Quarterly Time-Series Forecasting With Neural Networks

G. Peter Zhang and Douglas M. Kline

***Abstract*—Forecasting of time series that have seasonal and other variations remains an important problem for forecasters. This paper presents a neural network (NN) approach to forecasting quarterly time series. With a large data set of 756 quarterly time series from the M3 forecasting competition, we conduct a comprehensive investigation of the effectiveness of several data preprocessing and modeling approaches. We consider two data preprocessing methods and 48 NN models with different possible combinations of lagged observations, seasonal dummy variables, trigonometric variables, and time index as inputs to the NN. Both parametric and nonparametric statistical analyses are performed to identify the best models under different circumstances and categorize similar models. Results indicate that simpler models, in general, outperform more complex models. In addition, data preprocessing especially with deseasonalization and detrending is very helpful in improving NN performance. Practical guidelines are also provided.**

***Index Terms*—Forecasting, neural networks (NNs), quarterly time series, seasonality.**

I. Introduction

FORECASTING of time series that have seasonal variations remains an important problem for forecasters. Seasonality is observed in many forecasting problems in business, economics, and naturally occurring phenomena [1], [2]. In some applications, seasonality can drive a major part of movements in the quarterly or monthly time series [2].

There are numerous models and many different ways to analyze and forecast seasonal time series. Unfortunately, no single model or modeling approach is best for all seasonal time series under different conditions as suggested by a large number of theoretical and empirical studies including the recent M3 forecasting competition [3]. Traditional approaches to modeling seasonal time series such as the classic decomposition method require seasonal factors be removed before other factors can be analyzed. Seasonal autoregressive integrated moving average (SARIMA) models also require that the data be seasonally differenced first to achieve stationarity. This practice of seasonal adjustment or removal is due to the belief that seasonal fluctuations may dominate the remaining variations in a time series, causing difficulty in effectively dealing with other time-series components. On the other hand, the seasonal adjustment approach has been cautioned or criticized by several researchers in recent years [4], [5]. Ghysels *et al.* [6] suggest that seasonal adjustment might lead to undesirable nonlinear properties in univariate time series. Ittig [7] shows that the traditional method for generating seasonal indexes is biased when there is a trend component. In addition, different forms of the trend can impact the estimate of the seasonal components and affect the level of overestimation in the seasonal variation [8]. Hyndman [9] argues that the interaction between trend and seasonality is built into many seasonality models, which multiplies the task of choosing the correct model form, and can further confound the selection of seasonal methodologies. Furthermore, several empirical studies find that seasonal fluctuations are not always constant over time and at least in some time series, seasonal components are not independent of nonseasonal components, and thus may not be separable. The difficulty in distinguishing seasonal from nonseasonal fluctuations is the major motivation behind the recent development of seasonal unit root models and periodic models that take explicit consideration of seasonal variations [5]. de Gooijer and Franses [10] point out that "although seasonally adjusted data may sometimes be useful, it is typically recommended to use seasonally unadjusted data."

As a consequence of these conflicting results and recommendations, the practical issues of how to best deal with seasonal time series and which seasonal model is the most appropriate for a given time series are largely unsolved. In fact, adjustment for systematic events including seasonality is considered to be an area that still has a strong need for further research in developing and advancing forecasting principles [11].

This paper aims to provide some evidence on the effectiveness of neural network (NN) models on forecasting seasonal time series. More specifically, we explicitly investigate the practical issue of how to best use NNs to forecast *quarterly* time series using a large set of data from the M3 competition. Our research is motivated by the following observations. First, during the last decade, NNs have received enormous attention from both practitioners and academics across a wide range of disciplines. They are found to be a viable contender to various linear and nonlinear time-series models [12]–[14]. NNs, being nonlinear and data-driven in nature, may be well suited to model seasonality interacting with other components, and may relieve the practical burden of *a priori* model selection. Although there are several studies focusing on seasonal time-series forecasting,

Manuscript received May 4, 2006; revised January 18, 2007; accepted February 7, 2007. This work was supported in part by the International Institute for Forecasters and SAS under grant to support research on principles of forecasting. The work of G. P. Zhang was supported by the J. Mack Robinson College of Business, Georgia State University, Atlanta. The work of D. M. Kline was supported by the Cameron School of Business Research Fund, University of North Carolina at Wilmington.

G. P. Zhang is with the J. Mack Robinson College of Business, Georgia State University, Atlanta, GA 30303 USA (e-mail: gpzhang@gsu.edu).

D. M. Kline is with the Cameron School of Business Research, University of North Carolina at Wilmington, Wilmington, NC 28403 USA (e-mail: klined@uncw.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNN.2007.896859

findings are mixed. For example, among the major studies of NNs used for seasonal time series, Sharda and Patil [15] find that NNs are able to directly model seasonality, and preseasonal adjustment of data is not necessary. Nelson *et al.* [16], however, conclude just the opposite and NNs trained on deseasonalized data perform significantly better than those with raw data. Considering both seasonal and trend components in real time series, Zhang and Qi [17] find that not only preadjustment of seasonality is important, but a combined approach of detrending and deseasonality is most effective in forecasting performance.

Second, most published NN studies use monthly data. Quarterly data have characteristics that are different from monthly data. Little systematic studies have been conducted on quarterly time series with NNs. Swanson and White [18], [19] investigate the performance of NNs in modeling and forecasting nine quarterly seasonally adjusted U.S. macroeconomic time series and report positive results with NN models. However, these studies focus on the model selection issue and do not consider a number of modeling issues such as seasonality and trend treatment. Callen *et al.* [20] report a study on NN forecasting of quarterly accounting earnings and conclude that "NN models are not necessarily superior to linear time-series models even when the data are financial, seasonal, and nonlinear." This paper uses the seasonally unadjusted data and does not consider alternative approaches to dealing with both trend and seasonal components in the data, which may explain the negative findings toward NNs as recent studies tend to indicate that properly modeling seasonality is the key to the improved forecasting performance.[1]

Third, previous studies are either application specific or limited in scope and size (i.e., the number of data sets) and no systematic effort has been devoted to studying the general issue of how to use NNs to best model and forecast seasonal time series. That is, these studies focus on either a single application or on data sets that are relatively homogeneous. Therefore, findings from them may not be generalizable. For example, Alon *et al.* [21] and Chu and Zhang [22] consider forecasting issues with one aggregate retail sales time series. Swanson and White [18] use nine macroeconomic time series such as unemployment rate, industrial production index, gross national product, and net exports of goods and services, etc. In [17], ten aggregate economic time series in retail sales, industrial production, and housing starts are employed. In [20], a large sample size of 296 time series is used, but all of the same type of accounting earnings. Although in [14]–[16] relatively large sample sizes from the M- and M3-competitions are used, these studies are limited with regard to the number of models considered and the number of issues in dealing with seasonal and trend time series. In this paper, we aim to provide a more comprehensive and systematic study on how NNs can be used for quarterly time series with many more models and data sets from diverse areas.

Last, we would like to evaluate the effects of incorrectly estimating NN models for quarterly time series. Since there are numerous approaches to building NNs to deal with different time series, some approaches may not be appropriate. For example, if a time series contains a trend component, an NN structure that does not take this into consideration may not be the best model. On the other hand, an NN model that has a seasonal lag input may not be the best for nonseasonal time series.

The rest of this paper is organized as follows. In Section II, we review several relevant studies in the NN literature on modeling and forecasting quarterly or monthly time series. In Section III, we describe the research methodology used in our empirical investigation. Results are reported in Section IV. Section V provides the summaries and conclusions.

II. Forecasting Seasonal Time Series With NNs

A large body of literature exists in seasonal time-series analysis and forecasting. Some of the recent developments in seasonality modeling can be found in [4], [5], and [23]. In this section, our focus is on recent research efforts in seasonal time-series modeling using the NNs.

In an early effort of using NNs for seasonal time-series forecasting, Sharda and Patil [15] conduct a comparative study between NNs and ARIMA models. Among the 111 time series selected from the M-competition [24], 13 are annual series, 20 are quarterly, and 68 are monthly. They use a direct modeling approach without considering any specific issue of seasonality modeling. Results show that for quarterly and monthly time series, the performance of NNs is similar to that of ARIMA models. Thus, they conclude that NNs are able to "incorporate seasonality automatically."

Hill *et al.* [12] use a very similar set of quarterly and monthly time series used in [15] and obtains much better results with NNs in terms of statistical significance when compared with the traditional models. This difference in performance between two studies may be attributed to the prior seasonal adjustment or deseasonalization before NN model building, indicating the importance of removing seasonality in improving NN forecasting performance. However, only one NN architecture is considered and employed in [12] for all quarterly or monthly time series.

In [20], a sample of 296 quarterly accounting earnings series is used to compare the performance of NNs with that of several linear time-series models. Although the size of the data set is quite large, all the time series are quite short, containing 89 data points. It is found that on average, linear models are significantly better than NNs with a rolling sample approach of 40 quarters each in length for model building. However, as discussed earlier, Callen *et al.* [20] do not consider ways to handle seasonal variations and raw data are directly modeled in NNs, which may explain the inferiority of NNs. In addition, the use of a relatively small portion of sample for NN training may cause instability in model estimation especially when considering the direct seasonality modeling approach.

