Expert Systems With Applications 137 (2019) 59–73

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Hierarchical time series forecasting via Support Vector Regression in the European Travel Retail Industry

Juan Pablo Karmy, Sebastián Maldonado*

Facultad de Ingeniería y Ciencias Aplicadas, Universidad de los Andes, Mons. Álvaro del Portillo 12455, Las Condes, Santiago, Chile

ARTICLE INFO

Article history:
Received 3 September 2018
Revised 5 May 2019
Accepted 26 June 2019
Available online 27 June 2019

Keywords:
Hierarchical time series
Support Vector Regression
Time series analysis
Sales forecasting

ABSTRACT

Times series often offers a natural disaggregation in a hierarchical structure. For example, product sales can come from different cities, districts, or states; or be grouped by categories and subcategories. This hierarchical structure can be useful for improving the forecast, and this strategy is known as hierarchical time series (HTS) analysis. In this work, a novel strategy for sales forecasting is proposed using Support Vector Regression (SVR) and hierarchical time series. We formalize three different hierarchical time series approaches: bottom-up SVR, top-down SVR, and middle-out SVR, and use them in a sales forecasting project for the Travel Retail Industry. Various hierarchical structures are proposed for the retail industry in order to achieve accurate product-level predictions. Experiments on these datasets demonstrate the virtues of SVR-based hierarchical time series in terms of predictive performance when compared with the traditional ARIMA and Holt-Winters approaches for this task.

1. Introduction

Hierarchical time series (HTS) was recently formalized as a forecasting approach for time series that follow a hierarchical structure (Hyndman & Athanasopoulos, 2013). This is particularly useful in multi-category demand forecasting when the bottom level (SKU-level product sales) has scarce or incomplete data, yielding noisy predictions. In such cases, intermediate levels, such as the product category, can be used to enhance predictions via a bottom-up approach (Athanasopoulos, Hyndman, Kourentzes, & Petropoulos, 2017).

Demand forecasting is a very important application of expert systems for decision-making (Franses & Legerstee, 2011; Huber, Gossmann, & Stuckenschmidt, 2017; Jiang, Chin, Wang, Qu, & Tsui, 2017; Martínez, Frías, Pérez-Godoy, & Rivera, 2018). Demand forecasting has usually been addressed by traditional time series methods, such as the ARIMA (Auto-Regressive Moving Average) model, or the Holt-Winters approach (Makridakis & Wheelwright, 1977). However, machine learning methods, such as Support Vector Regression (SVR), and Artificial Neural Networks (ANN), have gained interest due to their ability to model non-linear patterns, enhancing predictive performance (Ahmed, Atiya, El-Gayar, & El-Shishiny, 2010; Crone, Lessmann, & Pietsch, 2006; Hansen, McDonald, & Nelson, 2006).

In this work, we formalized three hierarchical time series approaches based on Support Vector Regression, and applied them in sales forecasting for the Duty Free and Travel Retail Industry. An accurate forecast can be extremely useful in Travel Retail, since it aids in making both operational decisions, such as product assortment and replenishment, and tactical/strategic decisions, such as the opening of new stores and risk management (Wood & Tasker, 2008).

Our contribution is twofold. The first one is methodological: We propose novel SVR approaches for time series forecasting, in which traditional econometric methods used for this task are extended to non-linear modeling. Kernel methods confer flexibility to the modeling process, and our experiments demonstrate that they yield better predictive results. The second contribution is an applied one: We use our proposal in a real-world problem for multi-category demand forecasting in the Travel Retail Industry, developing an expert system for inventory replenishment and other operative and strategic decisions.

The contents of the remainder of this work are as follows: in Section 2 we describe the methodological background that is relevant for our proposal, which includes hierarchical time series analysis, and Support Vector Regression. The proposed framework for sales forecasting using SVR-based hierarchical time series is presented in Section 3. Experimental results using data from Generation Research AB are given in Section 4. Finally, Section 5 provides the main conclusions and addresses future developments.

* Corresponding author.
E-mail addresses: jpkarmy@miuandes.cl (J.P. Karmy), smaldonado@uandes.cl (S. Maldonado).

https://doi.org/10.1016/j.eswa.2019.06.060
0957-4174/

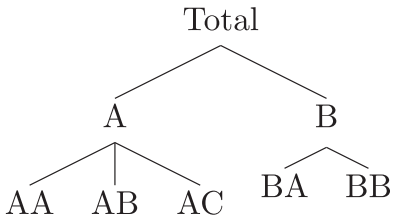


Fig. 1. Example of hierarchy.

2. Literature overview

In this section, the two main methodological aspects of the proposal are discussed: Hierarchical time series methods are presented in Section 2.1, while in Section 2.2 SVR for time series analysis is discussed. Applications in sales forecasting are presented for both topics.

2.1. Hierarchical time series

Univariate time series are sequences of objects o_1, o_2, ..., o_n declared at successive points t_1, t_2, ..., t_n in time (Steffen, Sarda, Artz-Beielstein, Zaefferer, & Strok, 2015). In order to generalize current patterns to predict future values, time series are decomposed on a trend-cycle component, a seasonal component, and a remainder (Hyndman & Athanasopoulos, 2013). The two best-known time series forecasting methods are, the Autoregressive Integrated Moving Average (ARIMA), and the Holt-Winters models (Hyndman & Athanasopoulos, 2013). The ARIMA model can be formulated as follows:

$$D_t = -(\Delta^d D_t - D_t) + \sum_{i=1}^{p} \alpha_i \cdot \Delta^d D_{t-i} + \sum_{j=1}^{q} \beta_j \cdot e_{t-j} + e_t, \tag{1}$$

where α_i $(i = 1, \ldots, p)$ and β_j $(j = 1, \ldots, q)$ are the coefficients related to the auto-regressive and moving average processes, respectively, and e_t corresponds to the error term, that is assumed to be white noise. The expression $\Delta^i D_t$ is the difference between observation t and its predecessor $(D_t - D_{t-1})$.

The Holt-Winters, on the other hand, is an extension of exponential smoothing that includes terms for modeling trend and seasonality. The forecast is made by the following equations:

$$A_t = \alpha \frac{D_t}{C_{t-L}} + (1-\alpha)(A_{t-1} - T_{t-1}), \qquad \alpha \in [0,1] \tag{2}$$

$$T_t = \beta(A_t - A_{t-1}) + (1-\beta)T_{t-1}, \qquad \beta \in [0,1] \tag{3}$$

$$C_t = \gamma \frac{D_t}{A_t} + (1-\gamma)C_{t-L}, \qquad \gamma \in [0,1] \tag{4}$$

$$F_{t+k} = (A_t + kT_t)C_{t-L+((k-1) \mod L)} \tag{5}$$

The parameter, L, represents the frequency of the seasonality; for example, $L = 4$ means quarterly data, while $L = 12$ represents monthly data. The smoothing parameters, α, β, and γ, are used to control the trade-off between level, trend, and seasonality, respectively.

Regarding hierarchical time series (HTS), Hyndman and Athanasopoulos (2013) formalized the idea of forecasting related time series that follow a hierarchical aggregation structure. Common examples are sales forecasts that can be disaggregated by product type or geographical location.

Fig. 1 presents a simple example of a hierarchical structure (Hyndman & Athanasopoulos, 2013). The "Total" time series is disaggregated at the first level, with series *A* and *B*. These new time series can also be disaggregated, generating a three-level hierarchy, in which there are two nodes at the first level, and five nodes at the lowest one.

The time series related to each node is then represented as the sum of the time series of all its "child" nodes. Defining y_t as the fully-aggregated time series, and the time series associated with node n as $y_{n,t}$, the example in Fig. 1 can be formalized as follows:

$$\begin{bmatrix} y_t \\ y_{A,t} \\ y_{A,t} \\ y_{B,t} \\ y_{AA,t} \\ y_{AB,t} \\ y_{AC,t} \\ y_{BA,t} \\ y_{BB,t} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_{AA,t} \\ y_{AB,t} \\ y_{AC,t} \\ y_{BA,t} \\ y_{BB,t} \end{bmatrix}$$

There are various forecasting approaches that can be used for hierarchical time series. The best-known strategies are bottom-up (BU), top-down (TD), and middle-out (MO). The bottom-up approach aims to forecast each bottom-level node of the hierarchy, and then forecast the higher levels by adding the values obtained previously. Following the example presented in Fig. 1, an h-step-ahead prediction using BU can be performed for each node, n, as $\hat{y}_{n,h}$, based on the prediction for the leaf nodes, *AA*, *AB*, *AC*, *BA*, and *BB*. This is illustrated as follows:

$$\hat{y}_{A,h} = \hat{y}_{AA,h} + \hat{y}_{AB,h} + \hat{y}_{AC,h}$$
$$\hat{y}_{B,h} = \hat{y}_{BA,h} + \hat{y}_{BB,h}$$
$$\hat{y}_h = \hat{y}_{A,h} + \hat{y}_{B,h}$$

This method has the advantage of having almost no information loss during aggregation. However, the data at the last hierarchy level could be noisy due to lack of information, leading to potentially poor predictive results (Hyndman & Athanasopoulos, 2013).

The top-down approach aims to perform a forecast for the top level of the hierarchy $(\hat{y}_h)$, and then disaggregate it to the different nodes by using a predefined proportion. The most common approach is the use of the average for each node, j, relative to its "parent" node as a proportion (Hyndman & Athanasopoulos, 2013), given by the following formula:

$$p_j = \frac{1}{T} \sum_{t=1}^{T} \frac{y_{j,t}}{y_t} \tag{6}$$

In the example of Fig. 1, the predictions for the different nodes based on $\hat{y}_h$ are given by:

$$\hat{y}_{A,h} = p_A \cdot \hat{y}_h$$
$$\hat{y}_{B,h} = p_B \cdot \hat{y}_h$$
$$\hat{y}_{AA,h} = p_{AA} \cdot \hat{y}_{A,h}$$
$$\hat{y}_{AB,h} = p_{AB} \cdot \hat{y}_{A,h}$$
$$\hat{y}_{AC,h} = p_{AC} \cdot \hat{y}_{A,h}$$
$$\hat{y}_{BA,h} = p_{BA} \cdot \hat{y}_{B,h}$$
$$\hat{y}_{BB,h} = p_{BB} \cdot \hat{y}_{B,h}$$

One of the disadvantages of this method is that there is potential information loss in the node branching process, and when the branching process is repeated, the information loss may increase (Hyndman & Athanasopoulos, 2013).

