Optimal forecast reconciliation with time series selection

Xiaoqian Wang*
Department of Econometrics & Business Statistics, Monash University, VIC 3800, Australia
and
Rob J Hyndman
Department of Econometrics & Business Statistics, Monash University, VIC 3800, Australia
and
Shanika L Wickramasuriya
Department of Econometrics & Business Statistics, Monash University, VIC 3800, Australia

October 15, 2024

Abstract

Forecast reconciliation ensures forecasts of time series in a hierarchy adhere to aggregation constraints, enabling aligned decision making. While forecast reconciliation can enhance overall accuracy in a hierarchical or grouped structure, the most substantial improvements generally occur in series with initially poor-performing base forecasts. Forecasts of certain series may get worse after reconciliation. In practical applications, some series in a structure often produce poor base forecasts due to model misspecification or low forecastability. To mitigate their negative impact, we propose two categories of forecast reconciliation methods that incorporate automatic time series selection based on out-of-sample and in-sample information, respectively. These methods keep "poor" base forecasts unused in forming reconciled forecasts, while adjusting the weights assigned to the remaining series accordingly when generating bottom-level reconciled forecasts. Additionally, our methods ameliorate disparities stemming from varied estimators of the base forecast error covariance matrix, alleviating challenges associated with estimator selection. Empirical evaluations through two simulation studies and applications using Australian labour force and domestic tourism data demonstrate the potential of the proposed methods to exclude series with high scaled forecast errors and show promising results.

Keywords: Forecasting, Hierarchical time series, Linear forecast reconciliation, Variable selection, Integer programming

I don't think it is correct to say that this method excludes series with "high FE" because FE depends on the scale. The top level series might have the largest FE due to its scale but this doesn't necessarily mean that the series produces poor forecasts. XQ — It should be scaled forecast errors as we report RMSSE in the results.

*Corresponding author. E-mail address: xiaoqian.wang@monash.edu (X. Wang).

1 Introduction

Forecast reconciliation is a post-processing method that ensures forecasts of multivariate time series adhere to known linear constraints present in the data (Hyndman et al. 2011). For example, the sum of regional unemployment forecasts should be equal to the national unemployment forecast.

Hyndman et al. (2011) introduced optimal forecast reconciliation, whereby "base" forecasts of all series are generated independently, and then adjusted to satisfy the constraints, leading to a set of coherent reconciled forecasts. Subsequent research has extended and developed the idea in the context of cross-sectional data (Hyndman et al. 2016, Wickramasuriya et al. 2019, Panagiotelis et al. 2021), temporal data (Athanasopoulos et al. 2017), and cross-temporal data (Di Fonzo & Girolimetto 2023). Athanasopoulos et al. (2024) provided a comprehensive introduction to the forecast reconciliation literature.

Reconciliation is known to improve overall forecast accuracy in collections of time series with aggregation constraints. On average, when the base forecasts are unbiased, the mean squared reconciled forecast error from the minimum trace reconciliation method (Wickramasuriya et al. 2019) is lower than that from the base forecasts (Wickramasuriya 2021). Most of the improvements attributed to reconciliation are observed in series with initially poor-performing base forecasts (Athanasopoulos et al. 2017). In practice, hierarchical time series can encompass hundreds or even thousands of individual series, making it impractical to focus on each model we fit. To address this complexity, we often rely on automated model selection methods. It is not uncommon for some series to have poor base forecasts due to challenges such as model misspecification or low signal-to-noise ratio (SNR). In such cases, it may be advantageous to exclude these poor-performing forecasts when performing reconciliation. This is the motivation for our proposed methods.

First, we propose constrained forecast reconciliation methods that incorporate time series selection based on out-of-sample information, assuming unbiased base forecasts. This is formulated as an optimization problem, using diverse penalty functions to control the number of nonzero column entries in the weighting matrix for linear forecast reconciliation. We show that the number of selected time series is at least equal to the number of series at the bottom level, and we can reconstruct the entire structure by aggregating/disaggregating the selected series. Second, we relax the unbiasedness assumption and introduce an additional unconstrained reconciliation method with selection, utilizing in-sample observations and their fitted values. This method leverages in-sample reconciliation performance for selection, potentially selecting fewer time series than the number of bottom-level series, and in extreme cases, resembling a top-down approach. Third, we conduct two Monte Carlo simulations to show the benefits of our methods under model misspecification and varying correlation levels among series. Finally, we assess the performance of the proposed methods in two real-world applications, a grouped hierarchy of Australian labour force data and a geographic hierarchy of Australian domestic tourism data.

The methods proposed in this paper incorporate time series selection to enhance forecast reconciliation. Our experiments demonstrate their potential to exclude series with high scaled forecast errors. In particular, the

don't think "high forecast error" is the right term.
XQ — It should be scaled forecas errors.

unconstrained reconciliation method with selection shows promising results in both simulations and real-world applications. Constrained methods with selection are particularly effective in addressing model misspecification. A remarkable feature of our methods is their ability to reduce disparities arising from using different estimators of the base forecast error covariance matrix, thereby mitigating challenges associated with estimator selection, which is a prominent concern in forecast reconciliation research.

Unlike Zhang et al. (2023), which keep good forecasts unchanged, we leave poor base forecasts unused in generating reconciled forecasts. Notably, our methods automate the selection process, relieving practitioners from the task of carefully choose which series to exclude. There may be other practical motivations to use our proposed approach. For example, in labour force forecasting, economists may be interested in various attributes such as gender, labor market region, and duration of job search, which create a complex hierarchical structure. Some series within this structure may be poorly specified for forecasting or exhibit weak patterns, resulting in poor base forecasts. Our methods selectively use some series into the reconciliation process, potentially improving overall accuracy while still providing reconciled forecasts for the entire hierarchy. Moreover, the selection procedure avoids the need to forecast the excluded series in future operations, at least for a short duration, and employs only the selected series along with the estimated weighting matrix for reconciliation, thereby improving computational efficiency. Finally, they also adapt well to scenarios with missing or extremely sparse data by adding additional constraints to exclude such series, enabling us to bypass their base forecasts while still returning reconciled outcomes.

Unsure from whe does we get com tational efficiency We need to fit m els to each series and get their fore casts to decide which series nee to be eliminated. XQ — All fore- casts are needed when using our reconciliation me ods. What I mea is that, after im- plementing our methods, we can avoid forecasting excluded series i "future operation at least for a sho period, and only use the selected series and the est mated G for reco ciliation.

The remainder of the paper is structured as follows. Section 2 presents notations and a review of linear forecast reconciliation methods. Section 3 introduces our proposed methods, which incorporate time series selection to strengthen forecast reconciliation, and provides theoretical insights. Section 4 and Section 5 show the results from two simulations and two real-world datasets, respectively. Section 6 discusses the practical significance, contributions, limitations, and potential future research. Finally, Section 7 provides the conclusion. The code and data for reproducing the results is available at https://github.com/xqnwang/hfs.

2 Preliminaries

2.1 Notation

A *hierarchical time series* is an n-dimensional multivariate time series that adheres to known linear constraints. Let $\boldsymbol{y}_t \in \mathbb{R}^n$ be a vector comprising observations from all time series in the hierarchy at time t, and $\boldsymbol{b}_t \in \mathbb{R}^{n_b}$ be a vector comprising observations of only the most disaggregated (bottom-level) series. The full hierarchy can be written as

$$\boldsymbol{y}_t = \boldsymbol{S}\boldsymbol{b}_t,$$

for $t = 1, 2, \ldots, T$, where T is the length of the time series, and $\boldsymbol{S}$ is an $n \times n_b$ *summing matrix* that defines the aggregation constraints. We can write the summing matrix as $\boldsymbol{S} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{I}_{n_b} \end{bmatrix}$, where $\boldsymbol{A}$ is an $n_a \times n_b$ *aggregation matrix* with $n = n_a + n_b$, and $\boldsymbol{I}_{n_b}$ is an n_b-dimensional identity matrix.

For example, Figure 1 shows a simple hierarchy with $n = 7$, $n_b = 4$, $n_a = 3$, $\boldsymbol{y}_t =$

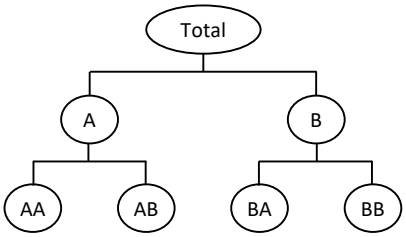


Figure 1: *An example of a two-level hierarchical time series.*

$[y_{\text{Total},t}, y_{\text{A},t}, y_{\text{B},t}, y_{\text{AA},t}, y_{\text{AB},t}, y_{\text{BA},t}, y_{\text{BB},t}]'$, $\boldsymbol{b}_t = [y_{\text{AA},t}, y_{\text{AB},t}, y_{\text{BA},t}, y_{\text{BB},t}]'$, and

$$\boldsymbol{S} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ & & \boldsymbol{I}_4 & \end{bmatrix}.$$

The notation is general enough to include aggregation constraints that are non-hierarchical. Please refer to Hyndman & Athanasopoulos (2021) for further details.

Hierarchical forecasting methods have been extensively applied across diverse domains. For instance, forecast reconciliation is widely implemented in tourism data (Athanasopoulos et al. 2009), where hierarchical time series arise due to geographic divisions. Total overnight trips for a whole nation can be disaggregated to states, and further subdivided into regions. In the context of a grocery retailer, the total sales of the "food" category can be subdivided into various subcategories and subsequently into distinct items (Zhang et al. 2023, Hollyman et al. 2021). In electricity load forecasting, consumption is measured using smart meters which naturally fall within a comprehensive geographic hierarchy (Ben Taieb et al. 2021). For additional interesting application examples, please refer to Athanasopoulos et al. (2024).

2.2 Linear forecast reconciliation

Let $\hat{\boldsymbol{y}}_{T+h|T} \in \mathbb{R}^n$ be a vector of h-step-ahead *base forecasts* for all time series in the hierarchy, given observations up to and including time T, and stacked in the same order as $\boldsymbol{y}_t$. We can use any method to generate these forecasts, but in general they will not be coherent (i.e., they won't satisfy the aggregation constraints). Let $\tilde{\boldsymbol{y}}_{T+h|T} \in \mathbb{R}^n$ denote a vector of h-step-ahead *reconciled forecasts* given by

$$\tilde{\boldsymbol{y}}_{T+h|T} = \boldsymbol{S}\boldsymbol{G}_h\hat{\boldsymbol{y}}_{T+h|T}, \tag{1}$$

where $\boldsymbol{G}_h$ is an $n_b \times n$ *weighting matrix*.

In general, forecast reconciliation methods consider the loss function (Ben Taieb & Koo 2019) given by

$$\begin{aligned} &\mathrm{E}\left[\left\|\boldsymbol{y}_{T+h} - \tilde{\boldsymbol{y}}_{T+h|T}\right\|_2^2 \mid \boldsymbol{I}_T\right] \\ &= \underbrace{\left\|\boldsymbol{S}\boldsymbol{G}_h\left(\mathrm{E}\left[\hat{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{I}_T\right] - \mathrm{E}\left[\boldsymbol{y}_{T+h} \mid \boldsymbol{I}_T\right]\right) + (\boldsymbol{S} - \boldsymbol{S}\boldsymbol{G}_h\boldsymbol{S})\mathrm{E}\left[\boldsymbol{b}_{T+h} \mid \boldsymbol{I}_T\right]\right\|_2^2}_{\text{bias}^2} + \underbrace{\mathrm{Tr}\left(\mathrm{Var}\left[\boldsymbol{y}_{T+h} - \tilde{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{I}_T\right]\right)}_{\text{variance}}, \end{aligned} \tag{2}$$

where $\|\cdot\|_2$ is the L_2 norm. This equation includes two parts in its decomposition, bias and variance of the reconciled forecasts.

Is it conditional MSE or unconditional MSE? In the 2019 paper, we used conditional expectation but later thought it should be unconditional MSE XQ — I think it follows the idea in the 2019 paper which is based on conditional MSE. Why do you think it should be unconditional MSE? If that is the case, should we remove

Minimum trace reconciliation

Let $\hat{\boldsymbol{e}}_{t+h|t} = \boldsymbol{y}_{t+h} - \hat{\boldsymbol{y}}_{t+h|t}$ denote the h-step-ahead in-sample *base forecast errors*, and $\tilde{\boldsymbol{e}}_{t+h|t} = \boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}_{t+h|t}$ denote the h-step-ahead *reconciled forecast errors*. Assuming the base forecasts are unbiased and imposing the constraint $\boldsymbol{G}_h\boldsymbol{S} = \boldsymbol{I}_{n_b}$ to preserve the unbiasedness of the reconciled forecasts, the bias term in Equation (2) equates to zero. Wickramasuriya et al. (2019) thus formulated the reconciliation problem as minimizing the trace (MinT) of the h-step-ahead covariance matrix of the reconciled forecast errors, leading to the unique solution given by

$$\boldsymbol{G}_h = \left(\boldsymbol{S}'\boldsymbol{W}_h^{-1}\boldsymbol{S}\right)^{-1}\boldsymbol{S}'\boldsymbol{W}_h^{-1}, \tag{3}$$

where $\boldsymbol{W}_h$ is the positive definite covariance matrix of the h-step-ahead base forecast errors.

