al., 2007) was selected to evaluate the ANN models on real time series. The T-competition dataset contains transportation time series of different frequencies. From the complete dataset of 161 monthly time series a subset that was tested for deterministic seasonality was selected.

Initially the presence of seasonality is verified. To accomplish this, a series of steps was performed. Firstly, for each time series a moving average filter of 12 periods was used to remove the trend from the time series. Following that, for each time series, all the seasonal indices were calculated and compared for statistically significant differences using the Friedman test. The time series that did not present significant differences were concluded to be not seasonal, i.e. all m_s for s = 1...12 were equal, and therefore were dropped from the final dataset.

Furthermore, not all seasonal time series are deterministic. Two different statistical tests were used to test for presence of deterministic seasonality. The first test is the Canova-Hansen test for seasonal stability (Canova and Hansen, 1995; Ghysels and Osborn, 2001). The null hypothesis is that the seasonal pattern is deterministic. Assuming a stochastic seasonal process for each m_s there is an associated residual term η_s ~ i.i.d. $N(0,\sigma_{\eta s}^2)$. If for any s in S the $\sigma_{\eta s}^2$ is greater than zero the process is stochastic. The Canova-Hansen test corresponds to jointly testing for all s in S if $\sigma_{\eta s}^2 = 0$. The second test is based on the definition of deterministic seasonality (1). After a low pass filter is applied to the time series, so that the seasonal component is separated, a regression model with S-1 binary dummies is fitted. The residuals are calculated and tested if they follow the assumptions of (1). This is done by an Augmented Dickey-Fuller (ADF) test. If the null is rejected then the residuals are stationary, i.e. (1) describes the data generating process of the time series. The order of the ADF test is selected automatically using the Bayesian Information Criterion (BIC) (Cheung and Lai, 1998).

The time series that pass both tests at a 5% significance level constitute the sample that is used for this empirical evaluation. The shortest selected time series is 87 months and the longest is 228 months long. Figure 4 provides a histogram of the length of the time series in the final sample, showing the distribution of short and long time series. The exact time series that were selecting can be found in table VI. For all the time series, the last 38 observations are split equally to validation and test sets, leaving all the remaining observations for the training set.

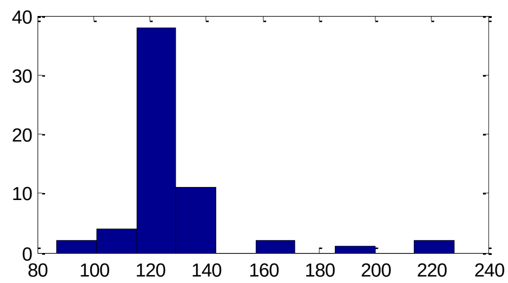

Fig. 4: The histogram reveals that most time series are between 120 and 140 months long and there are a few below 100 and above 160 months.

6.2. The Experimental Setup

The experimental design is similar to the one presented in section 5, with some differences in the model setup, which are discussed here. The ANN models have differences in the input vectors. In order to capture the trend and irregular components of the time series some additional non-seasonal time series lags are used for each model. These lags are identified using backward stepwise regression (Balkin and Ord, 2000; Swanson and White, 1997). The regression model is fitted to the time series and the significant lags are used as inputs to the ANNs. Only lags from t-1 up to t-11 are evaluated, therefore no seasonal lags are included. The resulting additional inputs are used together with the different approaches to model seasonality, as presented before in section 4.3. Note that for the *SRoot* model the identification of the additional inputs is done on the seasonally integrated time series.

Exponential smoothing family of models is used as a benchmark. The only difference in comparison to the previous experiment is that both seasonal and trend-seasonal exponential smoothing models are considered, according to the suggestions of Gardner (2006b).

6.3. Results

The competing MLP are tested for statistically significant differences using the Friedman test. At least one model is found to be different with a p-value = 0, so the post-hoc Nemenyi test is used to identify significant differences between the models and their ranking, as before. The results are provided in table IV.

Table IV: Summary of MLP nonparametric comparisons

Friedman p-value	0.000	
Models	Mean Rank*	Ranking
AR	166.81	2
Bin11	177.09	5
Bin12	172.44	4
Int	191.54	6
SinCos	170.53	3
Sindex	**139.77**	**1**
SRoot	210.33	7

All MLP have statistically significant differences under the Nemenyi test at 5% significance level; *the critical distance for the Nemenyi test at 1% significance level is 1.36, at 5% significance level is 1.16 and at 10% significance level is 1.06.

The results differ from the simulated time series presented before. *SIndex* is still ranked first with statistically significant better performance than the second best candidate. *AR* model follows, which outperforms *SinCos, Bin12* and *Bin11* in order of performance. This is in contrast to the results in table II, where the *AR* model ranked 5th. This can be attributed to the limited sample size as discussed in section 3. Note that the margin of difference between the *SinCos, Bin12* and *Bin11* is much smaller relatively to the difference of *SIndex* to *AR* or the difference of *SRoot* to the previous best model. *Int* and *SRoot* models perform as observed before, with the *SRoot* ranking last. This means that although the limited sample size affected the ranking between the AR model and the seasonal dummy models, deseasonalising for the case of deterministic seasonality still harms the performance significantly.