The middle-out approach performs the estimation for an intermediate level of the established hierarchy, and then uses the two previous approaches. The nodes above the estimated level are calculated based on the bottom-up method, while those below are calculated with the top-down approach (Hyndman & Athanasopoulos, 2013).

Finally, Hyndman and Athanasopoulos proposed another HTS approach, called the Optimal Reconciliation Method (Hyndman & Athanasopoulos, 2014). This approach aims at finding an optimal combination for all the independent forecasts at all levels via weighted averages. All these methods were presented empirically in Athanasopoulos et al. (2017), and implemented as an R software package (Hyndman, Athanasopoulos, & Shang, 2012).

2.2. Support vector regression for time series analysis

This section presents the ϵ-SVR method (Drucker, Burges, Kaufman, Smola, & Vapnik, 1997; Smola & Schölkopf, 1998), the extension of Support Vector Machines for regression, discussing its use in time series analysis.

Given a training set that encompasses a continuous dependent variable, $y_i \in \Re \ \forall i = 1, \ldots, m$, and covariates, $\mathbf{x}_i \in \Re^p$, $i = 1, \ldots, m$, that are lagged variables obtained from the dependent variable, SVR minimizes the objective function of the following quadratic programming problem:

$$\begin{aligned} \min_{\mathbf{w},b,\boldsymbol{\xi},\boldsymbol{\xi}^*} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}(\xi_i + \xi_i^*) \\ \text{s.t.} \quad & y_i - (\mathbf{w}^\top \mathbf{x}_i + b) \geq \epsilon + \xi_i, \qquad i = 1, \ldots, m, \\ & (\mathbf{w}^\top \mathbf{x}_i + b) - y_i \geq \epsilon + \xi_i^*, \qquad i = 1, \ldots, m, \\ & \xi_i \ \xi_i^* \geq 0, \qquad i = 1, \ldots, m, \end{aligned} \tag{7}$$

The introduction of the slack variables, $\boldsymbol{\xi}, \boldsymbol{\xi}^* \in \Re^m$, allows objects to lie outside the ϵ-insensitive tube. However, these objects are penalized in the objective function. The trade-off between model fit and complexity reduction is controlled by C (Smola & Schölkopf, 1998). The use of the l_2-regularization (the Euclidean norm of $\mathbf{w}$) guarantees that the regression function, $f(\mathbf{x})$, is as flat as possible (Smola & Schölkopf, 1998).

Formulation (7) can also be represented by its dual form, which leads to a kernel method for constructing non-linear surfaces. A kernel function, $K(\mathbf{x}_i, \mathbf{x}_s)$, is introduced to map the data to a higher dimensional feature space (Schölkopf & Smola, 2002). The dual representation of Formulation (7) follows:

$$\begin{aligned} \max_{\boldsymbol{\alpha},\boldsymbol{\alpha}^*} \quad & \sum_{i=1}^{m} y_i(\alpha_i - \alpha_i^*) - \epsilon\sum_{i=1}^{m}(\alpha_i + \alpha_i^*) \\ & -\frac{1}{2}\sum_{i,s=1}^{m}(\alpha_i - \alpha_i^*)(\alpha_s - \alpha_s^*)K(\mathbf{x}_i, \mathbf{x}_s) \\ \text{s.t.} \quad & \sum_{i=1}^{m}(\alpha_i - \alpha_i^*) = 0, \\ & 0 \leq \alpha_i \leq C, \qquad i = 1, \ldots, m, \\ & 0 \leq \alpha_i^* \leq C, \qquad i = 1, \ldots, m. \end{aligned} \tag{8}$$

For a new sample, $\mathbf{x}$, the decision rule, $f(\mathbf{x})$, is:

$$f(\mathbf{x}) = \sum_{i=1}^{m}(\alpha_i - \alpha_i^*)K(\mathbf{x}, \mathbf{x}_i) + b \tag{9}$$

There are several alternatives for kernel functions. In this paper we use the Linear and Gaussian kernels, which are arguably the most commonly used ones. The Linear kernel has the following form: $K(\mathbf{x}_i, \mathbf{x}_s) = \mathbf{x}_i^T\mathbf{x}_s$, while the Gaussian kernel (also known as the Radial Basis Function (RBF)) is defined as $K(\mathbf{x}_i, \mathbf{x}_s) = \exp\left(\gamma - ||\mathbf{x}_i - \mathbf{x}_s||^2\right)$, where $\gamma > 0$ is the parameter controlling the width of the kernel (Schölkopf & Smola, 2002).

SVR has been used extensively for time series analysis. A related survey can be found in Sapankevych and Sankar (2009), who claim that SVR may perform better than other methods in time series forecasting mainly because of its non-linear nature, and the existence of a single global minimum.

Crone, Guajardo, and Weber (2006) analyzed the ability of SVR to forecast basic time series patterns, comparing both kernel functions. In their work, they state that SVR with Gaussian kernel outperforms other methods for non-trended data. For time series with trend components, they found that Linear SVR was better than the Gaussian version.

Several studies in sales forecasting with SVR have been reported. Levis and Papageorgiou (2009), for example, performed demand forecasting with a three-step algorithm that could handle non-linear patterns effectively, achieving superior prediction accuracy in all the cases studied. Lu (2014) used SVR for sales forecasting, combining it with multivariate adaptive regression splines. Similarly, Du, Leung, Zhang, and Lai (2013) used SVR for predicting the demand for farm products. Another approach presented by Lu and Wang (2010) was developed for demand forecasting via SVR, in which independent component analysis and self-organizing maps are used as the preprocessing tool.

3. Proposed framework for sales forecasting using hierarchical SVR

In this section, we extend the reasoning behind hierarchical time series forecasting to ϵ-SVR. The goal of our approach is to utilize the hierarchical structure of the problem to achieve better prediction, while the use of kernel methods allows modeling non-linear patterns adequately by constructing flexible regression functions. To the best of our knowledge, the use of ϵ-SVR for hierarchical time series has not yet been reported in the literature.

The reasoning behind using SVR is that this approach is able to construct a flexible nonlinear regression function while reducing the risk of overfitting. This is done by minimizing the Structural Risk, guaranteeing that the final solution depends on only a few data points, leading to functions that correctly represents the observations while being as flat as possible at the same time (Drucker et al., 1997). This is of utmost importance in HTS analysis since few periods are usually available for training, aggravating the issue of overfitting for machine learning approaches.

The bottom-up, top-down, and middle-out approaches are formalized next using SVR (SVR-BU, SVR-TD, and SVR-MO, respectively). First, the notation is introduced. The following sets and parameters are considered:

- $j = (0, \ldots, J)$: levels of hierarchy, in which level 0 represents the root node, while level J is the one containing the leaf nodes.
- $n = (1, \ldots, N)$: nodes of a given hierarchy.
- A_j: subset of nodes at level j.
- S_n: subset of nodes that stems from node n.
- D_n: parent node for node n.
- p_n: historical proportion of node n.
- F_n: forecast for node n.

First, the SVR-BU method trains an ϵ-SVR at each leaf node (Formulation (8)), computing their corresponding predictions, F_n, $(\forall n \in A_{|J|})$. Then, these forecasts are used to provide predictions for the upper levels of the hierarchy by aggregating them using $F_n = \sum_{i \in S_n} F_i$, i.e., the sum of the predictions of all child nodes that stem from node n. The algorithm for this method is presented in Algorithm 1.

The SVR-TD works as follows: First, ϵ-SVR is trained at the root node (Formulation (8)), computing the predictions F_0. Then, the forecast is disaggregated to the lower hierarchies by using the average proportions. The algorithm for this method is presented in Algorithm 2.

Algorithm 1 SVR-BU Algorithm.

1. Calculate predictions, F_n, $\forall n \in A_J$:
   (a) For each node at the bottom level, train ϵ-SVR by solving the Formulation (8) with a given hyperparameter configuration to obtain the solution tuple (α, α^*).
   (b) Compute each forecast, F_n, at the bottom level using Formulation (9): $F_n(\mathbf{x}) = \sum_{i=1}^{m} (\alpha_i - \alpha_i^*) K(\mathbf{x}, \mathbf{x}_i) + b$.
2. For each level, $j = (0, \ldots, J-1)$, compute predictions, F_n, $\forall n \in A_j$, by using $F_n = \sum_{i \in S_n} F_i$, i.e., the sum of the predictions related to the nodes that stem from node n.

Algorithm 2 SVR-TD algorithm.

1. Calculate predictions F_0 at the root level:
   (a) For the top level of the hierarchy, train ϵ-SVR by solving the Formulation (8) with a given hyperparameter configuration to obtain the solution tuple (α, α^*).
   (b) Compute the forecast, F_0, using the Formulation (9): $F_0(\mathbf{x}) = \sum_{i=1}^{m} (\alpha_i - \alpha_i^*) K(\mathbf{x}, \mathbf{x}_i) + b$
2. For each level, $j = (1, \ldots, J)$, compute predictions, F_n, $\forall n \in A_j$, by using $F_n = p_n \cdot F_{D_j}$, i.e., disaggregating the forecast of the parent node following the historical proportions.

Finally, the SVR-MO is described. The first step corresponds to the definition of the initial level, $0 < j < J$, in which ϵ-SVR is trained for each node of the level. The corresponding forecasts, F_n, $\forall n \in A_j$, are then obtained by using Formulation (9). These forecasts are subsequently aggregated to the upper levels, $k < j$, following the bottom-up approach: For each level, k, the predictions, F_n, $\forall n \in A_k$, are obtained by computing $F_n = \sum_{i \in S_n} F_i$, i. e., the sum of the predictions related to the nodes that stem from node n. For all the levels, $k > j$, that are below j in the hierarchy, the bottom-up approach is applied: The forecast is disaggregated to the lower hierarchies by using the average proportions. The algorithm for this method is presented in Algorithm 3.

Algorithm 3 SVR-MO algorithm.

