Applied Soft Computing 112 (2021) 107756

Contents lists available at ScienceDirect

Applied Soft Computing

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

Hierarchical forecast reconciliation with machine learning

Evangelos Spiliotis [a], Mahdi Abolghasemi [b], Rob J. Hyndman [c], Fotios Petropoulos [d,*], Vassilios Assimakopoulos [a]

[a] Forecasting and Strategy Unit, School of Electrical and Computer Engineering, National Technical University of Athens, Greece
[b] Department of Data Science & AI, Monash University, Melbourne, Australia
[c] Department of Econometrics and Business Statistics, Monash University, Melbourne, Australia
[d] School of Management, University of Bath, UK

ARTICLE INFO

Article history:
Received 1 June 2020
Received in revised form 7 June 2021
Accepted 22 July 2021
Available online 30 July 2021

Keywords:
Forecasting
Time series
Hierarchies
Non-linear coherence

ABSTRACT

Over the last 15 years, studies on hierarchical forecasting have moved away from single-level approaches towards proposing linear combination approaches across multiple levels of the hierarchy. Such combinations offer coherent reconciled forecasts, improved forecasting performance and aligned decision-making. This paper proposes a novel hierarchical forecasting approach based on machine learning. The proposed method allows for non-linear combinations of the base forecasts, thus being more general than linear approaches. We structurally combine the objectives of improved post-sample empirical forecasting accuracy and coherence. Due to its non-linear nature, our approach selectively combines the base forecasts in a direct and automated way without requiring that the complete information must be used for producing reconciled forecasts for each series and level. The proposed method is evaluated both in terms of accuracy and bias using two different data sets coming from the tourism and retail industries. Our results suggest that the proposed method gives superior point forecasts than existing approaches, especially when the series comprising the hierarchy are not characterized by the same patterns.

1. Introduction and background

Accurate forecasting helps decision-making, especially when the future is uncertain. For example, forecasting the future demand of stock keeping units (SKUs) helps in managing a supply chain, and forecasting tourist arrivals helps in capacity planning.

Frequently, the time series to be forecast are naturally organized in hierarchical structures. For instance, although the demand for an SKU could be recorded on a store-by-store level, it could also be aggregated to give the demand on a regional or national level. At the same time, the demand of similar SKUs could be included in the demand of larger categories of products. These structures led to the development of hierarchical forecasting (HF) approaches. Such approaches are proposed in the cross-sectional [1–3], temporal [4], and cross-temporal domains [5,6]. Petropoulos et al. [7], section 2.9, provide a short overview of forecasting by aggregation.

The observed demands at each level will always add up to the observed demand at higher levels. It is usually desirable that the same holds true for forecasts — that is, that the aggregate of the forecasts at a lower level is equal to the forecast of the aggregates at a higher level. This property is known as forecasting "coherence" [8]. If forecasting at the different levels is done independently, it is very likely to have forecast incoherence — the forecasts do not add up.

Until the late 2010s, the problem of forecast incoherence was bypassed by modeling and producing forecasts on a single hierarchical level:

- Some researchers [9,10] have argued for generating forecasts only on the lowest, most granular level of a hierarchy. If forecasts are needed at higher levels, these are not produced directly using the aggregated information; instead, the lower level forecasts are summed up. This approach is known as "bottom-up" (BU). The BU approach can be more suitable for short-term operational decisions, such as logistics and production planning [11]. A downside of the BU approach is the difficulty to model each bottom level series due to the high level of noise and computational concerns in the case of large hierarchies [1,12].

[*] Correspondence to: School of Management, University of Bath, Claverton Down, Bath, BA2 7AY, UK.
E-mail addresses: spiliotis@fsu.gr (E. Spiliotis), mahdi.abolghasemi@monash.edu (M. Abolghasemi), rob.hyndman@monash.edu (R.J. Hyndman), f.petropoulos@bath.ac.uk (F. Petropoulos), vassim@central.ntua.gr (V. Assimakopoulos).

https://doi.org/10.1016/j.asoc.2021.107756
1568-4946/© 2021 Elsevier B.V. All rights reserved.

- Other researchers [12,13] have suggested that only the top level of a hierarchy be directly forecasted, and then the forecasts are disaggregated to the lower levels using historical or forecasted [1] proportions. This approach is known as "top-down" (TD). TD is more appropriate when strategic plans and decisions such as budgeting are made. TD generally requires fewer resources and modeling decisions, with forecasts being made on a single (top) series. However, the accuracy of the forecasts drops at lower levels of the hierarchy, due to the information loss incurred while aggregating the lower-level data to the higher aggregation levels.
- A solution between BU and TD is offered by the "middle-out" (MO) approach. In MO, forecasts are produced on an intermediate level of the hierarchy. Lower and higher level forecasts are derived by disaggregation and aggregation of the MO forecasts respectively.

The BU, MO, and TD approaches are myopic in the sense that they focus on a particular aggregation level to produce forecasts, thus ignoring some useful information [14] available at other levels. In the last 12 years, hierarchical forecasting approaches have significantly evolved to include combination (COM) approaches that directly tackle the challenge of coherence. COM approaches have the advantage of using the information from all hierarchical levels to produce forecasts. These forecasts are consequently combined, using weights that are obtained either statistically (see [1–3]) or empirically (see the cross-validation approach in [15]). Simpler combinations based on equal weights have also been shown to be useful under some settings [16]. The application of hierarchical combination approaches has one direct advantage: it renders forecasts across hierarchical levels coherent, a property that is desirable in aligning decision-making across the different functions of an organization. Apart from its direct benefits, more often than not, COM also results in superior forecasting performance compared to simpler HF approaches. [17] offer a direct connection of hierarchical forecast reconciliation to the wider literature of forecast combinations.

The hierarchical combination approaches that have been explored so far in the literature are linear in nature. The only existing non-linear approach in HF, proposed by [18], uses ML models under the MO approach to dynamically forecast the proportions of the child nodes from their parent. However, this approach exploits information only from the parent node, ignoring the rest of the nodes that could be useful for obtaining more accurate results.

In all four aforementioned approaches (BU, TD, MO, and COM), the base forecasts can be generated using any statistical or judgmental forecasting method. Indeed, the method of choice might differ depending on the aggregation level of focus and the data availability. Popular choices include univariate forecasting models, such as exponential smoothing (ETS) and autoregressive integrated moving average (ARIMA) models. However, the baseline models could also allow for exogenous information, which may be crucial in, for example, retail settings where promotions often occur. Moreover, [19] showed that combining the forecasts across methods in order to obtain more accurate base forecasts will also increase the performance of the final, reconciled hierarchical forecasts. The efficacy of the different HF approaches depends on the time series features, the level of forecasting, the forecasting horizon, the structure of the hierarchy, and the relationships of the series. We may consider these variables when choosing the most appropriate HF approach [13,18,20–22].

In this paper we offer a non-linear perspective to the problem of hierarchical reconciliation and forecast coherence. Motivated by the recent advances in the field of machine learning (ML) and its successful use in the area of forecasting [23–25], including large-scale forecasting competitions [26,27] and applications that involve highly-correlated or hierarchically structured data [18,28,29], we propose the use of ML techniques to derive the combination weights for the forecasts across the various aggregation levels of a hierarchy. We focus on two ML models that have been shown to perform well in time series forecasting and cross-learning contexts: Random forests (RF) and XGBoost (XGB). Such decision tree models allow the exploitation of non-linear relationships across a number of series. This is particularly useful in hierarchical structures, especially when exogenous variables are available on only some hierarchical levels, the series are not all characterized by the same patterns, or the relationships of the series change through time. These advantages were recently highlighted by the results of the latest M competition, M5 [27], where LightGBM, an efficient variant of gradient boosted trees, managed to outperform numerous other approaches when used to forecast the hierarchical unit sales of Walmart. Nevertheless, LightGBM was not used in a "pure" hierarchical forecasting fashion, i.e. for forecast reconciliation purposes, producing forecasts just for the series at the bottom aggregation level of the data set. The contributions of this paper are threefold:

- We propose a non-linear approach to the problem of hierarchical forecast reconciliation. This approach is more general compared to its linear counterparts and is expected to enhance the forecasting performance across all hierarchical levels, especially when the relationships of the individual series are complex or change significantly through time.
- The majority of the existing HF reconciliation approaches are, strictly speaking, designed to result in coherence under particular assumptions, with improvements in terms of forecasting performance being a welcome side effect. In contrast, our proposed approach structurally combines the objectives of post-sample empirical forecast accuracy and coherence in the training phase of the ML model. The only other approach in the literature that has this property is HF via cross-validation [15]. Other methods that optimize forecast accuracy under the constraint of linear coherence, like the one proposed by [3], do so using the one-step-ahead in-sample errors of the baseline forecasting methods, which may not be representative of post-sample accuracy.
- Unlike existing HF approaches, our proposed approach selectively combines the forecasts across the different nodes of the hierarchy in a direct and automated way, without requiring that all forecasts need to be used. That said, our approach is expected to allow for more flexible combinations, mixing forecasts from different series only when correlations are present, thus moving from prescribed reconciliation approaches to data-driven ones.

We benchmark the performance of the proposed ML HF approach against various existing hierarchical methods on two data sets coming from the tourism and retail industries, using ARIMA-like models to estimate the base forecasts. The benchmarks we consider include single-level approaches (BU and TD), simple combinations (the arithmetic mean of BU and TD), and state-of-the-art linear combination (COM) approaches that use forecasts from all hierarchical levels. Our results suggest that ML reconciliation approaches are superior to existing, linear ones, both in terms of accuracy and bias.

The remainder of the paper is organized as follows. Section 2 describes the most popular HF methods found in the literature, while Section 3 presents the proposed ML reconciliation approach. Section 4 presents the two data sets used for the empirical evaluation of the proposed method, describes the experimental set-up, and discusses our results and findings. Section 5 concludes the paper.

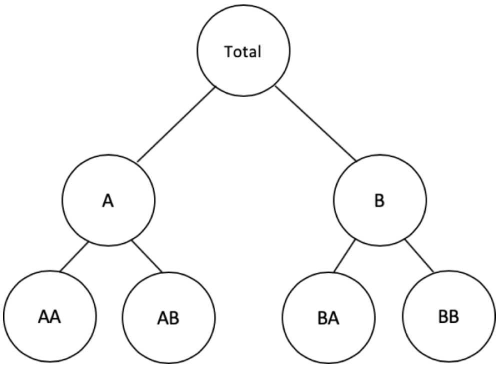


Fig. 1. A three level hierarchical structure.

2. Linear hierarchical forecasting approaches

In this section, we discuss the TD, BU, and COM methods as three well-established HF approaches. The following indices, notations, and parameters are used throughout this paper:

m = total number of series in the hierarchy;
m_i = total number of the series for level i;
k = total number of the levels in hierarchy;
n = number of the observations in each series;
$Y_{x,t}$ = the tth observation of series Y_x;
$\hat{Y}_{x,n}(h)$ = h-step-ahead independent base forecast of series Y_x based on n observations;
$\boldsymbol{Y}_{i,t}$ = the vector of all observations at level i;
$\hat{\boldsymbol{Y}}_{i,t}(h)$ = h-step-ahead forecast at level i;
$\boldsymbol{Y}_t$ = a column vector including all observations;
$\hat{\boldsymbol{Y}}_n(h)$ = h-step-ahead independent base forecast of all series based on n observations;
$\tilde{\boldsymbol{Y}}_n(h)$ = the final reconciled forecasts of all series.

