Neurocomputing 71 (2008) 3344–3352

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Forecasting models for interval-valued time series

André Luis S. Maia, Francisco de A.T. de Carvalho *, Teresa B. Ludermir

Centro de Informática, CIn/UFPE, Av. Prof. Luiz Freire, s/n, Cidade Universitária, CEP 50740-540 Recife, PE, Brazil

ARTICLE INFO

Available online 15 July 2008

Keywords:
Symbolic data analysis
Interval-valued data
Time series
ARIMA model
Artificial neural networks
Hybrid models

ABSTRACT

This paper presents approaches to interval-valued time series forecasting. The first and second approaches are based on the autoregressive (AR) and autoregressive integrated moving average (ARIMA) models, respectively. The third approach is based on an artificial neural network (ANN) model and the last is based on a hybrid methodology that combines both ARIMA and ANN models. Each approach fits, respectively, two models on the mid-point and range of the interval values assumed by the interval-valued time series in the learning set. The forecasting of the lower and upper bounds of the interval value of the time series is accomplished through a combination of forecasts from the mid-point and range of the interval values. The evaluation of the models presented is based on the estimation of the average behavior of the *mean absolute error* and *mean squared error* in the framework of a Monte Carlo experiment. The results demonstrate that the approaches are useful in forecasting alternatives for interval-valued time series and indicate that the hybrid model is an effective way to improve the forecasting accuracy achieved by any one of the models separately.

1. Introduction

Interval-valued data arise quite naturally in many situations in which such data represent uncertainty (for instance, confidence intervals), variability (minimum and maximum of daily temperature), etc. Interval-valued data have been considered from different points of view. The field of interval analysis [20,24] assumes that observations and estimations in the real world are usually incomplete or uncertain and, consequently, do not precisely represent the real data. According to this field, if precision is needed, data must be represented as intervals enclosing real quantities.

Interval-valued data have been also considered in the field of *symbolic data analysis* (SDA) [6]. This field, which is related to multivariate analysis, pattern recognition and artificial intelligence, aims to extend classical exploratory data analysis and statistical methods to symbolic data. Symbolic data allow multiple (sometimes weighted) values for each variable and new variable types (set-valued, interval-valued, and histogram-valued variables) have been introduced. These new variables make it possible to take into account the variability in the data. In the field of SDA, interval-valued data do not come from noise assumptions, but rather from aggregation or the expression of variation. They arise in situations such as recording daily interval temperatures at meteorological stations, daily interval stock prices, etc. Another source of interval-valued data is the aggregation of huge data bases in a reduced number of groups, the properties of which are described by interval-valued variables. Therefore, tools for interval-valued data analysis are very much required.

Different approaches have been introduced to analyze interval-valued data. A number of authors have considered neural network models for managing interval-valued data (see [2,12,18,25,27,28,30]). In the field of SDA, Billard and Diday [4] have introduced central tendency and dispersion measures that are suitable for interval-valued data. Cazes et al. [8] and Lauro and Palumbo [21] have introduced principal component analysis methods that are suitable for interval-valued data. Concerning supervised classification, Ichino et al. [17] have introduced a symbolic classifier as a region-oriented approach for interval-valued data. Linear regression models have also been considered by Billard and Diday [3] and Lima Neto and De Carvalho [22]. SDA also provides a number of clustering methods for interval-valued data (see [9,10,14]). These methods differ in the type of the symbolic data considered, their cluster structures and/or the clustering criteria considered.

Interval-valued time series are interval-valued data collected in a chronological sequence. Tools for interval-valued time series data analysis are also very much required. In the framework of SDA, this paper introduces models for forecasting interval-valued time series. We first present an extension of the autoregressive (AR), autoregressive integrated moving average (ARIMA) and artificial neural network (ANN) model estimation methodologies for the analysis of interval-valued time series. Next, we introduce a new methodology based on a hybrid ARIMA and ANN model following Zhang's proposal [35] in order to forecast time series of interval-valued data. A number of issues led us to consider a

* Corresponding author. Tel.: +55 81 21268430; fax: +55 81 21268438.
E-mail addresses: alsm3@cin.ufpe.br (A.L.S. Maia), fatc@cin.ufpe.br, francisco.carvalho@pq.cnpq.br (F.A.T. de Carvalho), tbl@cin.ufpe.br (T.B. Ludermir).

0925-2312/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2008.02.022

hybrid model. Firstly, it is very difficult in practice to determine when a time series is generated by either a linear or non-linear process or when a particular method is more efficient than another for forecasting the series. Secondly, real time series are rarely pure linear or non-linear processes and often contain both patterns. Thirdly, there is no single method that is best in all situations. Through the combination of the ARIMA and ANN models, complex autocorrelation structures in the data can be modeled so as to obtain greater forecasting accuracy. Examples of combinations of the ARIMA and ANN models can be found in [31–33].

In Section 2, we present a brief review of the AR, ARIMA and ANN models for time series forecasting. Hybrid models are also presented in Section 2. Section 3 presents an extension of these models in such a way that they are able to handle interval-valued time series. Section 4 describes the framework of Monte Carlo simulations and presents experiments with synthetic and real interval-valued data sets. Finally, Section 5 offers concluding remarks.

2. Time series forecasting models

For decades, a number of authors have used different statistical methods for modeling and forecasting time series. Such methods vary from a moving average and exponential smoothing to linear and non-linear regressions. Box and Jenkins [7] developed ARIMA models for time series forecasting. ARIMA models are used under linearity presuppositions, that is, the future value of a variable is assumed to be a linear function of several past observations and random errors. However, there are many series for which the linearity supposition is not satisfied. Consequently, ARIMA models cannot provide satisfactory results when used to capture the non-linear structure of data. This leads to an increase in forecasting errors.

A number of alternative methods have been developed to improve the forecasting of time series with non-linear patterns, such as the autoregressive conditional heteroscedastic (ARCH) model [13]. Despite the fact that such methods are superior to ARIMA models with regard to the forecasting of time series with non-linear patterns, they tend to be specific to particular applications. Neural network approaches have also been integrated into time series forecasting. Kaastra and Boyd [19] provide a general introduction to how an ANN model should be developed to model financial and economic time series. Neural network and traditional time series techniques have been compared in several studies. Sharda and Patil [29] found that neural networks performed as good as the automatic Box–Jenkins procedure. Maier and Dandy [23] suggest that the ARIMA model is better suited for short-term forecasts and that neural networks are better suited for longer-term forecasts.

This section presents traditional AR, ARMA, ARIMA time series models and *multilayer perceptron* ANN models for time series.

2.1. Box–Jenkins time series models

An often-used methodology in handling and predicting time series is known as the Box–Jenkins method or simply ARIMA. An autoregressive (AR) model is simply a model used to find an estimation of a variable based on previous input values of the variable. The actual equation for the AR model is as follows:

$$y_t = \theta_0 + \sum_{i=1}^{p} \phi_i y_{t-i} + \varepsilon_t,$$

where y_t is the current value of the time series at time t. The ϕ_i ($i = 1, 2, \ldots, p$) are the model parameters to be estimated. The model consists of three parts: a constant part θ_0, a random error part ε_t (white noise) and the AR summation. The parameter p represents the order of the model AR(p).