1. Define the initial level, $0 < j < J$.
2. Compute the predictions, F_n, $\forall n \in A_j$, at this middle level:
   (a) For each node at hierarchy j, train ϵ-SVR by solving the Formulation 8 with a given hyperparameter configuration to obtain the solution tuple, (α, α^*).
   (b) Compute the forecast, F_n, using Formulation 9: $F_n(\mathbf{x}) = \sum_{i=1}^{m} (\alpha_i - \alpha_i^*) K(\mathbf{x}, \mathbf{x}_i) + b$
3. For each (upper) level, $k < j$, compute predictions, F_n, $\forall n \in A_k$, by using $F_n = \sum_{i \in S_n} F_i$ (the bottom-up method).
4. For each (lower) level, $k > j$, compute predictions, F_n, $\forall n \in A_k$, by using $F_n = p_n \cdot F_{D_k}$ (the top-down method).

Regarding the choice of the hyperparameters, we suggest using a standard cross-validation approach for time-series analysis (for example, rolling forecast origin Hyndman & Athanasopoulos, 2013) as the model selection procedure. Given a training and a test set, the training set is further split into training and validation subsets in such a way that the training subset does not include samples that occur after the ones included in the validation subset. In order to obtain the hyperparameter configuration for each ϵ-SVR indicated in the previous algorithms, the following validation procedure is used:

4. Experimental results

We applied the proposed methodology to various hierarchical time series datasets, including well-known benchmark datasets for this task, and data from the Swedish company, Generation Research AB. We compared the SVR forecasting performances for each dataset with both Linear and Gaussian kernel, and the traditional forecasting methods, ARIMA and Holt-Winters. For each of the forecasts the three main HTS approaches were used: bottom-up, top-down, and middle-out.

This section is organized as follows: Section 4.1 provides a description of the different datasets used, while the experimental setting is described in Section 4.2. Finally, Section 4.3 summarizes the main results obtained by the various approaches studied in this project.

4.1. Description of the datasets

Ten different datasets were used in this project. Four of them correspond to well-known hierarchical time series datasets used for benchmarking in previous studies, while the remaining six are hierarchical time series datasets based on sales data of beauty products provided by the Swedish company, Generation Research AB, the leaders in data analysis for the Travel Retail Industry.

4.1.1. Benchmark datasets

The first test dataset is called “htseg2”, and corresponds to simulated data by Hyndman et al. (2012). This hierarchical time series has four levels, with a total of 17 individual time series, each of them of length 16.

The next benchmark dataset is known as “Austourism” (Hyndman, 2015), and contains the total quarterly visitor nights from 1998 to 2011 for various regions of Australia. In this case, a three-leveled hierarchical time series is built with the total time series in the top level (level zero), the total time series disaggregated by subsets of Australian political regions in level one, and the time series for each region mentioned above in level two.

The third dataset will be referred to “Departures”, and corresponds to monthly overseas departures from Australia (Hyndman, 2015). A three-leveled hierarchy is defined, in which level zero contains the total departures. This node is disaggregated into a first level that distinguishes between permanent, long-term, and short-term departures. This first level is disaggregated further to distinguish between residents and visitors for the cases of long-term and short-term departures.

Finally, the last HTS data contains the total number of weekly flight passengers between Melbourne and Sydney for the Ansett airline (the “Melsyd” dataset) (Hyndman, 2015). This time series dataset contains flights between 1987 and 1992, and is disaggregated by flight ticket class type (first class, business, and economy) using three hierarchies.

Table 1 presents a summary of all the benchmark datasets. Further description of each dataset is given in Appendix A, including the hierarchy trees and visualization of the various related time series.

Table 1
Summary for all benchmark datasets.

Dataset	# Series	Periodicity	# Observations	Time period
Htseg2	17	–	16	–
Austourism	13	Quarterly	56	1998–2011
Departures	8	Monthly	434	1976–2012
Melsyd	5	Weekly	283	1987–1992

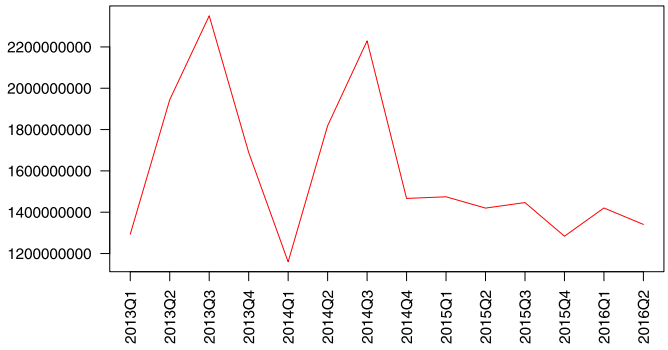


Fig. 2. Generation research's sales time series.

4.1.2. Travel retail datasets

In this study, we applied our proposal to a real case with sales data provided by Generation Research. This company is located in Örnsköldsvik, northern Sweden, from where the business is run. Even though Generation Research is a small company, it holds strong and lasting relationships with its clients, which allows it to hold sales data for almost 30% of the global market.

One of the services that Generation Research offers to its customers is sales forecasting, which helps them in managing uncertainty. Travel Retail is a very dynamic industry, and participants are eager for information to aid in making business decisions. This service is appreciated by Generation Research's clients, so the company invests continuously in novel sales forecasting strategies for improving its service.

The company has data from many different sub-industries of the Travel Retail Industry: Beauty, Confectionery & Fine food, Tobacco, and Beverage (Wines & Spirits). With this data, they provide services, such as reports, indicators, and audits, to their customers.

The data used in this study comes from the Beauty Industry, the largest one among the four sub-industries they manage. They have managed quarterly data based on sales data from retailers at locations all over the world for this industry since 2004. The data available corresponds to sales of beauty products reported in all channels and locations in Europe between the first quarter of 2013 and the second quarter of 2016 (14 quarters). Six different hierarchical structures were built based on this data, and the different product categories defined by the company.

Fig. 2 presents the total time series for the sales data provided by Generation Research. This time series includes sales of 20,802 different beauty products in the Travel Retail Industry, reported by 82 manufacturing companies in Europe between 2013 and 2016.

The product categories that were used for constructing the six hierarchical structures studied and reported in this paper are presented next.

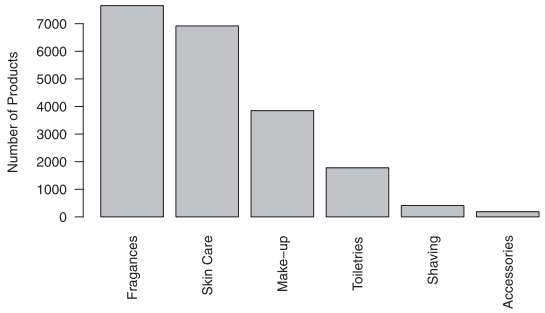
- Company: The company that reports the sale.
- Brand: The product's brand.
- Product line: A unique description of the product related to its SKU.
- First segmentation: First segmentation is defined based on the product category. The following categories were identified: fragrances, skin-care, make-up, toiletries, shaving, and accessories. The frequency plot for this segmentation is reported in Fig. 3(a).
- Second segmentation: The second segmentation defines new categories within each segment of the first segmentation. Shampoo, perfumes, eye, facial, and lip products are examples of categories in this segmentation. The "Rest" item contains all categories that have fewer than 225 products. The frequency plot for this segmentation is reported in Fig. 3(b).
- Third segmentation: Third segmentation is based on the product format, such as cream, stick, spray, or powder. The frequency plot for this segmentation is reported in Fig. 4(a). The "Rest" item contains all categories that have fewer than 200 products.
- Gender segmentation: Segmentation based on gender, which includes the following categories: Women, Men, Unisex, Children, and Mixed. The frequency plot for this segmentation is reported in Fig. 4(b).
- Target market segmentation: Whether the product corresponds to the premium or massive market. The frequency plot for this segmentation is reported in Fig. 5(a).
- G.R.-specific segmentation: This classification is defined by Generation Research with additional information of each product. Categories included in this segmentation are Travel Set, Seasonal Product, Miniatures, or Travel Retail Exclusives. The frequency plot for this segmentation is reported in Fig. 5(b).

Notice that information about the companies that provide the data, the product brands, and the product lines is not provided since it is confidential. However, general information is available:

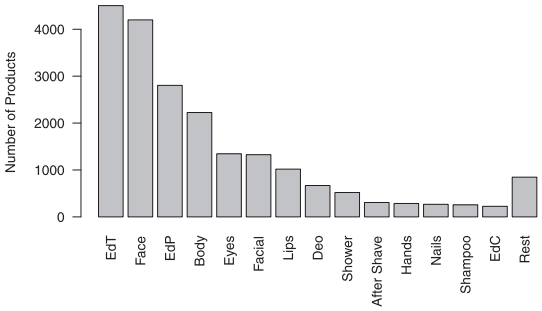
- 15 companies account for 80% of the products (16,706). The remaining 4605 products come from 66 different companies.
- There are 359 product brands. 80% of the products come from only 69 brands.
- 80% of the products contain 2855 different product lines, while the total amount of product lines is 6,217.

The dataset contains 20,801 different products. It is important to note that not all products were sold in every one of the fourteen quarters analyzed; there are some periods in which some companies did not register sales. Fig. 6 illustrates the total sales (in USD) for each period.

Based on this information, the six hierarchical structures at various levels are presented in Table 2. Our goal, then, is to take ad-

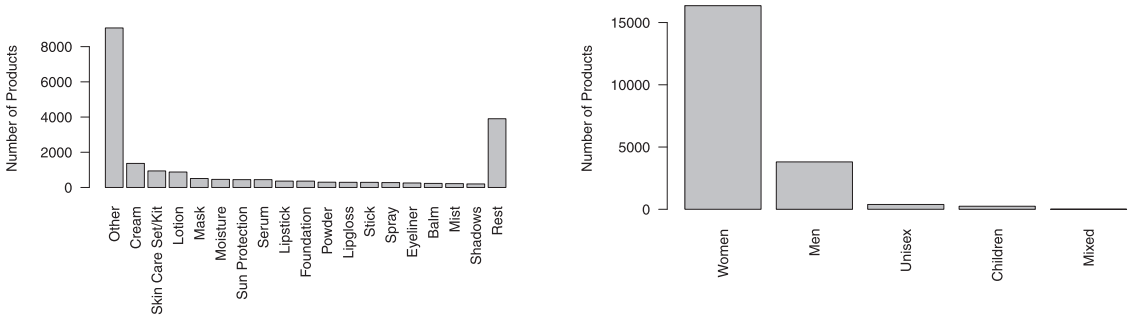


(a) First Segmentation



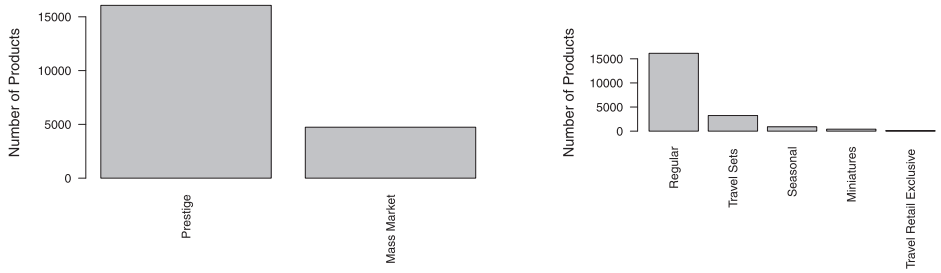
(b) Second Segmentation

Fig. 3. Descriptive analysis for the first and second segmentations.