A recent study [25] using a sample of 283 quarterly earnings series from a wide range of industries suggests that results reported in [20] can be dramatically improved. Although each series in this paper is still short with only 40 observations and the direct seasonal modeling approach is used, the authors are able to obtain significantly better forecasting results with NNs than those with linear time-series models, especially when fundamental accounting variables, such as accounts receivables, inventory, and capital expenditures, are incorporated in the NN modeling.

[1]We have attempted to replicate their study. However, we were not able to obtain the data sets after a few contacts with the authors.

In [26], a case study is presented for NNs in modeling and forecasting the well-known airline series popularized by [27]. The airline data contain 12 years of monthly totals of international airline passengers and have a clear upward trend with distinctive multiplicative seasonal patterns. This time series is well studied and documented with linear seasonal time-series methods and thus provides a good benchmark for NNs. The focus of [26] is to use a variety of in-sample model selection criteria including Akaike information criterion (AIC) and Bayesian information criterion (BIC) to select the best neural model for the *raw* data. The seasonality is modeled by considering appropriate input lag such as lag 12 in the modeling process. The results indicate that NNs in general do not perform better than the Box–Jenkins model in out-of-sample forecasting, even with the "best" NN models selected and several variations of modeling process including using logarithms, removing trend, and applying first and seasonal differencing for data preprocessing.

Swanson and White [18], [19] conduct several comparative forecasting experiments between NNs and linear econometric models on nine macroeconomic time series. Although the data are quarterly, most of them are seasonally adjusted and/or log differenced. Thus, their studies do not deal with seasonality directly. However, it is worth pointing out that they report positive results with NNs compared to other linear models examined.

In [30], 24 time series of annual change in monthly industrial production in three European countries are used for a comparative study between NNs and linear autoregressive models. Unlike in [18] and [19], where seasonally adjusted data are used in NN modeling, Heravi *et al.* [30] choose to use seasonally unadjusted series due to the concern of potential nonlinearity induced by the seasonal adjustment procedure. Although this direct seasonality modeling approach yields positive results with NNs in terms of the prediction of direction changes, linear models generally outperform NNs judging by root-mean-squared (rms) error.

Terasvirta *et al.* [31] examine a number of linear and nonlinear models including NNs for forecasting 47 monthly macroeconomic variables in seven developed economies. For those series that are not seasonally adjusted, monthly dummy variables and 12 lags of observations are used in the linear models and NNs, respectively, to model seasonality. The results for NNs are mixed with the model using Bayesian regularization having better forecasting accuracy than other models.

Nelson *et al.* [16] focus on the issue whether the data should be deseasonalized first in time-series forecasting using NNs. The study uses the 68 monthly series as in [12] and [15]. Forecasting performance is compared between NNs built on prior-deseasonalized data and those with raw data. The results clearly show the advantages of prior deseasonalization in improving NN performance. Thus, [16] points out that previous mixed results in seasonal time-series forecasting may be "due to inconsistent handling of seasonality".

Two studies [21], [22] report comparative results between NNs and a variety of linear models in forecasting monthly aggregate retail sales. In [21], NNs are used to directly model seasonal variation by using 12 lags of observations as input variables. Using two out-of-sample periods, they find that NNs perform the best in the first period which is characterized as more volatile in terms of supply push inflation, recessions, and high interest and unemployment rates and the Box-Jenkins models slightly outperform NNs in the second period which is more stable in terms of the macroeconomic factors. In contrast, [22] considers a variety of ways of modeling seasonality including deseasonalizing the time series and using seasonal dummies and trigonometric functions. Using five moving out-of-samples, they find that the overall best forecasting performance is achieved with a NN built on deseasonalized data.

Zhang and Qi [17] provide some further evidence on the effectiveness of prior seasonal adjustment in NN forecasting improvement based on both simulation and real-data results. It finds that NNs are not able to deal with seasonality and trend effectively with raw data and either deseasonalization or detrending can reduce forecasting errors dramatically. Furthermore, a combined approach of both detrending and deseasonalization is the most effective approach for NN modeling.

From the aforementioned review of the relevant literature, we make the following observations. First, mixed results have been reported on the relative merits of NNs in modeling and predicting seasonal time series. Different ways to deal with seasonality and/or model building can have dramatic impact on the performance of NNs. Second, no comprehensive studies have been performed with regard to large data set and various modeling considerations for seasonality. Third, a majority of studies use monthly data and only a few have focused on quarterly time series. Finally, deseasonalization is very effective. In almost all studies that report significantly better results with NNs, data are typically deseasonalized first before fitting an NN model.

III. Research Methodology

In order to have a comprehensive understanding of the effect of NN modeling on the forecasting ability of NNs, we have conducted a large-scale empirical study. A large set of quarterly time series from M3-competition is used in this investigation along with a large number of NN model structures. In addition, we examine the impact of several different data preprocessing approaches on NN performance. Several research questions are of interest to us as follows.

1) Is there an overall best way to model quarterly time series with NNs?
2) Are NNs able to directly model seasonality in quarterly time series? Given that NNs are data-driven and can model arbitrary functional forms, it is theoretically possible that an NN could directly model seasonality. However, there may be practical limitations, notably data sufficiency and nonlinear optimization issue that could make this approach unsuccessful.
3) Given the controversies around the seasonal adjustment approach, should the data be seasonally adjusted first? Should the data be preprocessed first, removing all significant patterns such as seasonality and trend?
4) Is inclusion of seasonality information such as seasonal dummy variables or trigonometric variables in NN modeling helpful in improving forecasting performance? Traditional seasonal methods utilize the information about seasonality regarding which season the data points are in.

TABLE I
CHARACTERISTICS OF QUARTERLY TIME SERIES FROM M3-COMPETITION

	Frequency		Sample Size		
Type	Count	Percent (%)	Min	Median	Max
Demographic	57	7.5	27	44	64
Finance	76	10.1	27	42	64
Industry	83	11.0	24	56	64
Macro	336	44.4	16	44	45
Micro	204	27.0	28	36	38
Total	756	100.0	16	44	64

NNs are known to be parameter-heavy. Including additional variables in an NN model can greatly increase the number of parameters in the model and cause data insufficiency. However, the added information may simplify the problem, and thus, they require fewer hidden nodes to approximate the underlying functional form. It is unclear how these concerns will play out in practice.

A. Data

In this paper, we use the 756 quarterly time series from the M3 forecasting competition [3]. The M3 competition data set is well studied, contains time series of various lengths and different types, and exhibits linear and nonlinear patterns [32]. Thus, the data set provides a sufficient test bed on which various models can be built and compared, and general conclusions may be obtained. Table I provides a summary of sample characteristics for the data set with regard to the frequency and size for each type of time series. It is clear that most series are macroeconomic (44%) and microeconomic variables (27%) and the sample size varies from 16 to 64 with the median length of 44 observations. In fact, 44 is not only the median, but also the mode of the data set with 249 time series having this length (33%).

For each series, we consider two data preprocessing approaches. One is the application of the natural logarithm to each observation, which we call "log," for the data transformation method and the other is the detrending and deseasonalization in addition to the log transformation, for which we call the "full" transformation method. For detrending, we fit a linear trend, and then subtract the estimated trend component from the raw data. For deseasonalizing, we employ the method of seasonal index based on centered moving averages, following the classic additive decomposition. The parameters for the detrending and deseasonalization are calculated with only the in-sample data. The estimated seasonal index is then used for seasonal adjustment of the time series and for out-of-sample forecasting. Note that there are other methods to remove seasonality. For example, Atiya *et al.* [33] subtract the time series from the seasonal average to obtain seasonally adjusted series. A novel algorithm based on Fourier transformation to deal with the seasonality is also proposed in [33].

In addition to these transformations, all data are scaled to be within $(-1, 1)$ before presenting to the NN. We make the distinction between a data transformation that is performed to address characteristics of a particular time series, and a data scaling that is applied to facilitate NN training. After NN modeling, the data are rescaled back following the reverse of the data transformation and scaling, and all the performance measures are calculated based on the original scale of the data.

In summary, we consider the following three data types: raw unprocessed data (raw), log-transformed data (log), and fully transformed data (full).

We also realize that although all 756 time series are quarterly, they are not necessarily all seasonal. Therefore, we try to distinguish seasonal time series from nonseasonal ones and then some insights may be obtained to see whether some models perform better than others on seasonal versus nonseasonal series. In this paper, we employ the following simple rule-of-thumb [34]: If the four-period autocorrelation is greater than $2/\sqrt{n}$, where n is the sample size, then the series is classified as seasonal; otherwise, it is nonseasonal. Using this rule, we find that 473 time series are judged as seasonal and the rest are nonseasonal.

The last 30% of each data series is retained as holdout sample or out-of-sample to measure forecast accuracy of each model. The remaining data set is used as the in-sample for model building. Although all M-competitions use the practice of leaving the last eight data points for out-of-sample testing, because of the sample size limit, we elect to choose the aforementioned rule in data splitting because the data set we have from the M3-competition does not include the last eight points reserved by the competition organizer for performance evaluation.