Autoregressive moving average (ARMA) models are created from a finite, linear combination of past values of the series and a finite linear combination of past errors. The particular model that will be used in the present paper, known as ARMA(p,q), is represented as follows:

$$y_t = \theta_0 + \sum_{i=1}^{p} \phi_i y_{t-i} + \varepsilon_t + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j},$$

where ε_t is the random error at time t; ϕ_i ($i = 1, 2, \ldots, p$) and θ_j ($j = 1, 2, \ldots, q$) are the model parameters to be estimated; p and q refer to the order of the model; the random errors ε_t are assumed to be independent and identically distributed with a zero mean and σ^2 constant variance. In practice, one applies the ARMA process not to the original time series, but to the transformed time series. Often, the time series of differences is stationary despite the non-stationarity of the underlying process. Stationary time series can be well estimated by the ARMA model. This leads to the definition of the ARIMA model:

$$\Delta^d y_t \text{ is ARMA}(p,q) \text{ process} \Longrightarrow y_t \text{ is ARIMA}(p,d,q) \text{ process},$$

where $\Delta^d y_t$ is the d order differencing operator. The differencing operator is applied to the time series until it becomes stationary. Thus, an ARIMA(p,d,q) process models the stationary differences of the order d of the time series y_t using the ARMA(p,q) process.

2.2. ANN models for time series

When the linearity restriction regarding the model form is relaxed, the number of possible models that can be used in time series forecasting for capturing non-linear structures is very large. For example, ANN models are able to approximate various forms of non-linearity in the data and, unlike ARIMA models, they do not require any presupposition regarding the form of the model. There is a huge variety of different ANN types, but the most popular one is the multilayer perceptron (MLP). MLP networks with two layers (one hidden and one output layer) connected acyclically are often used for modeling and forecasting time series. These MLP networks are similar to the AR process described in Section 2.1. The main advantage over other non-linear models is that MLPs are universal function approximators [16].

In the model we use here, the relation between the output y_t and inputs $y_{t-1}, y_{t-2}, \ldots, y_{t-p}$ is as follows:

$$y_t = \alpha_0 + \sum_{j=1}^{q} \alpha_j \cdot g\left(\beta_{0j} + \sum_{i=1}^{p} \beta_{ij} y_{t-i}\right) + \varepsilon_t,$$

where α's and β's are the model parameters (weights); p is the number of input nodes; q is the number of hidden nodes; and g is the transfer function. There are several types of transfer functions that can be used in the hidden layer, but the logistic function

$$g(x) = \frac{1}{1 + e^{-x}}$$

is the most often used function in MLPs. The reason for the popularity of this transfer function lies in the fact that its first derivative (which is needed for training the ANN) is a very simple expression.

Finally, notice that the MLP neural network seems very similar to the classical AR model. However, the MLP models are more powerful due to the non-linear functional mapping from past observations to the future value. From this standpoint, a neural network is a non-linear AR model. Compared to classical AR models, such a neural network has several advantages [11]: a neural network can model much more complex underlying characteristics of the series. However, the neural network approach also has some disadvantages [11]: neural networks require large numbers of sample data due to their large number of parameters (weights); neural networks can present a variety of problems such as overfitting, capture in local minima, etc.; neural networks do not necessarily include the linear case in a trivial way. Examples of neural networks used for time series prediction purposes can be found in [19,23,29].

2.3. Hybrid models

Box–Jenkins and ANN models have had considerable success in their linear and non-linear domains, respectively. However, the use of Box–Jenkins models for complex non-linear problems may not be adequate. Similarly, using ANNs to model linear problems has produced conflicting results in the literature [35]. Through a combination of different models, different series patterns can be captured. Thus, a hybrid methodology that can simulataneously model linear and non-linear processes seems to be a good strategy for practical use. For example, Su et al. [31] used a hybrid model to forecast a time series of reliability data with a growth trend. Their results showed that the hybrid model produced better forecasts than either the ARIMA model or the neural network alone. Wedding and Cios [34] proposed a hybrid model by combining radial basis function networks and traditional ARMA models. Hansen and Nelson [15] reported their success in combining neural networks, such as time-delay networks and back-propagation networks, with traditional time series models, such as ARIMA.

Zhang [35] states that a time series is composed of a linear autocorrelation structure and a non-linear component,

$$y_t = L_t + N_t, \tag{1}$$

where L_t and N_t, respectively denote the linear and non-linear components. The hybrid model that Zhang proposed [35] consists of two steps. First, ARIMA is used to model the linear component of Eq. (1). The residuals of the ARIMA model contain information on the non-linearity of the series,

$$e_t = y_t - \widehat{L}_t. \tag{2}$$

After adjusting the ARIMA model, the residuals are modeled through an ANN in order to capture the non-linear relation of the series using p input nodes:

$$e_t = f(e_{t-1}, e_{t-2}, \ldots, e_{t-p}) + \varepsilon_t.$$

In the precedent expression, f is a non-linear function determined by the neural network and ε_t is the random error. The prediction of the residual e_t is denoted through the ANN model as $\widehat{N}_t$. Thus, the combined forecast provided by the hybrid model is given by

$$\widehat{y}_t = \widehat{L}_t + \widehat{N}_t.$$

Note that this methodology does not require any presupposition regarding the correlation structure of the time series.

In summary, the methodology that Zhang proposes [35] consists of two steps. In the first step, an ARIMA model is used to analyze the linear part of the problem. In the second step, a neural network model is developed to model the residuals from the ARIMA model. Thus, it could be advantageous to model linear and non-linear patterns separately by using different models and then combine the forecasts to improve the overall modeling and forecasting performance. For further details on the methodology used, see [35].

3. Constructing models for interval-valued time series forecasting

In classical data analysis, each input variable assumes a single value. The need to consider data that contain information which cannot be represented by classical models has led to the development of SDA. Interval data are data in which the values of the variables are intervals in $\mathbb{R}$. Different methodologies have been developed to analyze interval-valued data. One way to represent this type of data is through the mid-point and range of interval [22].

When interval-valued data are collected in a chronological sequence, we say that we have a time series of interval-valued data. At each instant in time, $t = 1, 2, \ldots, n$, where n is the number of intervals observed in the time series, we have X_{U_t} and X_{L_t}, with $X_{L_t} \leqslant X_{U_t}$ as the upper and lower bounds of the interval, respectively,

$$[X_{L_1}; X_{U_1}], [X_{L_2}; X_{U_2}], \ldots, [X_{L_n}; X_{U_n}]. \tag{3}$$

In the methods presented here, two time series are considered: the interval mid-point series X^c; and the half-range interval series X^r. Consider the time series given by Eq. (3), we can, respectively, represent the mid-point and the half-range interval series by,

$$X_t^c = \frac{X_{U_t} + X_{L_t}}{2} \quad \text{and} \quad X_t^r = \frac{X_{U_t} - X_{L_t}}{2} \quad (t = 1, 2, \ldots, n).$$

In the extension of the Box–Jenkins and ANN models for interval-valued time series forecasting, we apply each of the models described in Section 2 to the mid-point interval-valued series X_t^c and to the half-range interval-valued series X_t^r. The values fitted for these two series will be used to predict future values of the upper and lower bounds of the intervals. Thus, the values predicted by these models for the lower and upper bounds of the interval, $\widehat{L}_{U_t}$ and $\widehat{L}_{L_t}$, are, respectively, given by

$$\widehat{L}_{U_t} = \widehat{X}_t^c + \widehat{X}_t^r \quad \text{and} \quad \widehat{L}_{L_t} = \widehat{X}_t^c - \widehat{X}_t^r,$$

where $\widehat{X}_t^c$ and $\widehat{X}_t^r$ represent the values predicted by the linear adjustment for the mid-point and the half-range interval-valued series, respectively.