(a) Third Segmentation (b) Gender Segmentation

Fig. 4. Descriptive analysis for the third and fourth segmentations.



(a) Target Market Segmentation (b) G.R.-Specific Segmentation

Fig. 5. Descriptive analysis for the fifth and sixth segmentations.

Table 2
All generation research hierarchical time series.

	Dataset	Level 1	Level 2	Level 3	Level 4	Level 5
1	Company	Company	Brand	Product Line	Product	–
2	Market	First Segmentation	Second Segmentation	Third Segmentation	Product	–
3	Gender	Gender	Product	–	–	–
4	LuxuryS	Target Market Segmentation	First Segmentation	Second Segmentation	Third Segmentation	Product
5	LuxuryC	Company	Target Market Segmentation	Brand	Product Line	Product
6	Other	G.R.-specific segmentation	Product	–	–	–

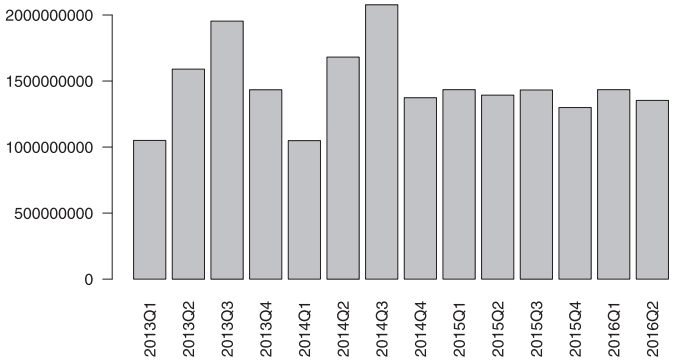


Fig. 6. Total sales per quarter for all the travel retail companies in Europe.

vantage of these structures in order to provide accurate level-based forecasts via hierarchical time series modeling, thus gaining insight into this application.

Notice that the hierarchies are not designed arbitrarily; they are inherent to the different applications considered in our study. Most hierarchical time series datasets have a natural hierarchical structure that makes sense for decision-making. See, for example, the hierarchical structures for the benchmark datasets in Appendix A. In the case study for travel retail, however, there are six different hierarchical structures, which are described in Table 2, that can be used for decision-making. We explored all six alternatives, assessing empirically the performance of our proposal in comparison with the benchmark approaches. All levels are also evaluated in terms of predictive performance.

4.2. Experimental setting

In this section, data preprocessing and model validation and selection are discussed. In regard to missing values, the benchmark datasets did not present missing information, but the travel retail data lacked data points for several products. Following the work by Liu, Yan, Yang, and Hauskrecht (2015), we explored four imputation methods that were indicated in that study: last value carried forward and backward, interpolation, predictive mean machine, and random forest. Notice that the first two methods perform missing value imputation for each time series independently, while the latter two consider the relations between products and hierarchies. Among the four approaches, random forest showed the best performance, and was therefore used for this task.

Validation and model selection were performed as follows: data was split into a training set and a test set. The training set was divided further into training and validation subsets in such a way that the training subset included only observations that occurred before the ones in the validation or test subsets (Hyndman & Athanasopoulos, 2013). This strategy is illustrated in Table 3.

For our experiments, the last value of the training set was used as the validation subset. The SVR parameters were estimated using

Table 3
Validation procedure.

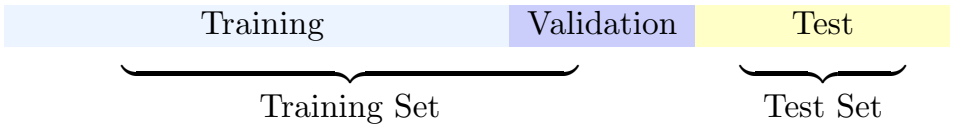


Table 4
Experimental setting for each dataset.

	Model	# training samples	# test samples	# Lags
1	Htseg2	14	2	2
2	Austourism	52	4	8
3	Departures	422	12	24
4	Melsyd	236	47	12
5	Generation datasets	12	2	2

the rolling forecasting strategy (Hyndman & Athanasopoulos, 2013) on the training and validation splits, as indicated in Algorithm 4. The following values for the SVR hyperparameters were studied:

- $\epsilon \in \{0, 0.1, 0.2, \ldots, 1\}$
- $\gamma \in \{0, 0.1, 0.2, \ldots, 1\}$
- $C \in \{2^{-7}, 2^{-6}, 2^{-5}, \ldots, 2^{10}\}$

Algorithm 4 Model selection and validation procedure.

1. Select the kernel function (Linear or Gaussian).
2. Split the training into training and validation subsets.
3. Define a range for each SVR hyperparameter:
   - ϵ and C for the Linear kernel.
   - ϵ, C and γ for the Gaussian kernel.
4. Train the model using the training set for each combination of parameters by solving the Formulation (8) to obtain (α, α^*).
5. Compute the prediction for the node on the validation set using the solution tuple, (α, α^*) and Formulation (9).
6. Calculate the Mean Absolute Percentage Error (MAPE) of the model by comparing the forecast with the true values.
7. Save the hyperparameter configuration with the lowest MAPE in the validation set.

A grid search was performed for parameters ϵ and C for the case of the Linear kernel. However, a reduced grid was used when the Gaussian kernel was chosen: A grid search was performed for parameters γ and C using $\epsilon = 0.1$, and, subsequently, we varied this last parameter for fixed γ and C.

The final performance is obtained for all methods by building the models with the full training set, and applying them to the test set, which remains unseen during the model selection process. The number of lags used as covariates, as well as the size of the training and test sets, depend on the available information. The detailed information is reported in Table 4.

For each dataset, the three hierarchical time series approaches (bottom-up, middle-out, and top-down) were tested using SVR, ARIMA, and Holt-Winters methods. the following models are reported:

- SVR BU: The best model between SVR with Linear and Gaussian Kernel with the bottom-up approach.
- SVR TD: The best model between SVR with Linear and Gaussian Kernel with the top-down approach.
- SVR MO: The best model between SVR with Linear and Gaussian Kernel with the middle-out approach, and the best among all possible start levels.
- ARIMA BU: The model built with the traditional ARIMA method and the bottom-up approach.
- ARIMA TD: The model built with the traditional ARIMA method and the top-down approach.
- ARIMA MO: The best model among all models built with the traditional ARIMA method and the middle-out approach with all possible start levels.
- HW BU: The model built with the traditional Holt-Winters method and the bottom-up approach.
- HW TD: The model built with the traditional Holt-Winters method and the top-down approach.
- HW MO: The best model among all models built with the traditional Holt-Winters method and the middle-out approach with all possible start levels.

For all models, the Mean Absolute Percentage Error (MAPE) is calculated for each node, i.e., for each time series of the hierarchy and the average MAPE is computed at each level. For each dataset, all models are ranked from 1 (best) to 9 (worst) in terms of their performance at each level, and a final model performance is obtained by averaging the rankings for all the levels. Formally, the average ranking, $\bar{R}_m$, for model m is computed as:

$$\bar{R}_m = \sum_{l=1}^{L} \frac{R_{m,l}}{L}, \tag{10}$$

where $R_{m,l}$ is the ranking achieved by model m on level l, and L is the number of levels that the dataset has.

4.3. Results summary

For each of the ten datasets, i.e., the four benchmark data sets and the six different hierarchical structures defined for the Generation Research data, the nine models were tested, computing the average MAPE per level and obtaining their respective ranks. The performance in each dataset is reported in Appendix B. The overall performance is reported in Table 5, in which the average ranking per model and dataset are presented. The overall ranking $\bar{\bar{R}}$ is computed as the average of all rankings $\bar{R}_m$ obtained by a model on all the datasets.

On average, the best performance was obtained by SVR with the bottom-up approach, followed by Holt-Winters with the bottom-up approach. Comparing all approaches, SVR achieved the best ranking not only with the bottom-up strategy, but also with the top-down approach. For the middle-out approach, Holt-Winters achieved a slightly better overall rank when compared with SVR-MO. This demonstrates the virtues of non-linear regression via SVR for hierarchical time series analysis. It can also be concluded that the bottom-up approach works best on average when compared with the top-down and middle-out approaches for the various models.

Next, the best parameters obtained for the SVR methods on each dataset are reported in Table 6.

Table 5
Result summary.

	Model	AUS	Htseg2	DEP	Melsyd	Generation datasets						$\bar{\bar{R}}$
						Other	Gender	Market	Company	LuxuryS	LuxuryC	
1	SVR BU	1	4.75	4	3.33	2.33	3.33	2.8	2.8	2.67	2.83	2.99
2	HW BU	3	3.75	1.67	8.67	2.33	2.33	3.6	1.8	2.33	1.67	3.12
3	HW MO	4.67	3.75	2	7.67	5.67	5.67	6	2	4.83	2	4.43
4	SVR TD	5.33	8.25	7	4.33	2.67	3	2.6	4.4	2.83	4.5	4.49
5	SVR MO	4	4.5	6.33	6.33	5	4.33	3.2	5.6	4.5	5.33	4.91
6	ARIMA BU	6	4	4	4.67	4.67	4	5	6	5	6	4.93
7	ARIMA MO	5.33	2.25	6.33	5.67	5.67	5.67	6.4	6	5.83	6.17	5.53
8	ARIMA TD	9	6.25	7.67	1.33	7.67	7.67	7	7.6	8	7.67	6.99
9	HW TD	6.67	7.5	6	3	9	9	8.4	8.8	9	8.83	7.62

Table 6
Best parameters for SVR models.