B. Models

We identify 48 NN models based on different possible combinations of lagged observations, seasonal dummy variables, trigonometric variables, and time index as inputs to the NN model. Since one-step-ahead forecasting is the focus of this paper, we use only one output in all NN structures. Table II summarizes the models used in this paper based on the relationship between the output variable (y_t) and a variety of possible inputs such as past lagged observations $(y_{t-1}, y_{t-2}, y_{t-3}, \ldots)$ and seasonal dummy variables. Models 1–6 consider six combinations of pure lagged observations as inputs to the NNs. Note that models 1–3 can be treated as nonseasonal while models 4–6 are seasonal as they include the observation four quarters before. These six models serve as the base models upon which we add a few more inputs to form other models. For example, the use of trigonometric or seasonal dummy variables may improve forecasting performance. Thus, we add seasonal dummy variables to each base model to form models 7–12. For quarterly data, we need only three dummy variables and quarters 1–4 are coded as (1, 0, 0), (0, 1, 0), (0, 0, 1), and (0, 0, 0), respectively. Models 13–18 are similar to models 7–12, except we use trigonometric variables of $\sin(2\pi t/4)$ and $\cos(2\pi t/4)$ instead of dummy variables. Models 19–24 add annual difference as one more input to

TABLE II
MODELS USED IN THIS PAPER

Model	Description
1	$y_t=f(y_{t-1})+\varepsilon_t$
2	$y_t=f(y_{t-1},y_{t-2})+\varepsilon_t$
3	$y_t=f(y_{t-1},y_{t-2},y_{t-3})+\varepsilon_t$
4	$y_t=f(y_{t-1},y_{t-2},y_{t-4})+\varepsilon_t$
5	$y_t=f(y_{t-1},y_{t-2},y_{t-3},y_{t-4})+\varepsilon_t$
6	$y_t=f(y_{t-1},y_{t-4})+\varepsilon_t$
7-12	*Models 1-6 with seasonal dummies*
13-18	*Models 1-6 with trig variables*
19-24	*Models 1-6 with annual difference*
25-30	*Models 1-6 with time index t*
31-36	*Models 1-6 with seasonal dummies and t*
37-42	*Models 1-6 with trig variables and t*
43-48	*Models 1-6 with annual difference and t*

the base models. Annual difference is defined as $(y_{t-1} - y_{t-5})$ and can be considered a rough estimate of trend [20]. Models 25–30 add the time index t to model the trend component. Models 31–48 are the same as models 7–24 except that, for models 31–48, we have added time index variable.

These 48 models represent a wide range of possible NN input–output structures for modeling seasonal time series. Although there could be more possible NN structures with different input values, we believe these models are representative in practice to model and forecast quarterly time series. Some of them are suitable for time series without trend and/or seasonal patterns. Some will be useful for trend or seasonal time series while still others may be able to handle multiple components such as trend, seasonality, and other significant patterns. These layered models also allow us to see whether one particular group of models (such as the six base models) is more effective than others.

C. NNs

For each time series, NN models are built with an in-sample selection approach. We use the standard three-layer feedforward NNs, which is by far the most popular type of NN model for time-series forecasting, although other types of NNs may be equally competent [35]–[37]. Node biases are used at the hidden and output layers. A sigmoid transfer function is employed at each hidden node and a linear transfer function is used at each output node. As mentioned earlier, we use one output node for one-step-ahead forecasting. The number of input nodes is based on the models identified in the last section. As the number of hidden nodes is not possible to determine in advance, empirical experimentations are needed to determine this parameter. Because of the very small sample size for many of the series, we limit our experimentation to only six possible values of hidden nodes: 0, 1, 3, 5, 7, and 9. The value of 0 is included to have a benchmark linear autoregressive model. The best number of hidden nodes is determined by using the original Akaike's information criterion (AIC).[2]

[2]As an early try, we used the generalized cross-validation (GCV) metric as an in-sample selection criterion [14]. GCV has a cost parameter that has to be estimated subjectively. We used 2.0 as in [14]. However, with small sample size, some technical difficulty can arise in using GCV such as the division by zero and a small change in the parameter causing a big difference in how sensitive the measure is to the size of the model. This is the reason we abandoned this criterion. A modified version of AIC was recently proposed in [38].

We use the Matlab NN toolkit for building NNs and making forecasts. The Levenburg–Marquardt training algorithm is used to minimize the sum of the squared errors (SSE) on each network. Training is stopped after 300 epochs, or until the algorithm stops improving the SSE. Each NN architecture is trained five times using different random starting weights. Then the best one is kept based on the lowest SSE. All in-sample data presented to the NN (inputs and targets) are scaled to between $(-1, 1)$ using Matlab's "premnmx" function, which returns parameters to accomplish the reverse transformation. All out-of-sample data are transformed using the in-sample parameters determined in the training stage.

Our experimental design can be represented by the following model:

$$y = f(D, T, M)$$

where

- y the performance measure;
- D the data series used (756 levels);
- T the data transformation performed on the data series (three levels);
- M the model form used for the forecast (48 levels).

That is, our study generates for each series three different data sets based on whether transformation is used and if so which one is used, and for each data set, 48 different models are built, each with six levels of different hidden nodes. The total number of observations we obtain with the best NN architecture is 108864 $(= 756 \times 3 \times 48)$.

The following pseudocode describes the methodology used in training NNs.

For each data series {

  For each data transformation {

    Linearly scale in-sample data to $(-1, 1)$, retaining parameters

    For each model form {

      For NN architectures with hidden nodes of $(0, 1, 3, 5, 7, 9)$ {

        Train five NNs from random starting parameter weights.

        Keep the best of the five based on SSE

      }

      Linearly scale the out-of-sample data using parameters from in-sample (from above)

      Using the best network architecture and parameter set, perform forecast on out-of-sample data

      Unscale the forecasts

      Untransform the forecasts using the appropriate inverse transformation

```
            Record the MAPE, RMSE, MAE on the unscaled
            out-of-sample forecasts and actual observations
        }
    }
}
```

IV. Empirical Results

Although we use a variety of performance measures in this paper including root-mean-squared error (RMSE), mean-absolute error (MAE), and mean-absolute-percentage error (MAPE), general results do not change much with these measures. Therefore, to save space, we report only the results with MAPE in this section.

Because the two types of data transformation used in this paper are based on the in-sample data, the transformations applied to the out-of-sample yield a few "outliers." For example, to facilitate NN training, we scale all in-sample data to be within $(-1, 1)$ based on the minimum and maximum values in the in-sample data. However, when we apply the same formulas to observations in the out-of-sample, a few observations are outside the range of $(-1, 1)$. When we later apply antilogarithm or other inverse transformations, the errors become inflated, causing a few very large outliers in the performance measure. For example, we find that the largest MAPE is more than 8.8×10^8. For this reason, we decide to remove the results with MAPE greater than 300%. In addition, some cases have a very small sample size (e.g., 16) and the NN model has a large number of parameters (e.g., model with seven or nine hidden nodes), resulting in many more parameters than observations. In these situations, we decide not to fit the model, and thus, no observations are obtained for these cases. The previous discussion yields a useful total number of observations of 106 831.

Table III shows the overall ANOVA result with two main factors of model (MODEL) and data preparation (PREP) and the data set (SERIES) as the blocking factor. It is clear that the blocking factor is highly significant, suggesting the usefulness of the blocking design. MODEL is significant at the 0.05 significance level while PREP is highly significant with the p-value less than 0.0001. There is no significant interaction effect between these two main factors.

A number of planned contrasts have been performed for different groups of models. Specifically, we look at the following six paired contrasts: 1) models with dummy variables versus models with trig variables, 2) models with trend t (the last 24 models) versus those without t (the first 24 models), 3) base models (the first six models) versus base models plus dummies (models 7–12), 4) base model versus base models plus trig variables (models 13–18), 5) base models versus base models plus annual differences (models 19–24), and 6) base models versus base models plus t (models 26–30). The contrast results across all three data preparations are reported in Table IV. The only two significant contrasts involve the use of trend index t. First, there is a significant difference between models using t and those without $t (p-\text{value} < 0.0001)$. The positive sign of the contrast estimate indicates that models without t provide, on average, more accurate forecasts than models with t. The second significant contrast occurs between the base models and the base models with t. The negative sign of the estimate shows that the base models are more accurate than the models with t. Table IV may suggest that, with all three data preparations considered together, using t is not a good strategy.