Concerning the hybrid model for interval-valued time series forecasting, the idea here is to use a methodology based on the hybrid system that Zhang proposed [35] for modeling the mid-point and the half-range interval-valued series. According to the equation of the residuals from the linear model (Eq. (2)), we can denote the residuals of the mid-point and half-range adjustments, respectively, as

$$e_{X_t^c} = X_t^c - \widehat{X}_t^c \quad \text{and} \quad e_{X_t^r} = X_t^r - \widehat{X}_t^r,$$

where $\widehat{X}_t^c$ and $\widehat{X}_t^r$ are, respectively, the values predicted by the linear adjustment for the mid-point and half-range interval-valued time series. Moreover, X_t^c and X_t^r are the corresponding observed values. Thus, after the linear adjustment of the interval-valued series, we have two new series: one from the non-linear residuals of the adjustment of the interval mid-point series $e_{X_t^c}$ and the other from the non-linear residuals of the adjustment of the half-range interval series $e_{X_t^r}$.

It should be pointed out that there is no need for strict pressupositions in this approach regarding the model (linearity, same correlation structure for the interval bounds of the series, etc.). Thus, the final forecast of the bounds of the interval time series is given by

$$\widehat{y}_{U_t} = \widehat{L}_{U_t} + \widehat{N}_{U_t} \quad \text{and} \quad \widehat{y}_{L_t} = \widehat{L}_{L_t} + \widehat{N}_{L_t},$$

where $\widehat{N}_{U_t}$ and $\widehat{N}_{L_t}$ are the errors predicted by the ANN model for the upper and lower bounds of the interval, respectively.

These errors are obtained from the following expression:

$$\widehat{N}_{U_t} = \widehat{e}_{X_t^c} + \widehat{e}_{X_t^r} \quad \text{and} \quad \widehat{N}_{L_t} = \widehat{e}_{X_t^c} - \widehat{e}_{X_t^r},$$

where $\widehat{e}_{X_t^c}$ and $\widehat{e}_{X_t^r}$, respectively, represent the error for the mid-point series and the error for the half-range interval series at time t.

4. Empirical results

This section shows the usefulness of the models presented through experiments with synthetic interval-valued time series data simulated with different degrees of forecasting difficulties and the study of two time series of interval-valued data from a meteorological station in China and stock prices from the Brazilian Petroleum Company (Petrobras).

MLP networks with two layers (one hidden and one output layer) connected acyclically are used in the ANN and hybrid models. We train the MLP using a conjugate gradient error minimization. In summary, the conjugate gradient learning algorithm has a second-order convergence property without complex calculation of the Hessian matrix. The conjugate gradient approach finds the optimal weight vector along the current gradient by doing a line-search. It computes the gradient at the new point and projects it onto a subspace defined by the complement of the space defined by all previously chosen gradients. The new direction is orthogonal to all previous search directions. Details on the conjugate gradient algorithm can be found in [5].

The selection of the best number of hidden units in the ANN involves experimentation. In this paper, a group of neural networks with different numbers of hidden units are trained and each is evaluated in the training set using a reasonable number of randomly selected starting weights. To perform the experiments, we used 50 iterations of the conjugate gradient algorithm, in which the initial weights were randomly generated to reduce the local minimum problem [26]. To determine the best number of hidden units, we used the mean squared error of the 50 iterations. The ANN with the best performance (regarding the mean squared error) in the training set is selected for modeling the time series.

4.1. Synthetic interval-valued time series

In order to assess the performance of the AR, ARIMA, ANN and hybrid models in terms of accuracy in the adjustment and forecasting of interval-valued time series, we have simulated some time series configurations with 200 observations. The simulations of the interval-valued time series were executed as follows:

(1) Let the mid-point interval time series X_t^c $(t = 1, 2, \ldots, n)$, which is obtained from the interval-valued time series $X_t = [X_{L_t}; X_{U_t}]$ $(t = 1, 2, \ldots, n)$, be a process generated with a known structure, such as an AR(1) process.
(2) After Step 1, we construct the half-range interval series. Suppose the half-range interval time series X_t^r $(t = 1, 2, \ldots, n)$ obtained from the interval-valued time series $X_t = [X_{L_t}; X_{U_t}]$ $(t = 1, 2, \ldots, n)$ is uniformly distributed in the interval $[a, b]$, for example, $X_t^r \sim U[10, 20]$.
(3) In the construction of the interval time series, we know that the series $X_t = [X_{L_t}; X_{U_t}]$ presents the following relationship to X_t^c and X_t^r: $X_{L_t} = X_t^c - X_t^r$ and $X_{U_t} = X_t^c + X_t^r$.
(4) At each replication, the simulated interval-valued time series is partitioned in a training set and test set.

Table 1 displays five different configurations for the generation of the interval-valued time series that were used to compare the performances of the AR, ARIMA, ANN and hybrid models. These configurations consist of a combination of the mid-point and range series generated. The first configuration, C_0, is the so-called random walk process. One implication of the random walk process is that the history of the process has no relevance to its future course. Configuration C_0 and configurations C_1 and C_2 present a linear relationship between the future value and past value of the mid-point series plus a random error term, ε_t, normally distributed with a zero mean and constant variance, $\varepsilon_t \sim N(0, 1)$. The non-linear configurations C_3 and C_4 are examples that present complex, chaotic behavior. An example of each of the simulated interval-valued time series is presented in Fig. 1. In this figure, each vertical line segment represents an observed interval-valued data; the extremes correspond to the minimum and maximum interval values.

It is expected that the use of the AR and ARIMA models for complex non-linear problems does not lead to satisfactory forecasting results and, therefore, the ANN and hybrid models are expected to provide greater forecasting accuracy.

4.2. Experimental evaluation

The performance evaluation of the presented interval-valued time series forecasting models, AR, ARIMA, ANN and hybrid model, was accomplished through the following measures: upper bound mean absolute error (MAD_U), lower bound mean absolute error (MAD_L), upper bound mean squared error (MSE_U) and lower bound mean squared error (MSE_L). The values of the error measures were obtained from the observed values $X_t = [X_{L_t}; X_{U_t}]$ $(t = 1, 2, \ldots, n)$ and corresponding predictive values $\widehat{X}_t = [\widehat{X}_{L_t}; \widehat{X}_{U_t}]$.

The forecasting error measures were calculated for the AR, ARIMA, ANN models and the hybrid model in the framework of a Monte Carlo experiment with 1000 replications in the training set and test set. For the results evaluated in the test set, two forecast horizons are considered; 6 and 12 steps ahead. The consistency or variation of the results in the out-of-sample sets is an important performance measure [19]. At the end of the experiments, the average and standard deviation were calculated for the MAD_U, MAD_L, MSE_U and MSE_L measures in the 1000 Monte Carlo replications. The selection of the best AR model and best ARIMA model for adjusting the mid-point and half-range of interval series was accomplished through the minimization procedure of the *Akaike Information Criterion* (AIC) [1]. These parameters were estimated for maximum likelihood.