The hierarchical time series can be expressed as $\boldsymbol{Y}_t = \boldsymbol{S}\boldsymbol{Y}_{k,t}$, where $\boldsymbol{S}$ is a summing matrix of order $m \times m_k$ that aggregates the bottom level series. Consider the hierarchy shown in Fig. 1 that shows a three level hierarchy.

The hierarchy shown in Fig. 1 can be expressed as:

$$\begin{bmatrix} Y_t \\ Y_{A,t} \\ Y_{B,t} \\ Y_{AA,t} \\ Y_{AB,t} \\ Y_{BA,t} \\ Y_{BB,t} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ & & \boldsymbol{I}_4 & \end{bmatrix} \times \begin{bmatrix} Y_{AA,t} \\ Y_{AB,t} \\ Y_{BA,t} \\ Y_{BB,t} \end{bmatrix}$$

The various HF approaches can then be expressed with a unified structure $\tilde{\boldsymbol{Y}}_n(h) = \boldsymbol{S}\boldsymbol{G}\hat{\boldsymbol{Y}}_n(h)$, where $\boldsymbol{G}$ is a matrix of order $m \times m_k$ which elements depend on the type of the HF method used [30].

2.1. Bottom-up

The BU approach considers just the base forecasts produced on the bottom level of the hierarchy and sums them appropriately to obtain forecasts at higher levels. In this approach, $\boldsymbol{G} = [\boldsymbol{0}_{m_k \times (m-m_k)} | \boldsymbol{I}_{m_k}]'$, where $\boldsymbol{0}_{i \times j}$ is an $i \times j$ null matrix. Thus, $\boldsymbol{G}$ extracts the bottom level forecasts and combines them with the summing matrix $\boldsymbol{S}$ to generate the final forecasts of the hierarchy.

2.2. Top-down

In the TD approach, base forecasts are produced just at the top level of the hierarchy and are then disaggregated to the lower levels with an appropriate factor. Gross and Sohl [21] investigated 21 different disaggregation methods for the TD approach. They concluded that Eqs. (1) and (2) indicate two disaggregation methods that give reasonable forecasts at the bottom level.

$$p_j = \frac{1}{n}\sum_{t=1}^{n}\frac{Y_{j,t}}{Y_t} \qquad j = 1, \ldots, m_k \tag{1}$$

$$p_j = \frac{\sum_{t=1}^{n} Y_{j,t}}{\sum_{t=1}^{n} Y_t} \qquad j = 1, \ldots, m_k \tag{2}$$

In Eq. (1), each proportion p_j reflects the average of the historical proportions of the bottom level series $Y_{j,t}$, while in Eq. (2), each proportion p_j reflects the average of the historical value of the bottom level series $Y_{j,t}$ relative to the average value of the total aggregate Y_t. These proportions can be used to form the vector $\boldsymbol{g} = [p_1, p_2, p_3, \ldots, p_{m_k}]$ so that $\boldsymbol{G} = [\boldsymbol{g} \mid \boldsymbol{0}_{m_k \times (m-1)}]'$. In this regard, $\boldsymbol{G}$ disaggregates the forecast at the top level to the lower levels.

[1] proposed the TD forecasted proportions (TDFP) approach that disaggregates the top level forecasts based on the forecasted proportions of lower level series rather than the historical proportions. According to this method,

$$p_j = \prod_{i=0}^{k-1}\frac{\hat{Y}_{j,n}^{(i)}(h)}{\sum(\hat{Y}_{j,n}^{(i+1)}(h))},$$

for $j = 1, \ldots, m_k$, where $\hat{Y}_{j,n}^{(i)}(h)$ is the h-step ahead forecast of the series that corresponds to the node which is i levels above j, and $\sum \hat{Y}_{i,n}(h)$ is the sum of the h-step ahead forecasts below node i that corresponds directly to the node i. These will form the vector $\boldsymbol{g} = [p_1, p_2, p_3, \ldots, p_{m_k}]$ so that $\boldsymbol{G} = [\boldsymbol{g} \mid \boldsymbol{0}_{m_k \times (m-1)}]'$. Similarly to the aforementioned TD methods, the TDFP approach will generate biased forecasts even if the base forecasts are unbiased [1]. [17] relaxed the assumption that a TD approach must adhere to a $\boldsymbol{G}$ of the form $[\boldsymbol{g} \mid \boldsymbol{0}_{m_k \times (m-1)}]'$, and offered a way to break down the top-level forecasts using the forecasts from the lower aggregation levels, taking into account the variance/covariance of the aggregated and disaggregated series and allowing them to retain the unbiasedness of the base forecasts, while still using the same information (in terms of forecasts) as the TDFP approach.

We use the `td` function in *hts* package to implement the TD method [31] that utilizes the proportions of Eq. (1).

2.3. Linear combination

The COM method uses a completely different approach for HF. This approach was developed over a series of papers by [2,32] and [3]. Let the h-step reconciled forecasts be given by

$$\tilde{\boldsymbol{Y}}_n(h) = \boldsymbol{S}\boldsymbol{G}\hat{\boldsymbol{Y}}_n(h).$$

They showed that the covariance matrix of the h-step-ahead reconciled forecast errors is given by

$$\boldsymbol{V}_h = \mathrm{Var}[\boldsymbol{y}_{n+h} - \tilde{\boldsymbol{Y}}_n(h)] = \boldsymbol{S}\boldsymbol{G}\boldsymbol{W}_h\boldsymbol{G}'\boldsymbol{S}',$$

where $\boldsymbol{W}_h$ is the variance–covariance matrix of the h-step ahead base forecast errors. Moreover, they demonstrate that if the base forecasts are unbiased, these reconciled forecasts will also be unbiased if and only if $\boldsymbol{S}\boldsymbol{G}\boldsymbol{S} = \boldsymbol{S}$. Finally, they showed that the $\boldsymbol{G}$ matrix that minimizes the trace of $\boldsymbol{V}_h$ such that $\boldsymbol{S}\boldsymbol{G}\boldsymbol{S} = \boldsymbol{S}$ is given by

$$\boldsymbol{G} = (\boldsymbol{S}'\boldsymbol{W}_h^{\dagger}\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{W}_h^{\dagger},$$

where $\boldsymbol{W}_h^{\dagger}$ is the generalized inverse of $\boldsymbol{W}_h$. Hence, the optimal unbiased forecasts from a linear reconciliation are given by

$$\tilde{\boldsymbol{Y}}_n(h) = \boldsymbol{S}(\boldsymbol{S}'\boldsymbol{W}_h^{\dagger}\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{W}_h^{\dagger}\hat{\boldsymbol{Y}}_n(h).$$

This is known as MinT (minimum trace) reconciliation . Note that it can be considered a generalized least squares estimator for a corresponding regression problem.

The challenge is the estimation of $\boldsymbol{W}_h$, especially for very large hierarchies, and different approximate estimates have been proposed.

1. Set $\boldsymbol{W}_h = k_h\boldsymbol{I}$. This is known as the OLS estimator [2]. This ignores the scale of each series and the relationships between the series.
2. Set $\boldsymbol{W}_h = k_h\text{diag}(\hat{\boldsymbol{W}}_1)$ where $\hat{\boldsymbol{W}}_1 = \frac{1}{T}\sum_{t=1}^{T}\hat{\boldsymbol{e}}_T(1)\hat{\boldsymbol{e}}_T'(1)$ is the sample covariance matrix of the one-step-ahead base forecast errors given by $\hat{\boldsymbol{e}}_T(1) = \boldsymbol{y}_{T+1} - \hat{\boldsymbol{y}}_T(1)$. This is known as the WLS estimator [32]. It ignores the relationships between the series, but takes account of the scale of each series.
3. Set $\boldsymbol{W}_h = k_h\text{diag}(\boldsymbol{S}\boldsymbol{1})$ where $\boldsymbol{1}$ is a unit n vector. This method ignores the relationships between the series and assumes that the bottom level series have errors with equal variances [4]. Because this method only depends on the structure of the hierarchy, it is known as structural scaling. It is particularly useful when residuals are not available.
4. Set $\boldsymbol{W}_h = k_h\left(\lambda_D\hat{\boldsymbol{W}}_{1,D} + (1-\lambda_D)\hat{\boldsymbol{W}}_1\right)$. This is a shrinkage estimator with diagonal target $\hat{\boldsymbol{W}}_{1,D} = \text{diag}(\hat{\boldsymbol{W}}_1)$, and shrinkage parameter
$$\lambda_D = \frac{\sum_{i\neq j}\text{Var}(\hat{r}_{ij})}{\sum_{i\neq j}\hat{r}_{ij}^2},$$
where $\hat{r}_{ij}$ is the $(i,j)^{th}$ element of the one-step-ahead in-sample correlation matrix [33]. The main advantage of this method is that it considers the relationships between the series.

In this paper, we consider the latter two methods: structural scaling (COM-SS) and shrinkage (COM-SHR). We use the MinT function in *hts* package in R to implement the COM-SHR and COM-SS methods [31].

3. ML hierarchical forecasting approach

In this section we present an ML reconciliation approach that exploits the potential of decision tree-based models. It is designed to deal with the limitations of the existing HF methods, highlighted in Section 1, and allow for the base forecasts produced for the complete hierarchy to be effectively combined in a non-linear fashion to yield coherent forecasts. We consider the random forest (RF) and the XGBoost (XGB) models as they are intuitively easy to understand and have shown promising results in time series forecasting, especially in applications where information is extracted from large time series data sets in order for the relationships of the series to be learned and the overall forecasting accuracy to be enhanced [34].

3.1. Proposed approach

The proposed ML reconciliation method uses time series cross-validation [30] to measure the out-of-sample forecast accuracy, which is then used in an optimization procedure to tune the ML model.