The MinT problem can be reformulated as a least squares problem with linear constraints:

$$\min_{\tilde{\boldsymbol{y}}_{T+h|T}} \quad \frac{1}{2}(\hat{\boldsymbol{y}}_{T+h|T} - \tilde{\boldsymbol{y}}_{T+h|T})'\boldsymbol{W}_h^{-1}(\hat{\boldsymbol{y}}_{T+h|T} - \tilde{\boldsymbol{y}}_{T+h|T}) \qquad \text{s.t.} \quad \tilde{\boldsymbol{y}}_{T+h|T} = \boldsymbol{S}\tilde{\boldsymbol{b}}_{T+h|T}, \tag{4}$$

where $\tilde{\boldsymbol{b}}_{T+h|T} \in \mathbb{R}^{n_b}$ comprises the h-step-ahead bottom-level reconciled forecasts, made at time T. The intuition behind MinT reconciliation is that the larger the estimated variance of the base forecast errors, the larger the range of adjustments permitted for forecast reconciliation.

It is challenging to estimate $\boldsymbol{W}_h$, especially for $h > 1$. It is common to assume $\boldsymbol{W}_h = k_h\boldsymbol{W}_1, \forall h$, where $k_h > 0$; then the MinT solution for $\boldsymbol{G}$ remains unchanged across different forecast horizons, h. Hence, we will drop the subscript h for ease of exposition. Table 1 lists the most popularly used candidate estimators for $\boldsymbol{W}_h$. In principle, all optimization methods can be based on either in-sample or out-of-sample data. The methods discussed here are considered "out-of-sample" as they use genuine forecasts, $\hat{\boldsymbol{y}}_{T+h|T}$, rather than "in-sample" fitted values for the optimization process.

Table 1: *Forecast reconciliation methods for which different estimators of $\boldsymbol{W}_h$ are used.*

Reconciliation method	$\boldsymbol{W}_h \propto$
OLS (Hyndman et al. 2011)	$\boldsymbol{I}$
WLSs (Athanasopoulos et al. 2017)	$\text{diag}(\boldsymbol{S}\mathbf{1})$
WLSv (Hyndman et al. 2016)	$\text{Diag}(\hat{\boldsymbol{W}}_1)$
MinT (Wickramasuriya et al. 2019)	$\hat{\boldsymbol{W}}_1$
MinTs (Wickramasuriya et al. 2019)	$\lambda\,\text{Diag}(\hat{\boldsymbol{W}}_1) + (1-\lambda)\hat{\boldsymbol{W}}_1$

Note: $\mathbf{1}$ is a vector of 1s of size n_b, $\text{diag}(\cdot)$ constructs a diagonal matrix using a given vector, $\hat{\boldsymbol{W}}_1$ denotes the unbiased covariance estimator based on the in-sample one-step-ahead base forecast errors (i.e., residuals), and $\text{Diag}(\cdot)$ forms a diagonal matrix using the diagonal elements of the input matrix. $\lambda \in [0,1]$ is the shrinkage intensity parameter.

Relaxation of the unbiasedness assumptions

Ben Taieb & Koo (2019) proposed a reconciliation method relaxing the assumption of unbiasedness. Their goal was to achieve a tradeoff between bias and variance by directly minimizing the mean squared reconciled forecast errors in Equation (2). By expanding the training window incrementally, one observation at a time, they formulated the

reconciliation problem as a regularized empirical risk minimization (RERM) problem:

$$\min_{\boldsymbol{G}_h} \frac{1}{(T-T_1-h+1)n} \left\| \boldsymbol{Y}_h^* - \hat{\boldsymbol{Y}}_h^* \boldsymbol{G}_h' \boldsymbol{S}' \right\|_F^2 + \lambda \| \operatorname{vec}(\boldsymbol{G}_h) \|_1,$$

where T_1 denotes the minimum number of observations used for model training, $\|\cdot\|_F$ is the Frobenius norm, $\|\cdot\|_1$ is the L_1 norm, $\operatorname{vec}(\cdot)$ denotes the vectorization of a matrix (stacking the columns of the matrix), $\boldsymbol{Y}_h^* = [\boldsymbol{y}_{T_1+h}, \ldots, \boldsymbol{y}_T]'$, $\hat{\boldsymbol{Y}}_h^* = \left[\hat{\boldsymbol{y}}_{T_1+h|T_1}, \ldots, \hat{\boldsymbol{y}}_{T|T-h}\right]'$, and $\lambda \geq 0$ is a regularization parameter.

When $\lambda = 0$, the problem reduces to an empirical risk minimization (ERM) problem without regularization. Assuming that the series in the structure are jointly weakly stationary and $\hat{\boldsymbol{Y}}_h^{*\prime}\hat{\boldsymbol{Y}}_h^*$ is invertible, it has a closed-form solution given by

$$\hat{\boldsymbol{G}}_h = \boldsymbol{B}_h^{*\prime}\hat{\boldsymbol{Y}}_h^* \left(\hat{\boldsymbol{Y}}_h^{*\prime}\hat{\boldsymbol{Y}}_h^*\right)^{-1},$$

where $\boldsymbol{B}_h^* = [\boldsymbol{b}_{T_1+h}, \ldots, \boldsymbol{b}_T]'$. If $\hat{\boldsymbol{Y}}_h^{*\prime}\hat{\boldsymbol{Y}}_h^*$ is not invertible, a generalized inverse can be applied. When $\lambda > 0$, imposing the L_1 penalty on $\boldsymbol{G}_h$ introduces sparsity and reduce estimation variance, albeit at the cost of introducing some bias.

Relaxing the assumption of unbiasedness of base forecasts, Wickramasuriya (2021) proposed an empirical MinT (**EMinT**) solution by minimizing the trace of the covariance matrix of the reconciled forecast errors. Assuming the series are jointly weakly stationary, the solution is given by

$$\hat{\boldsymbol{G}}_h = \boldsymbol{B}_h'\hat{\boldsymbol{Y}}_h \left(\hat{\boldsymbol{Y}}_h'\hat{\boldsymbol{Y}}_h\right)^{-1},$$

where $\boldsymbol{B}_h = [\boldsymbol{b}_h, \ldots, \boldsymbol{b}_T]'$, and $\hat{\boldsymbol{Y}}_h = \left[\hat{\boldsymbol{y}}_{h|0}, \ldots, \hat{\boldsymbol{y}}_{T|T-h}\right]'$.

The difference between EMinT and ERM lies in the data sources. EMinT is an "in-sample" method in the sense that $\hat{\boldsymbol{Y}}_h$ are predictions in the form of fitted values, while ERM (and also RERM) is an "out-of-sample" method, with $\hat{\boldsymbol{Y}}_h^*$ being genuine forecasts generated on a holdout validation set. Both EMinT and ERM consider an estimate of $\boldsymbol{G}$ that changes over the forecast horizon, which is why we keep the subscript h here.

In practical settings, some series in a hierarchy could have poor base forecasts due to model misspecification or low forecastability. Specifically, within a hierarchical structure, the influence of unforeseen events may prompt a forecaster to make a bad decision, leading to the use of a misspecified forecasting model for a specific time series and, consequently, yielding inferior forecasts. Moreover, lower-level time series are normally characterized by less apparent trend and seasonality, large intermittence, and volatility, rendering them more challenging to predict and resulting in poor forecasts.

A challenge in forecast reconciliation arises when some base forecasts perform poorly, as the weighting matrix $\boldsymbol{G}$ assimilates *all* base forecasts and maps them into bottom-level forecasts, which are subsequently summed by $\boldsymbol{S}$. While the RERM method introduces sparsity by shrinking some elements of $\boldsymbol{G}$ towards zero, it remains incapable of mitigating the adverse impact of underperforming base forecasts. Moreover, the method is time-consuming because it uses expanding windows to recursively generate out-of-sample base forecasts.

In addition to Ben Taieb & Koo (2019), several other contributions have incorporated diverse forms of shrinkage or penalization in forecast reconciliation methodologies. For example, Pang et al. (2022) introduced a group Lasso

penalty on weights assigned to clusters artificially added in a hierarchy to select ideal clusters. Their objective function focuses on a new hierarchical structure encompassing geographic and data cluster hierarchies, while disregarding forecast errors associated with zero-weighted clusters. Furthermore, they derive the optimal weight vector and optimal bottom level forecasts by solving the objective successively, leading to a time-consuming method that does not permanently mitigate the negative impact of poorly performing clusters on reconciliation performance. To address the insufficient emphasis on coherence in machine learning methods, Mishchenko et al. (2019) and Gleason (2020) included a regularization term to penalize forecast incoherence. However, these soft constraints do not ensure coherence. Nystrup et al. (2020) and Nystrup et al. (2021) considered the autocorrelation in forecast errors and used a shrinkage estimator or eigendecomposition of the cross-correlation matrix, effectively overcoming estimation inefficiencies in approximating $\boldsymbol{W}$ within a temporal hierarchy. Nonetheless, none of the aforementioned contributions achieve time series selection in forecast reconciliation, failing to alleviate their adverse impact on forecast performance, while maintaining consideration for forecast errors across the entire initial hierarchy.

We therefore propose two types of forecast reconciliation methods involving time series selection: constrained "out-of-sample" reconciliation, which adheres to the constraint $\boldsymbol{GS} = \boldsymbol{I}$, and unconstrained "in-sample" reconciliation, which operates without this constraint. These methods aim to address the negative effect of some poor base forecasts on the overall performance of the reconciled forecasts. Additionally, through the incorporation of regularization in the objective function, our method improves reconciliation outcomes produced with a "poor" choice of $\boldsymbol{W}$.

XQ — I've rewri ten this sentence to clarify the dif- ference between constrained and unconstrained me ods.

3 Forecast reconciliation with time series selection

In this section, we introduce our methods for forecast reconciliation while automatically achieving time series selection. Section 3.1 introduces constrained "out-of-sample" reconciliation methods, formulated based on genuine forecasts, while Section 3.2 presents an unconstrained "in-sample" reconciliation method, where the problem is formulated using in-sample observations and predictions in the form of fitted values.

3.1 Series selection under the unbiasedness assumption

As $\boldsymbol{S}$ is fixed and $\hat{\boldsymbol{y}}_{T+h|T}$ is given, $\boldsymbol{G}_h$ determines the linear reconciliation performance, as shown in Equation (1). We drop the subscript h here as we assume $\boldsymbol{W}$ and $\boldsymbol{G}$ do not vary with the forecast horizon. Recognizing that reconciled forecasts $\tilde{\boldsymbol{y}}$ are simply linear combinations of base forecasts $\hat{\boldsymbol{y}}$ via $\tilde{\boldsymbol{y}} = \boldsymbol{SG}\hat{\boldsymbol{y}}$, setting columns of $\boldsymbol{G}$ to zero can eliminate heavily misspecified forecasts from the combination. This leads to a generalization of the MinT optimization problem with an additional penalty term:

$$\min_{\boldsymbol{G}} \quad \frac{1}{2}\left(\hat{\boldsymbol{y}} - \boldsymbol{SG}\hat{\boldsymbol{y}}\right)' \boldsymbol{W}^{-1} \left(\hat{\boldsymbol{y}} - \boldsymbol{SG}\hat{\boldsymbol{y}}\right) + \lambda \mathfrak{g}(\boldsymbol{G}) \qquad \text{s.t.} \quad \boldsymbol{GS} = \boldsymbol{I}, \tag{5}$$

where $\hat{\boldsymbol{y}} := \hat{\boldsymbol{y}}_{T+1|T}$, $\mathfrak{g}(\cdot)$ penalizes the columns of $\boldsymbol{G}$ towards zero, and $\lambda \geq 0$ is a penalty parameter. The methods developed within this framework are "out-of-sample" in the sense that $\hat{\boldsymbol{y}}$ are genuine one-step-ahead forecasts. This

can be considered *a grouped variable selection problem*, with each group corresponding to a column of $\boldsymbol{G}$. When $\lambda = 0$, the problem reduces to the MinT optimization problem in Equation (4) with a closed-form solution.

The constraint $\boldsymbol{GS} = \boldsymbol{I}$ guarantees that the reconciled forecasts remain unbiased if the base forecasts are unbiased. Under this assumption and constraint, minimizing the loss function in Equation (2) simplifies to the MinT problem formulated in Equation (4), which underpins the constrained "out-of-sample" reconciliation methods within the framework in equation (5).

Proposition 1. *If the assumption that forecast reconciliation preserves unbiasedness of the reconciled forecasts is imposed by enforcing $\boldsymbol{GS} = \boldsymbol{I}$, then the number of nonzero column entries of $\hat{\boldsymbol{G}}$ (the solution to Equation (5)) will be no less than n_b. Moreover, the constraint $\boldsymbol{GS} = \boldsymbol{I}$ enforces that the selected columns of $\hat{\boldsymbol{G}}$ will correspond to variables that can "restore" the hierarchy.*

Proof. See Appendix A, supplementary materials. □

For example, for the simple hierarchy shown in Figure 1, the selected columns of $\hat{\boldsymbol{G}}$ will be at least $n_b = 4$. Our constrained reconciliation methods might simultaneously zero out the columns of $\boldsymbol{G}$ corresponding to series AA and BA, but not to series AA and AB.

Proposition 2. *The optimization problem in Equation (5) can be reformulated as a least squares problem with regularization and linear equality constraint as follows:*

$$\begin{aligned} \min_{\text{vec}(\boldsymbol{G})} \quad & \frac{1}{2}\left(\hat{\boldsymbol{y}} - \left(\hat{\boldsymbol{y}}' \otimes \boldsymbol{S}\right)\text{vec}(\boldsymbol{G})\right)' \boldsymbol{W}^{-1}\left(\hat{\boldsymbol{y}} - \left(\hat{\boldsymbol{y}}' \otimes \boldsymbol{S}\right)\text{vec}(\boldsymbol{G})\right) + \lambda\, \mathfrak{g}\left(\text{vec}(\boldsymbol{G})\right) \\ s.t. \quad & \left(\boldsymbol{S}' \otimes \boldsymbol{I}_{n_b}\right)\text{vec}(\boldsymbol{G}) = \text{vec}(\boldsymbol{I}_{n_b}), \end{aligned} \tag{6}$$

which is characterized as a high-dimensional problem in which the number of features, denoted as $p = n_b \times n$, is much larger than the number of observations, n.