Tables 2, 4 and 6 display the results of the Monte Carlo experiments for the five configurations. The standard deviations for the error measures considered are in parentheses.

Significance of the differences between the average of the error measures in the framework of this Monte Carlo experiment was tested using a suitable one-side Student's t-test for independent samples and a 5% significance level was adopted. The results of

Table 1
Simulated interval-valued time series configurations

Configuration	X^c process	X^r process
C_0	$X_t^c = X_{t-1}^c + \varepsilon_t$	$U[5, 10]$
C_1	$X_t^c = 10 + 0.7X_{t-1}^c + \varepsilon_t$	$U[10, 1]$
C_2	$X_t^c = 1 + X_{t-1}^c + \varepsilon_t$	$U[8, 10]$
C_3	$X_t^c = 4X_{t-1}^c(1 - X_{t-1}^c)$	$U[2, 5]$
C_4	$X_t^c = 0.2\dfrac{X_{t-17}^c}{1 + (X_{t-17}^c)^{10}} - 0.1X_{t-1}^c$	$U[2, 4]$

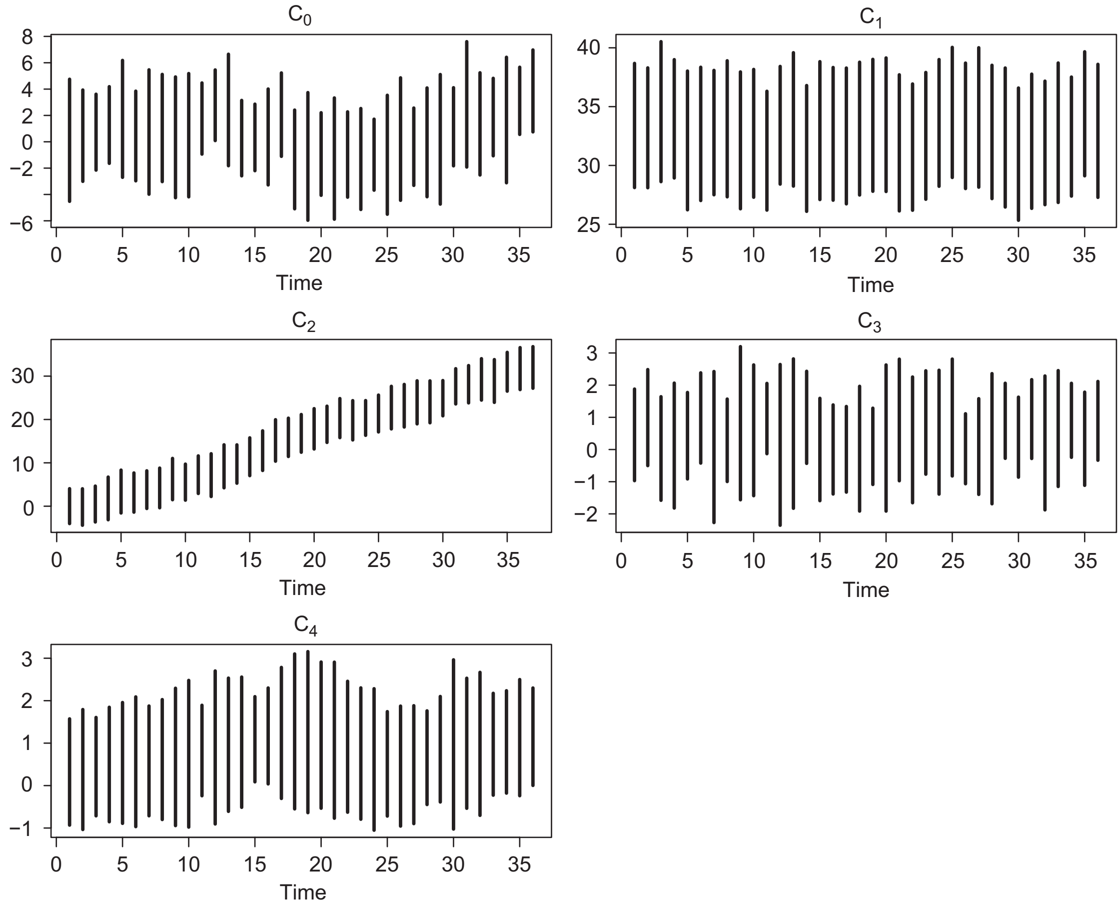


Fig. 1. Examples of simulated interval-valued time series.

these comparisons between models are displayed in Tables 3, 5 and 7. In these tables, the symbol "=" means that the null hypothesis has not been rejected (the difference between the average of the error measures concerning a pair of models is not statistically significant) and that the respective models present the same performance in terms of accuracy in the adjustement and forecasting of interval-valued time series. Moreover, concerning the comparison of the performance of the AR and ARIMA models for configuration C_2 through the measure *lower bound mean absolute error* in Table 3, the symbol "−" means that the null hypothesis has been rejected and that the performance of the ARIMA model is inferior to that of the AR model. This remark illustrates the meaning of the symbol "-". Finally, concerning the comparison of the performance of the AR and ARIMA models for configuration C_3 through the measure *lower bound mean absolute error* in Table 3, the symbol "+" means that the null hypothesis has been rejected and that the performance of the ARIMA model is superior to that of the AR model. This last remark illustrates the meaning of the symbol "+".

Table 2 presents the experimental results evaluated in the training set and Table 3 presents the comparison between models according to a Student's *t*-test for independent samples at a 5% significance level for the training set. We can see that the hybrid model achieved a better average performance than the AR and ARIMA models in forecasting interval-valued time series in the training set for nearly all the situations considered (except in configuration C_0 for the *upper bound mean absolute error* measure). The performance of the hybrid model was also superior to that of the ANN model (except for configuration C_0), specially for the series with a non-linear correlation structure (C_3 and C_4). Note that, even for the series with a linear correlation structure (C_0, C_1 and C_2), the hybrid model achieved better accuracy in the predictions than the AR and ARIMA models. For this training set, the ANN model had the second best performance. AR and ARIMA models had a similar performance for configurations C_0 and C_1. For the others configurations, ARIMA performed better regarding the *mean absolute error* measure, whereas AR performed better regarding the *mean squared error* measure.

Table 4 displays the results evaluated in the test set over the forecast horizon of 6 periods ahead. Table 5 presents the comparison between models according to a Student's *t*-test for independent samples at a 5% significance level in the test set over the forecast horizon of 6 periods ahead. Considering this forecast horizon, the ANN model achieved the best average performance among the four models for configurations C_0 and C_1; as expected, in configurations C_3 and C_4 the ANN and hybrid models were superior to the AR and ARIMA models. Moreover, the hybrid model outperformed the other models for configuration C_4 (see Table 5).

Table 6 displays the results evaluated in the test set over the forecast horizon of 12 periods ahead. Table 7 presents the comparison between models according to a Student's *t*-test for independent samples at a 5% significance level in the test set over the forecast horizon of 12 periods ahead. Considering this forecast horizon, the hybrid model presented a higher predictive performance in all configurations except configuration C_0, for which the ANN model was better. The performance of the ARIMA model was superior to that of the AR model concerning configuration C_0 and the performance of the AR model was superior to that of the ARIMA model for configurations C_2 and C_4.

