Expert Systems with Applications 40 (2013) 1981–1992

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

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

Forecasting seasonal time series with computational intelligence: On recent methods and the potential of their combinations

Martin Štěpnička [a,*], Paulo Cortez [b], Juan Peralta Donate [c], Lenka Štěpničková [a]

[a] IRAFM, Centre of Excellence IT4Innovations, Division University of Ostrava, 30. dubna 22, 701 03 Ostrava, Czech Republic
[b] Centro Algoritmi, Departamento de Sistemas de Informação, Universidade do Minho, Campus Azurém, 4800-058 Guimarães, Portugal
[c] Centro de Visión por Computación, Universidad Autónoma de Barcelona, Edifici O, 08193-Bellaterra, Barcelona, Spain

ARTICLE INFO

Keywords:
Time series
Computational intelligence
Neural networks
Support vector machine
Fuzzy rules
Genetic algorithm

ABSTRACT

Accurate time series forecasting is a key issue to support individual and organizational decision making. In this paper, we introduce novel methods for multi-step seasonal time series forecasting. All the presented methods stem from computational intelligence techniques: evolutionary artificial neural networks, support vector machines and genuine linguistic fuzzy rules. Performance of the suggested methods is experimentally justified on seasonal time series from distinct domains on three forecasting horizons. The most important contribution is the introduction of a new hybrid combination using linguistic fuzzy rules and the other computational intelligence methods. This hybrid combination presents competitive forecasts, when compared with the popular ARIMA method. Moreover, such hybrid model is more easy to interpret by decision-makers when modeling trended series.

1. Introduction

Forecasting the future is an important tool to support individual and organizational decision making. Time Series Forecasting (TSF) predicts the behavior of a given phenomenon based solely on the past patterns of the same event. In particular, an interesting TSF variant addresses seasonal data (e.g. monthly sales). Under such analysis, multi-step ahead prediction, i.e. forecasting several periods in advance, is highly relevant (e.g. for setting early production plans) in distinct domains, such as Agriculture, Finance, Sales and Production (Makridakis, Wheelwright, & Hyndman, 2008).

Computational Intelligence (CI) denotes a branch of the Artificial Intelligence field that relies on heuristic algorithms inspired in biological and natural intelligence. These CI algorithms include elements of learning and adaptation (e.g. neural networks, fuzzy rules and evolutionary computation) that facilitate intelligent behavior in complex real-world problems (Engelbrecht, 2007).

Although mainly statistical TSF methods (e.g. Holt-Winters exponential smoothing or ARIMA methodology) are widely used in practice (Makridakis et al., 2008), several computational intelligence techniques have been recently proposed for TSF as well (Palit & Popovic, 2005). For instance, some examples of CI applied to TSF are: Artificial Neural Networks (ANN) (Crone, Hibon, & Nikolopoulos, 2011), evolutionary computation (Cortez, Rocha, & Neves, 2004), Support Vector Machines (SVM) (Miiller et al., 1997), immune systems (Nunn & White, 2005), fuzzy techniques (Aznarte, Benítez, & Castro, 2007), or their combinations (Kasabov & Song, 2002; Peralta, Xiaodong Li, Gutierrez, & Sanchis, 2010).

While CI methods were successfully employed in different real-world tasks and several papers on their use in TSF were published, they became more standard in data mining applications rather than in time series, where statistical methods still dominate the market (Makridakis et al., 2008). Such preference for established statistical methods is due to several factors, such as conservatism of some forecasting community members (Price, 2009), but mainly due to a heritage of inferior performance of the first attempts to apply CI to TSF. Moreover, recent CI approaches to TSF often ignore very important issues such as hyperparameter selection (e.g. optimal choice of ANN topology), although it has been proved that an appropriate feature and model selection for a CI TSF model is crucial in order to provide constantly better performance (Crone & Kourentzes, 2010). Similarly, some typical arguments in favor of CI models, such as interpretability and linguistic nature of fuzzy models may seem to be a sort of an unsupported claim or even an empty cliché (Bodenhofer & Bauer, 2005).

These observations are among the main motivations for this paper, which has a fourfold goal: (1) to provide readers with a kind of tasting of distinct methods that may serve as an alternative to standard statistical methods and that may even outperform them; (2) to introduce how these methods may be enhanced, e.g., by using the sensitivity analysis to improve a feature selection for

* Corresponding author. Tel.: +420 59 709 1403.
E-mail addresses: Martin.Stepnicka@osu.cz (M. Štěpnička), pcortez@dsi.uminho.pt (P. Cortez), jperalta@cvc.uab.es (J.P. Donate), Lenka.Stepnickova@osu.cz (L. Štěpničková).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2012.10.001

SVM or by a genetic algorithm to search for the optimal ANN; (3) to introduce purely new combinations of interpretable linguistic fuzzy rules with improved ANN and SVM that provide both – accurate forecasting models and easy to interpret and understand descriptions of the data generating processes; (4) and finally, to challenge prior evidence on the inferior forecasting accuracy of CI in operational forecasting (Crone et al., 2011).

Therefore, we present three novel[1] CI approaches for multi-step seasonal TSF: the *Automatic Design of Artificial Neural Networks* (ADANN), which uses genetic algorithms to evolve ANN structures; the *SVM with time lag selection based on a sensitivity analysis procedure*; and the *linguistic fuzzy approach* to the trend-cycle analysis and forecasts. The first two methods from different perspectives focus on feature and model selection process for CI methods that is often omitted (Crone & Kourentzes, 2010). The latter method focuses on the interpretability issue of fuzzy models. Moreover, we propose the very new hybrid combinations of these CI methods, such that the fuzzy approach to the trend-cycle forecasts is complemented by the earlier two approaches that forecast seasonal components. The main contribution is the presentation of these novel methods and the experimental justification of their potential. Besides the achieved high quality accuracy, such models are more easy to interpret by decision-makers when modeling trended series.