TABLE III
Overall ANOVA Result

Source	DF	Mean Square	F-value	P-value
SERIES	755	19.6020	15.74	<.0001
MODEL	47	1.7441	1.40	0.0363
PREP	2	120.6193	96.85	<.0001
MODEL*PREP	94	1.3980	1.12	0.1962

We perform multiple comparisons for the two significant factors of MODEL and PREP with the Duncan's multiple range test. Fig. 1 summarizes the overall difference among 48 models. In general, except for the obvious outlier occurred at model 39, there is an increasing trend from model 1 to model 48. For each model or data preparation method, the Duncan grouping procedure assigns one or more letters to represent the group(s) in which the mean performance measure belongs. Different letters indicate that groups are significantly different at the 0.05 significance level. The largest mean is always associated with letter A, the second largest is denoted B, and so on. From the figure, we find that model 1 (indicated by letter C in the figure) is the most accurate model with the lowest average MAPE, which is significantly lower than that of all other models. Model 39 (with A) is the least accurate model, followed by model 44 (with B). All other models perform similarly with no significant difference between them (all with letters B and C). On the other hand, Table V shows the multiple comparisons among the three data preparation methods of raw, log, and full. Although on average the models built on the log-transformed data perform better than those on the raw data, their performances are similar or are not significantly different. However, the full transformation yields significantly lower average MAPE than both the raw and log transformation methods.

Table VI gives the separate ANOVA results for each level of PREP with SERIES as the blocking factor. For the raw data and fully preprocessed data, MODEL is highly significant, while, for the log-transformed data, MODEL is not significant at the 0.05 level.

Fig. 2 plots the forecasting performance (MAPE) of various models with regard to three data preparation approaches: raw in Fig. 2(a), log transformation in Fig. 2(b), and full transformation in Fig. 2(c). Overall, we see that when data are unprocessed, model performance varies quite dramatically while with log and full transformations, variations in performance among different models are much smaller. When the data are log transformed, models 37–45 exhibit higher variations with model 39 performing significantly worse than all other models. When the data are fully transformed, models 9–19 vary considerably with models 12 and 18 performing significantly worse than the other models. For unprocessed raw data, models 21 and 22

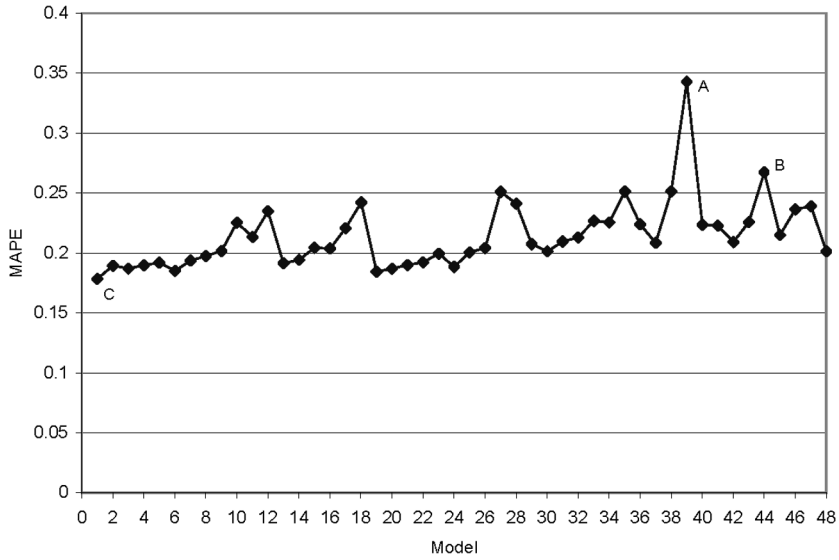

Fig. 1. Overall model performance with Duncan grouping.

TABLE IV
CONTRAST RESULT: OVERALL

Contrast	Estimate	Standard Error	t-value	P-value
dum vs trig	-0.1338	0.1164	-1.15	0.2505
trend t vs no t	0.6517	0.1641	3.97	<.0001
base vs base+dummy	-0.1209	0.0817	-1.48	0.1390
base vs base+trig	-0.1226	0.0814	-1.51	0.1320
base vs base+annDiff	-0.0076	0.0814	-0.09	0.9254
base vs base+t	-0.1839	0.0812	-2.27	0.0235

TABLE V
MULTIPLE COMPARISON: OVERALL

Data	Mean	Duncan Grouping
Raw	0.2448	A
Log	0.2504	A
Full	0.1466	B

(with three lagged values and one annual difference as inputs) perform the best. For the full transformed data, models 1–3 are the overall best performers. However, when the data are log transformed, except for the model 39, all other models do not perform significantly differently according to the Duncan's multiple range test, although simple models such as the base models perform slightly better than others. It is important to note that the scales used in Fig. 2(a)–(c) are different. In fact, with raw data, MAPE values are around a mean of 0.2448 across 48 models. For log-transformed data, this mean is about 0.25 and for the fully transformed data, the mean is 0.1466. Thus, it is clear that the full transformation can significantly reduce forecasting error for all models. It is also evident from these figures that when the data are not preprocessed, relatively more complex models are needed while for preprocessed data especially those with full transformation, simple models predict much better than more complex models.

The contrast results by data preparation are given in Table VII. When considering different data preparation strategies, we see somewhat different results compared to the overall results reported in Table IV. If the data are unprocessed raw series, then there are three significant contrasts between models using t and those without t(estimate $= 0.0253, p$-value $= 0.0278$), between the base models and the base models plus dummies (estimate $= -0.0131, p$-value $= 0.0227$), and between the base models and the base models with annual difference (estimate $= 0.0311, p$-value < 0.0001). When the data are log transformed, the only significant contrast occurs between models using t and those without t(p-value < 0.0125). If the data are fully transformed, there are four highly significant contrasts between models using t and those without t(estimate $= 0.8129, p$-value < 0.0001), between the base models and the base models plus dummies (estimate $= -0.2830, p$-value $= 0.0034$), between the base models and the base models plus trig variables (estimate $= -0.3061, p$-value $= 0.0015$), and between the base models and the base models with t(estimate $= -0.3341, p$-value $= 0.0005$).

Table VIII reports several descriptive statistics including minimum, maximum, and average MAPE of all models based on the classification of data type and data preparation. Several observations can be made. First, from the mean MAPE perspective, it is clear that across all five data types, full transformation of data is the most effective in terms of the model forecasting performance. The log transformation alone does not provide much advantage over the raw data. In fact, in almost all types, the average MAPE associated with the log transformation is

TABLE VI
ANOVA RESULTS BY PREP

PREP	Source	DF	Mean Square	F-value	P-value
Full	SERIES	755	13.1701	22.75	<.0001
	MODEL	47	0.8780	1.52	0.0128
Log	SERIES	755	7.5291	2.44	<.0001
	MODEL	47	3.6659	1.19	0.1755
Raw	SERIES	755	4.7686	2336.7	<.0001
	MODEL	47	0.0130	6.38	<.0001

TABLE VII
CONTRAST RESULT BY PREP

PREP	Contrast	Estimate	Standard Error	t-value	P-value
Raw	dummy vs trig	0.0000	0.0082	0	0.9988
	trend t vs no t	0.0253	0.0115	2.2	0.0278
	base vs base+dummy	-0.0131	0.0057	-2.28	0.0227
	base vs base+trig	-0.0085	0.0057	-1.5	0.1347
	base vs base+annDiff	0.0311	0.0057	5.45	<.0001
	base vs base+t	-0.0086	0.0057	-1.52	0.129
Log	dummy vs trig	-0.3726	0.3172	-1.17	0.2402
	trend t vs no t	1.1164	0.4470	2.5	0.0125
	base vs base+dummy	-0.0669	0.2226	-0.3	0.7637
	base vs base+trig	-0.0532	0.2219	-0.24	0.8103
	base vs base+annDiff	0.0053	0.2219	0.02	0.9811
	base vs base+t	-0.2089	0.2212	-0.94	0.3449
Full	dummmy vs trig	-0.0284	0.1375	-0.21	0.8362
	trend t vs no t	0.8129	0.1937	4.2	<.0001
	base vs base+dummy	-0.2830	0.0965	-2.93	0.0034
	base vs base+trig	-0.3061	0.0962	-3.18	0.0015
	base vs base+annDiff	-0.0593	0.0962	-0.62	0.5377
	base vs base+t	-0.3341	0.0959	-3.48	0.0005

TABLE VIII
MODEL PERFORMANCE BY TYPE OF SERIES AND DATA PREPARATION

	Data Preparation								
	Raw			Log			Full		
Type	Min	Max	Mean	Min	Max	Mean	Min	Max	Mean
Demographic	0.1867	0.2144	0.1999	0.1317	1.8979	0.2372	0.0983	0.2557	0.1454
Finance	0.2138	0.2471	0.2292	0.1921	0.3629	0.2373	0.0935	0.4000	0.1724
Industry	0.3116	0.3816	0.3552	0.2290	1.9966	0.3213	0.1157	0.2467	0.1773
Macro	0.1818	0.2145	0.1948	0.1447	1.0619	0.2056	0.0310	0.0845	0.0582
Micro	0.2892	0.3066	0.2982	0.2525	0.7149	0.3020	0.1781	0.7512	0.2654

worse than that with the raw data. Second, data transformations can help improve the best model performance but at the same time also increases the variability of the model performance. The minimum MAPE for both log and full transformations is much smaller than that for raw data across all data types. However, the range between the minimum and the maximum is often higher when data are log or fully transformed than when the data are unpreprocessed. One explanation for why the worst model with the transformed data is worse than that with the raw data is the issue with data transformation formulas used in in-sample applied to out-of-sample, which may result in several unusual outliers. Finally, NNs do perform differently with different types of data. For the raw data, the best mean MAPE $(= 0.1948)$ is for the macrodata, and the worst mean MAPE $(= 0.2982)$ is for the microdata. For the log-transformed data, the best MAPE is 0.2056 for the macrodata versus the worst