Dataset		Gaussian kernel			Linear kernel	
		ϵ	γ	c	ϵ	c
1	Htseg2	0.1	0.1	2^6	0.3	2^4
2	Austourism	0	0.1	2^0	0.3	2^0
3	Departures	0.1	0.2	2^{-1}	0.1	2^{-3}
4	Melsyd	0.1	0.9	2^{-1}	0.5	2^{-1}
5	All Generation datasets	0.1	0.1	2^9	0	2^{-7}

Finally, Table 7 presents the average running times for all the methods and datasets considering the best parameter configurations obtained during the model selection procedure. All the methods were implemented on a MacBook Pro with 8GB 2133 MHz LPDDR3 Memory, 128 GB SSD, a 2.3 GHz dual-core 7th-generation Intel Core i5 processor and using macOS Mojave 10.14.4 Operating System (64-bits). The R software was used as the programming language. For the middle-out approaches (MO), only the training times when level 1 was used as the middle level are reported because the training times are relatively similar when other levels are used as the pivot. For the SVR approaches, only the Gaussian kernel is reported since the running times are relatively similar when the linear kernel is used. For the SVR approaches, we split the total running time into two parts: (1) all SVR trainings, and (2) all mathematical operations for obtaining the aggregations and disaggregations.

It can be observed in Table 7 that our proposal is the fastest strategy for the small datasets, but the slowest one for the Generation datasets. It should be noted that the large running times reported for the SVR models are due to the construction of the aggregations and disaggregations of the forecasts; training all the SVR approaches is tractable and below one minute for all cases. In particular, the large running times reported for the SVR BU are due to our implementation of the aggregation process, which is inefficient when compared with the 'forecast' toolbox used for the ARIMA and Holt-Winter models.

5. Conclusions

This paper presents forecasting methods for hierarchical time series (HTS) based on Support Vector Regression (SVR). The three classic HTS approaches (bottom-up, top-down, and middle-out) were formalized and empirically tested on ten datasets. Six of these ten datasets represent a real-world project for Generation Research AB, an important consulting company for the Travel Retail Industry.

It is somewhat expected that our method is not always best in terms of predictive performance given the "no free lunch" of machine learning. This is particularly true in time series modeling, in which models are trained using few data samples. Nevertheless, our experiments show that our proposals perform consistently better on average than the alternatives based on statistical methods. It can be concluded that SVR is an excellent alternative for HTS modeling given not only its ability to construct flexible regression functions via kernel functions, but also to reduce the risk of overfitting by finding an adequate compromise between model fit and regularization.

Some managerial insights into the travel retail application can be gained. First, the bottom-up approaches tend to predict better than top-down and middle-out methods, despite the fact that the datasets are rather noisy at the bottom level. This confirms that SKU-level forecasting is a better strategy than using historical proportion for sales forecasting, and also demonstrating the virtues of

Table 7
Running times, in seconds, for all methods and datasets.

Model	AUS	Htseg2	DEP	Melsyd	Generation datasets					
					Other	Gender	Market	Company	LuxuryS	LuxuryC
ARIMA BU	2″.7	0″.569	9″.7	26″.7	716″.6	717″.5	748″.1	694″.2	693″.4	697″.6
ARIMA MO	4″.6	0″.817	15″.4	47″.0	739″.2	738″.0	787″.3	1023″.9	748″.5	1025″.7
ARIMA TD	0″.264	0″.053	0″.911	10″.4	0″.264	0″.263	0″.246	0″.246	0″.236	0″.238
HW BU	1″.1	0″.068	5″.8	0″.045	120″.3	120″.4	121″.7	120″.2	118″.4	120″.5
HW MO	1″.7	0″.122	9″.313	0″.088	126″.0	126″.4	131″.6	177″.5	130″.3	178″.9
HW TD	0″.110	0″.008	0″.877	0″.015	0″.240	0″.241	0″.213	0″.212	0″.206	0″.186
SVR BU train	0″.045	0″.025	0″.146	0″.037	40″.5	40″.6	41″.0	38″.9	38″.2	38″.6
SVR BU op	0″.007	0″.010	0″.005	0″.004	90″.4	90″.1	6439″.2	98,857″.3	10,827″.9	95,578″.4
SVR BU total	0″.052	0″.035	0″.151	0″.041	130″.9	130″.6	6480″.2	98,896″.2	10,866″.1	95,617″.0
SVR MO train	0″.012	0″.005	0″.082	0″.029	0″.013	0″.013	0″.015	0″.214	0″.005	0″.231
SVR MO op.	0″.011	0″.026	0″.009	0″.007	592″.0	590″.7	1782″.4	1983″.6	2279″.8	2657″.6
SVR MO total	0″.023	0″.031	0″.091	0″.036	592″.0	590″.7	1782″.5	1983″.8	2279″.8	2657″.8
SVR TD train	0″.003	0″.002	0″.020	0″.014	0″.002	0″.002	0″.003	0″.003	0″.003	0″.003
SVR TD op.	0″.008	0″.012	0″.005	0″.004	584″.5	584″.4	587″.3	661″.0	567″.7	659″.3
SVR TD total	0″.011	0″.014	0″.025	0″.018	584″.5	584″.4	587″.3	661″.0	567″.7	659″.3

adequate SKU-level data preprocessing using state-of-the-art techniques, such as random forests. Kernel-based SVR performs best in terms of prediction accuracy, confirming that there is an intrinsic nonlinearity in this type of data that can be modeled adequately with machine learning models.

The main limitation of the proposed approaches, and time series in general, is the lack of data availability. There is a trade-off between constructing models with only a few recent data points and including more samples from sales made in past periods that may not be relevant for predicting new sales. This issue is aggravated with machine learning models since these techniques require a parameter tuning step, reducing the training set further. Therefore, we chose SVR as the modeling approach since it has fewer hyperparameters than other machine learning methods, such as Artificial Neural Networks (ANNs).

Regarding future developments, there are some research opportunities that could complement this work. For example, machine learning methods different from SVR can be used for HTS. They could be designed and implemented, given the virtues shown by SVR in the present study. For example, ANNs have gained increasing attention in recent years, ever since the success of deep learning in tasks such as computer vision, Natural Language Processing (NLP), and Internet of Things (IoT) Analytics. However, the main limitation of the current ANN research is that it requires large datasets in order to show important gains when compared with other predictive approaches. This is usually not the case in HTS forecasting. Additionally, feature selection can be studied for the SVR approaches to define the relevant lags for each time series model. There are several studies that discuss feature selection in SVR (see e.g. Dai, Shao, & Lu, 2013; Maldonado & Weber, 2010), and some of these strategies can be useful in an HTS context. Finally, our study can be extended to more sophisticated HTS strategies besides the classic ones (bottom-up, top-down, and middle-out). For example, SVR can be used with the Optimal Reconciliation Approach (Hyndman & Athanasopoulos, 2014), or with an intelligent top-down approach that takes the evolution of the proportions over time into consideration.

Conflict of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Credit authorship contribution statement

Juan Pablo Karmy: Data curation, Conceptualization, Validation. **Sebastián Maldonado:** Conceptualization, Data curation, Writing - original draft.

Acknowledgments

The second author was supported by FONDECYT project 1160738. This research was partially funded by the Complex Engineering Systems Institute, ISCI (ICM-FIC: P05-004-F, CONICYT: FB0816).

Appendix A. Description of the satasets

The first benchmark dataset is called “htseg2”, and has four levels with a total of 17 individual time series. This dataset has two time series on the first level: *A* and *B*. Each node, *A* and *B*, is disaggregated into two new nodes, from which the last ten nodes of the lowest level are born. This is shown in Fig. A.1.

Fig. A.2 illustrates all the individual time series by level.

The next dataset is known as “Austourism” (Hyndman, 2015), and contains the total of the quarterly visitor nights from 1998 to 2011 for various regions of Australia. This dataset has the following structure (see Fig. A.3):

where

- Sydney: the Sydney metropolitan area.
- NSW: New South Wales other than Sydney.
- Melbourne: the Melbourne metropolitan area.
- VIC: Victoria other than Melbourne.
- BrisbaneGC: the Brisbane and Gold Coast area.
- QLD: Queensland other than Brisbane and the Gold Coast area.
- Capitals: the other five capital cities (Adelaide, Hobart, Perth, Darwin, and Canberra).
- Other: all other areas of Australia.

Fig. A.4 presents all the time series of this dataset by level:

The third dataset (“Departures”) studies the monthly overseas departures from Australia (Hyndman, 2015), and has the following structure (Fig. A.5):

where

- Permanent: permanent departures from Australia.
- Reslong: long-term resident departures from Australia.
- Vislong: long-term visitor departures from Australia.
- Resshort: short-term resident departures from Australia.
- Visshort: short-term visitor departures from Australia.

Fig. A.6 illustrates all the time series for the various hierarchy levels.

Finally, the “Melsyd” dataset contains the total number of weekly flight passengers between Melbourne and Sydney for the Ansett airline (Hyndman, 2015). Fig. A.7 presents the hierarchical structure of the dataset, while Fig. A.8 illustrates all the time series plotted by level.

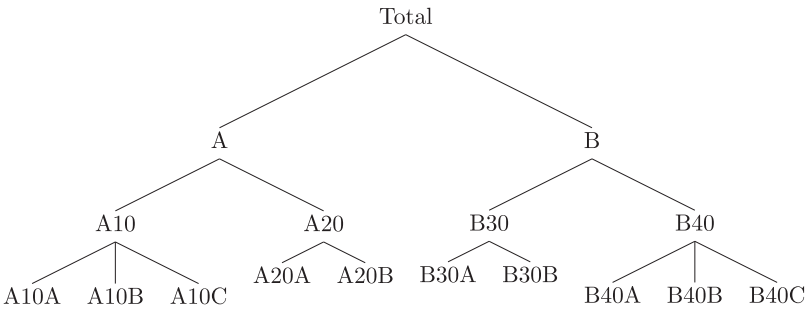


Fig. A.1. Tree diagram for the Htseg2 dataset.