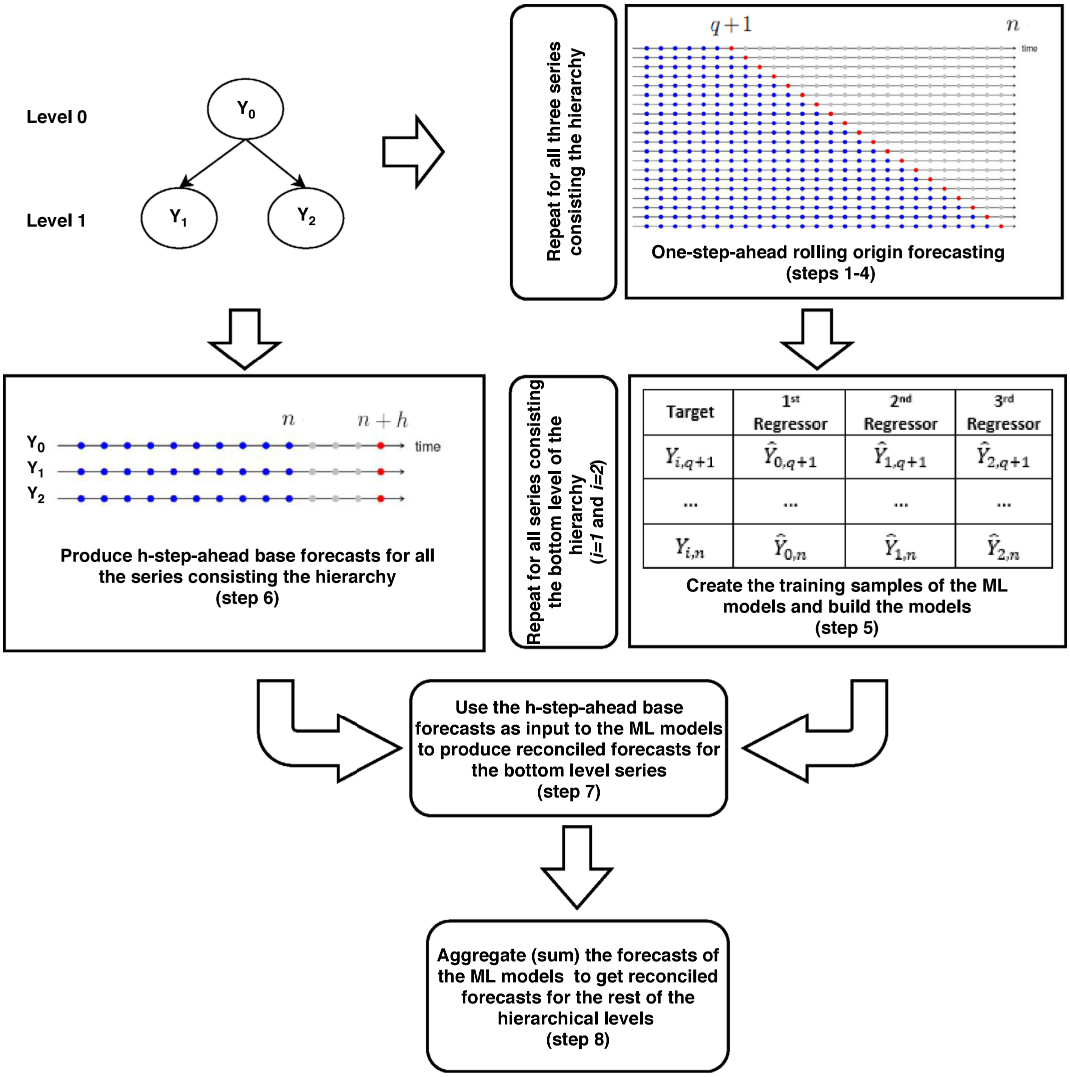
Assume a time series hierarchy that consists of k levels and m series, with each series of length n. We summarize our approach as follows.

1. The series are split into a series of training sets and test sets, with each training set comprising the first $p < n$ observations (for $p = q, q+1, \ldots, n-1$) and the corresponding test set comprising only the observations at time $p+1$.
2. A forecasting model is fitted to each series in each training set and one-step-ahead forecasts are produced for each test set.
3. A separate ML model (either a RF or XGB) is built for predicting each of the m_k bottom series of the hierarchy. The training set of each model consists of $n-p$ observations and $m+1$ variables. The first m variables (used as predictors or inputs) are the one-step-ahead forecasts produced during the rolling origin process for the m series of the hierarchy, and the last variable (the response or target) is the actual value of the bottom-level series at the corresponding times. The loss function of the models is the sum of squared errors, and the hyper-parameters of the ML models are determined either arbitrarily by the user or through an optimization procedure.
4. The complete sample of the series (all n observations) is used to produce h-step-ahead base forecasts for the m series of the hierarchy, where h is the forecasting horizon of interest.
5. The m_k models that were built in Step 3 are used to provide forecasts for the series of the bottom level of the hierarchy, using the base forecasts produced in Step 4 as input. This process is repeated h times, each time for a different forecasting horizon.
6. The forecasts produced by the ML models in step 5 are aggregated (summed) so that reconciled forecasts are produced for the rest of the hierarchical levels.

The proposed approach is demonstrated in Fig. 2 for the case of a simple, two-level hierarchy with one parent and two child nodes.

As seen, the proposed ML HF approach provides coherent forecasts by exploiting the information available at all hierarchical levels, following the approach used by the COM methods. The main difference between the COM methods and the proposed approach is that the base forecasts are not all necessarily used for deriving the reconciled ones, being selectively handled by the ML models built for this purpose. Moreover, even if all base forecasts are to be used by the ML models, the combination of the base forecasts will be done in a non-linear fashion with the weights not being directly related to the structure of the hierarchy or the residuals reported for the individual series/levels. Most importantly, since the ML models are trained with the explicit objective of minimizing the forecasting error for each series of the bottom level of the hierarchy, the reconciliation performed may lead to more accurate forecasts when compared to standard HF methods. Finally, note that each bottom-level series is predicted by a separate ML model, meaning that the reconciliation performed is highly specialized and, therefore, able to adapt to different patterns in each series.

Observe that the proposed approach is easy to generalize and is model independent. For example, a neural network (NN) or a support vector machine (SVM) could be used to replace RF and XGB. Similarly, any model of choice could be used for producing the base forecasts being reconciled. Moreover, the one-step-ahead forecasts produced for constructing the training sets of the ML models, could be easily expanded to h-step-ahead ones to better simulate the forecasting task under examination. Our proposal of using one-step-ahead forecasts is mainly based on the fact that by increasing the forecasting horizon of the base models, the observations of the training set, i.e. $n-p$, are


Level 0
Level 1
Y_0
Y_1
Y_2
Repeat for all three series consisting the hierarchy
$q+1$
n
time
One-step-ahead rolling origin forecasting (steps 1-4)
n
$n+h$
time
Produce h-step-ahead base forecasts for all the series consisting the hierarchy (step 6)
Repeat for all series consisting the bottom level of the hierarchy (i=1 and i=2)
Create the training samples of the ML models and build the models (step 5)
Use the h-step-ahead base forecasts as input to the ML models to produce reconciled forecasts for the bottom level series (step 7)
Aggregate (sum) the forecasts of the ML models to get reconciled forecasts for the rest of the hierarchical levels (step 8)

Target	1st Regressor	2nd Regressor	3rd Regressor
$Y_{i,q+1}$	$\hat{Y}_{0,q+1}$	$\hat{Y}_{1,q+1}$	$\hat{Y}_{2,q+1}$
...	...	...	...
$Y_{i,n}$	$\hat{Y}_{0,n}$	$\hat{Y}_{1,n}$	$\hat{Y}_{2,n}$

Fig. 2. Demonstration of the proposed machine learning hierarchical forecasting approach for the case of a two-level hierarchy consisting of one parent and two child nodes.

accordingly decreased. Thus, when dealing with low frequency data (e.g. monthly or quarterly) or relatively short time series, such an approach could significantly reduce the potential of the developed ML models.

The following subsections present the ML models used in this study for reconciliation. This includes information about the way the models were trained, optimized, and implemented.

3.2. XGBoost

XGB is an ensembling method based on decision trees that uses a gradient boosting approach to generate unbiased and robust forecasts [35]. This method has been applied to various forecasting and classification problems with promising results [36–38].

XGB uses a number of hyper-parameters that play a critical role in generating the final forecasts. There are various techniques for optimizing these hyper-parameters, including grid search, sequential model based algorithm configuration, and Bayesian optimization. Since grid search is computationally expensive, we tuned the hyper-parameters using a Bayesian optimization approach with 10-fold cross-validation [39]. The Bayesian approach starts with a priori values for the hyper-parameters and then iteratively updates to identify the best values for the investigated problem. We considered intervals with different lower and upper bounds for each hyper-parameter. We set the prior values of the learning rate, `eta`, between (0.01, 0.05), `sub sample` size prior values between (0.3, 1), `colsample-bytree` prior values between (0.3, 1), `min-child weight` between (0, 10), `max-depth` between (2, 10), and `gamma` between (0, 5). The values for the maximum number of boosting iterations rolled over the range of 50 and 200. We used a linear regression model as the objective function and chose the best results by minimizing the root mean squared error (RMSE). We tuned the hyper-parameters using the *rBAyesianOptimization* package for R [40].

3.3. Random forest

RF is an ensembling method that combines a large number of decision trees and takes an average of the trees to generate the final forecast [41]. Each tree of the RF is based on a random draw from the training data set. The trees are built using the bootstrapping method and splitting criteria in nodes. We consider the weighted variance as the splitting criteria which minimizes the sum of squared errors. This method has been successfully applied to numerous forecasting problems such as energy [42,43] and sales [44] forecasting.

Table 1
Number of time series per level of hierarchy in the "Tourism" data set.

Hierarchical level	Number of series
Level 0	1
Level 1	7
Level 2	27
Level 3	76
Total	111

Table 2
Number of time series per level of hierarchy in the "Sales" data set. The hierarchical structure is the same for all 55 products of the data set.

Hierarchical level	Number of series
Level 0	1
Level 1	2
Level 2	12
Total	15

RF is fast to run and it only has a few hyper-parameters: the number of trees (ntree), node size (nodesize), and number of variables sampled at each split (mtry). Of these, the number of constructed trees is the most important feature to be tuned. The problem of optimally selecting the number of trees has been intensively discussed in the literature [41,45–47]. The main problem is that although creating more trees is computationally more demanding, it does not guarantee a better forecast. This is because each tree is trained individually and so by adding more trees, over-fitting may occur [41]. On the other hand, since the individual trees constructed do not have the learning capacity of XGB, RF is typically more robust to outliers and over-fitting, especially for limited samples of data [48]. The hyper-parameter mtry denotes the number of variables sampled at each split and controls the randomness of the model. The nodesize hyper-parameter determines the minimal number of observations in a terminal node to be split.

We used grid search, an automated method that explores a set of different hyper-parameters values and computes the error on the validation set, with 10-fold cross-validation to find the optimal number of trees by minimizing the RMSE. We ran ntree on a sequence of intervals of width 5 ranging from 50 to 150 and fitted the best model using the *randomForest* package for R [49]. We tuned the other two hyper-parameters, mtry and nodesize, using the *mlr* package in R [50]. The lower and upper bound values for mtry were set between 2 and 6, respectively. The lower and upper bound values for nodesize were set on 10 and 50, respectively.

4. Empirical results

4.1. Data

In order to empirically evaluate the performance of the proposed ML HF methods, we consider two different data sets, to be named the "Tourism" and the "Sales" data set.

The "Tourism" data set involves a four-level hierarchy with the domestic visitor nights of Australia, measured in millions, across 76 regions (level 3). The regions can be grouped into 27 zones (level 2), which can be further aggregated into 7 states and territories (level 1), as well as into the total domestic visitor nights (level 0). Thus, based on these geographic divisions, the "Tourism" data set comprises 111 time series. The series have a duration of 240 months (20 years) and span from January 1998 to December 2017.