Proof. See Appendix A, supplementary materials. □

Next, we present three constrained "out-of-sample" reconciliation methods: (i) group best-subset selection with ridge regularization, (ii) parsimonious method with L_0 regularization, and (iii) group lasso method. These methods perform forecast reconciliation with series selection under the unbiasedness assumption, differing only in the regularization term employed.

Group best-subset selection with ridge regularization

In a high-dimensional context with $p \gg n$, it is common to assume the true regression coefficient (i.e., $\text{vec}(\boldsymbol{G})$ in our problem) is sparse. We apply a combination of L_0 and L_2 regularization to control the nonzero column entries in $\boldsymbol{G}$:

$$\begin{aligned} \min_{\text{vec}(\boldsymbol{G})} \quad & \frac{1}{2}\left(\hat{\boldsymbol{y}} - \left(\hat{\boldsymbol{y}}' \otimes \boldsymbol{S}\right)\text{vec}(\boldsymbol{G})\right)' \boldsymbol{W}^{-1}\left(\hat{\boldsymbol{y}} - \left(\hat{\boldsymbol{y}}' \otimes \boldsymbol{S}\right)\text{vec}(\boldsymbol{G})\right) + \lambda_0 \sum_{j=1}^{n} 1\left(\boldsymbol{G}_{\cdot j} \neq \boldsymbol{0}\right) + \lambda_2 \left\|\text{vec}\left(\boldsymbol{G}\right)\right\|_2^2 \\ \text{s.t.} \quad & \left(\boldsymbol{S}' \otimes \boldsymbol{I}_{n_b}\right)\text{vec}(\boldsymbol{G}) = \text{vec}(\boldsymbol{I}_{n_b}), \end{aligned} \tag{7}$$

where $1(\cdot)$ is the indicator function, $\lambda_0 \geq 0$ controls the number of nonzero columns of $\boldsymbol{G}$, and $\lambda_2 \geq 0$ controls the strength of the ridge regularization. In a hierarchical or grouped time series context, $\text{vec}(\boldsymbol{G})$ has an inherent non-overlapping grouping structure, wherein each group corresponds to a single column of $\boldsymbol{G}$, each of size n_b. Hence, we call this reconciliation method *group best-subset selection with ridge regularization*. In the results that follow, we label the **Subset** method differently based on various $\boldsymbol{W}$ estimators, referring to them as **OLS-subset**, **WLSs-subset**, **WLSv-subset**, **MinT-subset**, and **MinTs-subset**, respectively.

The best-subsets estimator, derived from an L_0-regularized least squares problem, is a natural and direct candidate for sparse learning. The L_0 penalty leads to models that have a subset of coefficients exactly equal to zero, effectively performing variable selection. The statistical properties of the best-subsets estimator have been extensively studied; see, for example, Greenshtein (2006), Zhang & Zhang (2012), and the references therein. However, Mazumder et al. (2022) argued that the vanilla L_0 penalization could suffer from overfitting in low SNR settings. To address the issue, we incorporate a ridge regularization in Equation (7), motivated by earlier work on best-subset selection (e.g., Hazimeh & Mazumder 2020, Mazumder et al. 2022), which suggests that additional ridge regularization helps mitigate the poor predictive performance of best-subset selection method in low SNR regimes.

We present a Big-M based mixed integer programming (MIP) formulation for the problem in Equation (7):

$$
\begin{aligned}
\min_{\text{vec}(\boldsymbol{G}),\boldsymbol{z},\check{\boldsymbol{e}},\boldsymbol{g}^+} \quad & \frac{1}{2}\check{\boldsymbol{e}}'\boldsymbol{W}^{-1}\check{\boldsymbol{e}} + \lambda_0 \sum_{j=1}^{n} z_j + \lambda_2 \boldsymbol{g}^{+\prime}\boldsymbol{g}^+ \\
\text{s.t.} \quad & \left(\boldsymbol{S}' \otimes \boldsymbol{I}_{n_b}\right) \text{vec}(\boldsymbol{G}) = \text{vec}\left(\boldsymbol{I}_{n_b}\right) \\
& \hat{\boldsymbol{y}} - \left(\hat{\boldsymbol{y}}' \otimes \boldsymbol{S}\right) \text{vec}(\boldsymbol{G}) = \check{\boldsymbol{e}} \\
& \sum_{i=1}^{n_b} g^+_{i+(j-1)n_b} \leqslant \mathscr{M} z_j, \quad j \in [n] \\
& \boldsymbol{g}^+ \geqslant \text{vec}(\boldsymbol{G}) \\
& \boldsymbol{g}^+ \geqslant -\text{vec}(\boldsymbol{G}) \\
& z_j \in \{0,1\}, \quad j \in [n],
\end{aligned}
\tag{8}
$$

where $\mathscr{M}$ is a Big-M parameter (specified a-priori) that is sufficiently large that the optimal solution to Equation (8), $\boldsymbol{g}^{+*}$, satisfies $\max_{j\in[n]} \sum_{i=1}^{n_b} g^+_{i+(j-1)n_b} \leqslant \mathscr{M}$, and $[n] = \{1,2,\ldots,n\}$. The binary variable $z_j = 0$ implies that $\boldsymbol{G}_{\cdot j} = \boldsymbol{0}$, and $z_j = 1$ implies that $\sum_{i=1}^{n_b} g^+_{i+(j-1)n_b} \leqslant \mathscr{M}$. Such Big-M formulations are commonly used in MIP problems to model relations between discrete and continuous variables, and have been recently explored in regression with L_0 regularization (Bertsimas et al. 2016). The problem is a mixed integer quadratic program (MIQP) that can be solved using commercial MIP solvers, e.g., Gurobi and CPLEX.

Parameter tuning. To avoid computationally expensive cross-validation, we tune the parameters to minimize the sum of squared reconciled forecast errors on the truncated training set, comprising only the $\max\{h,s\}$ observations closest to the forecast origin, where s is the seasonal period for seasonal data and $s = T$ for non-seasonal data. Let $\lambda_0^1 = \frac{1}{2}\left(\hat{\boldsymbol{y}} - \tilde{\boldsymbol{y}}^{\text{bench}}\right)' \boldsymbol{W}^{-1} \left(\hat{\boldsymbol{y}} - \tilde{\boldsymbol{y}}^{\text{bench}}\right)$, which captures the scale of the first term in the objective function,

where $\tilde{\boldsymbol{y}}^{\text{bench}}$ is a vector of reconciled forecasts obtained using Equation (3) with the same estimator of $\boldsymbol{W}$, and define $\lambda_0^k = 0.0001\lambda_0^1$. For the parameter λ_0, we consider a grid of $k+1$ values, $\{\lambda_0^1,\ldots,\lambda_0^k,0\}$, where $\lambda_0^j = \lambda_0^1\left(\lambda_0^k/\lambda_0^1\right)^{(j-1)/(k-1)}$ for $j \in [k]$. So $\lambda_0^1,\ldots,\lambda_0^k$ is a sequence decreasing on the log scale. We use a grid of six values for the parameter λ_2, $\{0, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2\}$. Thus, we tune over a two-dimensional grid of $(k+1)\times 6$ values to find the optimal combination of λ_0 and λ_2.

Computation details. The MIQP problem in Equation (8) is NP-hard and computationally intensive. Bertsimas et al. (2016) showed that commercial MIP solvers are capable of tackling problem instances for p up to a thousand. To address larger instances, there has been impressive work on developing MIP-based approaches for solving L_0-regularized regression problem; e.g., Bertsimas et al. (2016), Hazimeh & Mazumder (2020), and Hazimeh et al. (2022). However, it is challenging to extend these approaches to accommodate additional constraints in the optimization problem. Despite potential challenges in handling large instances with commercial MIP solvers, in our experiments, we use Gurobi to solve Equation (8) by configuring parameters such as MIPGap = 0.001 and TimeLimit = 600 seconds for cases with $p > 1000$. This allows to terminate the solver before reaching the global optimum and return a suboptimal solution instead. This strategy is motivated by our need to consider numerous parameter candidates, and the final solution will be validated against the training set, which prevents the use of a poor estimate of $\boldsymbol{G}$.

Parsimonious method with L_0 regularization

Instead of estimating the entire matrix $\boldsymbol{G}$ as above, we leverage the MinT solution in Equation (3) to streamline the optimization problem under consideration. Specifically, we define $\bar{\boldsymbol{S}} = \boldsymbol{A}\boldsymbol{S}$, where $\boldsymbol{A} = \text{diag}(\boldsymbol{z})$ is an $n\times n$ diagonal matrix, and $\boldsymbol{z}$ is an n-dimensional vector with elements either equal to 0 or 1. Taking the MinT solution in Equation (3), we have $\bar{\boldsymbol{G}} = (\boldsymbol{S}'\boldsymbol{A}'\boldsymbol{W}^{-1}\boldsymbol{A}\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{A}'\boldsymbol{W}^{-1}$. Given fixed $\boldsymbol{S}$ and estimation of $\boldsymbol{W}$, $\bar{\boldsymbol{G}}$ is entirely determined by $\boldsymbol{A}$. Thus, when the jth diagonal element of $\boldsymbol{A}$ is zero, the jth column of $\bar{\boldsymbol{G}}$ becomes entirely composed of zeros. Therefore, the optimization problem can be reduced to an integer quadratic programming problem where all of the variables are restricted to being integers:

$$\begin{aligned} \min_{\boldsymbol{A}} \quad & \frac{1}{2}\left(\hat{\boldsymbol{y}} - \boldsymbol{S}\bar{\boldsymbol{G}}\hat{\boldsymbol{y}}\right)'\boldsymbol{W}^{-1}\left(\hat{\boldsymbol{y}} - \boldsymbol{S}\bar{\boldsymbol{G}}\hat{\boldsymbol{y}}\right) + \lambda_0\sum_{j=1}^{n}\boldsymbol{A}_{jj} \\ \text{s.t.} \quad & \bar{\boldsymbol{G}} = (\boldsymbol{S}'\boldsymbol{A}'\boldsymbol{W}^{-1}\boldsymbol{A}\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{A}'\boldsymbol{W}^{-1} \qquad \text{and} \qquad \bar{\boldsymbol{G}}\boldsymbol{S} = \boldsymbol{I}, \end{aligned}$$

where $\lambda_0 \geq 0$ controls the number of nonzero diagonal elements in $\boldsymbol{A}$, consequently affecting the number of nonzero columns (i.e., selected time series) in $\boldsymbol{G}$. We call this reconciliation method the *parsimonious method with L_0 regularization* due to its appeal in reducing the number of parameters. In the results that follow, we label the **Parsimonious** method differently based on various estimators for $\boldsymbol{W}$, referring to them as **OLS-parsim**, **WLSs-parsim**, **WLSv-parsim**, **MinT-parsim**, and **MinTs-parsim**, respectively.

In the Parsimonious method, the unknown matrix $\boldsymbol{A}$ is restricted to elements of 0 or 1. Thus the L_2 penalty is excluded from its optimization problem, unlike in the Subset method. We note that achieving time series selection

with this optimization problem can be challenging, as identifying a solution $\hat{\boldsymbol{A}}$ with some zero diagonal elements while satisfying both the MinT solution and the constraint may be difficult. Thus, the resulting solution tends to be dense and may not have zero columns.

To ensure the invertibility of $\boldsymbol{S}'\boldsymbol{A}'\boldsymbol{W}^{-1}\boldsymbol{A}\boldsymbol{S}$, and make the problem compatible with Gurobi, we reformulate the problem as

$$\begin{aligned}
\min_{\boldsymbol{A},\bar{\boldsymbol{G}},\boldsymbol{C},\check{\boldsymbol{e}},\boldsymbol{z}} \quad & \frac{1}{2}\check{\boldsymbol{e}}'\boldsymbol{W}^{-1}\check{\boldsymbol{e}} + \lambda_0 \sum_{j=1}^{n} z_j \\
\text{s.t.} \quad & \bar{\boldsymbol{G}}\boldsymbol{S} = \boldsymbol{I} \\
& \hat{\boldsymbol{y}} - \left(\hat{\boldsymbol{y}}' \otimes \boldsymbol{S}\right) \operatorname{vec}(\bar{\boldsymbol{G}}) = \check{\boldsymbol{e}} \\
& \bar{\boldsymbol{G}}\boldsymbol{A}\boldsymbol{S} = \boldsymbol{I} \\
& \bar{\boldsymbol{G}} = \boldsymbol{C}\boldsymbol{S}'\boldsymbol{A}'\boldsymbol{W}^{-1} \\
& z_j \in \{0,1\}, \quad j \in [n].
\end{aligned} \tag{9}$$

Parameter tuning. Similar to the setup in the group best-subset selection, we select the tuning parameter, λ_0, by minimizing the sum of squared reconciled forecast errors on a truncated training set, comprising only the $\max\{h,s\}$ observations that occurred prior to the forecast origin. Let $\lambda_0^1 = \frac{1}{2}\left(\hat{\boldsymbol{y}} - \tilde{\boldsymbol{y}}^{\text{bench}}\right)' \boldsymbol{W}^{-1} \left(\hat{\boldsymbol{y}} - \tilde{\boldsymbol{y}}^{\text{bench}}\right)$, and $\lambda_0^k = 0.0001\lambda_0^1$, the collection of candidate values for λ_0 we consider is $\{\lambda_0^1, \ldots, \lambda_0^k, 0\}$, where $\lambda_0^j = \lambda_0^1 \left(\lambda_0^k/\lambda_0^1\right)^{(j-1)/(k-1)}$ for $j \in [k]$.