Table 2
Average and standard deviation of the mean squared errors and mean absolute errors calculated from 1000 replications in the framework of a Monte Carlo experiment for the training set

Model	*MAD*		*MSE*	
	X_U	X_L	X_U	X_L
Configuration C_0				
AR	1.2421 (1.1655)	1.2198 (1.1366)	1.2233 (1.1232)	1.2234 (1.1232)
ARIMA	1.2234 (1.1599)	1.1989 (1.1367)	1.2500 (1.1232)	1.2893 (1.1252)
ANN	1.1756 (1.1276)	1.0354 (1.1489)	1.1006 (1.1565)	1.0998 (1.1555)
Hybrid	1.1567 (1.1276)	1.0033 (1.1488)	1.1104 (1.1559)	1.1330 (1.1554)
Configuration C_1				
AR	1.9875 (1.1178)	1.8918 (1.1177)	1.5447 (1.1431)	1.4481 (1.1448)
ARIMA	1.9871 (1.1175)	1.9271 (1.1175)	1.5434 (1.1438)	1.4458 (1.1431)
ANN	1.8817 (1.1813)	1.8778 (1.1177)	1.3531 (1.1448)	1.3951 (1.1437)
Hybrid	1.7573 (1.1888)	1.7575 (1.1811)	1.1788 (1.1437)	1.1731 (1.1435)
Configuration C_2				
AR	1.3231 (1.1340)	1.3332 (1.1232)	1.2938 (1.1234)	1.2158 (1.1227)
ARIMA	1.4008 (1.1207)	1.4321 (1.1235)	1.4433 (1.1232)	1.3432 (1.1234)
ANN	1.2343 (1.4945)	1.2291 (1.5433)	1.1754 (1.1125)	1.1703 (1.1023)
Hybrid	1.1381 (1.1212)	1.1226 (1.1349)	1.0843 (1.1146)	1.0864 (1.1298)
Configuration C_3				
AR	0.6654 (0.1345)	0.6656 (0.0102)	0.2455 (0.0171)	0.2123 (0.0180)
ARIMA	0.6545 (0.1245)	0.6447 (0.0202)	0.2468 (0.0113)	0.2166 (0.0120)
ANN	0.4227 (0.1748)	0.4067 (0.0106)	0.1489 (0.0122)	0.1762 (0.0154)
Hybrid	0.3996 (0.1669)	0.4006 (0.0143)	0.1467 (0.0129)	0.1486 (0.0127)
Configuration C_4				
AR	0.4688 (0.1145)	0.4746 (0.1153)	0.1211 (0.0677)	0.1387 (0.0658)
ARIMA	0.4477 (0.1453)	0.4566 (0.1344)	0.1218 (0.0653)	0.1223 (0.0670)
ANN	0.3776 (0.1130)	0.4009 (0.1154)	0.1190 (0.0634)	0.1044 (0.0553)
Hybrid	0.2064 (0.1144)	0.2100 (0.1134)	0.1008 (0.0525)	0.1065 (0.0676)

Table 3
Comparison between models according to a Student's t-test for independent samples at a 5% significance level for the training set

Model	Measure	C_0		C_1		C_2		C_3		C_4	
		X_U	X_L	X_U	X_L	X_U	X_L	X_U	X_L	X_U	X_L
AR											
ARIMA	*MAD*	=	=	=	=	=	−	+	+	+	+
	MSE	=	=	=	=	−	−	−	−	=	+
ANN	*MAD*	=	+	+	=	=	+	+	+	+	+
	MSE	+	+	+	=	+	=	+	+	=	+
Hybrid	*MAD*	+	+	+	+	+	+	+	+	+	+
	MSE	+	+	+	+	+	+	+	+	+	+
ARIMA											
ANN	*MAD*	=	+	+	=	+	+	+	+	+	+
	MSE	+	+	+	=	+	+	+	+	=	+
Hybrid	*MAD*	=	+	+	+	+	+	+	+	+	+
	MSE	+	+	+	+	+	+	+	+	+	+
ANN											
Hybrid	*MAD*	=	=	+	+	=	+	+	+	+	+
	MSE	=	=	+	+	+	+	+	+	+	=

4.3. *Meteorological station in China interval-valued data set*

The forecasting models presented for interval-valued time series have been applied to an interval-valued data set extracted from the Long-Term Instrumental Climatic Database of the

Table 4
Average and standard deviation of mean squared errors and mean absolute errors calculated from 1000 replications in the framework of a Monte Carlo experiment for the test set: 6-step ahead forecast

Model	*MAD*		*MSE*	
	X_U	X_L	X_U	X_L
Configuration C_0				
AR	1.2422 (0.1538)	1.2444 (0.1302)	1.6707 (0.6095)	1.6787 (0.5723)
ARIMA	1.1343 (0.1404)	1.1467 (0.1129)	1.6487 (0.5014)	1.6445 (0.4234)
ANN	1.0554 (0.1323)	1.0049 (0.1743)	1.3987 (0.3051)	0.3343 (0.3973)
Hybrid	1.1087 (0.1475)	1.1164 (0.1643)	1.4326 (0.3709)	0.4565 (0.3310)
Configuration C_1				
AR	1.3329 (0.1345)	1.3236 (0.1235)	1.4655 (0.4305)	1.5645 (0.4497)
ARIMA	1.0132 (0.1134)	1.3235 (0.1124)	1.4445 (0.3402)	1.4641 (0.4566)
ANN	1.0431 (0.1074)	1.1975 (0.1034)	1.2664 (0.4068)	1.2365 (0.4045)
Hybrid	1.1332 (0.1545)	1.2388 (0.1349)	1.4246 (0.3567)	1.4423 (0.4657)
Configuration C_2				
AR	2.4034 (0.2332)	2.4534 (0.2246)	8.4320 (0.3443)	8.4861 (0.3459)
ARIMA	2.0545 (0.2439)	2.0765 (0.2246)	8.4342 (0.3454)	8.4327 (0.4157)
ANN	0.5452 (0.2767)	0.5666 (0.2857)	0.2423 (0.4534)	0.2646 (0.3416)
Hybrid	0.4335 (0.2565)	0.4436 (0.2342)	0.2248 (0.3789)	0.2366 (0.2256)
Configuration C_3				
AR	1.2880 (0.2314)	1.2876 (0.2388)	1.8844 (0.2578)	1.8676 (0.3455)
ARIMA	1.2897 (0.2332)	1.2645 (0.2309)	1.8234 (0.2788)	1.8789 (0.2576)
ANN	1.0487 (0.2736)	1.1332 (0.2658)	1.5456 (0.2656)	1.6619 (0.2637)
Hybrid	1.0263 (0.2312)	1.1213 (0.2323)	1.6789 (0.2254)	1.6762 (0.3876)
Configuration C_4				
AR	0.9489 (0.1244)	0.9403 (0.1641)	1.1433 (0.3442)	1.2398 (0.3475)
ARIMA	0.9726 (0.1245)	0.9863 (0.1436)	1.1659 (0.3564)	1.2656 (0.3355)
ANN	0.8390 (0.1456)	0.8664 (0.1836)	1.1376 (0.3435)	1.2167 (0.3434)
Hybrid	0.8112 (0.1245)	0.7945 (0.1850)	0.9782 (0.3343)	0.9998 (0.3334)