Table 1 summarizes the number of series present per hierarchical level, while Fig. 3 visualizes some indicative series from each level. Observe that the trend and seasonal patterns differ among the series, especially for different states and territories. Moreover, the trend of some series (e.g. A, C, and E) changes through the years, in contrast to others (e.g. G) that remain quite constant. This indicates that considering a dynamic, non-linear HF method instead of a linear one, could prove beneficial for predicting these series. For more information about the data set, please see [5].

The "Sales" data set involves 55 three-level hierarchies that present the sales of the cereal and breakfast products sold by a company in various locations of Australia, along with the corresponding prices. Each hierarchy refers to a different product, with the first level (level 0) representing the total sales of the manufacturer, the second level (level 1) the way these sales are disaggregated into two retailers, and the third level (level 2) the sales reported for each of the six distribution centers (DCs) used by each retailer. Thus, the "Sales" data set includes 55 hierarchies, each consisting of 15 time series. The series have a duration of 120 weeks and span from September 2016 to December 2018.

Table 2 summarizes the number of series present per hierarchical level, while Fig. 4 visualizes the series of each level for one indicative product of the data set. Note that, although the retailers display different demand patterns, DCs have a similar pattern to their retailers in terms of promotions. Moreover, different entities of the hierarchy may experience different levels of uplifts in sales. Thus, an ML HF method, which effectively captures sales variations, could be more effective for reconciling the base forecasts of these series than a traditional, linear one.

Due to the notable variations present in the "Sales" data set, the optimization of the hyper-parameters was performed for each hierarchical time series and set of child–parents separately, while for the "Tourism" data set we optimized the hyper-parameters for the hierarchy as a whole in order to accelerate the whole process.

4.2. Base forecasts and benchmarks

Given that each data set displays its own particular characteristics, we consider different forecasting models for producing the base forecasts in each case.

More specifically, for the "Tourism" data set we consider ARIMA models, as implemented in the *forecast* package for R [51]. We search for the optimal ARIMA model per series in the space of both non-seasonal and seasonal ARIMA (SARIMA) models. In more detail, the search for the optimal ARIMA form takes place in the in-sample data in a step-wise fashion. First, unit root tests are used to identify the appropriate degree of non-seasonal and seasonal differencing to render the data stationary. Consequently, four initial model forms are fitted and compared against each other by means of the corrected Akaike Information Criterion (AICc), which accounts for model performance and complexity, thus avoiding overfitting. The best of the four models (smallest AICc) is considered as the "temporary best" model. Then, the search expands so that the autoregressive (AR) and moving-average (MA) orders components of the temporary best model (p and q for the non-seasonal components; P and Q for the seasonal components) change by one. If a better model is identified, the search continues; otherwise, the search stops. The algorithm is described in detail in [52].

On the other hand, for the "Sales" data set we use regression models with ARIMA errors (RegARIMA) using price as a regressor variable. By using RegARIMA model, the effect of the promotions, which typically increase sales and drive major changes in the underlying demand behavior, is effectively taken into account [53,

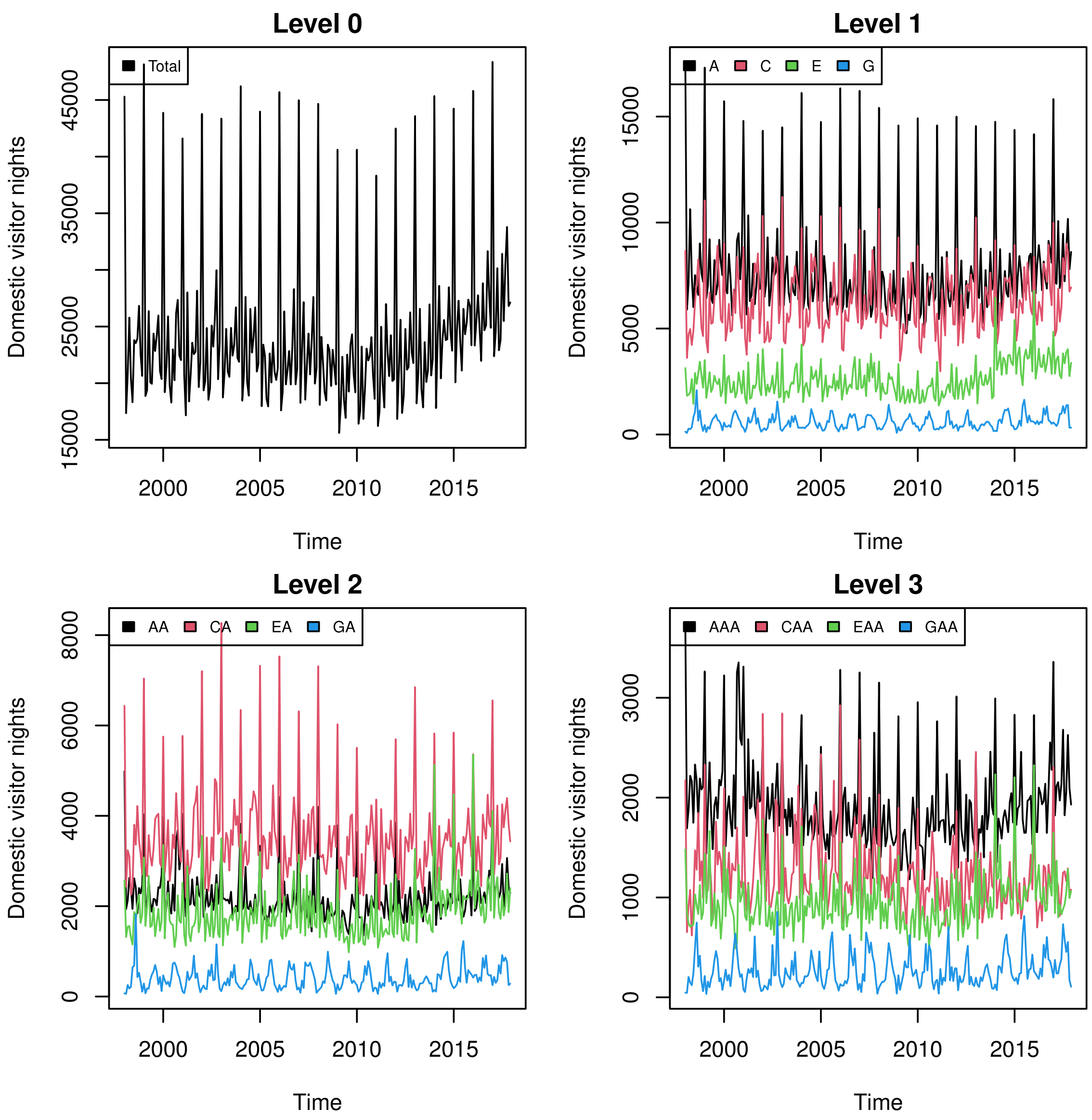


Fig. 3. Domestic visitor nights, measured in millions, for selected geographic divisions of Australia. A sample of indicative series is used for representing each level of the "Tourism" data set, i.e. total demand (level 0, top-left), 4 out of 7 states and territories (level 1, top-right), 4 out of 27 zones (level 2, bottom-left), and 4 out of 76 regions (level 3, bottom-right). According to the notation used, CAA denotes the first region of CA, which is the first zone of C, being the third state of Australia in the data set, and so on.

54]. RegARIMA, as shown in Eq. (3), is a regression model that utilizes the exogenous regressor to fit a model to time series data and an ARIMA(p, d, q) model to the error terms. The parameter d determines the degree of differencing, if required. The best model is chosen based on AICc.

$$y_t = \beta_0 + \beta_1 x_t + \eta_t, \tag{3}$$

where y_t and x_t are the forecast regressor value at time t, η_t is an ARIMA(p, d, q) error, and β_0 and β_1 the coefficients of the regression model.

We use the ARIMA models for producing 12-step-ahead forecasts for the monthly series of the "Tourism" data set and the RegARIMA models for producing 8-step-ahead forecasts for the weekly series of the "Sales" data set. We should note that ETS [55] and Theta [56] were also tested for producing the base forecasts for the case of the "Tourism" data set, providing similar insights to the ones reported for ARIMA. Thus, for reasons of brevity, and in order for the baseline models used in both cases to be similar in nature, we proceed by reporting the results only for the ARIMA models.

We benchmark our proposed reconciliation approaches based on machine-learning, ML-RF and ML-XGB, against five benchmarks. The two main benchmarks used in this study are the BU and TD approaches that were described in Sections 2.1 and 2.2 . Effectively, the base forecasts of the respective levels (bottom or top) are assumed to be the final forecasts for that level, with forecasts at every other level being calculated through aggregation (bottom-up) or disaggregation (top-down) using the disaggregation method depicted in Eq. (1). BU and TD approaches can be seen as an extreme case of forecast reconciliation in the sense that the forecasts for other levels are computed so that the direct base forecasts for the target level remain unchanged. Given that BU and TD do not involve any forecast combination per se, they are considered the basic benchmarks in the hierarchical forecasting literature, against which all other reconciliation methods should be compared.

Another benchmark that has been shown to perform well in the literature is an equal-weighted combination across the various hierarchical levels [17]. This is reasonable since combining forecasts has long been considered an effective practice