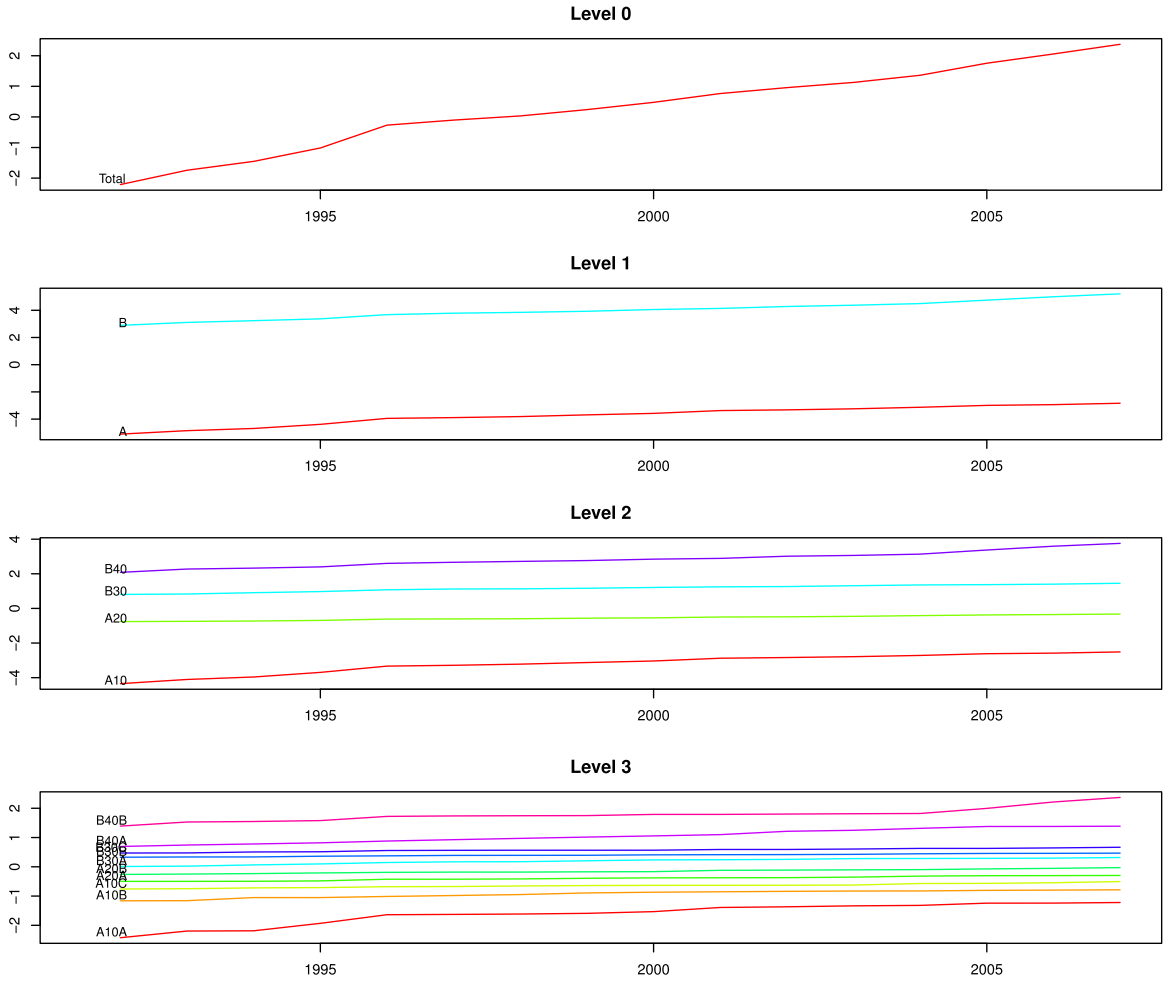


Fig. A.2. Timeplots for the Htseg2 dataset.

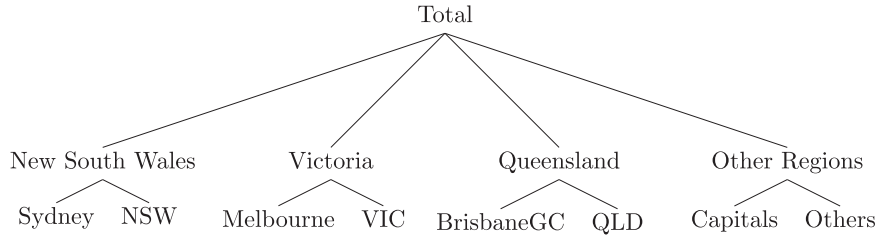


Fig. A.3. Tree diagram for the Austourism dataset.

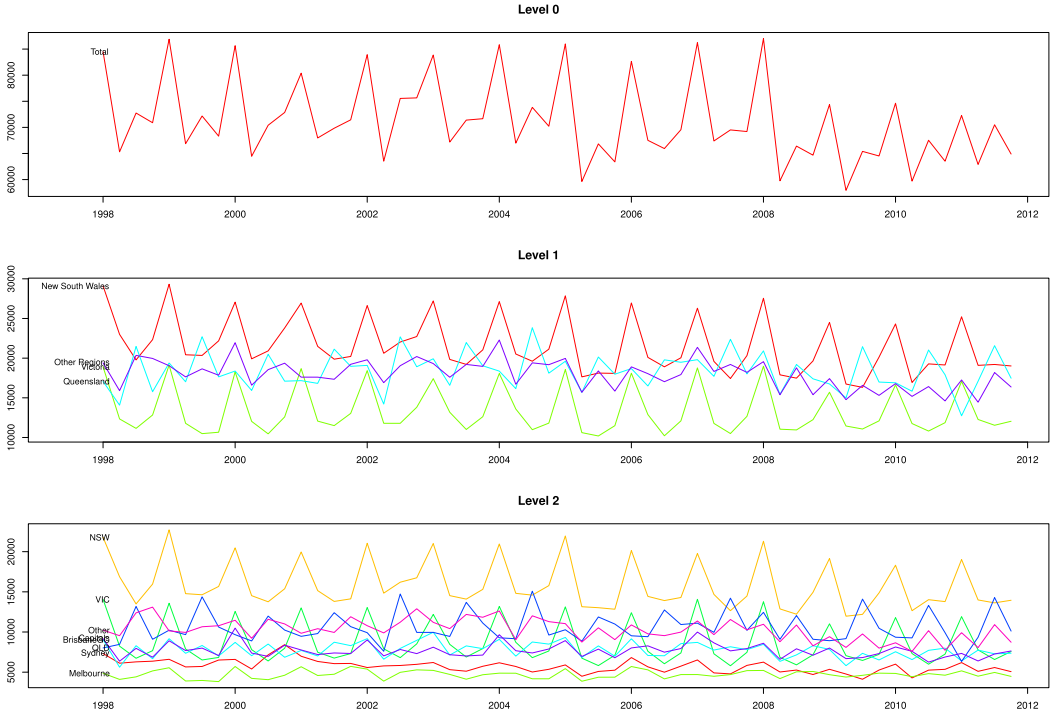


Fig. A.4. Timeplots for the Austourism dataset.

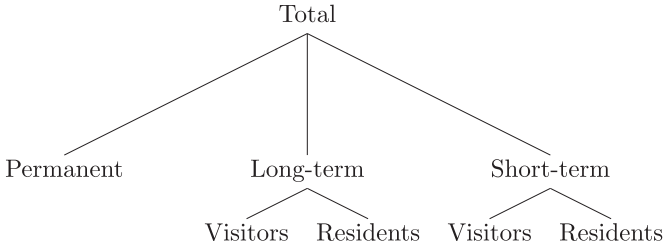


Fig. A.5. Tree diagram for the Departures dataset.

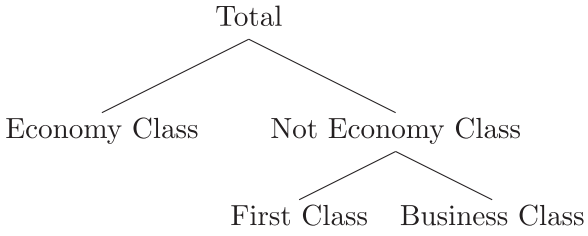


Fig. A.7. Tree diagram for the Melsyd dataset.

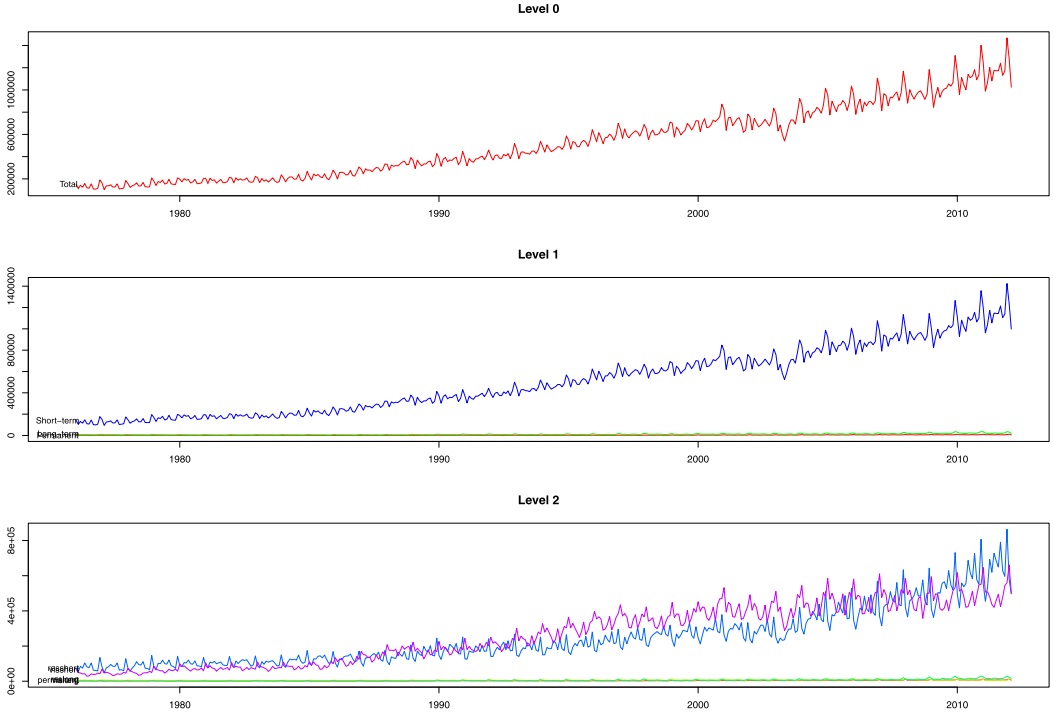


Fig. A.6. Timeplots for the Departures dataset.