Using sMAPE the ANN models are compared against the benchmarks and the results across all time series are presented in table V.

Table V: Summary sMAPE across all time series

Model	Training	Validation	Test
AR	<u>16.30%</u>	13.08%	<u>20.10%</u>
Bin11	<u>15.80%</u>	12.53%	17.51%
Bin12	**13.87%**	12.49%	16.85%
Int	14.92%	12.47%	<u>17.85%</u>
SinCos	14.40%	12.07%	17.53%
Sindex	14.61%	**11.92%**	**16.70%**
SRoot	<u>19.44%</u>	15.49%	<u>20.69%</u>
EXSM	14.80%	17.58%	17.64%

The best performing model in each set is marked with bold numbers. The models that are outperformed by the EXSM benchmark are underlined

The *SIndex* model performs best, in agreement with table III for the simulated time series. On the test set the *AR*, *Int* and *SRoot* models fail to outperform the benchmarks. This shows that although the best trained *AR* model is less accurate than the *Bin11, Bin12* and *SinCos* in all training validation and test sets, its error has less extreme values, resulting in the lower mean rank observed in table IV. The *SRoot* model is consistently worse than all other ANN models providing more evidence that seasonal differences for the case of deterministic seasonality has a negative effect on accuracy. Overall, the results of the evaluation of the real time series dataset agree with the synthetic data evaluation.

7. Conclusions

Different methodologies to model time series with deterministic seasonality were evaluated. By exploring the theoretical properties of deterministic seasonality it was shown that the current debate in the literature, on how to model seasonality with ANN, does not address the problem correctly for this type of seasonality. Seven competing approaches to model the seasonality were evaluated and compared against exponential smoothing model on two datasets, a set of synthetic time series with known properties and a subset of the T-competition that has real transportation time series.

We found that for deterministic seasonality it is not advisable to deseasonalise the time series. Deseasonalising (through seasonal differences) hindered the model to accurately estimate the m_s and therefore affected forecasting accuracy negatively. The *SRoot* model performed consistently worse compared to all other ANN models and several times failed to outperform the exponential smoothing benchmarks.

Using S-1 or S dummy variables to code the seasonality did not have important differences for ANN models. For the synthetic time series, where the properties of the time series were controlled, the differences proved to be insignificant, while for the real time series using S dummy variables proved marginally better. Furthermore, we found that a sine-cosine encoding of the time series performed more robustly than binary seasonal dummy variables, resulting in significantly lower mean rank for the transportation dataset and minimal differences in the synthetic dataset. The sine-cosine encoding that was used here is not the equivalent to the trigonometric representation of seasonality, which uses sine and cosine waves of several frequencies. The degrees of freedom of the model were reduced by using a pair of sine and cosine of fixed frequency, making use of the approximation capabilities of MLPs, through the use of several hidden nodes. Note that the same did not seem to work well when a single integer dummy variable was used to code the seasonality. This seems to be the case due to the monotonic coding of each season.

We propose a coding that is based on seasonal indices. This approach used as a single explanatory variable a series of seasonal indices. This model outperformed significantly all competing ANN and the benchmarks for both datasets. Furthermore, this model was the most parsimonious, requiring a single additional input to model the deterministic seasonality. This can have significant implications for high frequency data that have long seasonal periods and the dimensionality of the input vector can become a problem for ANN training.

This study does not address thoroughly the issue of how to best estimate the seasonal indices. In the literature several methods have been suggested on how to estimate the seasonal indices of a time series. Here a very simple approach is employed that is found to be adequate. Under the assumption of deterministic seasonality the seasonal indices remain constant thus making the estimation easier. However, in real time series sample size and irregularities can possibly affect adversely their estimation, evidence of which was not found in this analysis, but has not been examined in detail. Similar difficulties would arise in the presence of multiple overlaying seasonalities. It is important to evaluate the robustness of the findings with different approaches to estimate the seasonal indices. In this study we focused on monthly time series. In future research, this study will be extended to a wider range of seasonal frequencies to validate the findings and provide a reliable solution for a range of practical applications.

References

Adya, M., & Collopy, F. 1998. How effective are neural networks at forecasting and prediction? A review and evaluation. *Journal of Forecasting*, 17: 481-495.

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

Anders, U., Korn, O., & Schmitt, C. 1998. Improving the pricing of options: A neural network approach. *Journal of Forecasting*, 17: 369-388.

Armstrong, J.S. 2006. Findings from evidence-based forecasting: Methods for reducing forecast error. *International Journal of Forecasting*, 22: 583-598.

Balkin, S.D., & Ord, J.K. 2000. Automatic neural network modeling for univariate time series. *International Journal of Forecasting*, 16: 509-515.

Callen, J.L., Kwan, C.C.Y., Yip, P.C.Y., & Yuan, Y.F. 1996. Neural network forecasting of quarterly accounting earnings. *International Journal of Forecasting*, 12: 475-482.