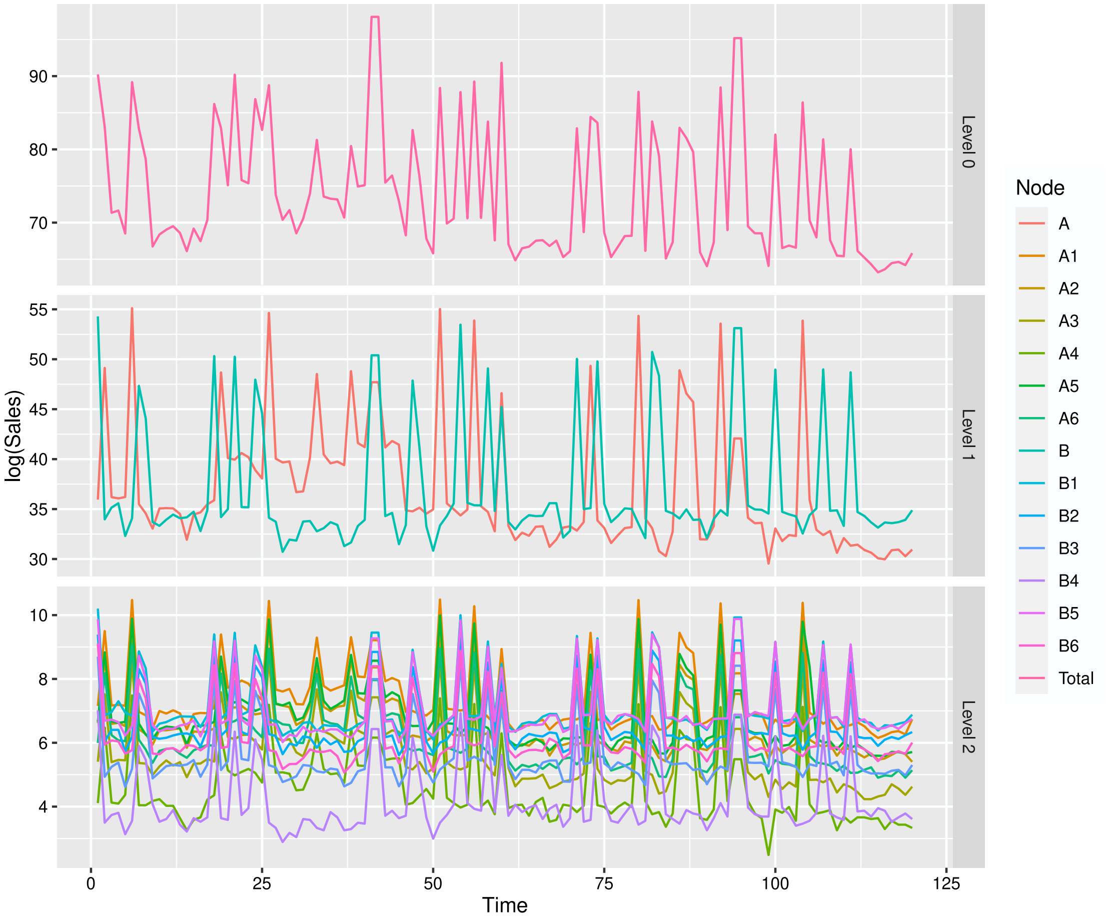


Fig. 4. Sales of an indicative cereal/breakfast product sold in Australia. The sales are presented in total, as well as per retailer (A, B) and twelve distribution center (A1, . . . , A6, B1, . . . , B6). This is an indicative example of the hierarchies involved in the "Sales" data set.

for improving forecasting accuracy. We implement the simple average (arithmetic mean) of the forecasts derived by the BU and TD approaches as this particular combination reporting promising results for the case of freight earnings [16] and retail sales [27] forecasting. Hereafter, we call this approach "TD-BU". Finally, we benchmark our proposed approaches against state-of-the-art linear reconciliation approaches, namely COM-SS, and COM-SHR, as described in 2.3.

4.3. Evaluation

We evaluate the forecasting performance of the HF methods both in terms of accuracy (absolute deviation of the forecasts around the true values) and bias (consistent distance observed between the forecasts and the true values), using the mean absolute scaled error (MASE) [57], as well as the root mean squared scaled error (RMSSE) and absolute mean scaled error (AMSE). The measures can be calculated as

$$\text{MASE} = \frac{n-s}{h}\frac{\sum_{t=n+1}^{n+h}|y_t - f_t|}{\sum_{t=s+1}^{n}|y_t - y_{t-s}|},$$

$$\text{RMSSE} = \sqrt{\frac{n-s}{h}\frac{\sum_{t=n+1}^{n+h}(y_t - f_t)^2}{\sum_{t=s+1}^{n}(y_t - y_{t-s})^2}},$$

$$\text{AMSE} = \frac{n-s}{h}\frac{|\sum_{t=n+1}^{n+h}(y_t - f_t)|}{\sum_{t=s+1}^{n}|y_t - y_{t-s}|},$$

where y_t and f_t are the observation and the forecast for period t, n is the sample size (observations used for training the forecasting model), s is the length of the seasonal period, and h is the forecasting horizon. In all cases, lower values indicate better forecasts.

Note that all measures are scale-independent, meaning that averaging across series is possible. Moreover, given that the median minimizes the sum of the absolute errors [58], while the mean minimizes the sum of the squares of these errors [59], MASE and RMSSE are appropriate for evaluating the accuracy of the examined HF method in approximating the median and the mean of the future values, respectively. Accordingly, AMSE is appropriate for measuring the bias of the reconciled forecasts.

In order for our results to represent reality as close as possible and approximate the actual performance of the examined HF methods in a long-term run, we consider the rolling-origin evaluation approach [60]. According to this approach, the first N observations of each series are used for producing h-step-ahead forecasts, with the following $N+1\dots N+h$ observations used for evaluating them. Then, the forecasting origin is increased by one and new forecasts are produced from the updated origin, this time using $N+1$ observations for training the forecasting model and the following $N+2\dots N+h+1$ ones for testing. This process is repeated K times, until there are no observations left for evaluating the forecasts, i.e. while $N+h+K-1 \leq n$.

Given that the length and the frequency of the series of the two data sets differ, we consider a different, yet indicative implementation of the rolling-origin evaluation approach per case. Specifically, in the "Tourism" data set we begin to produce forecasts at the end of the 14th year of the data set ($N_1 = 168$ months) and use the remaining 6 years for testing, thus performing a total of $K_1 = 61$ evaluations. Accordingly, in the "Sales" data set, we start producing forecasts at the end of the 1st year of the data set ($N_2 = 52$ weeks) and use the remaining 60 weeks of each sales time series for testing, thus performing a total of $K_2 = 61 \times 55 = 3355$ evaluations. The overall performance of the HF methods in each data set is computed by averaging the scores reported across all K_1 and K_2 evaluation periods.

Note that in order for the ML HF methods to be effectively trained to derive accurate reconciled forecasts when provided with a set of base forecasts, we require a data set that includes an adequate sample of past, actual time series values (target variables) and the corresponding base forecasts produced for these periods by the forecasting model (regressor variables). In order to obtain such a data set, we produce multiple one-step-ahead forecasts in a rolling-origin fashion, starting from an initial point, p, and finishing at the forecast origin considered in each repetition of the rolling-origin evaluation approach, as described in Section 3 (steps 1–4). We set p equal to $p_1 = 60$ and $p_2 = 26$ for the "Tourism" and "Sales" data set, respectively, so that a reasonable amount of full seasonal periods is available for producing the base forecasts to be used for training the ML HF methods. In this regard, in the first evaluation performed for the "Tourism" data set, a sample of $N_1 - p_1 = 108$ records will be available for training the ML HF methods, with the records becoming $N_1 + 61 - 1 - p_1 = 168$ in the last evaluation. Accordingly, a sample of $N_2 - p_2 = 26$ records will be available in the first evaluation of the "Sales" data set for each of the 55 hierarchies, with their length reaching $N_2 + 60 - 1 - p_2 = 85$ records in the last evaluation.

4.4. Results

Tables 3 and 4 summarize the performance of the HF methods considered in this study in terms of accuracy (MASE and RMSSE) and bias (AMSE) for the "Tourism" and the "Sales" data set, respectively. The first column of each table indicates the HF methods considered, while the rest of the columns present the performance of the method for each aggregation level separately, as well as across all levels (average of measure values reported for Level 0 to Level k). All levels are weighted equally since we do not focus on a particular decision-making problem, aimed at a specific hierarchical level.

Before proceeding with the evaluation of the results, we highlight that two of the benchmarks employed, namely COM-SS and COM-SHR, are considered state-of-the-art in the field of hierarchical time series forecasting as they have been proven to significantly improve the base forecasts provided to them as input. Moreover, although much more simplistic in nature, the BU and TD methods are highly competitive and, in some applications, difficult benchmarks to beat. Thus, further improving the performance of HF based on ML approaches becomes a promising, yet challenging task.

Table 3
Forecasting performance reported for various HF methods in the "Tourism" data set after applying the rolling-origin evaluation approach (average of 61 evaluations of 12-month-ahead forecasts). The performance, as measured by MASE, RMSSE, and AMSE, is estimated both per hierarchical level and across all levels.

Method	Level 0	Level 1	Level 2	Level 3	Average
MASE					
BU	1.184	1.050	0.923	0.857	1.003
TD	1.048	1.297	1.124	0.978	1.112
TD-BU	1.076	1.085	0.935	0.857	0.988
COM-SS	1.094	0.968	0.887	0.843	0.948
COM-SHR	1.047	**0.956**	0.872	0.824	0.925
ML-RF	1.045	0.964	0.859	0.812	0.920
ML-XGB	**1.043**	0.965	**0.859**	**0.812**	**0.920**
RMSSE					
BU	1.439	1.314	1.186	1.124	1.266
TD	**1.238**	1.630	1.460	1.297	1.406
TD-BU	1.295	1.361	1.207	1.127	1.247
COM-SS	1.308	1.225	1.137	1.109	1.195
COM-SHR	1.265	1.214	1.120	1.086	1.171
ML-RF	1.261	**1.208**	1.104	1.066	1.159
ML-XGB	1.255	1.208	**1.101**	**1.064**	**1.157**
AMSE					
BU	1.066	0.639	0.443	0.350	0.624
TD	0.845	0.594	0.404	0.341	0.546
TD-BU	0.956	0.582	0.390	0.324	0.563
COM-SS	0.988	0.611	0.426	0.349	0.593
COM-SHR	0.935	0.599	0.417	0.337	0.572
ML-RF	0.780	**0.526**	0.366	0.319	0.498
ML-XGB	**0.779**	0.526	**0.365**	**0.317**	**0.497**

Table 4
Forecasting performance reported for various HF methods in the "Sales" data set after applying the rolling-origin evaluation approach (average of 330 evaluations of 8 week ahead forecasts). The performance, as measured by MASE, RMSSE, and AMSE, is estimated both per hierarchical level and across all levels.

Method	Level 0	Level 1	Level 2	Average
MASE				
BU	0.491	0.516	0.540	0.516
TD	0.522	0.785	0.971	0.759
TD-BU	0.490	0.572	0.822	0.690
COM-SS	0.497	0.529	0.629	0.552
COM-SHR	0.495	0.520	0.542	0.519
ML-RF	**0.433**	0.449	**0.465**	**0.449**
ML-XGB	0.447	**0.447**	0.473	0.455
RMSSE				
BU	0.653	0.710	0.741	0.701
TD	0.684	1.118	1.314	1.039
TD-BU	0.650	0.798	0.934	0.813
COM-SS	0.655	0.720	0.844	0.739
COM-SHR	0.654	0.713	0.742	0.703
ML-RF	**0.625**	**0.675**	**0.703**	**0.668**
ML-XGB	0.654	0.719	0.759	0.711
AMSE				
BU	0.300	0.323	0.330	0.318
TD	0.320	0.423	0.627	0.456
TD-BU	0.300	0.323	0.531	0.412
COM-SS	0.301	0.327	0.372	0.334
COM-SHR	0.305	0.327	0.331	0.321
ML-RF	**0.290**	0.312	0.308	0.303
ML-XGB	0.308	**0.301**	**0.299**	**0.303**

The results for the "Tourism" data set presented in Table 3 show that, on average, ML-XGB is the most accurate HF method in terms of MASE, doing slightly better than ML-RF. Specifically, ML-XGB is 17% and 8% more accurate on average when compared to the TD and the BU method, respectively, being also 3% and 0.6% more precise than the COM-SS and COM-SHR methods. The same stands in general for the individual hierarchical levels, with the exceptions of the TD method for which results are comparable to the ones of the ML methods at the top level, as well as the COM-SHR that displays the best performance at Level 1. This can be partially explained by reviewing the particularities of these two methods: TD builds on the base forecasts produced for the top level of the hierarchy, thus omitting any information provided from the rest of the series, while COM-SHR combines the forecasts from all the series of the hierarchy linearly. As a result, if the fully aggregated series is predictable enough, the TD method will provide accurate results at Level 0. Accordingly, if the information required for accurately predicting a level in the middle of the hierarchy, like Level 1, is not complicated and sufficiently provided by the neighboring levels (Levels 0 and 2), COM-SS and COM-SHR will result in improved forecasting accuracy. Note however that COM-SS is always outperformed by COM-SHR due to the latter incorporating information about the correlation structure of the series. Moreover, as expected, the simple combination of TD and BU leads on average to more accurate forecasts than the individual methods being combined by mixing their advantages when it comes to modeling series at higher and lower cross-sectional levels, respectively. Yet, both ML methods manage to outperform TD-BU at all aggregation levels, being on average 7% more accurate.