Computation details. We employ Gurobi to solve Equation (9) by configuring parameters such as MIPGap = 0.001 and TimeLimit = 600 seconds for problems with $p > 1000$.

Group lasso method

Yuan & Lin (2006) introduced the group lasso method, which extends lasso to situations with a grouped structure among variables. Similar to lasso, group lasso induces sparsity, but at the group level, leading to more interpretable models by reducing the number of non-zero groups of coefficients. Lounici et al. (2011) demonstrated that group lasso enhances prediction and estimation properties compared to the traditional lasso method. The statistical properties of the group-lasso estimator have been extensively studied in the literature (e.g., Nardi & Rinaldo 2008).

When the problem of forecast reconciliation with time series selection is reframed as a least squares problem, our goal is to perform group-wise variable selection. Specifically, the unknown parameter $\operatorname{vec}(\boldsymbol{G})$ possesses an inherent grouping structure, with each group corresponding to a single column of $\boldsymbol{G}$, each of size n_b. Thus, we consider *a group lasso problem under the unbiasedness assumption* given by

$$\begin{aligned}
\min_{\boldsymbol{G}} \quad & \frac{1}{2}\left(\hat{\boldsymbol{y}} - \left(\hat{\boldsymbol{y}}' \otimes \boldsymbol{S}\right) \operatorname{vec}(\boldsymbol{G})\right)' \boldsymbol{W}^{-1} \left(\hat{\boldsymbol{y}} - \left(\hat{\boldsymbol{y}}' \otimes \boldsymbol{S}\right) \operatorname{vec}(\boldsymbol{G})\right) + \lambda \sum_{j=1}^{n} w_j \left\|\boldsymbol{G}_{\cdot j}\right\|_2 \\
\text{s.t.} \quad & \left(\boldsymbol{S}' \otimes \boldsymbol{I}_{n_b}\right) \operatorname{vec}(\boldsymbol{G}) = \operatorname{vec}\left(\boldsymbol{I}_{n_b}\right),
\end{aligned} \tag{10}$$

where $\lambda \geq 0$ is a tuning parameter, $w_j \geq 0$ is the penalty weight assigned in $\boldsymbol{G}_{\cdot j}$ to make the model more flexible,

and the second term in the objective is the penalty function that is intermediate between the L_1-penalty that is used in the lasso and the L_2-penalty that is used in ridge regression. In the results that follow, we label the **Lasso** method based on various estimators for $\boldsymbol{W}$, referring to them as **OLS-lasso**, **WLSs-lasso**, **WLSv-lasso**, **MinT-lasso**, and **MinTs-lasso**, respectively.

Next, we present the second order cone programming (SOCP) formulation for the group lasso based estimators:

$$\begin{aligned} \min_{\text{vec}(\boldsymbol{G}),\check{\boldsymbol{e}},\boldsymbol{g}^+} \; & \frac{1}{2}\check{\boldsymbol{e}}'\boldsymbol{W}_h^{-1}\check{\boldsymbol{e}} + \lambda \sum_{j=1}^{n} w_j c_j \\ \text{s.t.} \quad & \left(\boldsymbol{S}' \otimes \boldsymbol{I}_{n_b}\right) \text{vec}(\boldsymbol{G}) = \text{vec}\left(\boldsymbol{I}_{n_b}\right) \\ & \hat{\boldsymbol{y}} - \left(\hat{\boldsymbol{y}}' \otimes \boldsymbol{S}\right) \text{vec}(\boldsymbol{G}) = \check{\boldsymbol{e}} \\ & c_j = \sqrt{\sum_{i=1}^{n_b} g_{i+(j-1)n_b}^{+2}}, \quad j \in [n]. \end{aligned} \tag{11}$$

Equation (11) includes additional auxiliary variables $c_j \in \mathbb{R}_{\geq 0}$, $j \in [n]$, and second order cone constraints, $c_j = \sqrt{\sum_{i=1}^{n_b} g_{i+(j-1)n_b}^{+2}}$ for $j \in [n]$.

Compared to the previous methods, group lasso is computationally friendlier. Nonetheless, Hazimeh et al. (2023) demonstrated, both empirically and theoretically, that the group L_0-regularized method exhibits advantages over its group lasso counterpart across a range of regimes. Group lasso can either be highly dense or possess non-zero coefficients that are overly shrunk. This issue becomes more pronounced when the groups are correlated with each other, as group lasso tends to retain all correlated groups instead of seeking a more concise model.

Penalty weights and parameter tuning. In the context of group lasso, the default choice for the penalty weight, w_j, is $\sqrt{p_j}$, where p_j is the size of each group (in our case, $p_j = n_b$). In our experiments, we allocate different penalty weights to each group using $w_j = 1/\|\boldsymbol{G}_{\cdot j}^{\text{bench}}\|_2$, which allows us to account for variations in scale across different time series in the structure.

We compute the group lasso over $k+1$ values of the tuning parameter λ, and select the parameter by optimizing the sum of squared reconciled forecast errors on a truncated training set, consisting only of $\max\{h,s\}$ observations occurred prior to the forecast origin. The collection of candidate values for λ is $\{\lambda^1,\ldots,\lambda^k,0\}$, where $\lambda^1 = \max_{j=1,\ldots,n} \left\| -\left((\hat{\boldsymbol{y}}' \otimes \boldsymbol{S})_{\cdot j^*}\right)' \boldsymbol{W}^{-1}\hat{\boldsymbol{y}}\right\|_2 / w_j$, $\lambda^k = 0.0001\lambda^1$, and $\lambda^j = \lambda^1(\lambda^k/\lambda^1)^{(j-1)/(k-1)}$ for $j \in [k]$.

Proposition 3. *Relaxing the constraint $\boldsymbol{G_h S} = \boldsymbol{I}_{n_b}$, we define λ^1 as the smallest λ value such that all predictors in the group lasso problem have zero coefficients. Then we have*

$$\lambda^1 = \max_{j=1,\ldots,n} \left\| -\left((\hat{\boldsymbol{y}}' \otimes \boldsymbol{S})_{\cdot j^*}\right)' \boldsymbol{W}^{-1}\hat{\boldsymbol{y}}\right\|_2 / w_j,$$

where j^* denotes the column index of $\hat{\boldsymbol{y}}' \otimes \boldsymbol{S}$ that corresponds to the jth column of $\boldsymbol{G}$.

Proof. See Appendix A, supplementary materials. □

Computation details. Due to the incorporation of the constraint, we can not directly use some open-source packages designed for group lasso. Consequently, we employ Gurobi to solve the SOCP problem, configuring it by

setting OptimalityTol = 0.0001.

3.2 Series selection relaxing the unbiasedness assumption

In this section, we relax the unbiasedness assumption, and introduce a reconciliation method with selection that relies on in-sample observations and fitted values. Let $\boldsymbol{Y} \in \mathbb{R}^{T\times n}$ denote a matrix comprising observations from all time series on the training set in the structure, and $\hat{\boldsymbol{Y}} \in \mathbb{R}^{T\times n}$ denote a matrix of in-sample one-step-ahead forecasts (i.e., fitted values) for all time series. The proposed *empirical group lasso* method considers the optimization problem

$$\min_{\boldsymbol{G}} \quad \frac{1}{2T}\left\|\boldsymbol{Y}-\hat{\boldsymbol{Y}}\boldsymbol{G}'\boldsymbol{S}'\right\|_F^2+\lambda\sum_{j=1}^{n} w_j\left\|\boldsymbol{G}_{\cdot j}\right\|_2,$$

where $\lambda \geq 0$ is a tuning parameter, $w_j \geq 0$ is the penalty weight assigned in $\boldsymbol{G}_{\cdot j}$ to make a more flexible model. We rewrite the problem as

$$\min_{\mathrm{vec}(\boldsymbol{G})} \quad \frac{1}{2T}\left\|\mathrm{vec}(\boldsymbol{Y})-(\boldsymbol{S}\otimes\hat{\boldsymbol{Y}})\,\mathrm{vec}\left(\boldsymbol{G}'\right)\right\|_2^2+\lambda\sum_{j=1}^{n} w_j\left\|\boldsymbol{G}_{\cdot j}\right\|_2,$$

which becomes a standard group lasso problem, with $\mathrm{vec}(\boldsymbol{Y})$ serving as the dependent variable and $\boldsymbol{S}\otimes\hat{\boldsymbol{Y}}$ as the covariate matrix. We denote this as **Elasso** in the results that follow.

Unlike the methods introduced in Section 3.1, Elasso relaxes the unbiasedness conditions on both base and reconciled forecasts and operates as an "in-sample" method because $\hat{\boldsymbol{Y}}$ are predictions in the form of fitted values (i.e., $\boldsymbol{Y}$ is in training data when base forecasts are computed). Thus Elasso aims to directly minimize the mean squared reconciled forecast errors, as described in Equation (2), rather than focusing solely on the variance term. This clarifies two points: (1) why the Elasso method omits the $\boldsymbol{GS}=\boldsymbol{I}$ constraint, which ensures the unbiasedness of reconciled forecasts when base forecasts are unbiased, and (2) why the Elasso method does not use a $\boldsymbol{W}$ matrix, which is introduced when deriving from $\mathrm{Var}(\tilde{\boldsymbol{e}}_{T+h|T})$ under the constraint.

We note that both the "in-sample" Elasso and EMinT methods require only one round of model training. Similarly, the Subset, Parsimonious, and Lasso methods also need only one round of training and forecasting, despite their reliance on genuine out-of-sample forecasts. In contrast, the "out-of-sample" RERM and ERM methods use an iterative approach with expanding windows for out-of-sample forecasts, demanding extensive rounds of model training and significant computation time. To ensure a fair comparison, we exclude RERM and ERM methods from simulation studies and empirical results.

Relaxing the unbiasedness assumption may result in fewer non-zero column entries in the $\boldsymbol{G}$ solution than the number of series at the bottom level. This differs from constrained reconciliation methods detailed in Section 3.1. In an extreme scenario, the solution may take the form of a top-down $\boldsymbol{G}_{TD}=[\boldsymbol{p}\mid\boldsymbol{O}_{n_b\times(n-1)}]$, where $\boldsymbol{p}=(p_1,p_2,\ldots,p_{n_b})$ is a proportionality vector obtained based on in-sample reconciled forecast errors, and $\boldsymbol{O}_{n_b\times(n-1)}$ is a matrix of zeros of order $n_b\times(n-1)$. Only the column corresponding to the top level (most aggregated level) of the $\boldsymbol{G}$ matrix retains non-zero values.

We also explored the empirical version of group best-subset selection with ridge regularization and the parsi-

monious method with L_0 regularization in which we omit the unbiasedness assumption. It is worth mentioning that Hazimeh et al. (2023) presented an algorithmic framework for formulating the group L_0 problem with ridge regularization and provided the **L0Group** Python package for implementation. However, our experiments showed that this algorithm can not terminate within five hours for typical instances with $p \sim 10^4$. Therefore, in this paper, we only present the empirical group lasso method for series selection without the unbiasedness assumption.

Penalty weights and parameter tuning. Similar to the setup in the group lasso method, we assign different penalty weights to each group by setting $w_j = 1/\|\boldsymbol{G}_{\cdot j}^{\text{OLS}}\|_2$, where $\boldsymbol{G}^{\text{OLS}}$ is the solution obtained by the OLS estimator of $\boldsymbol{W}$. Given a fixed tuning parameter, we solve the target optimization problem by considering the initial $T - T_v$ observations, where $T_v = \max\{h, s\}$ for seasonal time series and $T_v = \lfloor \frac{1}{10} T \rfloor$ for non-seasonal time series. Then the tuning parameter, λ, is selected by minimizing the sum of squared reconciled forecast errors on a truncated training set, comprising only the T_v observations closest to the forecast origin. Specifically, for λ values, we consider $\{\lambda^1, \ldots, \lambda^k, 0\}$, where $\lambda^1 = \max_{j=1,\ldots,n} \left\| -\frac{1}{T} \left(\left(\boldsymbol{S} \otimes \hat{\boldsymbol{Y}}\right)_{j*} \right)' \text{vec}(\boldsymbol{Y}) \right\|_2 / w_j$, $\lambda^k = 0.0001\lambda^1$, and $\lambda^j = \lambda^1 \left(\lambda^k/\lambda^1\right)^{(j-1)/(k-1)}$ for $j \in [k]$. Following the derivation in the proof of Proposition 3, λ^1 is the smallest λ value such that all predictors in the empirical group lasso problem have zero coefficients, i.e., $\boldsymbol{G} = \boldsymbol{O}$. Note that we need to resolve the optimization problem based on the whole training set by using the optimal tuning parameter to obtain the final solution.

define what is N. XQ — It should T.

Computation details. While there are open-source packages available for solving group lasso problems, they are still relatively slow when handling large instances. For example, given a specific value for the parameter, λ, our experiments observed that, using the **gglasso** R package, we can not obtain a solution within five hours for typical instances with $p \sim 10^4$. Instead, we use Gurobi to solve the problem using the SOCP formulation for the empirical group lasso which aligns with Equation (11) but omits the constraint.

4 Monte Carlo simulations

To assess the proposed reconciliation methods with selection outlined in Section 3, we carry out two simulations with different designs. Both simulations consider a hierarchy comprising two levels of aggregation, as shown in Figure 1. Bottom-level series are first generated and then summed to obtain the aggregated series at higher levels.