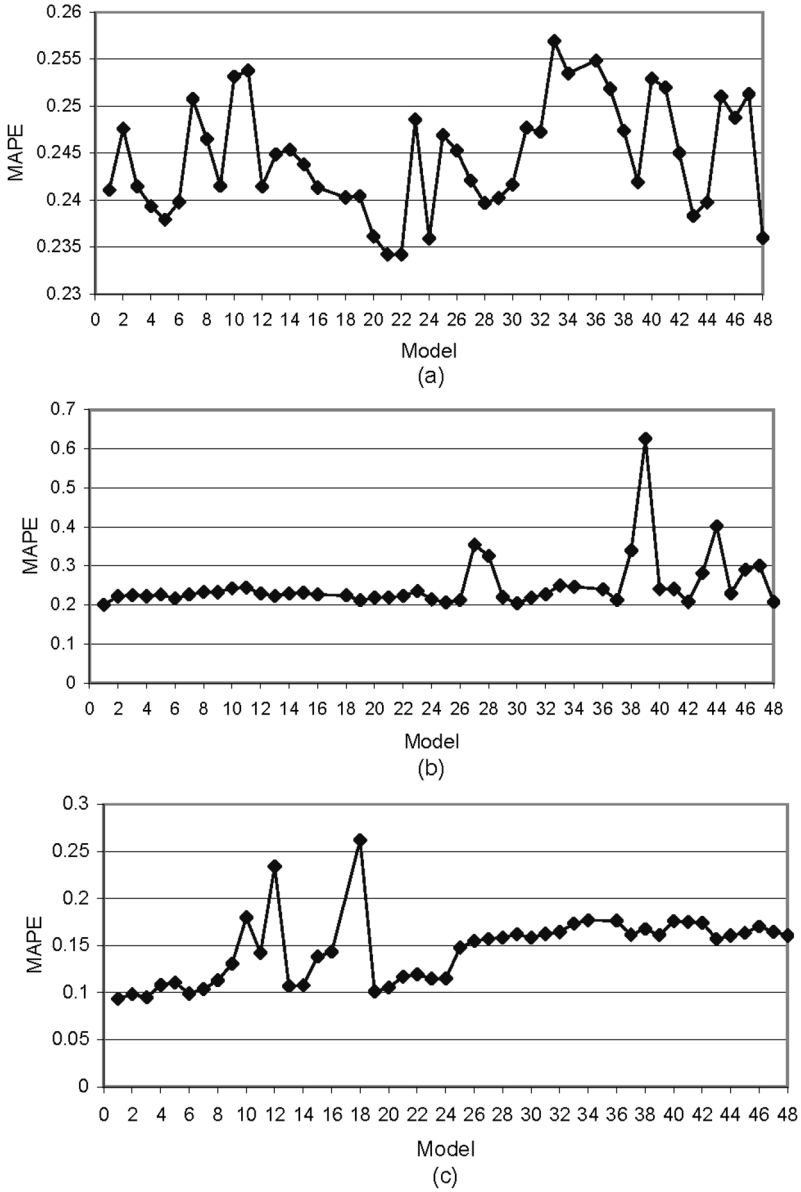

Fig. 2. Model performance by data preparation method. (a) Model performance with raw data. (b) Model performance with log transformation. (c) Model performance with full transformation.

TABLE IX
MODEL PERFORMANCE BY SEASONALITY AND DATA PREPARATION

	Data Preparation								
	Raw			Log			Full		
Seasonality	Min	Max	Mean	Min	Max	Mean	Min	Max	Mean
Yes	0.2420	0.2663	0.2529	0.2086	0.7176	0.2540	0.0878	0.1943	0.1461
No	0.2212	0.2429	0.2319	0.1789	0.6979	0.2447	0.0960	0.5018	0.1485

MAPE of 0.3213 for the industry data. For the fully transformed data, the best MAPE is 0.0582 for the macrodata versus the worst of 0.2654 for the microdata. The macrodata are the only ones that consistently give the best forecasting performance with all three data preparations.

In Table IX, we give forecasting performance of all models with regard to the data preparation method and whether the data exhibit some seasonality based on the rule of thumb discussed earlier. Although full transformation again is very effective in improving NN forecasting performance, log transformation does not provide much advantage over raw data. In addition, we find that the model performance is quite similar for those series that have seasonality and those that do not.

Table X summarizes the distribution of the best hidden node selected across three data preparations. In general, we see a clear decreasing order of frequency or relative frequency from 0 to 9 hidden nodes regardless of whether data are preprocessed or which data transformation method is used. The only exception

TABLE X
HIDDEN NODE DISTRIBUTION

	Data Preparation							
	Raw		Log		Full		Total	
Hidden	Frequency	Percent	Frequency	Percent	Frequency	Percent	Frequency	Percent
0	28490	80	18179	51.05	15033	42.21	61702	57.76
1	6795	19.08	15537	43.63	19752	55.47	42084	39.39
3	291	0.82	1606	4.51	788	2.21	2685	2.51
5	31	0.09	267	0.75	37	0.1	335	0.31
7	4	0.01	17	0.05	1	0	22	0.02
9		0	2	0.01	0	0	3	0.00
Total	35612		35608		35611		106831	100

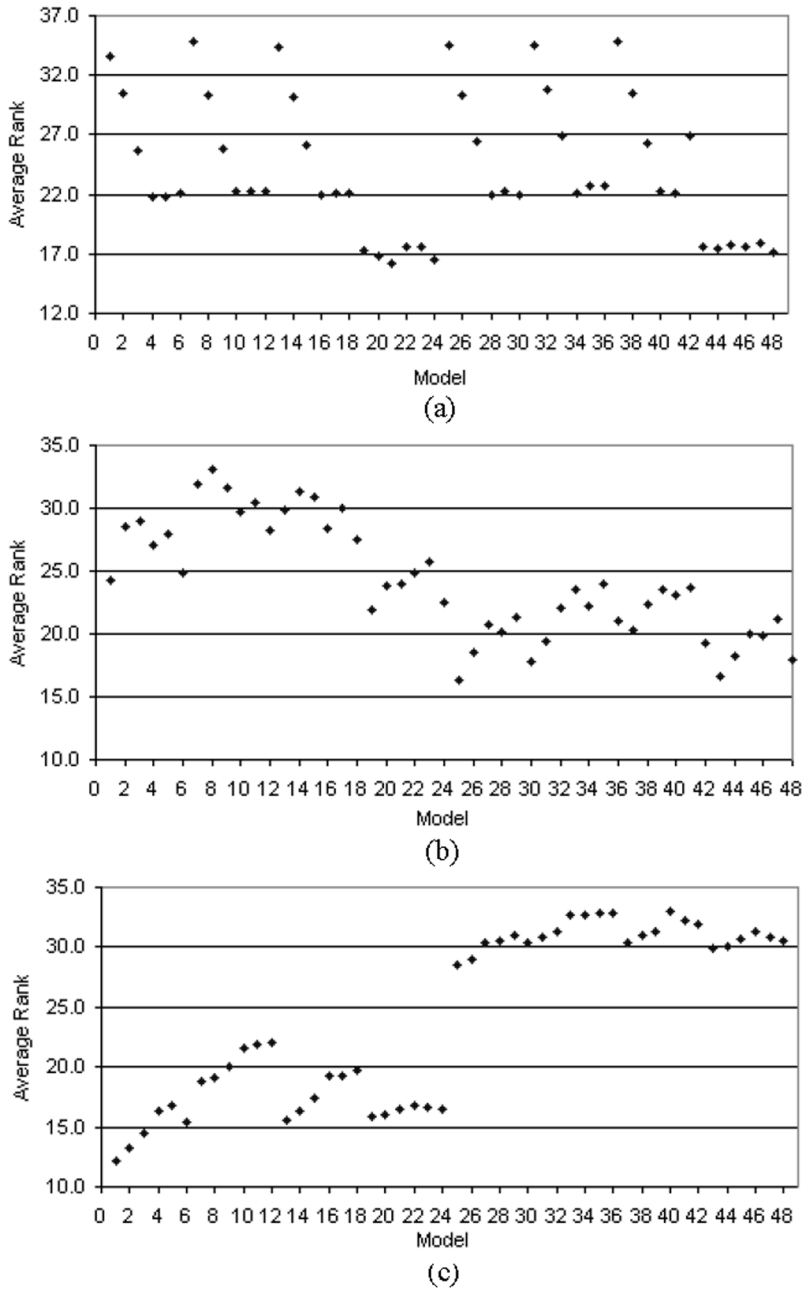

Fig. 3. Average ranks of 48 models. (a) Raw data. (b) Log-transformed data. (c) Fully transformed data.