The results are similar in terms of RMSSE, with just two differences worth reporting. First, in this case, the performance of the TD method at the top level is not only comparable to the one of the ML methods, but actually better by about 2%. However, TD continues to produce significantly less accurate results for the rest of the hierarchical levels. Second, at Level 1, COM-SHR is no longer the best performing method, being outperformed by both ML approach to a similar extent. Thus, we conclude that ML approaches are generally better in approximating the mean of the future values of the series than their median, a phenomenon which can be possibly attributed to the way these methods learn: Both RF and XGB are optimized by minimizing the sum of squared errors produced. Thus, these models learn how to properly approximate the mean and not necessarily the median of the series being predicted.

This last argument may also explain the bias reported for each method, as measured by AMSE. Given that mean squared error can be decomposed into a bias and an accuracy term [61], both ML-RF and ML-XGB are indirectly trained so that they minimize the bias of the reconciled forecasts. In this regard, in contrast to MASE and RMSSE, the ML HF methods always provide significantly less biased forecasts than the benchmarks, especially for the higher levels of the hierarchy. In particular, ML-XGB, the best performing method in terms of AMSE, is on average 15% better than the benchmark methods, being also less biased by 8% for the bottom level, 18% for the top level, and 14% for the two levels in the middle. Observe also that the worst performing method in terms of AMSE is the BU, with the TD doing also much better than the TD-BU, COM-SS, and COM-SHR methods. This indicates that when the base forecasts produced at the bottom level of the hierarchy are biased, reconciliation methods should put more emphasis on the top level where forecasts are more likely to be robust and, therefore, less biased. On the other hand, the fact that ML methods, which exploit the base forecasts produced at all hierarchical levels in a similar fashion to COM-SS and COM-SHR, are still able to provide unbiased results, highlights the potential of dynamic, non-linear reconciliation approaches.

The results are even more encouraging for the case of the “Sales” data set. According to MASE, the ML-RF method is considered the most accurate approach on average, being also the best HF method for all levels apart from Level 1. However, even at Level 1, ML-RF is outperformed only to a small extent by ML-XGB, which is also an ML approach. Moreover, in this data set, the differences reported between the ML methods and the benchmarks are always significant, with the improvements being around 14% at the top level, 21% at the middle, and 26% at the bottom. In other words, not only the improvements reported for the “Sales” data set are greater than those of the “Tourism” data set, but can be also observed across all levels, becoming more significant for the lower levels of the hierarchy. This could be due to the major differences reported in the “Sales” data set between the retailers, meaning that combining the base forecasts from the complete hierarchy to produce forecasts for a particular series is inappropriate when the series do not share the same patterns, at least to some extent. On the contrary, the results highlight that when an ML HF method is utilized for this purpose, being able to selectively combine the base forecasts, the information from the complete hierarchy could still be relevant. This conclusion is also supported by observing that COM-SS and COM-SHR do similarly in terms of MASE to the relatively much simpler BU method. Similarly, although the TD-BU performs better than the TD method, it is outperformed by both proposed ML models at all levels, being also less accurate than BU.

The results of MASE are in a general agreement to those reported for the case of the RMSSE. Again, ML-RF, the best performing ML HF method, outperforms all the benchmarks, with the improvements reported being higher for the lower levels of the hierarchy (6%, 10%, and 20% on average for Levels 0, 1 and 2, respectively). However, ML-XGB manages to provide slightly less biased results than ML-RF at all levels apart from the top one. Again, the differences between the two ML approaches are small, with their performance being also much better than that of the benchmarks. For example, according to AMSE, ML-RF is on average 6% less biased than the benchmarks at the top level, 14% at the middle level, and 18% at the bottom level.

Fig. 5 provides further insight about the relative accuracy of the HF methods for 55 hierarchical sales data at different levels in terms of MASE, AMSE, and RMSSE. It demonstrates that both ML HF methods generate more accurate forecasts than their counterparts with ML-RF being the top performing method in terms of MASE. While ML-XGB has performed more consistently across different series at Levels 0 and 1, the ML-RF method has generated more consistent forecasts at Level 2. This notion also holds for RMSSE. This might be due to different features of time series, such as seasonality, entropy, and the trend of the base time series [18,53]. Finally, it is apparent that the RF and XGB methods performed quite similarly in terms of AMSE.

To further validate our findings, we also examine the significance of the differences reported between the various HF methods using the multiple comparisons with the best (MCB) test, as proposed by [62]. The MCB test ranks the performance of the examined methods across the series being forecast using a measure of choice and then compares their average ranks by considering a critical difference, determined through a confidence interval. In our case, the confidence was set to 95% and the significance of the results was evaluated for the MASE, RMSSE, and AMSE measures, separately.

The results of the MCB test for the “Tourism” and “Sales” data sets are presented in the graphs located on left and right side of Fig. 6, respectively. In these graphs, if the intervals of two methods do not overlap, this indicates a statistically different performance. Thus, methods that do not overlap with the gray zone of the figures are significantly worse than the best, and vice versa. As seen, our results suggest that, in both data sets, the ML HF methods display lower average ranks, i.e. provide better forecasts than the examined alternatives for more instances, with ML-XGB and ML-RF being also, in the vast majority of the cases, significantly better than the benchmarks considered. The only exceptions refer to the RMSSE measure in the “Sales” data set, where ML-XGB is not significantly better than COM-SHR and BU, and the MASE measure in the “Tourism” data set, where ML-XGB and ML-RF are not significantly better than COM-SHR. Moreover, we find that the differences between the RF and XGB approaches are, in most cases, insignificant, indicating that the merits of the proposed HF method are mainly driven by the non-linear optimization approach used for reconciling the base forecasts and not by the particular ML algorithm utilized for performing this task. Therefore, we conclude that ML HF methods can be effectively used to provide significantly more accurate, reconciled forecasts and improve the overall forecasting performance in hierarchical settings.

By summarizing the results of both data sets, the following conclusions can be drawn:

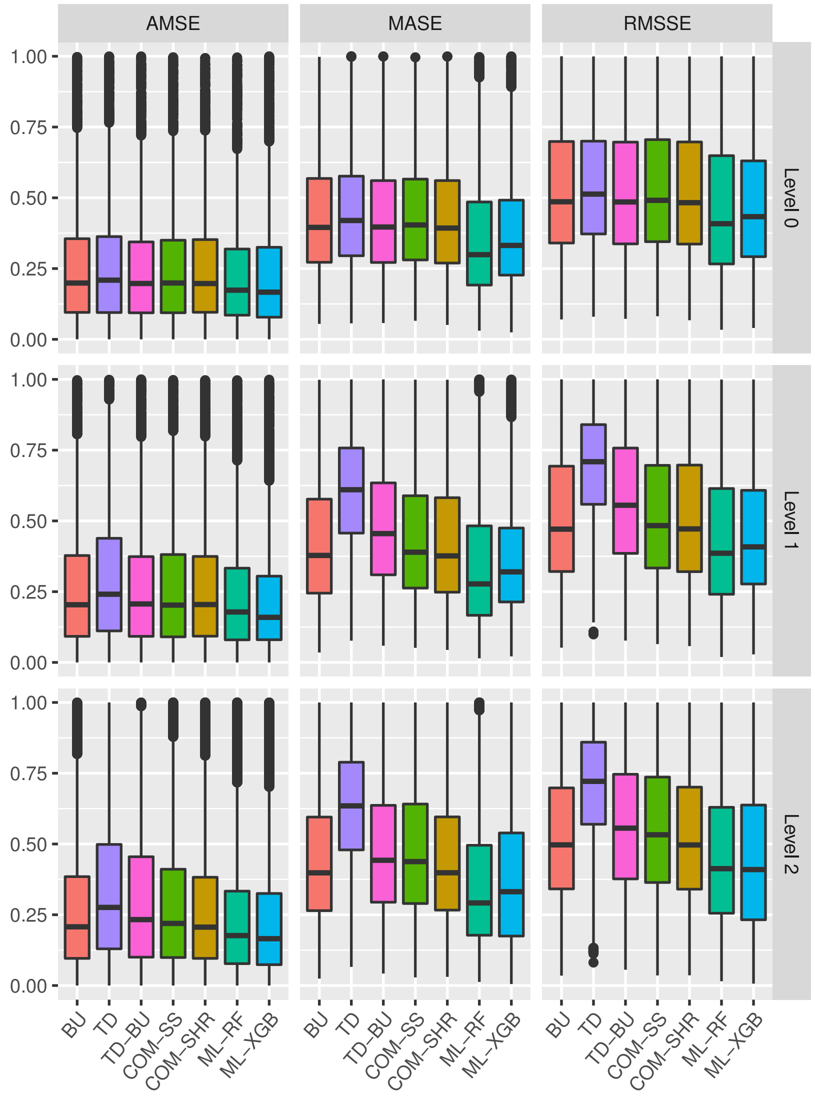


Fig. 5. The accuracy of the examined HF methods on the "Sales" data set at different aggregation levels.

- ML HF methods, combining the base forecasts for the complete hierarchy in a non-linear way, provide on average significantly better forecasts than existing methods, both in terms of accuracy and bias. Whether these results can be generalized to other data sets remains to be seen.
- The information of other series at different levels of the hierarchy (cross-sectional information) can be useful in forecasting the future values of a series regardless of the reconciliation methodology used.
- The expected improvements from using an ML HF method instead of the existing linear methods are higher when the series in the hierarchy are characterized by different patterns. The greater the differences between the series, at all levels, the higher the potential of using a selective, non-linear reconciliation approach.
- When a particular ML approach is considered for reconciling the base forecasts produced for a hierarchy, the model selected for determining the combination weights of these forecasts does not greatly affect the final results. Note however that this conclusion is drawn based on an experiment where two decision tree models are used, both utilizing the same approach for performing the reconciliation. Thus, further investigation is required to confirm that this is also the case (i) when different types of ML models (e.g. NNs and SVMs) are used for combining the base forecasts and (ii) different reconciliation approaches are utilized.