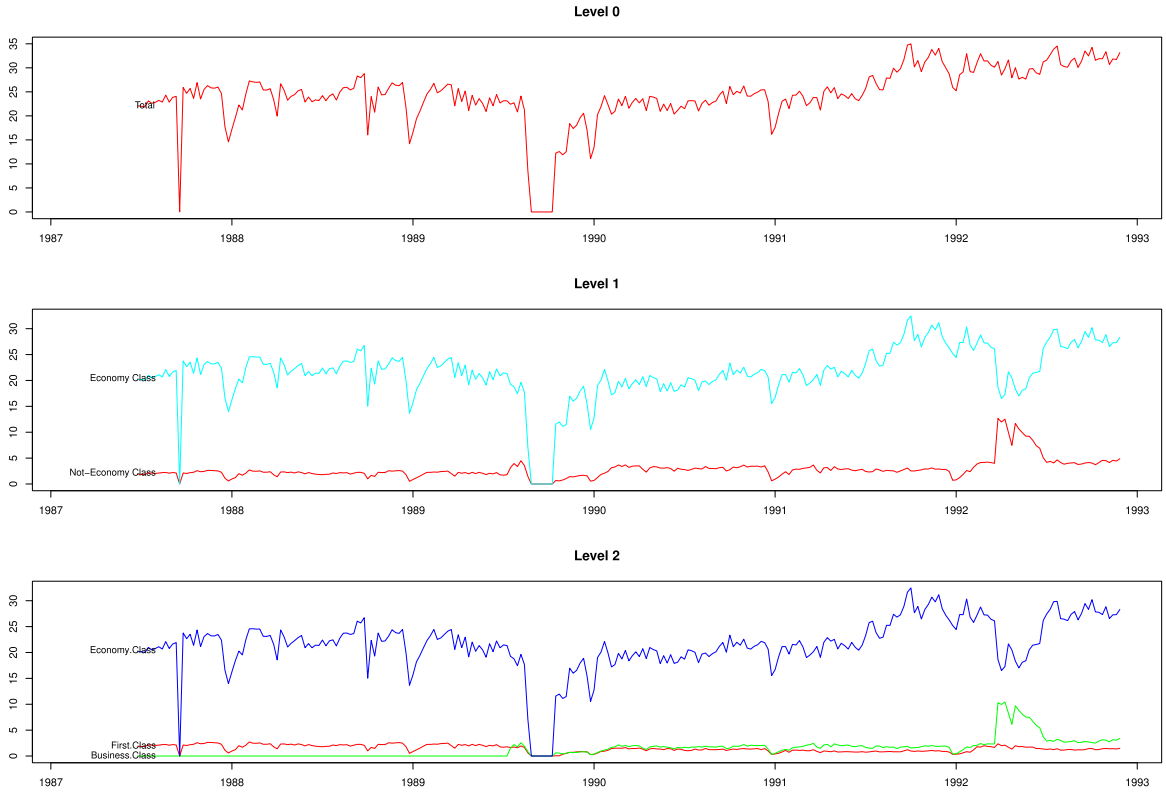


Fig. A.8. Timeplots for the Melsyd dataset.

Appendix B. Detailed results for each dataset

Table B.1 presents the results for the benchmark dataset "Htseg2". The best model for this dataset corresponds to ARIMA based on the middle-out approach, followed by Holt-Winters based on the bottom-up strategy. SVR models do not perform as well as these methods, probably because this dataset was simulated for comparing ARIMA and Holt-Winters models, and its construction does not consider complex non-linear patterns that are present in the real world.

Something similar occurs with the "Departures" dataset. The results shown in Table B.2 show the clear advantage of the traditional methods over SVR. In this dataset, SVR is superior to ARIMA, but is not better than Holt-Winters for each HTS approach.

For the "Melsyd" dataset (Table B.3), the model with the best performance is ARIMA with the top-down approach, followed by Holt-Winters with the same approach. SVR achieves the best performance among the bottom-up strategies, also performing better than the Holt-Winters strategy for the middle-out approach.

For the "Austourism" dataset (Table B.4), SVR with the bottom-up approach has the best performance, followed by Holt-Winters based on the same strategy. An important finding to note is the fact that SVR performs better than ARIMA and Holt-Winters for all the hierarchical approaches (bottom-up, top-down, and middle-out).

Table B.5 represents the first of the six hierarchical time series built with Generation Research's data. For this dataset, the best average ranking was achieved by Holt-Winters models given their positive performance at the lower levels, followed by all the SVR approaches.

For the "LuxuryS" dataset (Table B.6), the best ranked model is Holt-Winters, but SVR performs better than all the rest of the models.

Something similar occurs for the "Company" dataset. Table B.7 shows that Holt-Winters has a better ranking than SVR with both

Table B.1
Detailed results for the Htseg2 dataset.

		Average MAPE				Ranking				
	Model	Total	Level 1	Level 2	Level 3	Total	Level 1	Level 2	Level 3	Average ranking
1	ARIMA MO	0.23	2.52	2.34	7.58	2	3	2	2	2.25
2	HW MO	2.24	2.66	2.59	8.22	4	4	3	4	3.75
3	HW BU	10.12	2.45	2.07	8.36	7	2	1	5	3.75
4	ARIMA BU	0.23	3.78	3.05	7.82	3	6	4	3	4
5	SVR MO	4.9	1.6	8.53	27.98	5	1	6	6	4.5
6	SVR BU	12.63	3.49	3.53	6.49	8	5	5	1	4.75
7	ARIMA TD	0.232	318.288	355.287	448.31	1	8	8	8	6.25
8	HW TD	19.41	236.75	266.43	340.11	9	7	7	7	7.5
9	SVR TD	8.87	600.18	650.61	785.9	6	9	9	9	8.25

Table B.2
Detailed results for the Departures dataset.

		Average MAPE			Ranking			
	Model	Total	Level 1	Level 2	Total	Level 1	Level 2	Average ranking
1	HW BU	1.92	5	6.12	1	2	2	1.67
2	HW MO	2.32	3.72	5.15	4	1	1	2
3	SVR BU	2.29	5.89	7.26	3	4	5	4
4	ARIMA BU	2.34	5.67	6.58	6	3	3	4
5	HW TD	2.02	18	64.17	2	8	8	6
6	SVR MO	2.9	5.95	34.49	8	5	6	6.33
7	ARIMA MO	2.97	6.42	6.92	9	6	4	6.33
8	SVR TD	2.72	10.31	39.09	7	7	7	7
9	ARIMA TD	2.33	19.44	64.74	5	9	9	7.67

Table B.3
Detailed results for the Melsyd dataset.

		Average MAPE			Ranking			
	Model	Total	Level 1	Level 2	Total	Level 1	Level 2	Average ranking
1	ARIMA TD	9.99	24.46	32.21	1	1	2	1.33
2	HW TD	13.73	27.02	32.47	4	2	3	3
3	SVR BU	15.94	29.37	31.09	6	3	1	3.33
4	SVR TD	15.15	29.56	34.82	5	4	4	4.33
5	ARIMA BU	11.05	41.48	48.57	2	6	6	4.67
6	ARIMA MO	12.46	42.55	51.23	3	7	7	5.67
7	SVR MO	18.62	36.07	35.2	9	5	5	6.33
8	HW MO	16.66	48.53	57.71	7	8	8	7.67
9	HW BU	17.77	48.73	57.93	8	9	9	8.67

Table B.4
Detailed results for the Austourism dataset.

		Average MAPE			Ranking			
	Model	Total	Level 1	Level 2	Total	Level 1	Level 2	Average ranking
1	SVR BU	2.6	6.28	7.42	1	1	1	1
2	HW BU	5.2	6.78	8.81	5	2	2	3
3	SVR MO	3.32	6.98	9.86	3	3	6	4
4	HW MO	5.89	7.27	9.24	7	4	3	4.67
5	SVR TD	2.98	10.45	10.7	2	7	7	5.33
6	ARIMA MO	5.72	7.99	9.75	6	6	4	5.33
7	ARIMA BU	6.1	7.36	9.81	8	5	5	6
8	HW TD	5.08	10.61	11.37	4	8	8	6.67
9	ARIMA TD	6.26	11.25	11.83	9	9	9	9

Table B.5
Detailed results for the G.R. dataset, LuxuryC structure.

		Average MAPE						Ranking						
	Model	Total	Level 1	Level 2	Level 3	Level 4	Level 5	Total	Level 1	Level 2	Level 3	Level 4	Level 5	Average ranking
1	HW BU	6.12	647.15	571.2	753.91	2444.67	3847.16	3	1	1	2	2	1	1.67
2	HW MO	6.96	651.99	575.5	734.82	2444.64	3893.16	4	2	2	1	1	2	2
3	SVR BU	4.45	816.15	717.29	813.35	4246.08	6160.91	2	3	3	3	3	3	2.83
4	SVR TD	0.69	833.68	733.47	1084.45	8145.13	8763.1	1	5	5	4	6	6	4.5
5	SVR MO	8.32	832.09	732.65	1125.61	9171.24	9852.88	5	4	4	5	7	7	5.33
6	ARIMA BU	9.64	947.13	831.67	1309.65	5757.59	6753.8	6	6	6	9	4	5	6
7	ARIMA MO	10.01	948.65	833.13	1249.62	5757.66	6677.7	7	7	7	7	5	4	6.17
8	ARIMA TD	18.26	967.62	850.51	1230.84	10,708.3	11,238.4	8	8	8	6	8	8	7.67
9	HW TD	22.11	1001.55	880.36	1272.57	11,058.47	11,605.78	9	9	9	8	9	9	8.83

Table B.6
Detailed results for the G.R. dataset, LuxuryS structure.

		Average MAPE						Ranking						
	Model	Total	Level 1	Level 2	Level 3	Level 4	Level 5	Total	Level 1	Level 2	Level 3	Level 4	Level 5	Average ranking
1	HW BU	6.12	4.27	11.98	36.7	692.58	3847.16	4	3	2	3	1	1	2.33
2	SVR BU	4.45	3.55	12.76	33.49	850.3	6160.91	3	2	4	1	3	3	2.67
3	SVR TD	0.69	2.31	10.85	34.87	934.6	8763.39	1	1	1	2	6	6	2.83
4	SVR MO	0.88	4.85	12.7	37.32	969.67	8862.74	2	4	3	4	7	7	4.5
5	HW MO	17.57	12.05	14.99	38.56	698.25	4211.69	7	7	6	5	2	2	4.83
6	ARIMA BU	9.64	7.16	13.58	45.81	900.59	6753.8	5	5	5	7	4	4	5
7	ARIMA MO	15.14	11.45	15.04	45.69	904.46	7204.24	6	6	7	6	5	5	5.83
8	ARIMA TD	18.26	16.23	21.21	49.29	1115.57	11,238.4	8	8	8	8	8	8	8
9	HW TD	22.11	20.02	24.35	52.55	1153.23	11,605.78	9	9	9	9	9	9	9

Table B.7
Detailed results for the G.R. dataset, Company structure.