4.1 Setup 1: Exploring the effect of model misspecification

We follow a simulation setup similar to Wickramasuriya et al. (2019), assuming that the bottom-level time series are generated using the basic structural time series model

$$\boldsymbol{b}_t = \boldsymbol{\mu}_t + \boldsymbol{\gamma}_t + \boldsymbol{\eta}_t,$$

where $\boldsymbol{\mu}_t$ and $\boldsymbol{\gamma}_t$ are trend and seasonal components defined by

$$\boldsymbol{\mu}_t = \boldsymbol{\mu}_{t-1} + \boldsymbol{v}_t + \boldsymbol{\varrho}_t, \qquad \boldsymbol{\varrho}_t \sim \mathcal{N}\left(\boldsymbol{0}, \sigma_\rho^2 \boldsymbol{I}_4\right),$$

$$\boldsymbol{v}_t = \boldsymbol{v}_{t-1} + \boldsymbol{\zeta}_t, \qquad \boldsymbol{\zeta}_t \sim \mathcal{N}\left(\mathbf{0}, \sigma_\zeta^2 \boldsymbol{I}_4\right),$$
$$\boldsymbol{\gamma}_t = -\sum_{i=1}^{s-1} \boldsymbol{\gamma}_{t-i} + \boldsymbol{\omega}_t, \qquad \boldsymbol{\omega}_t \sim \mathcal{N}\left(\mathbf{0}, \sigma_\omega^2 \boldsymbol{I}_4\right),$$

$\boldsymbol{\varrho}_t$, $\boldsymbol{\zeta}_t$, and $\boldsymbol{\omega}_t$ are error terms independent of each other and over time, and $\boldsymbol{\eta}_t$ is generated independently from an ARIMA$(p,0,q)$ process, where p and q take values of 0 or 1 with equal probability. Coefficients in the ARIMA process are randomly sampled from a uniform distribution $U(0.5,0.7)$, and the contemporaneous error covariance matrix is given by

$$\begin{bmatrix} 5 & 3 & 2 & 1 \\ 3 & 4 & 2 & 1 \\ 2 & 2 & 5 & 3 \\ 1 & 1 & 3 & 4 \end{bmatrix},$$

which enables correlations among time series in a hierarchical structure.

We set $s = 4$ for quarterly data with error variances $\sigma_\rho^2 = 2$, $\sigma_\zeta^2 = 0.007$, and $\sigma_\omega^2 = 7$. Initial values for $\boldsymbol{\mu}_0$, $\boldsymbol{v}_0$, $\boldsymbol{\gamma}_0$, $\boldsymbol{\gamma}_1$, and $\boldsymbol{\gamma}_2$ are generated independently from a multivariate normal distribution with zero mean and identity covariance matrix. Each bottom-level series has 180 observations, with the last $h = 16$ forming the test set. The bottom-level series are then aggregated to form higher-level data. This process is repeated 500 times.

We use ETS models to generate base forecasts for each hierarchy using default settings from the **forecast** R package (Hyndman et al. 2023). To introduce model misspecification, we degrade the performance of series A at the middle level by applying a 1.5 multiplier to its in-sample fitted values and out-of-sample forecasts. We also repeated the analysis with model misspecification for series AA at the bottom level and series Total at the top level, respectively. The results for these two scenarios are similar and reported in Appendix B.

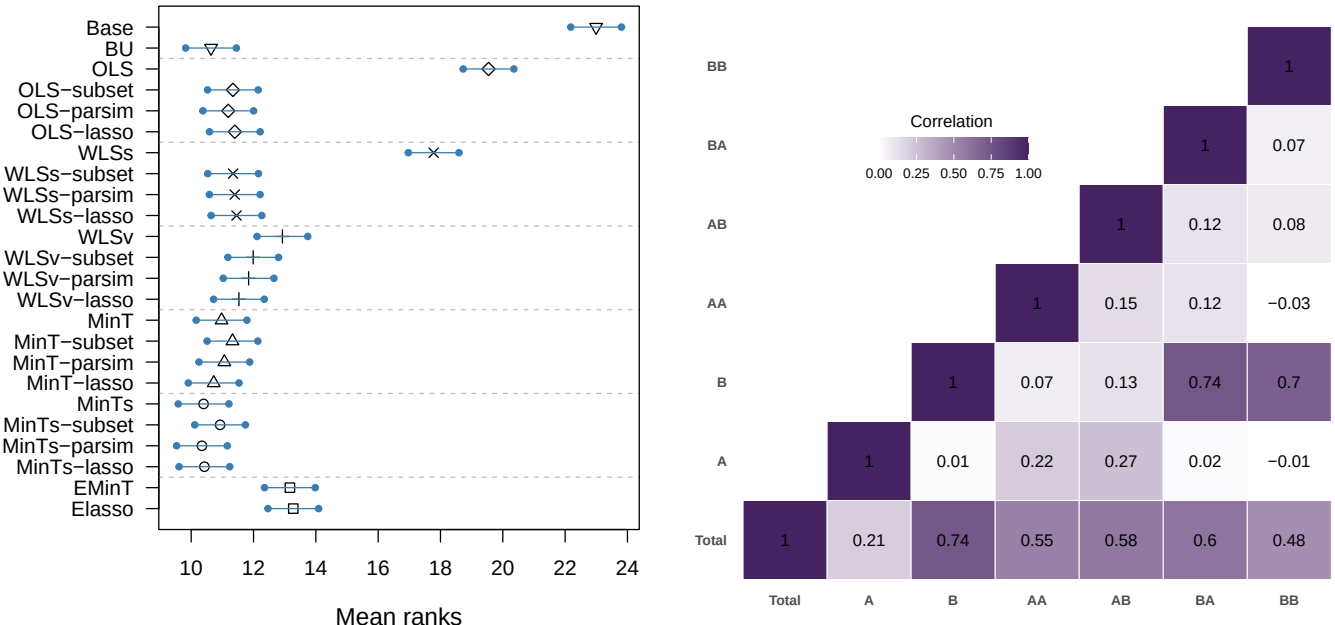
As elaborated following the introduction of each method in Section 3, we conduct parameter tuning by minimizing the sum of squared reconciled forecast errors on a truncated training set. We experimented with directly using the estimated weighting matrix $\boldsymbol{G}$ derived from various candidate parameter values for forecast reconciliation. The results indicated significant variability in reconciliation outcomes depending on the parameter values. However, the optimal parameters identified through our tuning method consistently yielded stable and well-performing reconciliation results.

Table 2 summarizes the percentage improvement in average root mean squared error (RMSE) for each level and the overall structure (denoted as Average), relative to the RMSE of the base forecasts. The BU row shows results for the "bottom-up" approach, where base forecasts at the bottom level are aggregated to generate higher-level forecasts. Multiple comparisons with the best (MCB) test at a 95% confidence level, shown in Figure 2a, assesses the statistical significance of differences among the methods in Table 2. With MCB, the ranking performances are statistically different if the intervals of two methods do not overlap.

We find that the BU method performs the best overall, which is unsurprising given that the aggregated series A is deteriorated in this setup, and the BU method does not use its information. WLSv, MinT, and MinTs also perform well, with no significant difference from BU, as they account for the in-sample covariance of base forecast errors,

Table 2: *Performance of proposed (gray-shaded) and benchmark methods for simulation Setup 1. The Base row shows average RMSE of base forecasts, while entries below show RMSE percentage decrease (negative) or increase (positive) for reconciliation methods. Blue entries highlight the best-performing methods; bold entries indicate proposed methods outperforming bechmarks.*

	Top				Middle				Bottom				Average			
Method	h=1	1–4	1–8	1–16	h=1	1–4	1–8	1–16	h=1	1–4	1–8	1–16	h=1	1–4	1–8	1–16
Base	9.6	10.7	12.6	15.6	12.1	14.4	15.3	17.0	4.2	4.9	5.9	7.5	7.2	8.5	9.6	11.4
BU	**–1.0**	0.4	0.6	0.7	**–47.7**	**–49.6**	**–43.6**	**–36.2**	**0.0**	**0.0**	**0.0**	**0.0**	**–23.0**	**–24.0**	**–19.8**	**–15.3**
OLS	8.5	13.9	10.4	7.6	–28.2	–29.4	–26.7	–23.1	22.9	23.9	17.0	11.3	–4.2	–3.8	–4.2	–4.1
OLS-subset	**–0.5**	**0.5**	**0.6**	**0.7**	**–46.3**	**–49.0**	**–43.2**	**–35.9**	**2.2**	**1.0**	**0.7**	**0.5**	**–21.5**	**–23.4**	**–19.4**	**–15.0**
OLS-parsim	**–0.5**	**0.5**	**0.6**	**0.6**	**–46.5**	**–49.0**	**–43.2**	**–36.0**	**2.2**	**1.2**	**0.7**	**0.5**	**–21.6**	**–23.4**	**–19.4**	**–15.0**
OLS-lasso	**–0.2**	**1.5**	**1.4**	**1.3**	**–46.9**	**–48.9**	**–43.1**	**–35.8**	**0.9**	**0.8**	**0.5**	**0.3**	**–22.1**	**–23.3**	**–19.3**	**–14.9**
WLSs	12.1	18.6	14.0	10.2	–34.4	–35.1	–31.7	–26.9	15.6	17.0	12.0	8.0	–9.0	–8.0	–7.6	–6.5
WLSs-subset	**–0.1**	**1.2**	**1.1**	**1.1**	**–46.7**	**–48.8**	**–43.1**	**–35.8**	**1.5**	**1.1**	**0.8**	**0.6**	**–21.8**	**–23.2**	**–19.2**	**–14.8**
WLSs-parsim	**0.0**	**1.2**	**1.0**	**0.9**	**–46.5**	**–48.8**	**–43.1**	**–35.9**	**1.7**	**1.3**	**0.9**	**0.6**	**–21.6**	**–23.1**	**–19.2**	**–14.9**
WLSs-lasso	**–0.1**	**1.5**	**1.5**	**1.3**	**–46.7**	**–48.9**	**–43.1**	**–35.8**	**0.9**	**0.8**	**0.5**	**0.3**	**–22.0**	**–23.2**	**–19.3**	**–14.9**
WLSv	–0.8	2.3	1.8	1.6	–46.3	–47.9	–42.3	–35.2	1.6	1.9	1.2	0.8	–21.7	–22.2	–18.6	–14.4
WLSv-subset	–0.7	**1.3**	**1.4**	**1.4**	**–46.9**	**–48.7**	**–42.9**	**–35.6**	**1.0**	**1.0**	**0.8**	**0.6**	**–22.2**	**–23.1**	**–19.1**	**–14.7**
WLSv-parsim	–0.4	**1.5**	**1.4**	**1.2**	**–46.9**	**–48.6**	**–42.8**	**–35.6**	**0.9**	**1.2**	**0.9**	**0.7**	**–22.2**	**–23.0**	**–19.0**	**–14.7**
WLSv-lasso	–0.6	**1.3**	**1.3**	**1.3**	**–47.2**	**–48.9**	**–43.0**	**–35.7**	**0.6**	**0.8**	**0.5**	**0.4**	**–22.4**	**–23.3**	**–19.2**	**–14.8**
MinT	0.2	0.5	0.6	0.5	–47.5	–49.4	–43.5	–36.1	1.1	0.5	0.3	0.1	–22.3	–23.7	–19.6	**–15.3**
MinT-subset	**–0.1**	0.8	0.9	0.9	–46.9	–49.1	–43.3	–36.0	1.7	0.9	0.5	0.3	–21.9	–23.4	–19.4	–15.1
MinT-parsim	0.2	0.5	0.6	0.5	–47.5	–49.4	–43.5	–36.1	1.1	0.5	0.3	0.1	–22.3	–23.7	–19.6	**–15.3**
MinT-lasso	**–0.3**	**0.3**	0.6	0.5	**–47.6**	–49.4	–43.5	–36.1	**0.8**	**0.3**	**0.2**	0.1	**–22.5**	**–23.9**	**–19.7**	**–15.3**
MinTs	–0.3	0.3	**0.4**	**0.4**	–47.6	–49.5	**–43.6**	**–36.2**	0.7	0.2	0.1	**0.0**	–22.6	–23.9	**–19.8**	**–15.3**
MinTs-subset	**–0.8**	0.5	0.8	0.8	–47.2	–49.2	–43.4	–36.0	1.0	0.7	0.4	0.3	–22.3	–23.6	–19.5	–15.1
MinTs-parsim	–0.3	0.3	**0.4**	**0.4**	–47.6	–49.5	**–43.6**	**–36.2**	0.7	0.2	0.1	**0.0**	–22.6	–23.9	**–19.8**	**–15.3**
MinTs-lasso	**–0.9**	**0.2**	0.5	0.5	**–47.7**	–49.5	**–43.6**	**–36.2**	**0.5**	0.2	0.1	0.1	**–22.8**	**–24.0**	**–19.8**	**–15.3**
EMinT	2.2	2.9	2.5	1.7	–46.2	–48.1	–42.4	–35.3	3.6	2.9	2.0	1.1	–20.5	–21.9	–18.2	–14.3
Elasso	**1.4**	**2.7**	**2.4**	**1.6**	**–46.4**	**–48.2**	–42.4	**–35.4**	**3.1**	3.2	2.1	1.2	**–20.9**	–21.9	–18.2	–14.3



(a) *MCB test for hierarchy*

(b) *Correlation between base forecast errors*

Figure 2: *MCB test result and correlation matrix heatmap for simulation Setup 1.*

allowing for larger adjustments in reconciliation for forecasts with higher error variance. However, OLS and WLSs significantly underperform the other benchmark methods. Our proposed methods show significant improvements when using OLS and WLSs estimators of $\boldsymbol{W}$, which do not consider in-sample covariance. A key advantage of our forecast reconciliation methods with selection is their ability to reduce differences introduced by using different

estimators of $\boldsymbol{W}$, thereby mitigating the risk of estimator selection. We can align the forecast accuracy achieved using different estimators, making them approach the best results we can obtain. By relaxing the unbiasedness assumption, Elasso performs similarly to EMinT overall, with slight improvements at aggregated levels, which is typically the primary concern for practitioners.