5. Conclusions

The challenge of hierarchical forecast reconciliation, to produce coherent forecasts across the various hierarchical levels, has so far been tackled with various linear approaches. Early solutions focused on producing forecasts at a single aggregation level with the forecasts of the other levels being derived by aggregation/disaggregation, thus essentially avoiding the incoherence issue. Current state-of-the-art solutions linearly combine the forecasts across all levels. In this study, we have proposed the use of non-linear combination approaches to achieve reconciliation using ML models.

Our results suggest that, on average, the proposed hierarchical reconciliation approaches based on ML perform well in practice, both in terms of forecast accuracy and bias. Not only can they outperform simple hierarchical approaches, such as BU, TD, and their simple average, but they also show improvements over robust state-of-the-art linear combination approaches. The good performance of HF ML is more evident on the "Sales" data set compared to the "Tourism" data set, possibly due to the importance of the bottom-level information where our algorithm

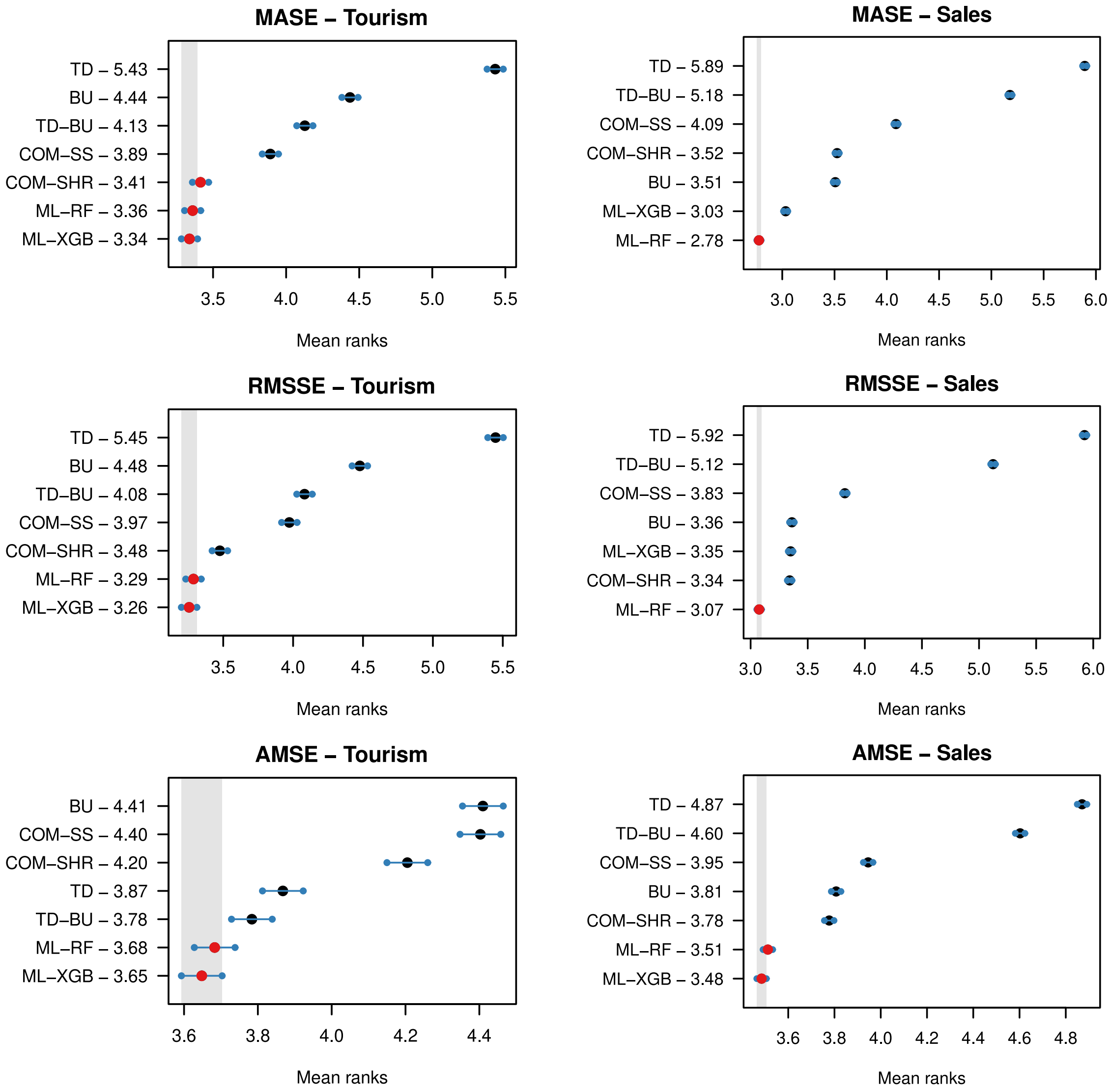


Fig. 6. MCB test conducted on the HF methods examined in this study using the time series of the "Tourism" and "Sales" data sets. MASE, RMSSE, and AMSE are used for computing the ranks and a 95% confidence level is considered. For the "Tourism" data set (graphs on the left), the ranks are computed considering all the series of the hierarchy, i.e. a total of 111 series × 61 rolling evaluations = 6771 instances. For the "Sales" data set (graphs on the right), the ranks are computed considering all the series of the hierarchies included in the data set, i.e. a total of 15 series × 61 rolling evaluations × 55 hierarchies = 50325 instances..

primarily focuses. The promising empirical results are driven from the design of our approach. HF ML not only results in consistent forecasts across aggregation levels, as is the case with more traditional hierarchical approaches, but also explicitly takes into account the out-of-sample forecast accuracy. The derived combination weights of the HF ML approach provide a selective pooling of the forecasts across the various aggregation levels.

It would be interesting to explore if our insights stand for other ML methods and other data sets. In the following, we discuss additional, alternative paths for future investigation.

- In this study, we focused on the case of cross-sectional hierarchical structures. However, forecasting with hierarchies has been extended to the temporal and the cross-temporal dimensions [4–6]. Future work could apply our approach to these dimensions as well and benchmark against standard, linear reconciliation approaches. One challenge, though, has to do with the size of the task, especially in the cross-temporal domain, and the ability to apply the ML approaches described here when the time series are not long enough.
- Our approach focused on optimizing the performance of the bottom-level series, building m_k models in total. Further research could generalize this optimization objective to other (or multiple) levels of aggregation.
- We showed that HF ML approaches perform better in the case of point forecasting. Future research could extend our results to include evaluation on the forecast uncertainty [15].
- Our empirical study included two data sets, sampled in monthly and weekly frequencies. We expect that the performance improvements observed by applying non-linear approaches to hierarchical forecast reconciliation would be amplified for higher data frequencies (e.g. daily or hourly).
- Despite the improved forecasting performance, the computational complexity should be also examined. It is important to trade off any gains on the forecast accuracy against additional computational cost/resources [63,64].
- In this study, we used tree-based models for reconciling the base forecasts as the results of the M5 forecasting competition suggest that models like LightGBM work well with

hierarchical time series data. Nevertheless, since the proposed approach is model independent, future studies could consider other nonlinear models, such as NNs, to forecast hierarchical time series. It should be noted, however, that NNs are more data-hungry in nature, requiring larger sample sizes for being effectively trained, meaning that they may not be as suitable as tree-based models for forecasting relatively small data sets such as those used in this study.

CRediT authorship contribution statement

Evangelos Spiliotis: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Mahdi Abolghasemi:** Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Rob J. Hyndman:** Conceptualization, Investigation, Methodology, Project administration, Supervision, Writing – original draft. **Fotios Petropoulos:** Conceptualization, Investigation, Methodology, Project administration, Writing – original draft, Writing – review & editing. **Vassilios Assimakopoulos:** Conceptualization, Investigation, Project administration, Supervision.

Declaration of competing 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.

References

[1] G. Athanasopoulos, R.A. Ahmed, R.J. Hyndman, Hierarchical forecasts for Australian domestic tourism, Int. J. Forecast. 25 (1) (2009) 146–166.

[2] R.J. Hyndman, R.A. Ahmed, G. Athanasopoulos, H.L. Shang, Optimal combination forecasts for hierarchical time series, Comput. Statist. Data Anal. 55 (9) (2011) 2579–2589.

[3] S.L. Wickramasuriya, G. Athanasopoulos, R.J. Hyndman, Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization, J. Amer. Statist. Assoc. 114 (526) (2019) 804–819.

[4] G. Athanasopoulos, R.J. Hyndman, N. Kourentzes, F. Petropoulos, Forecasting with temporal hierarchies, European J. Oper. Res. 262 (1) (2017) 60–74.

[5] N. Kourentzes, G. Athanasopoulos, Cross-temporal coherent forecasts for Australian tourism, Ann. Tour. Res. 75 (2019) 393–409.

[6] E. Spiliotis, F. Petropoulos, N. Kourentzes, V. Assimakopoulos, Cross-temporal aggregation: Improving the forecast accuracy of hierarchical electricity consumption, Appl. Energy 261 (2020) 114339.

[7] F. Petropoulos, D. Apiletti, V. Assimakopoulos, M.Z. Babai, D.K. Barrow, C. Bergmeir, R.J. Bessa, J.E. Boylan, J. Browell, C. Carnevale, J.L. Castle, P. Cirillo, M.P. Clements, C. Cordeiro, F.L.C. Oliveira, S. De Baets, A. Dokumentov, P. Fiszeder, P.H. Franses, M. Gilliland, M. Sinan Gönül, P. Goodwin, L. Grossi, Y. Grushka-Cockayne, M. Guidolin, M. Guidolin, U. Gunter, X. Guo, R. Guseo, N. Harvey, D.F. Hendry, R. Hollyman, T. Januschowski, J. Jeon, V.R.R. Jose, Y. Kang, A.B. Koehler, S. Kolassa, N. Kourentzes, S. Leva, F. Li, K. Litsiou, S. Makridakis, A.B. Martinez, S. Meeran, T. Modis, K. Nikolopoulos, D. Önkal, A. Paccagnini, I. Panapakidis, J.M. Pavía, M. Pedio, D.J. Pedregal, P. Pinson, P. Ramos, D.E. Rapach, J. James Reade, B. Rostami-Tabar, M. Rubaszek, G. Sermpinis, H.L. Shang, E. Spiliotis, A.A. Syntetos, P.D. Talagala, T.S. Talagala, L. Tashman, D. Thomakos, T. Thorarinsdottir, E. Todini, J.R.T. Arenas, X. Wang, R.L. Winkler, A. Yusupova, F. Ziel, Forecasting: Theory and Practice, Tech. Rep., 2020, URL arXiv:2012.03854.

[8] G. Athanasopoulos, P. Gamakumara, A. Panagiotelis, R.J. Hyndman, M. Affan, Hierarchical forecasting, in: P. Fuleky (Ed.), Macroeconomic Forecasting in the Era of Big Data: Theory and Practice, Springer International Publishing, 2020, pp. 689–719.

[9] B.J. Dangerfield, J.S. Morris, Top-down or bottom-up: Aggregate versus disaggregate extrapolations, Int. J. Forecast. 8 (2) (1992) 233–241.

[10] A. Zellner, J. Tobias, A note on aggregation, disaggregation and forecasting performance, J. Forecast. 19 (5) (2000) 457–465.