Table 5
Comparison between models according to a Student's t-test for independent samples at a 5% significance level for the test set: 6-step ahead forecast

Model	Measure	C_0		C_1		C_2		C_3		C_4	
		X_U	X_L	X_U	X_L	X_U	X_L	X_U	X_L	X_U	X_L
AR											
ARIMA	*MAD*	+	+	+	=	+	+	=	+	−	−
	MSE	=	=	=	+	=	=	+	=	=	−
ANN	*MAD*	+	+	+	+	+	+	+	+	+	+
	MSE	+	+	+	+	+	+	+	+	=	=
Hybrid	*MAD*	+	+	+	+	+	+	+	+	+	+
	MSE	+	+	+	+	+	+	+	+	+	+
ARIMA											
ANN	*MAD*	+	+	−	+	+	+	+	+	+	+
	MSE	+	+	+	+	+	+	+	+	+	+
Hybrid	*MAD*	+	+	−	+	+	+	+	+	+	+
	MSE	+	+	=	=	+	+	+	+	+	+
ANN											
Hybrid	*MAD*	−	−	−	−	+	+	+	=	+	+
	MSE	−	−	−	−	=	+	−	=	+	+

People's Republic of China.[1] Among other variables, this database contains temperatures observed at meteorological stations in

[1] Available at http://dss.ucar.edu/datasets/ds578.5/data/

Table 6
Average and standard deviation of mean squared errors and mean absolute errors calculated from 1000 replications in the framework of a Monte Carlo experiment for the test set: 12-step ahead forecast

Model	*MAD*		*MSE*	
	X_U	X_L	X_U	X_L
Configuration C_0				
AR	1.7732 (0.2656)	1.7655 (0.2643)	3.3420 (0.5564)	3.3348 (0.5852)
ARIMA	1.4456 (0.2533)	1.4325 (0.2975)	3.2426 (0.6564)	3.0447 (0.5587)
ANN	0.9844 (0.2344)	0.9825 (0.2987)	1.1543 (0.4056)	1.9409 (0.4976)
Hybrid	1.2904 (0.2328)	1.2538 (0.2458)	2.0337 (0.4508)	2.0700 (0.4554)
Configuration C_1				
AR	0.8565 (0.0553)	0.8578 (0.0575)	0.9545 (0.1531)	0.9678 (0.1596)
ARIMA	0.8569 (0.0577)	0.8479 (0.0556)	0.9573 (0.1532)	0.9600 (0.1467)
ANN	0.8303 (0.0786)	0.8428 (0.0527)	0.9648 (0.1466)	0.9505 (0.1688)
Hybrid	0.8104 (0.0676)	0.8217 (0.0675)	0.8392 (0.1690)	0.8532 (0.1558)
Configuration C_2				
AR	3.3321 (0.2321)	3.3670 (0.2654)	11.2578 (0.9342)	11.1776 (0.9423)
ARIMA	3.3312 (0.2759)	3.4656 (0.2773)	11.4575 (1.1564)	11.4980 (1.0553)
ANN	1.1213 (1.2675)	1.1554 (1.2879)	2.0252 (1.3533)	2.0300 (1.4672)
Hybrid	0.8977 (0.1679)	0.9052 (0.1578)	0.7874 (0.4349)	0.7480 (0.3937)
Configuration C_3				
AR	1.2560 (0.2440)	1.2549 (0.2980)	1.6447 (0.3074)	1.6557 (0.3097)
ARIMA	1.2864 (0.2453)	1.2554 (0.2566)	1.6164 (0.3097)	1.6672 (0.2983)
ANN	1.1367 (0.2653)	1.1692 (0.2323)	1.2855 (0.1344)	1.3504 (0.1479)
Hybrid	1.0974 (0.2864)	1.0999 (0.2553)	1.2578 (0.1373)	1.2748 (0.1249)
Configuration C_4				
AR	0.7533 (0.1324)	0.7786 (0.1534)	0.2340 (0.0344)	0.2353 (0.1100)
ARIMA	0.8237 (0.1580)	0.8064 (0.1686)	0.2554 (0.0575)	0.2498 (0.0976)
ANN	0.6481 (0.1736)	0.6344 (0.1456)	0.1878 (0.0654)	0.1786 (0.0437)
Hybrid	0.6100 (0.1843)	0.6256 (0.1469)	0.1755 (0.0643)	0.1578 (0.0729)

Table 7
Comparison between models according to a Student's *t*-test for independent samples at a 5% significance level for the test set: 12-step ahead forecast

Model	Measure	C_0		C_1		C_2		C_3		C_4	
		X_U	X_L	X_U	X_L	X_U	X_L	X_U	X_L	X_U	X_L
AR											
ARIMA	*MAD*	+	+	=	+	=	−	−	=	−	−
	MSE	+	+	=	=	−	−	+	=	−	−
ANN	*MAD*	+	+	+	+	+	+	+	+	+	+
	MSE	+	+	=	+	+	+	+	+	+	+
Hybrid	*MAD*	+	+	+	+	+	+	+	+	+	+
	MSE	+	+	+	+	+	+	+	+	+	+
ARIMA											
ANN	*MAD*	+	+	+	+	+	+	+	+	+	+
	MSE	+	+	=	=	+	+	+	+	+	+
Hybrid	*MAD*	+	+	+	+	+	+	+	+	+	+
	MSE	+	+	+	+	+	+	+	+	+	+
ANN											
Hybrid	*MAD*	−	−	+	+	+	+	+	+	+	=
	MSE	−	−	+	+	+	+	+	+	+	+

China. We considered an average temperature series from the BaoDing meteorological station. A natural representation of the BaoDing station is given by the description of each month through an interval of minimum and maximum daily average temperatures. Here, we worked with the monthly temperatures for the period 1974 to 1988, with a total of 180 observations (intervals). Fig. 2 displays part of the interval-valued time series of the

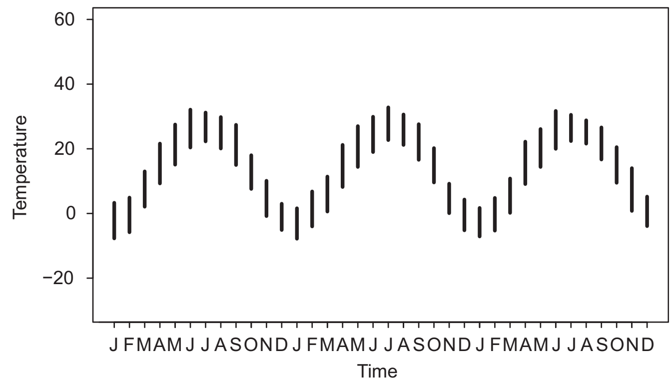


Fig. 2. Part of the interval-valued time series of the temperatures observed at the BaoDing station in China.

Table 8
Comparison for the BaoDing meteorological interval data set: training data set

Training data set				
Model	*MAD*		*MSE*	
	X_U	X_L	X_U	X_L
AR	1.5705	1.3966	4.1494	3.0068
ARIMA	1.5232	1.2530	3.7570	2.5581
ANN	1.0311	0.9322	1.6378	1.3218
Hybrid	1.1237	1.0220	2.3550	1.7202

temperatures observed at the BaoDing station from January 1986 to December 1988.