TABLE XI
COMPARISON OF MODELS WITH THE BEST METHOD—RAW DATA

model	Average rank	Rank interval lower limit	Rank interval upper limit	Significantly worse than the best?
1	33.527	32.866	34.189	Yes
2	30.550	29.888	31.211	Yes
3	25.622	24.961	26.284	Yes
4	21.801	21.139	22.462	Yes
5	21.789	21.128	22.450	Yes
6	22.146	21.484	22.807	Yes
7	34.798	34.137	35.460	Yes
8	30.368	29.707	31.030	Yes
9	25.877	25.216	26.538	Yes
10	22.295	21.634	22.957	Yes
11	22.201	21.540	22.862	Yes
12	22.234	21.573	22.896	Yes
13	34.356	33.694	35.017	Yes
14	30.195	29.534	30.857	Yes
15	26.094	25.432	26.755	Yes
16	21.944	21.283	22.606	Yes
17	22.158	21.497	22.820	Yes
18	22.110	21.449	22.772	Yes
19	17.330	16.669	17.991	No
20	16.849	16.188	17.511	No
21	**16.144**	15.482	16.805	No
22	17.529	16.868	18.191	Yes
23	17.557	16.895	18.218	Yes
24	16.488	15.827	17.150	No
25	34.459	33.798	35.120	Yes
26	30.343	29.682	31.005	Yes
27	26.436	25.774	27.097	Yes
28	21.974	21.312	22.635	Yes
29	22.314	21.653	22.976	Yes
30	21.977	21.315	22.638	Yes
31	34.540	33.878	35.201	Yes
32	30.802	30.140	31.463	Yes
33	26.972	26.310	27.633	Yes
34	22.140	21.478	22.801	Yes
35	22.732	22.070	23.393	Yes
36	22.715	22.054	23.377	Yes
37	34.765	34.103	35.426	Yes
38	30.484	29.823	31.146	Yes
39	26.347	25.686	27.009	Yes
40	22.178	21.517	22.840	Yes
41	22.053	21.391	22.714	Yes
42	26.956	26.295	27.618	Yes
43	17.593	16.931	18.254	Yes
44	17.368	16.707	18.030	No
45	17.712	17.050	18.373	Yes
46	17.548	16.887	18.210	Yes
47	17.964	17.303	18.626	Yes
48	17.110	16.448	17.771	No

TABLE XII
COMPARISON OF MODELS WITH THE BEST METHOD—LOG TRANSFORMATION

model	Average rank	Rank interval lower limit	Rank interval upper limit	Significantly worse than the best?
1	24.306	23.645	24.968	Yes
2	28.554	27.892	29.215	Yes
3	29.013	28.351	29.674	Yes
4	27.028	26.367	27.690	Yes
5	27.935	27.274	28.597	Yes
6	24.893	24.231	25.554	Yes
7	31.871	31.210	32.532	Yes
8	33.027	32.366	33.689	Yes
9	31.677	31.016	32.339	Yes
10	29.694	29.032	30.355	Yes
11	30.391	29.730	31.053	Yes
12	28.254	27.593	28.915	Yes
13	29.796	29.135	30.458	Yes
14	31.384	30.723	32.046	Yes
15	30.914	30.253	31.575	Yes
16	28.337	27.676	28.999	Yes
17	29.960	29.298	30.621	Yes
18	27.429	26.767	28.090	Yes
19	21.923	21.261	22.584	Yes
20	23.762	23.100	24.423	Yes
21	23.911	23.250	24.573	Yes
22	24.904	24.243	25.566	Yes
23	25.669	25.008	26.330	Yes
24	22.532	21.871	23.194	Yes
25	**16.268**	15.606	16.929	No
26	18.511	17.849	19.172	Yes
27	20.679	20.017	21.340	Yes
28	20.163	19.502	20.825	Yes
29	21.379	20.717	22.040	Yes
30	17.738	17.076	18.399	Yes
31	19.448	18.786	20.109	Yes
32	22.077	21.415	22.738	Yes
33	23.484	22.822	24.145	Yes
34	22.145	21.483	22.806	Yes
35	23.908	23.247	24.570	Yes
36	21.003	20.341	21.664	Yes
37	20.307	19.646	20.968	Yes
38	22.296	21.634	22.957	Yes
39	23.492	22.831	24.154	Yes
40	23.065	22.403	23.726	Yes
41	23.683	23.021	24.344	Yes
42	19.319	18.657	19.980	Yes
43	16.549	15.887	17.210	No
44	18.239	17.577	18.900	Yes
45	20.059	19.398	20.720	Yes
46	19.792	19.130	20.453	Yes
47	21.225	20.564	21.887	Yes
48	17.909	17.247	18.570	Yes

is with full transformation where one hidden node is the best for about 55% of the models. When the data are unprocessed or log transformed, a majority of the best models (80% for the raw data and 51% for the logged data) are in fact linear as zero hidden nodes are selected. One hidden node models are the next most commonly selected model (19% and 44%, respectively, for the raw and logged data). Overall, we find that about 58% of the models have zero hidden nodes and 40% have one hidden nodes. As the data are subject to more transformations especially detrending and deseasonalization, more nonlinear models are selected. The results may not be surprising as we have quite small sample size for most of the time series in this paper. In addition, most NN models have only one hidden node, indicating again that simpler models forecast better than more complex ones. It is worthwhile to note that in [30] the similar result regarding the dominance of one hidden node networks is obtained with 24 monthly time series.

Following [39], we also conduct a number of ranking tests to compare each model against the best and against the mean at the 0.05 significance level. These tests are about the null hypothesis that a single ranking does not differ from a random ranking and are based on the average rankings of various models. Overall, using the Friedman test, we find there is a significant difference among different models in performance rankings across for each of the three data types. In order to see which method is significantly different from other methods, two multiple comparison procedures are used. One is the multiple comparisons with the best (MCB), which determines which models are significantly worse than the best model. Another is the multiple comparisons with the mean or analysis of the means (ANOM), which allows us to see which models are statistically better (or worse) than the average. Results are reported in Tables XI–XIII for three data types: raw, log transformed, and fully transformed. We list average ranks, rank intervals, and whether one model is significantly worse than the best in these tables. If the intervals for two

TABLE XIII
COMPARISON OF MODELS WITH THE BEST METHOD—FULL TRANSFORMATION

model	Average rank	Rank interval lower limit	Rank interval upper limit	Significantly worse than the best?
1	**12.121**	11.460	12.782	No
2	13.264	12.602	13.925	No
3	14.409	13.748	15.071	Yes
4	16.365	15.704	17.027	Yes
5	16.724	16.063	17.386	Yes
6	15.366	14.705	16.028	Yes
7	18.759	18.097	19.420	Yes
8	19.122	18.461	19.784	Yes
9	20.083	19.421	20.744	Yes
10	21.605	20.944	22.267	Yes
11	21.930	21.268	22.591	Yes
12	22.052	21.390	22.713	Yes
13	15.542	14.881	16.204	Yes
14	16.309	15.647	16.970	Yes
15	17.338	16.677	17.999	Yes
16	19.303	18.641	19.964	Yes
17	19.246	18.585	19.908	Yes
18	19.728	19.066	20.389	Yes
19	15.929	15.268	16.591	Yes
20	16.075	15.414	16.737	Yes
21	16.558	15.896	17.219	Yes
22	16.782	16.120	17.443	Yes
23	16.709	16.047	17.370	Yes
24	16.544	15.883	17.206	Yes
25	28.534	27.873	29.196	Yes
26	29.026	28.365	29.688	Yes
27	30.309	29.647	30.970	Yes
28	30.476	29.814	31.137	Yes
29	30.915	30.253	31.576	Yes
30	30.408	29.747	31.070	Yes
31	30.858	30.196	31.519	Yes
32	31.224	30.562	31.885	Yes
33	32.621	31.960	33.283	Yes
34	32.658	31.996	33.319	Yes
35	32.813	32.151	33.474	Yes
36	32.916	32.255	33.578	Yes
37	30.327	29.666	30.989	Yes
38	30.990	30.329	31.652	Yes
39	31.222	30.560	31.883	Yes
40	32.980	32.319	33.642	Yes
41	32.262	31.601	32.924	Yes
42	31.979	31.318	32.641	Yes
43	29.919	29.257	30.580	Yes
44	30.059	29.397	30.720	Yes
45	30.720	30.059	31.382	Yes
46	31.245	30.584	31.906	Yes
47	30.905	30.243	31.566	Yes
48	30.536	29.874	31.197	Yes

models do not overlap, then these models do not belong to the same group of ranking. For the raw data (Table XI), we find the best average ranking is 16.1. Five models (models 19, 20, 24, 44, and 48) do not perform significantly worse than the best model (model 21). It is noted that all these models in the best model group contain annual difference, suggesting that this input variable is quite useful in modeling and forecasting raw data. All other models perform significantly worse than the best model. For the logged data, Table XII suggests that the best model is model 25 with an average ranking of 16.3, although model 43 is the only one that is not significantly worse than the best model. Both models contain input variables of y_{t-1} and t. For the fully transformed data, model 1 is the best with an average ranking of 12.1, and model 2 does not perform significantly worse than the best. All others are significantly worse than the best. Fig. 3 plots the average rankings of all models for the three types of data.