Table 3: *Proportion of time series being selected using proposed reconciliation methods for simulation Setup 1. RMSSE values of base forecasts for each series are in parentheses. The last column displays a stacked barplot of the total selected series from 500 hierarchies, with darker sub-bars indicating higher counts.*

(RMSSE)	Total (1.48)	A (3.25)	B (1.55)	AA (1.57)	AB (1.62)	BA (1.62)	BB (1.57)	Summary
OLS-subset	0.55	0.04	0.41	0.74	0.78	0.79	0.83	
OLS-parsim	0.61	0.04	0.52	0.75	0.69	0.69	0.83	
OLS-lasso	0.04	0.35	0.02	1.00	1.00	1.00	1.00	
WLSs-subset	0.45	0.06	0.36	0.81	0.84	0.81	0.87	
WLSs-parsim	0.61	0.06	0.48	0.75	0.71	0.73	0.84	
WLSs-lasso	0.02	0.33	0.02	1.00	1.00	1.00	1.00	
WLSv-subset	0.54	0.29	0.46	0.91	0.94	0.86	0.89	
WLSv-parsim	0.59	0.32	0.53	0.82	0.86	0.77	0.86	
WLSv-lasso	0.27	0.42	0.26	1.00	1.00	1.00	1.00	
MinT-subset	0.69	0.64	0.66	0.95	0.96	0.90	0.90	
MinT-parsim	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinT-lasso	0.82	0.74	0.83	1.00	0.99	0.97	0.97	
MinTs-subset	0.62	0.63	0.58	0.95	0.96	0.90	0.86	
MinTs-parsim	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinTs-lasso	0.68	0.75	0.68	1.00	1.00	1.00	1.00	
Elasso	0.78	0.95	0.68	1.00	1.00	1.00	1.00	

In addition, we report the proportion of time series being selected by our proposed methods across 500 simulation instances, along with the average root mean squared scaled error (RMSSE) for base forecasts of each series, as shown in Table 3. Our methods select fewer time series, while enhancing forecast accuracy compared to benchmarks. Subset methods, in particular, select fewer time series than Parsimonious and Lasso, which aligns with our expectations that the Parsimonious and Lasso methods tend to produce dense estimates. Notably, the misspecified series A shows higher RMSSE and is selected least frequently, except in cases where all bottom-level series are retained so series A leads to only slight deterioration. Series Total and B are also selected less often maybe due to the high correlation of their forecast errors with other series, as shown in Figure 2b.

4.2 Setup 2: Exploring the effect of correlation

We now simulate a hierarchical structure with correlated series, using a similar simulation to Wickramasuriya (2021), and the same hierarchical structure as shown in Figure 1. We use a stationary VAR(1) data generating process for the time series at the bottom level:

$$\boldsymbol{b}_t = \boldsymbol{c} + \begin{bmatrix} \boldsymbol{A}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A}_2 \end{bmatrix} \boldsymbol{b}_{t-1} + \boldsymbol{\varepsilon}_t,$$

where $\boldsymbol{c}$ is a constant vector with all entries set to 1, $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are 2×2 matrices with eigenvalues $z_{1,2} = 0.6[\cos(\pi/3) \pm i\sin(\pi/3)]$ and $z_{3,4} = 0.9[\cos(\pi/6) \pm i\sin(\pi/6)]$, respectively, $\boldsymbol{\varepsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$, where

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_1 & 0 \\ 0 & \boldsymbol{\Sigma}_2 \end{bmatrix}, \quad \text{and} \quad \boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \begin{bmatrix} 2 & \sqrt{6}\rho \\ \sqrt{6}\rho & 3 \end{bmatrix},$$

and $\rho \in \{0, \pm 0.2, \pm 0.4, \pm 0.6, \pm 0.8\}$ controls the error correlation in the simulated hierarchy.

For each time series at the bottom level, we generate a total of 101 observations, with the last observation serving as the test set, i.e., $T = 100$ and $h = 1$. Once again, the data at the higher levels are obtained by aggregating the bottom-level series. The process is repeated 500 times for each candidate correlation, ρ.

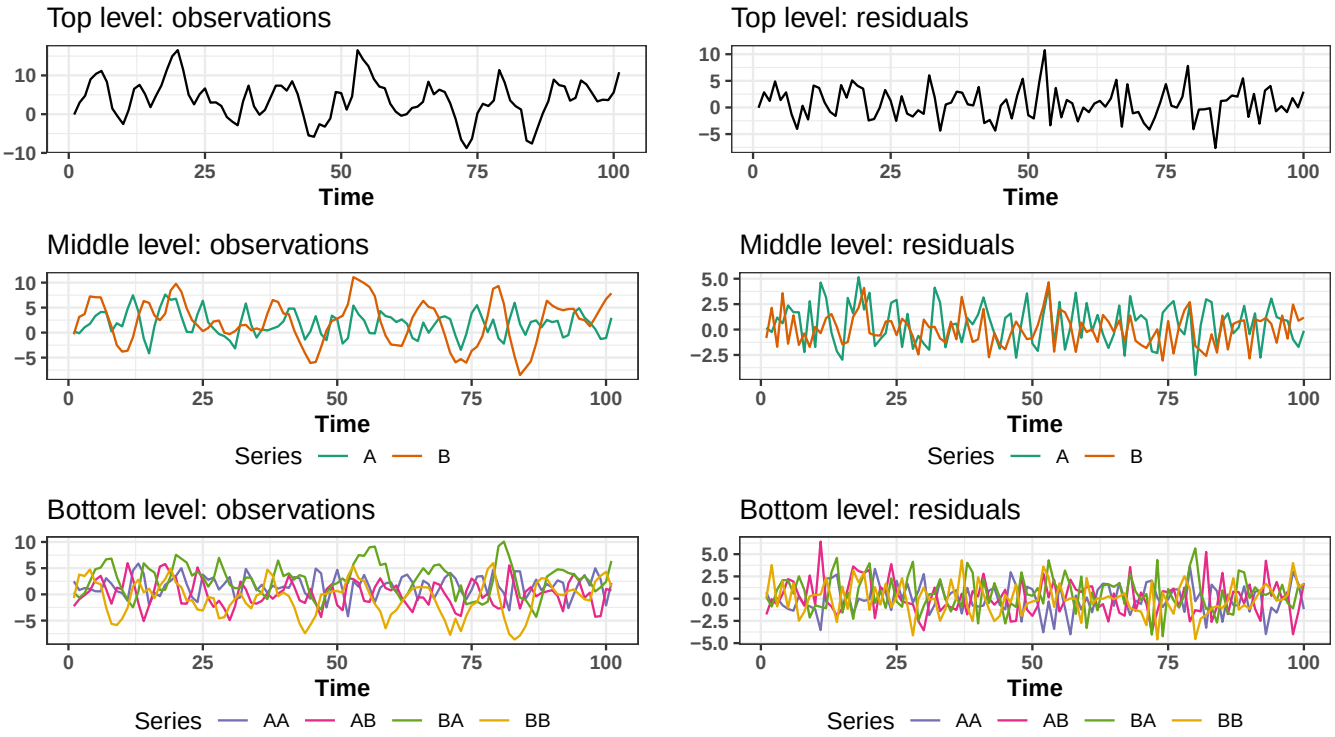


Figure 3: *An example hierarchical time series and its in-sample residuals in Setup 2.*

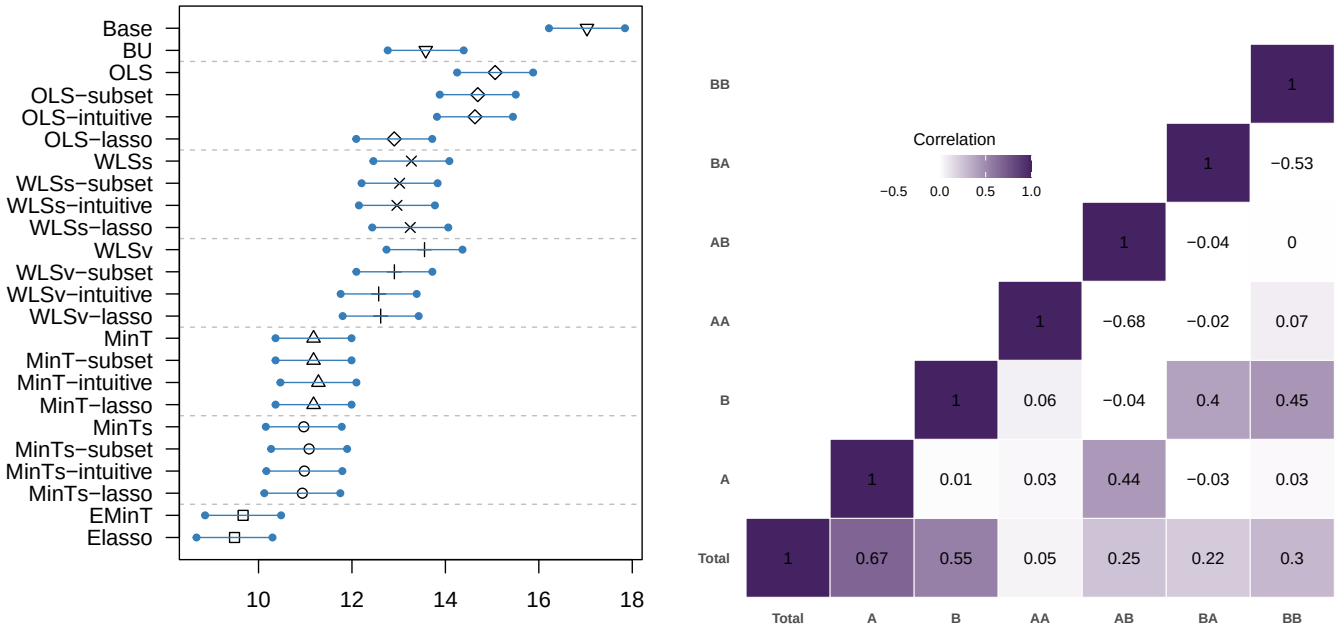
For each series, base forecasts are generated from ARMA models. We identify the best ARMA model using the automated algorithm implemented in the **forecast** R package (Hyndman & Khandakar 2008). Additionally, when fitting ARMA models for time series Total, A, and BA, we introduce a slight bias by omitting the constant term. Figure 3 presents an illustrative example of a simulated hierarchical time series. The left panels depict time plots for each series at different levels of the structure, while the right panels show the residuals obtained from forecasting each series using the fitted ARMA model. Notably, despite our omission of the constant term when fitting ARMA models to series Total, A, and BA, the residuals derived from the identified optimal models still exhibit fluctuations around zero and do not display significant deviations in comparison to the residuals from other series. This is because the influence of the constant term is minimal, i.e., it is much smaller compared to the data variability. Thus, it may be challenging to identify the "poor" base forecasts and exclude them from reconciliation in this setup.

Similarly to Section 4.1, we experimented with using the estimated weighting matrix obtained from different parameter values for forecast reconciliation. The results showed noticeable variability in the reconciliation outcomes depending on the parameter values. However, our tuning method consistently produced stable and well-performing reconciliation results.

The results across various error correlations are presented in Table 4, with the MCB test for a correlation of -0.8 shown in Figure 4a. The conclusion from the MCB test results for other error correlations are qualitatively

Table 4: *Performance of proposed (gray-shaded) and benchmark methods for simulation Setup 2. The Base row shows average RMSE of base forecasts, while entries below show RMSE percentage decrease (negative) or increase (positive) for reconciliation methods. Blue entries highlight the best-performing methods; bold entries indicate proposed methods outperforming bechmarks.*