The paper is organized as follows. First, in Section 2, we introduce the used forecasting methods and principles. Next, in Section 3 we describe the seasonal datasets, introduce the forecasting accuracy metrics and finally, introduce a benchmark that serves as a comparison baseline. Then, in Section 4 we present and discuss the obtained results. Finally, we conclude the paper in Section 5.

2. Forecasting methods

Before we introduce the used forecasting methods, we briefly recall the problem. Let $\{y_t | t = 1, \ldots, T\} \subset \mathbb{R}$ be the past values (called *in-samples*) of a given time series. TSF task is to built a model that analyzes the in-samples in order to forecast the future values (so-called *out-of-samples*). Thus, the task is to determine

$$\{F_t = y_t - e_t | t = T+1, \ldots, T+h\} \subset \mathbb{R}, \quad h \geqslant 1, \tag{1}$$

where e_t denotes the forecasting error that should be minimized according to an accuracy measure (see Section 3.2) and h denotes the forecasting horizon. We assume that only in-sample data is used to build such TSF model. After fitting (also known as training) a given time series model, the last known values are fed into the model and it determines the out-of-sample. In case of $h > 1$, either the model directly outputs multi-step ahead forecasts or the out-of-samples are forecasted iteratively by using 1-ahead forecasts (and the remaining up to the $h - 1$ predicted values) as inputs of the model (Cortez, Rio, Rocha, & Sousa, 2006).

2.1. Automatic design of artificial neural networks (ADANN)

Time series processes often exhibit temporal and spatial variability and suffer by issues of nonlinearity of physical processes, conflicting spatial and temporal scale and uncertainty in parameter estimates. ANNs are flexible models that have the capability to learn the underlying relationships between the inputs and outputs of a process, without needing the explicit knowledge of how these variables are related. We recall typical examples in market predictions (Edwards, Tansley, Frank, & Davey, 1997) or in meteorological (Frank, Davey, & Hunt, 2001) and network traffic forecasting (Cortez et al., 2006).

As mentioned above, finding an adequate ANN model for a particular time series is a key issue. Different studies have treated with the design of an ANN from three different points of view.

- Connection weights: values for each connection in an ANN.
- Topology: number of hidden layers, hidden nodes in each layer, etc.
- Learning rules: learning factor and momentum values.

Related to the estimation of the connection weights, it is well known that learning algorithms like backpropagation usually got stuck in a local minimum (Riedmiller & Braun, 1993). Whitley and Hanson (1989) proposed the use of evolutionary computation to search for appropriate connection weights and avoiding the local minimum problem by means of a global search. Later, Belew, Mcinerney, and Schraudolph (1990) proposed a hybrid approach carrying out a global search by a genetic algorithm and tuning better the connection weights obtained through a backpropagation-like learning algorithm. Distinct constructive/destructive methods for the evolution of topologies of ANNs have been presented (Frean, 1990), but those based on evolutionary computation obtain better results (Miller, Todd, & Hegde, 1989). At last, there are some works that try to evolve the learning rules (Jacobs, 1988).

In this paper, a novel evolving hybrid system that uses both, a genetic algorithm and the backpropagation learning, is proposed. This approach involves an evolution of the ANN topology and backpropagation learning parameter, with multiple initializations.

Normalization of the time series data has to be done as an initial step and after fitting the ANN, the inverse process is carried out. This step is important as ANN with logistic activation functions output values within the range [0,1]. Time series in-samples are transformed into a pattern set with I inputs. A single neuron is placed at the output layer and multi-step forecasts are often performed using an iterative feedback of the previous forecasts (Cortez, Rocha, & Neves, 2006). Therefore, each time series is transformed into a patterns set where each pattern consists of:

$$(N_{t-I}, \ldots, N_{t-2}, N_{t-1}) \rightarrow N_t,$$

where all N_i values correspond to the normalized y_i ones. This pattern set is used to train and validate each ANN generated during the Genetic Algorithm (GA) execution. Thus, the data is split into training (with the first X% data) and validation sets (with the remaining patterns), as shown in Fig. 1.

The search for the best ANN design can be performed by a GA (Fogel, 2005) using exploitation and exploration. When using such GA, there are three crucial issues: (i) the solution space and what is included into a chromosome; (ii) how each solution is codified into a chromosome, i.e. encoding schema; and (iii) what is the fitness function.

In this work, we opted for a multilayer perceptron as the base forecasting model, with one hidden layer and backpropagation as the learning algorithm, according to Cybenko (1989). Regarding the backpropagation choice, we note that we use multiple initializations (as distinct seeds are used, see Eq. (2)) and also evolve its learning factor. Under such scheme, backpropagation is unlikely to fall into a local minima. Moreover, backpropagation is the most used algorithm in the TSF domain and studies presenting learning algorithms that outperform backpropagation should be viewed critically, since there is a bias to publish only algorithms that outperform the standard backpropagation (Kourentzes & Crone, 2010). Also, a majority of such papers do not report all details about training parameters and use few distinct initializations.

A direct encoding schema for fully connected multilayer perceptron is considered. For this encoding scheme the information placed into the chromosome is: two decimal digits, i.e., two genes to codify the number of inputs nodes (I); two genes for the number

[1] All three CI methods were separately proposed in the 2010 IEEE World Congress on Computational Intelligence (WCCI), under the special session "Computational Intelligence in Forecasting".