Fig. 4 shows the average rankings of 48 models over 756 series compared with the average ranking for each of the three different data types. In each case, the average rank (the solid line) along with the upper and lower bounds (dotted lines) from the ANOM procedure are plotted. Fig. 4(a) shows some interesting pattern for the raw data with the average ranking around 24. It is clear that models with annual differences (models 19–24 and 43–48) perform significantly better than the average. While models without annual differences but containing inputs such as $y_{t-1}, y_{t-2},$ and y_{t-3} perform significantly worse than the average, those models with inputs of y_{t-4} are all close to the lower bound of the average model. For logged data, Fig. 4(b) suggests that using time index t helps improve model performance as most models with t (models 25–48) are significantly better than average while models without t (models 1–24) are worse than the average. On the other hand, when the data are fully transformed, we see a different picture in Fig. 4(c) as all models without t performed significantly better than the average while all models with t are significantly worse than the average. These observations are reasonable because when the data are fully transformed, trend and other significant components may have been removed from the data, resulting in better performance for models without t.

V. CONCLUSION

How to effectively forecast quarterly time series is an important yet challenging task not only for the traditional modelers, but also for NN forecasters. In this paper, we have conducted a comprehensive evaluation of NNs in modeling and forecasting quarterly time series. With a large sample from the M3-competition, we examined a number of modeling issues an NN forecaster may encounter. We considered 48 systematic models with a variety of possible input variables along with three possible data types based on data preparations. Both parametric and nonparametric statistical tests were applied to the results for the performance measures and the rankings.

Our main conclusions are summarized as follows.

1) Different neural models perform differently especially from the input variable selection perspective. Our results clearly show that different combinations of input variables can have significant impact on the model performance. Therefore, in applying NNs, it is critical to identify a set of important input variables to be included in the modeling process, rather than treating them as given and focusing only on the selection of the hidden nodes which seems to be a common approach in the literature. In addition, the NN performance on different types of data differs. Overall, NNs perform the best on the macrodata.
2) Data preparation or transformation is the key to improving NN performance for quarterly time series. We find that by removing significant patterns such as trend and seasonality, NN models perform significantly better than those with raw data. In other words, if the time-series data contains seasonality and trend, NNs are not able to handle these components simultaneously or directly model them. This observation suggests that mixed previous research findings may be due to the failure of considering different preprocessing transformation approaches which can be critical in improving NN modeling and forecasting. In addition, with preprocessed data without complicated patterns, simpler or more parsimonious NN models can be constructed. In this

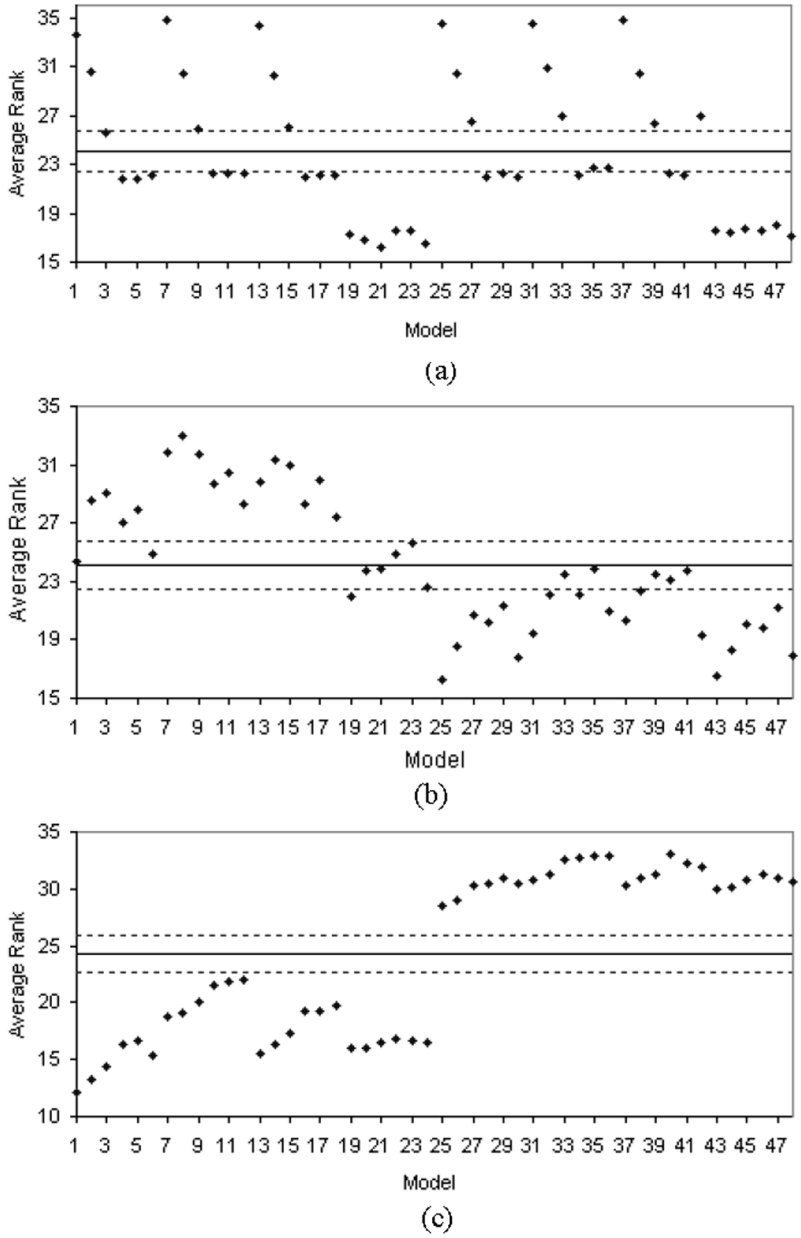

Fig. 4. Average ranks of 48 models with average ranking. (a) Raw data. (b) Logged data. (c) Fully transformed data.

paper, we do not consider the issue of whether detrending or deseasonalization is more responsible for performance improvement. We believe that the combined approach should be the best because of the following: a) both detrending and deseaonalization make time series more stationary, thus simplifying the modeling task, and b) in [17], for monthly time series, it has been found that either approach can dramatically reduce forecasting errors, but using both preprocessing methods yield the best forecasting results.

3) Overall, simple models perform better than more complicated models. Thus, our results confirm one of the most significant findings in all of the M-competitions. Here, simplicity means both the number of input nodes used and the number of hidden nodes selected. For example, the first six base models with only the past lagged observations as inputs perform, in general, better than all other models which contain more input variables. In addition, a majority of the best models uses zero or one hidden node, indicating simple linear autoregressive or NN models perform the best. However, it is equally important to note that when considering different data types, simple models with regard to input variables do not always perform better than more sophisticated models, especially when the data are unprocessed raw observations. That is, in order for NNs to capture all significant components of trend and seasonality, more complex models may be necessary.
4) The inclusion of seasonality information such as seasonal dummy variables or trigonometric variables and trend variable in NN modeling is not generally helpful in improving forecasting performance. One possible explanation is that these variables are deterministic and too simple to capture the dynamic and complex seasonal or trend structures in the data. Another reason is that the time series used in

this paper is very short and using these additional variables makes the NN models more complex, and thus does not provide benefits in improving performance. However, using annual difference variable seems to be helpful in some cases where the data are not fully processed.

Our paper has several limitations. First, as noted earlier, although we used a large number of real time series, all of them can be categorized as small because the maximum length is 64. For seasonal time-series modeling, Box and Jenkins [27] recommend the minimum sample size of 50 or higher should be used in ARIMA modeling. As NNs have more parameters to estimate, larger sample sizes may be required in order to avoid overfitting problems and get better forecasting outcomes. This sample size limitation may be the reason that zero hidden node networks are the most frequently selected models in our paper. Second, although research in forecasting tends to suggest that the in-sample model selection criterion such as AIC or BIC may not be a reliable guide from the forecasting perspective [18], [28], [29], we are not able to use holdout sample or cross-validation approach due to the short time-series nature. Third, all the models in this paper are preselected in terms of the input variables, representing some common practices used in the literature for quarterly time series. It may be better to let NNs to select some important relevant variables from a number of potential inputs. Balkin and Ord [14] describe one such method. Finally, although all time series are quarterly, they are not necessarily seasonal or trending. We have only applied a simple rule of thumb to test if there is a significant correlation between time-series values separated by four lags. This approach may not be effective to identify the true seasonality complicated with the trend factor. We do not perform formal statistical tests regarding whether a time series contains seasonality and/or trend, if so, whether this component is stochastic or deterministic because of the following: 1) we are not aware of whether such tests are available and 2) if available, how effective these tests are. It is ideal if we can have a test regarding whether a series contains certain significant components and if so what data transformation techniques should be applied.