	Top					Middle					Bottom					Average				
Method	ρ=–0.8	–0.4	0	0.4	0.8	ρ=–0.8	–0.4	0	0.4	0.8	ρ=–0.8	–0.4	0	0.4	0.8	ρ=–0.8	–0.4	0	0.4	0.8
Base	2.4	2.9	3.4	4.1	4.0	1.5	1.8	2.1	2.4	2.5	1.5	1.5	1.5	1.5	1.4	1.6	1.8	2.0	2.1	2.1
BU	–17.0	–9.0	–6.7	–7.0	–7.4	–6.8	0.4	4.8	5.7	2.8	0.0	0.0	0.0	0.0	0.0	–5.3	–1.9	–0.2	–0.1	–1.0
OLS	–11.0	–8.2	–7.7	–8.2	–8.0	–3.5	–0.7	3.1	2.5	0.8	0.7	–0.6	–2.0	–2.3	–2.1	–2.8	–2.4	–1.8	–2.4	–2.7
OLS-subset	**–11.4**	**–8.4**	**–8.1**	**–8.4**	**–8.8**	**–3.7**	–0.7	3.2	2.5	**0.4**	**0.3**	**–0.8**	–2.0	–1.7	**–2.6**	**–3.2**	**–2.5**	**–1.9**	–2.2	**–3.2**
OLS-parsim	**–11.6**	–8.0	**–7.8**	–8.0	**–8.4**	**–3.6**	–0.4	3.7	2.5	**0.3**	**0.6**	–0.2	–1.3	–0.4	–1.5	**–3.0**	–2.0	–1.3	–1.6	**–2.8**
OLS-lasso	**–19.2**	**–9.8**	–7.2	**–8.7**	**–8.2**	**–10.5**	**–1.7**	**2.9**	**2.4**	0.8	**–0.8**	**–0.8**	–1.6	–2.3	–2.1	**–7.1**	**–3.1**	–1.6	**–2.5**	**–2.8**
WLSs	–16.8	–11.1	–9.6	–10.4	–10.2	–8.1	–2.8	1.5	1.2	–0.4	–0.3	–1.1	–2.4	–2.9	–2.9	–5.7	–3.9	–3.0	–3.6	–4.0
WLSs-subset	**–17.3**	**–11.4**	**–9.9**	**–11.1**	**–10.8**	**–8.3**	–2.8	**1.4**	**0.7**	**–0.9**	**–0.7**	**–1.3**	–2.4	**–3.2**	**–3.3**	**–6.1**	**–4.0**	**–3.1**	**–4.1**	**–4.5**
WLSs-parsim	**–16.9**	**–11.5**	**–9.8**	–10.0	**–10.6**	**–8.5**	–2.8	**1.4**	1.5	**–0.7**	**–0.7**	**–1.2**	–2.3	–2.7	**–3.0**	**–6.1**	**–4.0**	–3.0	–3.3	**–4.3**
WLSs-lasso	**–18.3**	–11.1	–9.2	**–10.5**	–9.8	**–9.3**	–2.4	**1.4**	1.2	–0.1	**–0.8**	–1.0	–2.4	–2.9	–2.8	**–6.6**	–3.7	–2.9	**–3.7**	–3.7
WLSv	–16.5	–11.9	–10.0	–10.6	–10.6	–7.6	–3.4	0.9	1.1	–0.5	–0.5	–1.2	–2.3	–2.9	–3.0	–5.7	–4.3	–3.2	–3.7	–4.2
WLSv-subset	**–16.8**	**–12.1**	–9.8	**–10.8**	**–10.7**	**–7.8**	**–3.5**	1.1	1.2	**–1.0**	**–1.1**	**–1.3**	–2.2	–2.9	**–3.2**	**–6.1**	**–4.4**	–3.0	–3.7	**–4.4**
WLSv-parsim	**–17.6**	**–12.6**	**–10.1**	–10.5	–10.6	**–8.7**	**–3.8**	**0.7**	1.1	**–0.8**	**–1.9**	**–1.5**	–2.3	**–3.0**	–3.0	**–7.0**	**–4.7**	**–3.3**	–3.7	**–4.3**
WLSv-lasso	**–19.8**	–11.6	–9.7	–10.5	–10.6	**–10.5**	–3.0	1.2	1.2	–0.5	**–1.2**	–1.1	–2.2	–2.9	–3.0	**–7.5**	–4.1	–3.0	–3.7	–4.2
MinT	–25.4	–18.8	–12.4	**–15.3**	**–12.6**	–15.5	–7.0	0.0	–2.0	–2.0	–4.0	–4.6	–4.3	–5.8	–5.1	–11.4	–8.5	–5.0	–7.2	–6.0
MinT-subset	–25.4	–18.8	–12.4	**–15.3**	**–12.6**	–15.5	–7.0	0.0	–2.0	–2.0	–4.0	–4.6	–4.3	–5.8	–5.1	–11.4	–8.5	–5.0	–7.2	–6.0
MinT-parsim	–25.4	–18.8	–12.4	**–15.3**	**–12.6**	–15.5	–7.0	0.0	–2.0	–2.0	–4.0	–4.6	–4.3	–5.8	–5.1	–11.4	–8.5	–5.0	–7.2	–6.0
MinT-lasso	–25.4	–18.8	–12.4	**–15.3**	**–12.6**	–15.5	–7.0	0.0	–2.0	–2.0	–4.0	–4.6	–4.3	–5.8	–5.1	–11.4	–8.5	–5.0	–7.2	–6.0
MinTs	–25.4	–17.7	–12.1	–14.2	–12.5	–16.1	–6.8	–0.8	–1.6	**–2.4**	–4.0	–4.6	–4.9	–5.9	**–5.2**	–11.6	–8.2	–5.4	–6.8	**–6.2**
MinTs-subset	–25.2	–17.6	–12.1	–14.2	–12.5	–16.1	–6.8	–0.8	–1.6	**–2.4**	–3.9	–4.6	–4.9	–5.9	**–5.2**	–11.5	–8.2	–5.4	–6.8	**–6.2**
MinTs-parsim	–25.4	–17.7	–12.1	–14.2	–12.5	–16.1	–6.8	–0.8	–1.6	**–2.4**	–4.0	–4.6	–4.9	–5.9	**–5.2**	–11.6	–8.2	–5.4	–6.8	**–6.2**
MinTs-lasso	–25.4	–17.6	–12.1	–14.2	–12.5	–16.1	–6.7	–0.8	–1.6	**–2.4**	–4.0	–4.6	–4.9	–5.9	**–5.2**	–11.6	–8.2	–5.4	–6.8	**–6.2**
EMinT	**–31.2**	**–19.8**	**–12.5**	–14.1	–11.1	**–22.9**	**–10.9**	**–2.4**	**–3.2**	–1.0	**–7.4**	**–7.3**	**–6.9**	**–7.5**	–5.1	**–16.4**	**–11.2**	**–6.9**	**–7.9**	–5.3
Elasso	–31.0	–19.1	–11.1	–13.6	**–11.2**	–22.7	–9.7	–1.8	–2.4	**–1.7**	**–7.4**	–7.2	–6.1	–5.7	–3.5	–16.3	–10.6	–6.0	–6.8	–4.9



(a) *MCB test for hierarchy*

(b) *Correlation between base forecast errors*

Figure 4: *MCB test result and correlation matrix heatmap for simulation Setup 2, with error correlation being -0.8.*

similar, thus we report them in Appendix C. The results show that our proposed methods slightly outperform or are comparable to their respective benchmarks at all levels when using OLS, WLSs, and WLSv estimators. However, only the OLS-lasso method significantly outperforms the OLS benchmark. It is important to note the challenge of identifying the "poor" base forecasts in this simulation design, given that the omission of the constant term has minimal impact relative to the data variability. In addition, the MinT and MinTs methods perform especially

well, and our proposed methods provide similar results. This is attributed to the use of in-sample covariance by MinT and MinTs, which allows for large adjustments in reconciliation for base forecasts with high estimated error variance. Elasso performs the best across the whole hierarchy, though not significantly better than EMinT due to the identification challenge. We have also considered alternative error correlation values, $\rho = -0.6, -0.2, 0.2, 0.4$, for this simulation setting, but to save space, we do not present all results. The omitted results follow a similar pattern and are available upon request.

Table 5 presents the proportion of time series selected using our methods, with an error correlation of -0.8. The correlation matrix heatmap for forecast errors is displayed in Figure 4b. To save space, the results for other error correlations are similar and given in Appendix C. Despite the challenges in identifying poor-performing series in this setup, Table 5 shows that Subset and Parsimonious methods, using OLS, WLSs, and WLSv estimators, are able to exclude the series Total, A, and BA (which have higher RMSSE) in some instances. Notably, Total and A are selected less frequently than BA, likely due to their forecast errors being more correlated with other series, as shown in Figure 4b. Subset methods outperform Parsimonious methods in selection. Lasso methods typically select all bottom-level series, given their tendency to yield dense estimates, as discussed in Section 3.1. When addressing higher correlation values, as shown in Appendix C, our methods maintain the ability to exclude series that should be omitted in reconciliation. However, their efficacy diminishes in the presence of positive correlation, particularly in removing the bottom-level series BA. Hence, our methods are especially preferable when the error correlation within the structure is negative.

Not sure of last t
lines. XQ — Ha
now been rewritte

Table 5: *Proportion of time series selected using proposed reconciliation methods for simulation Setup 2, with error correlation being -0.8. RMSSE values of base forecasts for each series are in parentheses. The last column displays a stacked barplot of the total selected series from 500 hierarchies, with darker sub-bars indicating higher counts.*

(RMSSE)	Total (0.84)	A (0.87)	B (0.62)	AA (0.68)	AB (0.81)	BA (0.91)	BB (0.77)	Summary
OLS-subset	0.32	0.34	0.95	0.98	1	0.74	1.00	
OLS-parsim	0.58	0.52	0.93	0.97	1	0.61	0.97	
OLS-lasso	0.61	0.34	0.38	1.00	1	1.00	1.00	
WLSs-subset	0.27	0.40	0.98	1.00	1	0.73	1.00	
WLSs-parsim	0.49	0.57	0.96	1.00	1	0.74	0.99	
WLSs-lasso	0.48	0.62	0.72	1.00	1	1.00	1.00	
WLSv-subset	0.30	0.42	1.00	1.00	1	0.68	1.00	
WLSv-parsim	0.49	0.53	0.99	1.00	1	0.47	1.00	
WLSv-lasso	0.35	0.70	0.85	1.00	1	1.00	1.00	
MinT-subset	1.00	1.00	1.00	1.00	1	1.00	1.00	
MinT-parsim	1.00	1.00	1.00	1.00	1	1.00	1.00	
MinT-lasso	1.00	1.00	1.00	1.00	1	1.00	1.00	
MinTs-subset	0.87	0.85	1.00	1.00	1	0.85	1.00	
MinTs-parsim	1.00	1.00	1.00	1.00	1	1.00	1.00	
MinTs-lasso	0.86	0.84	1.00	1.00	1	0.85	1.00	
Elasso	0.94	0.79	0.93	1.00	1	1.00	1.00	

5 Applications

In this section we describe two empirical applications: Section 5.1 focuses on a grouped hierarchy built using the Australian labour force survey data released by the Australian Bureau of Statistics, while Section 5.2 considers Australian domestic tourism flows with a natural geographic hierarchy.

5.1 Forecasting Australian labour force

The dataset from the Labour Force Survey, released by the Australian Bureau of Statistics, comprises monthly data on the number of unemployed persons in Australia from January 2010 to July 2023[1]. We address the few missing observations using linear interpolation. Analyzing unemployment data by labour market region and duration of job search offers valuable insights into regional disparities, and the structural nuances underlying unemployment. Forecast reconciliation is crucial in such a case to ensure aligned decision making.

We construct a grouped hierarchy by disaggregating the number of unemployed persons over two independent attributes, duration of job search (*Duration*), and State and Territory (*STT*). At the bottom level, the data are disaggregated by both attributes. We refer to the bottom level as the *Duration* $\times$ *STT* level. Specifically, there are six different groups of job search duration, under 1 month, 1–3 months, 3–6 months, 6–12 months, 1–2 years, and 2 years and over. Additionally, the number of unemployed persons in Australia can be disaggregated by eight states and territories, i.e., NSW, VIC, QLD, SA, WA, TAS, NT, and ACT. So the final grouped hierarchy consists of the top series, six series at the Duration level, eight series at the STT level, and 48 series at the Duration $\times$ STT level, giving 63 time series in total, each of length 163 observations.

The top panel in Figure 5 shows the total number of unemployed persons in Australia from January 2010 to July 2023, representing the top-level series in the hierarchy. The monthly series shows strong seasonality, marked by prominent peaks occurring every January, attributable to school-leavers. Lower peaks occur in July, perhaps impacted by the start of the financial year. Amidst the backdrop of COVID-19's non-essential service shutdowns and trading restrictions, March and April 2020 saw a notable surge in unemployment. However, as coronavirus cases dwindled significantly and restrictions eased in the aftermath, employment made a remarkable recovery, leading to a subsequent decline in unemployment. The bottom-left panel displays the breakdown of unemployed individuals by state and territory, while the bottom-right panel presents the breakdown by the duration of job search. The plots display diverse and rich dynamics both within and between different levels of the hierarchy. For example, there was noticeable growth observed during 2020 for some states such as NSW, VIC, and QLD, whereas other states did not experience such significant growth. Additionally, there is a resemblance in the seasonal patterns between NSW and QLD, while the seasonal pattern in VIC differs. Compared to the STT level, the seasonal patterns in the Duration-level series are more consistent and potentially easier to forecast.

We assess the forecast accuracy of base forecasts and various reconciliation methods through a rolling forecast

[1]The Labour Force Survey data is publicly available at https://www.abs.gov.au/statistics/labour/employment-and-unemployment/labour-force-australia-detailed/aug-2023.