		Average MAPE					Ranking					
	Model	Total	Level 1	Level 2	Level 3	Level 4	Total	Level 1	Level 2	Level 3	Level 4	Average ranking
1	HW BU	6.12	647.15	753.91	2444.67	3847.16	3	1	2	2	1	1.8
2	HW MO	6.96	651.99	743.4	2444.64	3893.16	4	2	1	1	2	2
3	SVR BU	4.45	816.15	813.35	4246.08	6160.91	2	3	3	3	3	2.8
4	SVR TD	0.72	833.38	1084.06	8142.21	8759.99	1	5	4	6	6	4.4
5	SVR MO	8.32	832.09	1125.61	9171.24	9852.88	5	4	5	7	7	5.6
6	ARIMA MO	10.01	948.65	1258.43	5757.66	6677.7	7	7	7	5	4	6
7	ARIMA BU	9.64	947.13	1309.65	5757.59	6753.8	6	6	9	4	5	6
8	ARIMA TD	18.26	967.62	1230.84	10,708.3	11,238.4	8	8	6	8	8	7.6
9	HW TD	22.11	1001.55	1272.57	11,058.47	11,605.78	9	9	8	9	9	8.8

Table B.8
Detailed results for the G.R. dataset, Market structure.

		Average MAPE					Ranking					
	Model	Total	Level 1	Level 2	Level 3	Level 4	Total	Level 1	Level 2	Level 3	Level 4	Average ranking
1	SVR TD	0.69	8.41	25.11	328.22	8764	1	1	1	3	7	2.6
2	SVR BU	4.45	11.17	28.68	357.62	6160.91	3	3	3	2	3	2.8
3	SVR MO	1.54	12.08	27.54	357.57	8397.93	2	5	2	1	6	3.2
4	HW BU	6.12	11.01	36.05	425.92	3847.16	4	2	5	6	1	3.6
5	ARIMA BU	9.64	11	41.03	418	6753.8	5	4	8	4	4	5
6	HW MO	18.33	16.03	38.2	429.91	4181.49	8	6	7	7	2	6
7	ARIMA MO	16.13	16.9	41.88	423.54	7241.2	6	7	9	5	5	6.4
8	ARIMA TD	18.26	17.01	35.28	479.47	11,238.4	7	8	4	8	8	7
9	HW TD	22.11	19.2	37.92	496.45	11,605.78	9	9	6	9	9	8.4

Table B.9
Detailed results for the G.R. dataset, Other structure.

		Average MAPE			Ranking			
	Model	Total	Level 1	Level 2	Total	Level 1	Level 2	Average ranking
1	SVR BU	4.45	10.1	6160.91	2	2	3	2.33
2	HW BU	6.12	10.47	3847.16	3	3	1	2.33
3	SVR TD	0.69	9.59	8763.26	1	1	6	2.67
4	ARIMA BU	9.64	12.88	6753.8	5	5	4	4.67
5	SVR MO	6.17	11.13	9297.06	4	4	7	5
6	ARIMA MO	17.06	15.23	7085.53	6	6	5	5.67
7	HW MO	20.85	15.88	4320.54	8	7	2	5.67
8	ARIMA TD	18.26	17.05	11,238.4	7	8	8	7.67
9	HW TD	22.11	19.99	11,605.78	9	9	9	9

Table B.10
Detailed results for the G.R. dataset, Gender structure.

		Average MAPE			Ranking			
	Model	Total	Level 1	Level 2	Total	Level 1	Level 2	Average ranking
1	HW BU	6.12	11.84	3847.16	4	2	1	2.33
2	SVR TD	0.68	6.36	8765.33	1	1	7	3
3	SVR BU	4.45	15.19	6160.91	3	4	3	3.33
4	ARIMA BU	9.64	13.19	6753.8	5	3	4	4
5	SVR MO	2.74	17.53	8143.58	2	5	6	4.33
6	ARIMA MO	18.26	18.17	7257.12	6	6	5	5.67
7	HW MO	21.92	18.71	4423.81	8	7	2	5.67
8	ARIMA TD	18.26	21.12	11,238.4	7	8	8	7.67
9	HW TD	22.11	25.07	11,605.78	9	9	9	9

the bottom-up and middle-out approaches, because it has better performance at the lower levels. At the higher levels, the best models are the SVR approaches.

For the “Market” dataset, SVR has the best performance for all HTS approaches. These results are presented in Table B.8.

For the “Other” dataset (Table B.9), the best model is SVR with the bottom-up approach. For each approach, SVR is the model with the best performance.

For the “Gender” dataset, the best ranked model is Holt-Winters with the bottom-up approach. SVR is better than ARIMA and Holt-Winters for all the other approaches. With bottom-up, SVR is better than ARIMA in general, and better than Holt-Winter at the top levels. These results are presented in Table B.10.

References

Ahmed, N. K., Atiya, A. F., El-Gayar, N., & El-Shishiny, H. (2010). An empirical comparison of machine learning models for time series forecasting. *Econometric Reviews, 29*(5–6), 594–621.

Athanasopoulos, G., Hyndman, R. J., Kourentzes, N., & Petropoulos, F. (2017). *Forecasting with temporal hierarchies*. Working Paper.

Crone, S., Guajardo, J., & Weber, R. (2006). Artificial intelligence in theory and practice. In M. Bramer (Ed.), *TFIP International Federation for Information Processing: 217* (pp. 149–158). Boston: Springer.

Crone, S., Lessmann, S., & Pietsch, S. (2006). Forecasting with computational intelligence – An evaluation of support vector regression and artificial neural networks for time series prediction. In *The 2006 IEEE international joint conference on neural network proceedings* (pp. 3159–3166).

Dai, W. S., Shao, Y. E., & Lu, C. (2013). Incorporating feature selection method into support vector regression for stock index forecasting. *Neural Computing and Applications, 23*(6), 1551–1561.

Drucker, H., Burges, C., Kaufman, L., Smola, A., & Vapnik, V. (1997). Support vector regression machines. In *Advances in neural information processing systems: 9* (pp. 155–161). MIT Press.

Du, X. F., Leung, S. C. H., Zhang, J. L., & Lai, K. K. (2013). Demand forecasting of perishable farm products using support vector machine. *International Journal of Systems Science, 44*(33), 556–567.

Franses, P. H., & Legerstee, R. (2011). Combining SKU-level sales forecasts from models and experts. *Expert Systems with Applications, 38*(3), 2365–2370.

Hansen, J., McDonald, J., & Nelson, R. (2006). Some evidence on forecasting time-series with support vector machines. *Journal of the Operational Research Society, 57*(9), 1053–1063.

Huber, J., Gossmann, A., & Stuckenschmidt, H. (2017). Cluster-based hierarchical demand forecasting for perishable goods. *Expert Systems with Applications, 76*, 140–151.

Hyndman, R. J. (2015). Package fpp. https://cran.r-project.org/web/packages/fpp/fpp.pdf.

Hyndman, R. J., & Athanasopoulos, G. (2013). *Forecasting: Principles and practice*. O Texts.

Hyndman, R. J., Athanasopoulos, G., & Shang, H. L. (2012). hts: An R package for forecasting hierarchical or grouped time series. https://cran.r-project.org/web/packages/hts/hts.pdf.

Jiang, S., Chin, K.-S., Wang, L., Qu, G., & Tsui, K. L. (2017). Modified genetic algorithm-based feature selection combined with pre-trained deep neural network for demand forecasting in outpatient department. *Expert Systems with Applications, 82*, 216–230.

Levis, A. A., & Papageorgiou, L. G. (2009). Customer demand forecasting via support vector regression analysis. *Chemical Engineering Research and Design, 83*(8), 1009–1018.

Liu, Z., Yan, Y., Yang, J., & Hauskrecht, M. (2015). Missing value estimation for hierarchical time series: A study of hierarchical web traffic. In *2015 IEEE international conference on data mining* (pp. 895–900).

Lu, C.-J. (2014). Sales forecasting of computer products based on variable selection scheme and support vector regression. *Neurocomputing, 128*, 491–499.

Lu, C.-J., & Wang, Y.-W. (2010). Combining independent component analysis and growing hierarchical self-organizing maps with support vector regression in product demand forecasting. *International Journal of Production Economics, 128*(2), 603–613.

Makridakis, S., & Wheelwright, S. C. (1977). Forecasting: Issues & challenges for marketing management. *Journal of Marketing, 41*(4), 24–38.

Maldonado, S., & Weber, R. (2010). Feature selection for support vector regression via kernel penalization. In *2010 international joint conference on neural networks proceedings* (pp. 1973–1979).

Martínez, F., Frías, M. P., Pérez-Godoy, M. D., & Rivera, A. J. (2018). Dealing with seasonality by narrowing the training set in time series forecasting with KNN. *Expert Systems with Applications, 103*, 38–48.

Hyndman, R. J., & Athanasopoulos, G. (2014). Optimally reconciling forecasts in a hierarchy. *Foresight*, 42–48.

Sapankevych, N., & Sankar, R. (2009). Time series prediction using support vector machines: A survey. *IEEE Computational Intelligence Magazine, 4*, 24–38.

Schölkopf, B., & Smola, A. J. (2002). *Learning with kernels*. Cambridge, MA, USA.: MIT Press.

Smola, A. J., & Schölkopf, B. (1998). A tutorial on support vector regression. *Technical Report, NeuroCOLT Technical Report NC-TR-98-030*. Royal Holloway College, University of London, UK.

Steffen, M., Sarda, A., Artz-Beielstein, T., Zaefferer, M., & Strok, J. (2015). Comparison of different methods for univariate time series imputation in R. arXiv:1510.03924.

Wood, S., & Tasker, A. (2008). The importance of context in store forecasting: The site visit in retail location decision-making. *Journal of Targeting, Measurement and Analysis for Marketing, 16*(2), 139–155.