[11] K.B. Kahn, Revisiting top-down versus bottom-up forecasting, J. Bus. Forecast. 17 (2) (1998) 14.

[12] C.W. Gross, J.E. Sohl, Disaggregation methods to expedite product line forecasting, J. Forecast. 9 (3) (1990) 233–254.

[13] G. Fliedner, An investigation of aggregate variable time series forecast strategies with specific subaggregate time series statistical correlation, Comput. Oper. Res. 26 (10–11) (1999) 1133–1149.

[14] C.L. Pennings, J. van Dalen, Integrated hierarchical forecasting, European J. Oper. Res. 263 (2) (2017) 412–418.

[15] J. Jeon, A. Panagiotelis, F. Petropoulos, Probabilistic forecast reconciliation with applications to wind power and electric load, European J. Oper. Res. (2019).

[16] W. Abouarghoub, N.K. Nomikos, F. Petropoulos, On reconciling macro and micro energy transport forecasts for strategic decision making in the tanker industry, Transp. Res. E Logist. Transp. Rev. 113 (2018) 225–238.

[17] R. Hollyman, F. Petropoulos, M.E. Tipping, Understanding forecast reconciliation, European J. Oper. Res. (2021) http://dx.doi.org/10.1016/j.ejor.2021.01.017.

[18] M. Abolghasemi, R.J. Hyndman, G. Tarr, C. Bergmeir, Machine learning applications in time series hierarchical forecasting, 2019, arXiv preprint arXiv:1912.00370.

[19] E. Spiliotis, F. Petropoulos, V. Assimakopoulos, Improving the forecasting performance of temporal hierarchies, PLoS One 14 (10) (2019) e0223422.

[20] G. Fliedner, Hierarchical forecasting: issues and use guidelines, Ind. Manage. Data Syst. 101 (1) (2001) 5–12.

[21] C.W. Gross, J.E. Sohl, Disaggregation methods to expedite product line forecasting, J. Forecast. 9 (3) (1990) 233–254.

[22] Z.D. Nenova, J.H. May, Determining an optimal hierarchical forecasting model based on the characteristics of the data set, J. Oper. Manage. 44 (2016) 62–68.

[23] B.N. Oreshkin, D. Carpov, N. Chapados, Y. Bengio, N-BEATS: neural basis expansion analysis for interpretable time series forecasting, 2019, CoRR abs/1905.10437, arXiv:1905.10437. URL http://arxiv.org/abs/1905.10437.

[24] D. Salinas, V. Flunkert, J. Gasthaus, T. Januschowski, DeepAR: Probabilistic forecasting with autoregressive recurrent networks, Int. J. Forecast. 36 (3) (2020) 1181–1191, http://dx.doi.org/10.1016/j.ijforecast.2019.07.001, URL http://www.sciencedirect.com/science/article/pii/S0169207019301888.

[25] A.-A. Semenoglou, E. Spiliotis, S. Makridakis, V. Assimakopoulos, Investigating the accuracy of cross-learning time series forecasting methods, Int. J. Forecast. (2020) http://dx.doi.org/10.1016/j.ijforecast.2020.11.009, URL http://www.sciencedirect.com/science/article/pii/S0169207020301850.

[26] S. Makridakis, E. Spiliotis, V. Assimakopoulos, The M4 competition: 100,000 time series and 61 forecasting methods, Int. J. Forecast. 36 (1) (2020) 54–74, http://dx.doi.org/10.1016/j.ijforecast.2019.04.014, M4 Competition. URL http://www.sciencedirect.com/science/article/pii/S0169207019301128.

[27] S. Makridakis, E. Spiliotis, V. Assimakopoulos, The M5 accuracy competition: Results, findings and conclusions, 2020, URL https://www.researchgate.net/publication/344487258_The_M5_Accuracy_competition_Results_findings_and_conclusions.

[28] S. Ma, R. Fildes, Retail sales forecasting with meta-learning, European J. Oper. Res. 288 (1) (2021) 111–128, http://dx.doi.org/10.1016/j.ejor.2020.05.038, URL http://www.sciencedirect.com/science/article/pii/S0377221720304847.

[29] S. Ma, R. Fildes, T. Huang, Demand forecasting with high dimensional data: The case of SKU retail sales forecasting with intra- and inter-category promotional information, European J. Oper. Res. 249 (1) (2016) 245–257, http://dx.doi.org/10.1016/j.ejor.2015.08.029, URL http://www.sciencedirect.com/science/article/pii/S0377221715007845.

[30] R.J. Hyndman, G. Athanasopoulos, Forecasting: Principles and Practice, second ed., OTexts, Melbourne, Australia, 2018, URL http://OTexts.com/fpp2.

[31] R. Hyndman, A. Lee, E. Wang, S. Wickramasuriya, hts: Hierarchical and grouped time series, 2020, R package version 6.0.0. URL https://CRAN.R-project.org/package=hts.

[32] R.J. Hyndman, A.J. Lee, E. Wang, Fast computation of reconciled forecasts for hierarchical and grouped time series, Comput. Statist. Data Anal. 97 (2016) 16–32.

[33] J. Schäfer, K. Strimmer, A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics, Stat. Appl. Genet. Mol. Biol. 4 (1) (2005).

[34] P. Montero-Manso, G. Athanasopoulos, R.J. Hyndman, T.S. Talagala, FFORMA: Feature-based forecast model averaging, Int. J. Forecast. 36 (1) (2020) 86–92.

[35] T. Chen, C. Guestrin, XGBoost: A scalable tree boosting system, in: Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, ACM, 2016, pp. 785–794.

[36] D. Nielsen, Tree Boosting with Xgboost-Why Does Xgboost Win "Every" Machine Learning Competition? (Master's thesis), NTNU, 2016.

[37] S.P. Chatzis, V. Siakoulis, A. Petropoulos, E. Stavroulakis, N. Vlachogiannakis, Forecasting stock market crisis events using deep and statistical machine learning techniques, Expert Syst. Appl. 112 (2018) 353–371.

[38] H. Demolli, A.S. Dokuz, A. Ecemis, M. Gokcek, Wind power forecasting based on daily wind speed data using machine learning algorithms, Energy Convers. Manage. 198 (2019) 111823.

[39] J. Snoek, H. Larochelle, R.P. Adams, Practical bayesian optimization of machine learning algorithms, in: Advances in Neural Information Processing Systems, 2012, pp. 2951–2959.

[40] Y. Yan, rBayesianOptimization: Bayesian optimization of hyperparameters, 2016, R package version 1.1.0. URL https://CRAN.R-project.org/package=rBayesianOptimization.

[41] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.

[42] S. Smyl, N.G. Hua, Machine learning methods for GEFCom2017 probabilistic load forecasting, Int. J. Forecast. 35 (4) (2019) 1424–1431.

[43] Y.-Y. Cheng, P.P. Chan, Z.-W. Qiu, Random forest based ensemble system for short term load forecasting, in: 2012 International Conference on Machine Learning and Cybernetics, 1, IEEE, 2012, pp. 52–56.

[44] F. Jiménez, G. Sánchez, J.M. García, G. Sciavicco, L. Miralles, Multi-objective evolutionary feature selection for online sales forecasting, Neurocomputing 234 (2017) 75–92.

[45] P. Probst, M.N. Wright, A.-L. Boulesteix, Hyperparameters and tuning strategies for random forest, Wiley Interdiscip. Rev. Data Min. Knowl. Discov. 9 (3) (2019) e1301.

[46] R.K. Barman, S. Saha, S. Das, Prediction of interactions between viral and host proteins using supervised machine learning methods, PLoS One 9 (11) (2014).

[47] L. Breiman, J. Friedman, C.J. Stone, R.A. Olshen, Classification and Regression Trees, CRC Press, 1984.

[48] J.H. Friedman, Stochastic gradient boosting, Comput. Statist. Data Anal. 38 (4) (2002) 367–378.

[49] A. Liaw, M. Wiener, Classification and regression by randomForest, R News 2 (3) (2002) 18–22, URL https://CRAN.R-project.org/doc/Rnews/.

[50] B. Bischl, M. Lang, L. Kotthoff, J. Schiffner, J. Richter, E. Studerus, G. Casalicchio, Z.M. Jones, mlr: Machine learning in R, J. Mach. Learn. Res. 17 (170) (2016) 1–5, URL http://jmlr.org/papers/v17/15-066.html.

[51] R. Hyndman, G. Athanasopoulos, C. Bergmeir, G. Caceres, L. Chhay, M. O'Hara-Wild, F. Petropoulos, S. Razbash, E. Wang, F. Yasmeen, forecast: Forecasting functions for time series and linear models, 2020, R package version 8.12. URL http://pkg.robjhyndman.com/forecast.

[52] R.J. Hyndman, Y. Khandakar, Automatic time series forecasting: The forecast package for R, J. Stat. Softw. 27 (3) (2008) 1–22, http://dx.doi.org/10.18637/jss.v027.i03.

[53] M. Abolghasemi, E. Beh, G. Tarr, R. Gerlach, Demand forecasting in supply chain: The impact of demand volatility in the presence of promotion, Comput. Ind. Eng. (2020) 106380.

[54] M. Abolghasemi, R. Hyndman, E. Spiliotis, C. Bergmeir, Model selection in reconciling hierarchical time series, 2020, arXiv preprint arXiv:2010.10742.

[55] R.J. Hyndman, A.B. Koehler, R.D. Snyder, S. Grose, A state space framework for automatic forecasting using exponential smoothing methods, Int. J. Forecast. 18 (3) (2002) 439–454.

[56] V. Assimakopoulos, K. Nikolopoulos, The theta model: A decomposition approach to forecasting, Int. J. Forecast. 16 (4) (2000) 521–530.

[57] R.J. Hyndman, A.B. Koehler, Another look at measures of forecast accuracy, Int. J. Forecast. 22 (4) (2006) 679–688.

[58] N.C. Schwertman, A.J. Gilks, J. Cameron, A simple noncalculus proof that the median minimizes the sum of the absolute deviations, Amer. Statist. 44 (1) (1990) 38–39.

[59] S. Kolassa, Evaluating predictive count data distributions in retail sales forecasting, Int. J. Forecast. 32 (3) (2016) 788–803.

[60] L.J. Tashman, Out-of-sample tests of forecasting accuracy: an analysis and review, Int. J. Forecast. 16 (4) (2000) 437–450.

[61] S.B. Taieb, A.F. Atiya, A bias and variance analysis for multistep-ahead time series forecasting, IEEE Trans. Neural Netw. Learn. Syst. 27 (1) (2016) 62–76.

[62] A.J. Koning, P.H. Franses, M. Hibon, H.O. Stekler, The M3 competition: Statistical tests of the results, Int. J. Forecast. 21 (3) (2005) 397–409.

[63] M. Gilliland, The value added by machine learning approaches in forecasting, Int. J. Forecast. 36 (1) (2020) 161–166.

[64] K. Nikolopoulos, F. Petropoulos, Forecasting for big data: Does suboptimality matter? Comput. Oper. Res. 98 (2018) 322–329.