We agree with Terasvirta *et al.* [31] that "in order to obtain acceptable results with nonlinear models, modeling has to be carried with care." This paper shows that although there is a large number of possible ways to model an NN, failing to consider important ones such as data transformation or preparation and appropriate input selection may result in considerably worse results.

REFERENCES

[1] T. Abeysinghe and G. Rajaguru, "Quarterly real GDP estimates for china and ASEAN4 with a forecast evaluation," *J. Forecasting*, vol. 23, pp. 431–447, 2004.

[2] P. H. Franses and D. van Dijk, "The forecasting performance of various models for seasonality and nonlinearity for quarterly industrial production," *Int. J. Forecasting*, vol. 21, pp. 87–102, 2005.

[3] S. Makridakis and M. Hibon, "The M3-competition: Results, conclusions and implications," *Int. J. Forecasting*, vol. 16, pp. 451–476, 2000.

[4] S. Hylleberg, , S. Hylleberg, Ed., "General introduction," in *Modelling Seasonality*. Oxford, U.K.: Oxford Univ. Press, 1992, pp. 3–14.

[5] P. H. Franses, "Recent advances in modelling seasonality," *J. Econom. Surv.*, vol. 10, no. 3, pp. 299–345, 1996.

[6] E. Ghysels, C. W. J. Granger, and P. L. Siklos, "Is seasonal adjustment a linear or nonlinear data filtering process?," *J. Business Econom. Statist.*, vol. 14, pp. 374–386, 1996.

[7] P. T. Ittig, "A seasonal index for business," *Decision Sci.*, vol. 28, no. 2, pp. 335–355, 1997.

[8] D. M. Miller and D. Williams, "Damping seasonal factors: Shrinkage estimators for the X-12-ARIMA program," *Int. J. Forecasting*, vol. 20, pp. 529–549, 2004.

[9] R. J. Hyndman, "The interaction between trend and seasonality," *Int. J. Forecasting*, vol. 20, pp. 561–563, 2004.

[10] J. G. de Gooijer and P. H. Franses, "Forecasting and seasonality," *Int. J. Forecasting*, vol. 13, pp. 303–305, 1997.

[11] J. S. Armstrong, *Principles of Forecasting: A Handbook for Researchers and Practitioners*. Norwell, MA: Kluwer, 2001.

[12] T. Hill, M. O'Connor, and W. Remus, "Neural network models for time series forecasts," *Manag. Sci.*, vol. 42, pp. 1082–1092, 1996.

[13] G. Zhang, B. E. Patuwo, and M. Y. Hu, "Forecasting with artificial neural networks," *Int. J. Forecasting*, vol. 14, pp. 35–62, 1998.

[14] S. D. Balkin and J. K. Ord, "Automatic neural network modeling for univariate time series," *Int. J. Forecasting*, vol. 16, no. 4, pp. 509–15, 2000.

[15] R. Sharda and R. B. Patil, "Connectionist approach to time series prediction: An empirical test," *J. Intell. Manuf.*, vol. 3, pp. 317–323, 1992.

[16] M. Nelson, T. Hill, T. Remus, and M. O'Connor, "Time series forecasting using NNs: Should the data be deseasonalized first?," *J. Forecasting*, vol. 18, pp. 359–367, 1999.

[17] G. P. Zhang and M. Qi, "Neural network forecasting for seasonal and trend time series," *Eur. J. Operat. Res.*, vol. 160, pp. 501–514, 2005.

[18] N. R. Swanson and H. White, "A model selection approach to real-time macroeconomic forecasting using linear models and artificial neural networks," *Rev. Econom. Statist.*, vol. 79, pp. 540–550, 1997.

[19] N. R. Swanson and H. White, "Forecasting economic time series using flexible versus fixed specification and linear versus nonlinear econometric models," *Int. J. Forecasting*, vol. 13, pp. 439–461, 1997.

[20] L. J. Callen, C. C. Y. Kwan, P. C. Y. Yip, and Y. Yuan, "Neural network forecasting of quarterly accounting earnings," *Int. J. Forecasting*, vol. 12, pp. 475–482, 1996.

[21] I. Alon, M. Qi, and R. J. Sadowsik, "Forecasting aggregate retail sales: A comparison of artificial neural networks and traditional methods," *J. Retailing Consumer Services*, vol. 8, no. 3, pp. 147–156, 2001.

[22] C.-W. Chu and G. P. Zhang, "A comparative study of linear and nonlinear models for aggregate retail sales forecasting," *Int. J. Prod. Econom.*, vol. 86, pp. 217–231, 2003.

[23] E. Ghysel and D. R. Osborn, *The Econometric Analysis of Seasonal Time Series*. Cambridge, U.K.: Cambridge Univ. Press, 2001.

[24] S. Makridakis, A. Anderson, R. Carbone, R. Fildes, M. Hibdon, R. Lewandowski, J. Newton, E. Parzen, and R. Winkler, "The accuracy of extrapolation (time series) methods: Results of a forecasting competition," *J. Forecasting*, vol. 1, pp. 111–153, 1982.

[25] W. Zhang, Q. Cao, and M. J. Schniederjans, "Neural network earning per share forecasting models: A comparative analysis of alternative methods," *Decision Sci.*, vol. 35, no. 2, pp. 205–237, 2004.

[26] J. Farway and C. Chatfield, "Time series forecasting with neural networks: A comparative study using the airline data," *Appl. Statist.*, vol. 47, pp. 231–250, 1995.

[27] G. E. P. Box and G. M. Jenkins, *Time Series Analysis: Forecasting, and Control*. San Francisco, CA: Holden Day, 1976.

[28] N. R. Swanson and H. White, "A model-selection approach to assessing the information in the term structure using linear models and artificial neural networks," *J. Business Econom. Statist.*, vol. 13, pp. 265–275, 1995.

[29] M. Qi and G. P. Zhang, "An investigation of model selection criteria for neural network time series forecasting," *Eur. J. Operat. Res.*, vol. 132, no. 3, pp. 188–102, 2001.

[30] S. Heravi, D. R. Osborn, and C. R. Birchenhall, "Linear versus neural network forecasts for European industrial production series," *Int. J. Forecasting*, vol. 20, pp. 435–446, 2004.

[31] T. Terasvirta, D. van Dijk, and M. C. Medeiros, "Linear models, smooth transition autoregressions, and neural networks for forecasting macroeconomic time series: A re-examination," *Int. J. Forecasting*, vol. 21, pp. 755–774, 2005.

[32] S. Balkin, "The value of nonlinear models in the M3-competition," *Int. J. Forecasting*, vol. 17, pp. 545–546, 2001.

[33] A. Atiya, S. El-Shoura, S. Shaheen, and M. El-Sherif, "A comparison between neural networks forecasting techniques–Case study: River flow forecasting," *IEEE Trans. Neural Netw.*, vol. 10, no. 2, pp. 402–409, Mar. 1999.

[34] P. Newbold, W. L. Carlson, and B. M. Thorne, *Statistics for Business and Economics*, 5th ed. Upper Saddle River, NJ: Prentice-Hall, 2003.

[35] M. C. Medeiros and A. Veiga, "A flexible coefficient smooth transition time series model," *IEEE Trans. Neural Netw.*, vol. 16, no. 1, pp. 97–113, Jan. 2005.
[36] K. Papagiannaki, N. Taft, Z.-L. Zhang, and C. Diot, "Long-term forecasting of internet backbone traffic," *IEEE Trans. Neural Netw.*, vol. 16, no. 5, pp. 1110–1124, Sep. 2005.
[37] Z. Zeng and J. Wang, "Improved conditions for global exponential stability of recurrent neural networks with time-varying delays," *IEEE Trans. Neural Netw.*, vol. 17, no. 3, pp. 623–635, May 2006.
[38] A.-K. Seghouane and S.-I. Amari, "The AIC criterion and symmetrizing the Kullback-Leibler divergence," *IEEE Trans. Neural Netw.*, vol. 18, no. 1, pp. 97–106, Jan. 2007.
[39] A. J. Koning, P. H. Franses, M. Hibon, and H. O. Stekler, "The M3 competition: Statistical tests of the results," *Int. J. Forecasting*, vol. 21, pp. 397–409, 2005.

G. Peter Zhang received the B.S. degree in mathematics and the M.S. degree in statistics from East China Normal University, Shanghai, China, in 1985 and 1987, respectively, and the Ph.D. degree in operations management from Kent State University, Kent, OH, in 1998.

Currently, he is an Associate Professor of Managerial Sciences at Georgia State University, Atlanta. His research interests include NNs, forecasting, and supply chain management.

Dr. Zhang currently serves as an Associate Editor of *Neurocomputing* and *Forecasting Letters* and is on the editorial review board of *Production and Operations Management* journal.

Douglas M. Kline received the B.S. degree in mathematics, the M.B.A. degree, and the Ph.D. degree in operations research and information systems from Kent State University, Kent, OH, in 1989, 1992, and 1995, respectively.

Currently, he is an Associate Professor of Information Systems at the University of North Carolina at Wilmington. His research interests include NNs, information systems architecture, and software development practices.