To assess the forecasting performance of the AR, ARIMA, ANN and hybrid models, the data set is divided into two samples of training and test sets, and two forecast horizons of 6 and 12 periods ahead were considered. The training data sets were used exclusively for model development and the test data sets were used to evaluate the established model.

Tables 8 and 9 give the error measures for BaoDing meteorological station interval-valued time series in the training set and test set, considering forecasts of 6 and 12 steps ahead. The performances of the AR, ARIMA, ANN and hybrid models are evaluated through the calculation of MAD_U, MAD_L, MSE_U and MSE_L measures. We conclude that, ANN model had the best performance in this database among the four models compared.

4.4. Stock prices of the Petrobras interval-valued data set

In this section, we analyze the interval-valued time series of historical stock prices for the Brazilian Petroleum Company S.A. (Petrobras).[2] This database contains the daily open, highest, lowest, and close prices of the Petrobras stocks. Here, we considered daily prices over the period from January 2005 to December 2006, with a total of 484 intervals representing highest and lowest daily prices. Fig. 3 displays the interval prices of the Petrobras stocks over part of the interval-valued time series.

To evaluate the forecasting performance of the four models, the interval-valued time series of Petrobras stock prices is divided into training and test sets, over two forecast horizons: 5 and 20 periods ahead. Tables 10 and 11 present the results of the error measures. In general, the ANN and hybrid models outperform the

[2] Available at http://finance.yahoo.com/q/hp?s=PBR-A (URL accessed on January, 2007).

Table 9
Forecasting comparison for the BaoDing meteorological station interval data set, 6-step and 12-step ahead forecasts: test set

Model	*MAD*		*MSE*	
	X_U	X_L	X_U	X_L
6-steps ahead				
AR	0.9784	1.2689	2.8297	1.7290
ARIMA	0.9598	0.7831	2.7687	0.6970
ANN	1.0064	0.5642	1.8001	0.6884
Hybrid	1.1354	0.6842	2.6761	0.8239
12-steps ahead				
AR	1.5353	1.2444	4.5351	1.7363
ARIMA	1.4598	0.9357	3.8211	1.1112
ANN	1.0034	0.8942	1.7549	0.8680
Hybrid	1.1356	0.9821	2.9723	1.3628

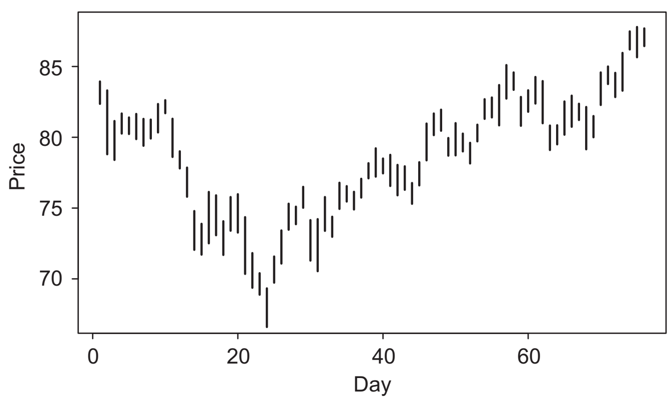


Fig. 3. Part of the interval-valued time series of Petrobras stock prices.

Table 10
Comparison for stock prices in the Petrobras interval data set: training data set

Training data set				
Model	*MAD*		*MSE*	
	X_U	X_L	X_U	X_L
AR	0.9944	1.0760	1.7791	2.1185
ARIMA	0.9758	1.0610	1.7248	2.0941
ANN	0.9486	1.0004	1.5439	1.7922
Hybrid	0.9210	0.8964	1.3566	1.6998

Table 11
Forecasting comparison for stock prices in the Petrobras interval data set, 5-step and 20-step ahead forecasts: test set

Model	*MAD*		*MSE*	
	X_U	X_L	X_U	X_L
5-steps ahead				
AR	0.9387	1.1098	1.3528	1.5392
ARIMA	0.9469	1.0275	1.3483	1.3440
ANN	1.1345	0.8982	1.4549	1.4598
Hybrid	0.8851	1.1009	0.7560	1.2350
20-steps ahead				
AR	1.7986	1.8335	5.1923	5.4265
ARIMA	1.7677	1.8068	4.8932	5.0949
ANN	0.8851	1.0977	1.3245	1.5789
Hybrid	1.1330	1.2589	2.0872	2.3697

AR and ARIMA models in this data set. This result is expected, since this time series has irregular behavior.

5. Concluding remarks

This paper presented four new methods for modeling and forecasting interval-valued time series. The first and second approaches are based on AR and ARIMA models, respectively. The third approach is based on ANN models and the last is based on a hybrid methodology that combines both ARIMA and ANN models. We adjusted the models on the mid-point and interval range series in the training set. The prediction of values for the lower and upper bounds of the intervals was accomplished through a combination of the mid-point and interval range forecasts.

The evaluation of the four methods was accomplished through the average behavior of the mean absolute error and mean squared error of the forecasts in the framework of a Monte Carlo experiment. The Monte Carlo simulations demonstrated that the four methods exhibited a satisfactory performance in forecasting interval series with either a linear or non-linear behavior and are a useful forecasting alternative to interval-valued time series. However, the hybrid model using ARIMA to model the linear component of the series and artificial neural networks to capture the nonlinearity aspects achieved the best average performance concerning the error measures considered. Note that the hybrid model outperformed the AR and ARIMA models even in situations in which the series had linear behavior. When the interval-valued time series exhibited chaotic behavior, the hybrid model and the ANN model were superior to the AR and ARIMA models. Two applications considering real data sets were also analyzed. For the BaoDing meteorological station interval-valued time series, the ANN model had the best performance among the four compared models. The results of the second application demonstrated the superiority of the ANN and hybrid models in the interval-valued time series of historical stock prices of the Petrobras company.

Acknowledgments

The authors would like to thank CNPq, CAPES and FACEPE (Brazilian Agencies) for their financial support, as well as the anonymous referees for their valuable comments and suggestions.

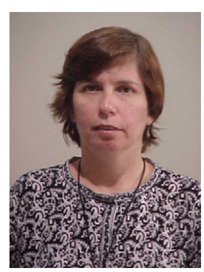
References

[1] H. Akaike, A new look at the statistical model identification, IEEE Trans. Autom. Control 19 (1974) 716–723.
[2] M. Beheshti, A. Berrached, A. de Korvin, C. Hu, O. Sirisaengtaksin, On interval weighted three-layer neural networks, in: SS'98: Proceedings of the 31st Annual Simulation Symposium, IEEE Computer Society, Washington, DC, USA, 1998.
[3] L. Billard, E. Diday, Regression analysis for interval-valued data, Data Analysis, Classification and Related Methods, in: Proceedings of the Seventh Conference of the International Federation of Classification Societies (IFCS'00), Springer, Belgium, 2000, pp. 369–374.
[4] L. Billard, E. Diday, From the statistics of data to the statistics of knowledge: symbolic data analysis, J. Am. Stat. Assoc. 98 (462) (2003) 470–487.
[5] C. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, Oxford, 1996.
[6] H.H. Bock, E. Diday, Analysis of Symbolic Data, Springer, Berlin, Heidelberg, 2000.
[7] G.E.P. Box, G.M. Jenkins, Time Series Analysis, Forecasting and Control, San Francisco, Holden Day, 1976.
[8] P. Cazes, A. Chouakria, E. Diday, S. Schektman, Extension de l'analyse en composantes principales à des données de type intervalle, Revue de Statistique Apliquée XLV (3) (1997) 5–24.
[9] M. Chavent, F.A.T. De Carvalho, Y. Lechevallier, R. Verde, New clustering methods for interval data, Comput. Stat. 21 (2) (2006) 211–229.
[10] F.A.T. De Carvalho, Fuzzy c-means clustering methods for symbolic interval data, Pattern Recognition Lett. 28 (2007) 423–437.