Canova, F., & Hansen, B.E. 1995. Are seasoanl patterns constant over time - a test for seasonal stability. *Journal of Business & Economic Statistics*, 13: 237-252.

Chen, Z., & Yang, Y. 2004. Assessing forecasting accuracy measures. *Working Paper*.

Cheung, Y.W., & Lai, K.S. 1998. Power of the Augmented Dickey-Fuller test with information-based lag selection. *Journal of Statistical Computation and Simulation*, 60: 57-65.

Crone, S.F., & Dahwan, R. 2007. Forecasting Seasonal Time Series with Neural networks: A Sensitivity Analysis of Architecture Parameters. *Proceedings of the International Joint Conference on Neural Networks, IJCNN'07, Orlando, USA, IEEE*.

Crone, S.F., & Kourentzes, N. 2007. Input variable selection for time series prediction with neural networks-an evaluation of visual, autocorrelation and spectral analysis for varying seasonality. *European Symposium on Time Series Prediction, ESTSP '07, Espoo, Finland*: 195-205.

Crone, S.F., & Kourentzes, N. 2009. Input-variable Specification for Neural Networks - an Analysis of Forecasting low and high Time Series Frequency. *Proceedings of the International Joint Conference on Neural Networks, IJCNN'09, Atlanta, USA, IEEE*: 3221-3228.

Curry, B. 2007. Neural networks and seasonality: Some technical considerations. *European Journal of Operational Research*, 179: 267-274.

Demšar, J. 2006. Statistical Comparisons of Classifiers over Multiple Data Sets. *J. Mach. Learn. Res.*, 7: 1-30.

Gardner, E.J. 2006a. Exponential smoothing: The state of the art--Part II. *International Journal of Forecasting*, 22: 637-666.

Gardner, E.S. 2006b. Exponential smoothing: The state of the art - Part II. *International Journal of Forecasting*, 22: 637-666.

Ghysels, E., & Osborn, D.R. 2001. *The Econometric Analysis of Seasonal Time Series*. Cambridge: Cambridge University Press.

Hagan, M.T., Demuth, H.B., & Beale, M.H. 1996. *Neural Network Design*. Boston: PWS Publishing.

Hibon, M., Young, P., & Scaglione, M. 2007. Forecasting Principles: T-competition.

Hill, T., O'Connor, M., & Remus, W. 1996. Neural network models for time series forecasts. *Management Science*, 42: 1082-1092.

Hippert, H.S., Bunn, D.W., & Souza, R.C. 2005. Large neural networks for electricity load forecasting: Are they overfitted? *International Journal of Forecasting*, 21: 425-434.

Hornik, K. 1991. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4: 251-257.

Hornik, K., Stinchcombe, M., & White, H. 1989. Multilayer feedforward networks are universal approximators *Neural Networks*, 2: 359-366.

Kvanli, A.H., Pavur, R.J., & Keeling, K. 2002. *Introduction to Business Statistics*. Mason, Ohio: Thomson/South-Western.

Lachtermacher, G., & Fuller, J.D. 1995. Backpropagation in Time-Series Forecasting. *Journal of Forecasting*, 14: 381-393.

Makridakis, S., & Hibon, M. 2000. The M3-Competition: results, conclusions and implications. *International Journal of Forecasting*, 16: 451-476.

Matas-Mir, A., & Osborn, D.R. 2004. Does seasonality change over the business cycle? An investigation using monthly industrial production series. *European Economic Review*, 48: 1309-1332.

Nelson, M., Hill, T., Remus, W., & O'Connor, M. 1999. Time series forecasting using neural networks: Should the data be deseasonalized first? *Journal of Forecasting*, 18: 359-367.

Osborn, D.R., Heravi, S., & Birchenhall, C.R. 1999. Seasonal unit roots and forecasts of two-digit European industrial production. *International Journal of Forecasting*, 15: 27-47.

Qi, M., & Zhang, G.P. 2001. An investigation of model selection criteria for neural network time series forecasting. *European Journal of Operational Research*, 132: 666-680.

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

Tashman, L. 2000. Out-of-sample tests of forecasting accuracy: an analysis and review. *International Journal of Forecasting*, 16: 437-450.

Vogl, T.P., Mangis, J.K., Rigler, A.K., Zink, W.T., & Alkon, D.L. 1988. Accelerating the convergence of the backpropagation method. *Biological Cybernetics*, 59: 257-263.

Zhang, G.P. 2001. An investigation of neural networks for linear time-series forecasting. *Computers & Operations Research*, 28: 1183-1202.

Zhang, G.P., & Kline, D.M. 2007. Quarterly time-series forecasting with neural networks. *Ieee Transactions on Neural Networks*, 18: 1800-1814.

Zhang, G.P., Patuwo, B.E., & Hu, M.Y. 2001. A simulation study of artificial neural networks for nonlinear time-series forecasting. *Computers & Operations Research*, 28: 381-396.

Zhang, G.P., & Qi, M. 2005. Neural network forecasting for seasonal and trend time series. *European Journal of Operational Research*, 160: 501-514.

Zhang, G.Q., Patuwo, B.E., & Hu, M.Y. 1998. Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, 14: 35-62.