[11] G. Dorffner, E. Leitgeb, H. Koller, Toward improving exercise ECG for detecting Ischemic heart disease with recurrent and feedforward neural nets, Neural Networks for Signal Process. IV, IEEE (1994) 499–508.
[12] G.P. Drago, S. Ridella, Possibility and necessity pattern classification using an interval arithmetic perceptron, Neural Comput. Appl. 8 (1) (1999) 40–52.
[13] R.F. Engle, Autoregressive conditional heteroscedasticity with estimates of the variance of UK inflation, Econometrica 50 (1982) 987–1008.
[14] K.C. Gowda, E. Diday, Symbolic clustering using a new dissimilarity measure, Pattern Recognition 24 (6) (1991) 567–578.
[15] J.V. Hansen, R.D. Nelson, Neural networks and traditional time series methods: a synergistic combination in state economic forecasts, IEEE Trans. Neural Networks 8 (1997) 863–873.
[16] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Networks 2 (1989) 359–366.
[17] M. Ichino, H. Yaguchi, E. Diday, A fuzzy symbolic pattern classifier, in: E. Diday, et al. (Eds.), Ordinal and Symbolic Data Analysis, Springer, Berlin, 1996, pp. 92–102.
[18] H. Ishibuchi, H. Tanaka, H. Okada, An architecture of neural networks with interval weights and its application to fuzzy regression analysis, Fuzzy Sets Syst. 57 (1993) 27–39.
[19] I. Kaastra, M. Boyd, Designing a neural network for forecasting financial and economic time-series, Neurocomputing 10 (1996) 215–236.
[20] R.B. Kearfott, Interval computations: introduction, uses, and resources, Euromath Bull. 1 (2) (1994).
[21] N.C. Lauro, F. Palumbo, Principal component analysis of interval data: a symbolic data analysis approach, Comput. Stat. 15 (1) (2000) 73–87.
[22] E.A. Lima Neto, F.A.T. De Carvalho, Centre and range method for fitting a linear regression model to symbolic interval data, Comput. Stat. Data Anal. 52 (2008) 1500–1515.
[23] H.R. Maier, G.C. Dandy, Neural network models for forecasting univariate time series, Neural Networks World 6 (5) (1996) 747–772.
[24] R.E. Moore, Interval Analysis, Prentice-Hall, Englewood Cliffs, NJ, 1966.
[25] R.E. Patiño-Escarcina, B.R.C. Bedregal, A. Lyra, Interval computing in neural networks: one layer interval neural networks, in: G. Das, V.P. Gulati (Eds.), Proceedings of the Seventh International Conference on Information Technology, CIT, Hyderabad, India, 2004, pp. 68–75.
[26] L. Pretchelt, A Set of Neural Network Benchmark Problems and Benchmarking Rules,Technical Report 21/94, University Karlsruhe, Germany, 1994.
[27] A.M. Roque, C. Maté, J. Arroyo, Á. Sarabia, iMLP: applying multi-layer perceptrons to interval-valued data, Neural Process. Lett. 25 (2) (2007) 157–169.
[28] F. Rossi, F.B. Conan-Guez, Multilayer perceptron on interval data, in: K. Jajuga, A. Sokolowski, H.H. Bock (Eds.), Classification, Clustering, and Data Analysis (IFCS 2002), Cracow, Poland, 2002, pp. 427–434.
[29] R. Sharda, R. Patil, Neural networks as forecasting exports: an empirical test, in: Proceedings of IJCNN Meeting 1990 2, 1990, pp. 491–494.
[30] S.J. Simoff, Handling uncertainty in neural networks: an interval approach, in: Proceedings of IEEE International Conference on Neural Networks, Washington, DC, USA, 1996, pp. 606–610.
[31] C.T. Su, L.I. Tong, C.M. Leou, Combination of time series and neural network for reliability forecasting modeling, J. Chin. Inst. Ind. Eng. 14 (4) (1997) 419–429.
[32] T. Taskaya-Temizel, K. Ahmad, Are ARIMA neural network hybrids better than single models? in: Proceedings of International Joint Conference on Neural Networks 2005, Montreal, Canada, 2005.
[33] F.M. Tseng, H.C. Yu, G.H. Tzeng, Combining neural network model with seasonal time series ARIMA model, Technol. Forecast. Soc. Change 69 (2002) 71–87.
[34] D.K. Wedding II, K.J. Cios, Time series forecasting by combining RBF networks certainty factors and the Box-Jenkins model, Neurocomputing 10 (3) (1996) 149–168.
[35] G. Zhang, Time series forecasting using a hybrid ARIMA and neural network model, Neurocomputing 50 (2003) 159–175.

André L. Santiago Maia is currently a Ph.D. candidate in Computer Science of the Informatics Center at Federal University of Pernambuco, Brazil. He received a B.S. in Statistics from Federal University of Bahia, Brazil, and a M.S. in Statistics from the Department of Statistics at Federal University of Pernambuco, Brazil. He joined the Fundação Joaquim Nabuco, Brazil, in 2007, where he is currently researcher. His current research interests include data modeling, time series and symbolic data analysis.

Francisco de A.T. de Carvalho received the Ph.D. degree in Computer Science in 1992 from Institut National de Recherche en Informatique et en Automatique (INRIA) and Paris- IX Dauphine University, France. From 1992 to 1998, he was a Lecturer at Statistical Department at Federal University of Pernambuco. He joined the Center of Informatics at Federal University of Pernambuco, Brazil, in 1999, where he is currently an Associated Professor. He has published over a 120 articles in scientific journals and conferences. He is one of the associated editors of the Electronic Journal of Symbolic Data Analysis. His research interests include symbolic data analysis, clustering, regression analysis, time series analysis, statistical pattern recognition, information filtering, web usage mining, recommender systems.



Teresa B. Ludermir received the Ph.D. degree in Artificial Neural Networks in 1990 from Imperial College, University of London, UK. From 1991 to 1992, she was a Lecturer at Kings College London. She joined the Center of Informatics at Federal University of Pernambuco, Brazil, in September 1992, where she is currently a Professor and the Head of the Computational Intelligence Group. She has published over a 150 articles in scientific journals and conferences, two books in NN and organized two of the Brazilian Symposium on Neural Networks. She is one of the Editors-in-Chief of the International Journal of Computation Intelligence and Applications. Her research interests include weightless NN, hybrid neural systems and applications of neural networks.