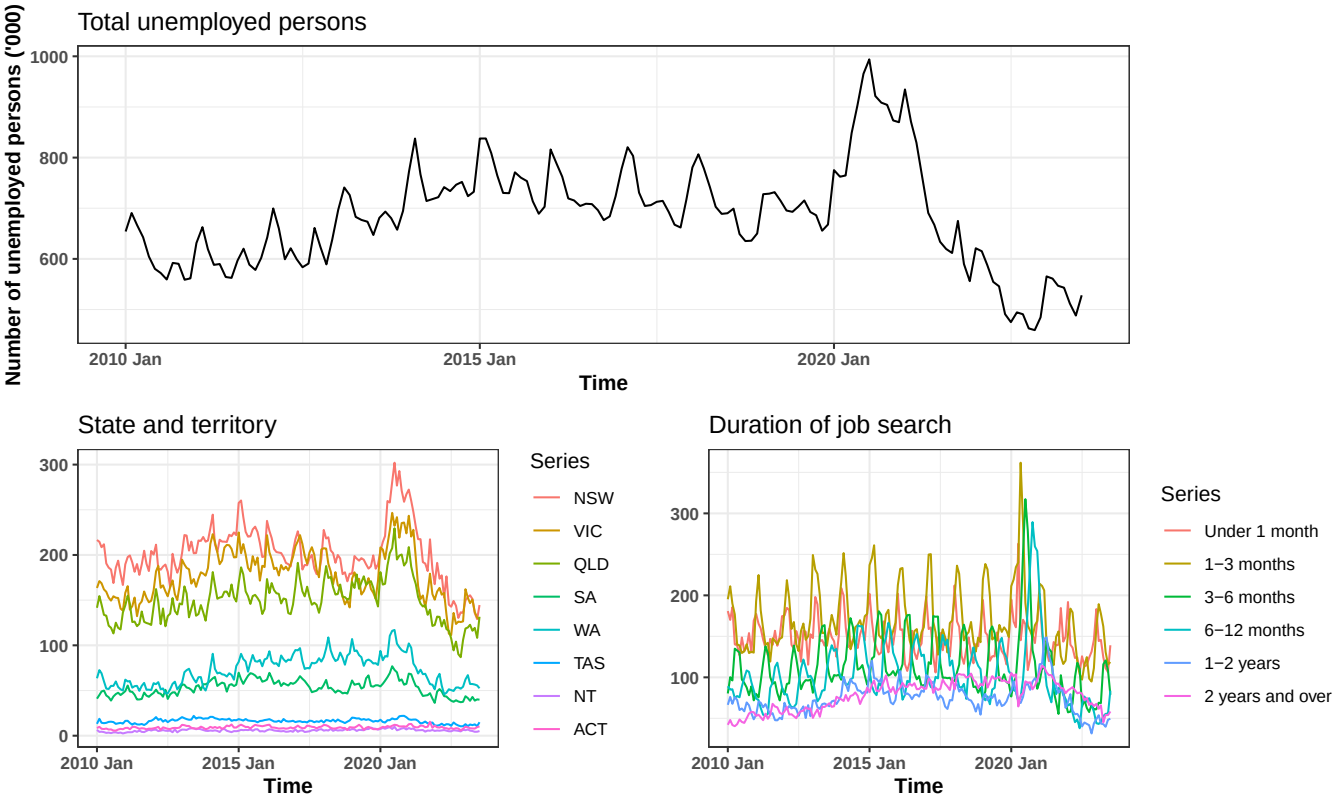


Figure 5: *Australia unemployed persons, disaggregated by state and territory, and by duration of job search.*

origin approach. Our aim is to generate 1- to 12-step-ahead forecasts for each of the 63 series while ensuring coherence. Given the limited data compared to the forecast horizon, we initiate the process with a training set of 139 observations for each series. The training set is used to select the optimal ETS model with the automatic algorithm implemented in the **forecast** package for R (Hyndman & Khandakar 2008). Using these fitted ETS models, we generate base forecasts, and perform diverse forecast reconciliation methods. Then we roll the forecast origin forward by one month and repeat the process, until July 2022. We note that it may be challenging to identify the series with "poor" forecasts due to structural changes in the data caused by the COVID-19 pandemic, which affect the accuracy of forecasts across all time series.

The average results are presented in Table 6. The MCB test and error correlation calculations are not performed due to the limited data length. Results of MinT and the respective proposed methods are excluded due to poor performance, likely caused by the poor sample covariance estimate given the small sample size relative to the number of series in the structure. Subset methods using different estimators of $\boldsymbol{W}$ generally improve forecast accuracy over their benchmarks, particularly at aggregation levels, which are of paramount concern to practitioners. The exception is WLSs-subset, which shows slight deterioration, though still improving top-level forecasts. Moreover, the Parsimonious and Lasso methods almost always yield results identical to their benchmarks, because they tend to provide dense estimates, and ETS models typically avoid extremely poor forecasts. However, OLS-parsim shows deterioration at all levels except the top level. When we drop the unbiasedness assumption, EMinT performs the worst across all levels, as it assumes joint weak stationarity among series, which is evidently not the case here. Elasso ranks the best overall and significantly outperforms EMinT, with the most accurate coherent forecasts observed at

different estimate of W? XQ — Co rected.

Table 6: *Performance of proposed (gray-shaded) and benchmark methods for Australian labour force data. The Base row shows average RMSE of base forecasts, while entries below show RMSE percentage decrease (negative) or increase (positive) for reconciliation methods. Blue entries highlight the best-performing methods; bold entries indicate proposed methods outperforming bechmarks.*

Method	Top	Duration	STT	Duration x STT	Average
Base	67.6	18.1	10.7	3.3	6.6
BU	24.2	0.8	10.4	0.0	6.3
OLS	1.0	–3.3	–0.1	0.7	–0.5
OLS-subset	**–2.0**	**–4.4**	**–1.2**	**0.2**	**–1.6**
OLS-parsim	**0.2**	–3.2	0.6	1.2	–0.2
OLS-lasso	1.0	–3.3	–0.1	0.7	–0.5
WLSs	7.0	–4.3	1.6	**–1.6**	–0.3
WLSs-subset	**4.2**	–3.4	**1.0**	0.3	0.1
WLSs-parsim	7.0	–4.3	1.6	**–1.6**	–0.3
WLSs-lasso	7.0	–4.3	1.6	**–1.6**	–0.3
WLSv	6.6	–4.5	0.2	–1.3	–0.6
WLSv-subset	**1.8**	**–4.6**	**–1.0**	–0.5	**–1.3**
WLSv-parsim	6.6	–4.5	0.2	–1.3	–0.6
WLSv-lasso	6.6	–4.5	0.2	–1.3	–0.6
MinTs	5.3	–3.7	–0.8	–0.7	–0.5
MinTs-subset	**4.5**	**–3.9**	**–0.9**	–0.7	**–0.7**
MinTs-parsim	5.3	–3.7	–0.8	–0.7	–0.5
MinTs-lasso	5.3	–3.7	–0.8	–0.7	–0.5
EMinT	10.2	24.1	6.2	27.9	19.6
Elasso	**–2.3**	**–3.6**	**–8.8**	**0.5**	**–2.9**

the top level and STT level.

Results based on the final test set spanning from August 2022 to July 2023 are detailed in Appendix D, allowing for model training with more data and analysis of post-COVID patterns. Table D.1 shows that the results are qualitatively similar to those in Table 6. Table D.2 presents the number of series selected at each level, their average RMSSE values, and the identified optimal tuning parameters. Only Subset and Elasso methods are presented, as they differ from benchmark results. The scale variation in optimal parameters across methods is due to differences in objective function scales. Notably, all Subset methods exclude some series, and Elasso, using only 11 series, performs best overall when assessed based on average results across the entire hierarchy and forecast horizon. With the exception of the series selected at the STT level by WLSs-subset and at the Duration level by WLSv-subset and Elasso, the time series chosen by the proposed methods show either lower or at least equivalent RMSSE values at each level compared to all series within that level.

is it true to say S level for WLSs-subset has RMSS > 1, don't think s XQ — Has now been rewritten.

5.2 Forecasting Australian domestic tourism

Australian domestic tourism flows are measured as the number of overnight trips Australians spend away from home. The data are sourced from the National Visitor Survey and collected through computer-assisted telephone interviews with approximately 120,000 residents aged 15 years and older. The data follow a geographic structure, with national total tourism flows at the top level, then disaggregated into seven states and territories (referred to as *State* level hereafter), further dividing into 27 zones, and finally into 76 regions. Thus, $n_b = 76$ and $n = 111$. Each series spans January 1998 to December 2017, with a total of 240 observations.

Figure 6 shows aggregate tourism flows for Australia and individual states, revealing pronounced seasonal patterns across both national and states levels, though patterns vary across series. Significant growth began around

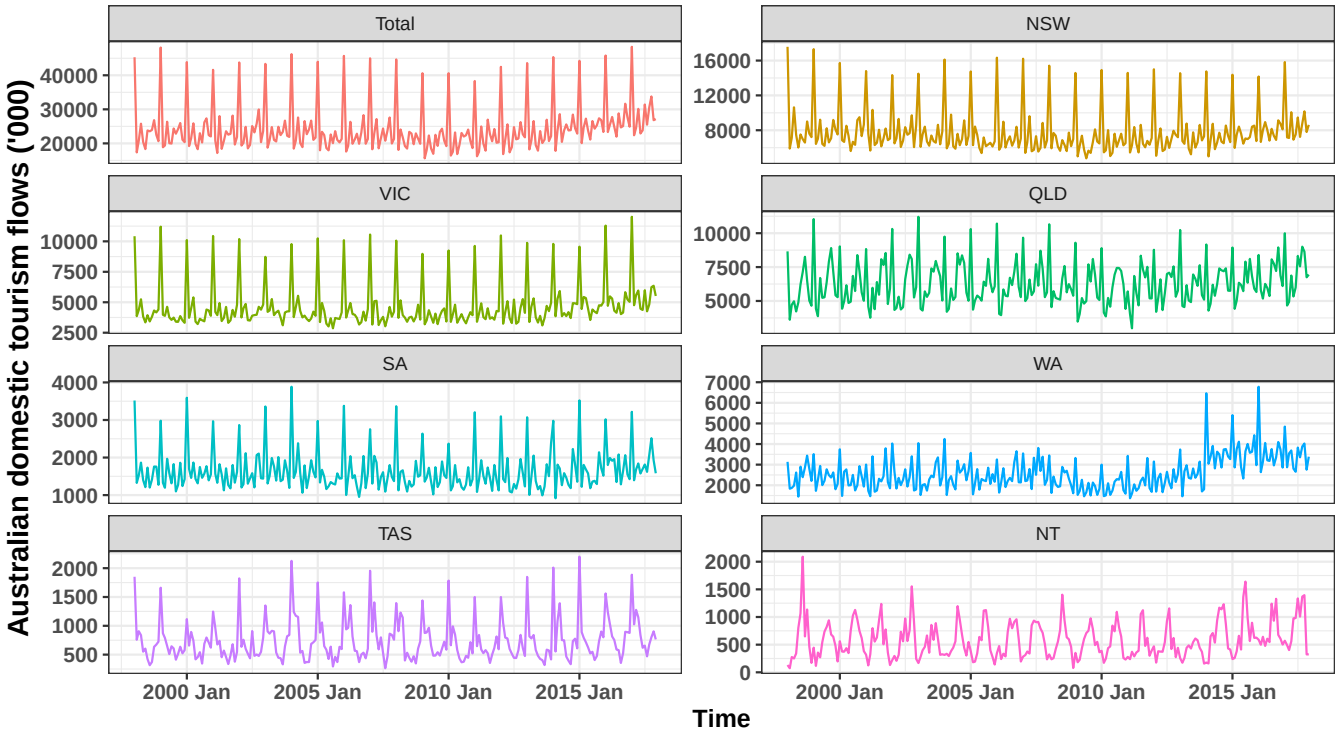


Figure 6: *Domestic tourism flows from January 1998 to December 2017 for Australia as well as the states.*

2010 for the national flow and some states such as NSW, VIC, QLD, and WA, while flows for SA, TAS, and NT are relatively flat. Moreover, there is a notable decline in tourism flows for WA in 2016.

Our objective is to forecast tourism flows for each series in the geographic hierarchy while ensuring coherence across all levels. We use a rolling forecast origin to evaluate the forecast accuracy of different methods. We start with a training set of 216 months for each series, and compute base forecasts from optimal ETS models. We then roll the forecast origin forward, month by month, until December 2016. The base forecasts are reconciled using our proposed methods and some existing reconciliation methods.

Table 7 reports forecast performance results. Similar to Section 5.1, MinT and the respective proposed methods are excluded due to poor performance, and the MCB test and error correlation calculations are omitted due to limited data. The results show that OLS outperforms other benchmarks, despite WLSv and MinTs accounting for in-sample covariance of base forecast errors, highlighting its effectiveness. Overall, the Subset methods outperform benchmarks, especially for aggregation levels. The only exception is OLS-subset, which slightly reduces overall accuracy while improving top-level forecasts. Parsimonious and Lasso methods produce results almost identical to their benchmarks, which is not surprising as ETS models rarely yield extremely poor forecasts, making them challenging to be selected out using methods that tend to return dense estimates. When we relax the unbiasedness assumption, EMinT consistently performs worst across all levels due to the lack of joint weak stationarity among the series in the hierarchy. Elasso, however, shows large improvement compared to EMinT, and ranks best across most levels except the bottom level.

We present the results based on the last test set from January 2017 to December 2017 in Appendix E. Table E.1 shows the percentage improvement in RMSEs for various reconciliation methods, and Figure E.1 displays the

Table 7: *Performance of proposed (gray-shaded) and benchmark methods for Australian domestic tourism data. The Base row shows average RMSE of base forecasts, while entries below show RMSE percentage decrease (negative) or increase (positive) for reconciliation methods. Blue entries highlight the best-performing methods; bold entries indicate proposed methods outperforming bechmarks.*

Method	Top	State	Zone	Region	Average
Base	1773.1	442.0	185.3	94.4	153.6
BU	38.8	15.7	1.6	0.0	7.4
OLS	1.9	–1.3	–5.6	–2.4	–2.7
OLS-subset	**–3.0**	0.9	–1.1	0.6	–0.2
OLS-parsim	1.9	–1.3	–5.6	–2.4	–2.7
OLS-lasso	2.9	0.3	–4.9	–2.2	–2.0
WLSs	18.1	5.2	–3.4	–1.8	1.1
WLSs-subset	**10.2**	**4.3**	–2.1	–0.5	**1.0**
WLSs-parsim	18.1	5.2	–3.4	–1.8	1.1
WLSs-lasso	18.5	5.5	–3.3	–1.7	1.2
WLSv	23.6	7.9	–2.2	–1.9	2.4
WLSv-subset	**5.7**	**0.7**	**–4.9**	**–2.3**	**–1.7**
WLSv-parsim	23.6	7.9	–2.2	–1.9	2.4
WLSv-lasso	23.6	7.9	–2.2	–1.9	2.4
MinTs	18.6	5.7	–3.4	**–2.7**	0.9
MinTs-subset	**3.2**	**–0.7**	**–5.4**	–2.5	**–2.5**
MinTs-parsim	18.6	5.7	–3.4	**–2.7**	0.9
MinTs-lasso	18.6	5.7	–3.4	**–2.7**	0.9
EMinT	59.7	70.4	81.4	85.9	79.0
Elasso	**–18.7**	**–19.6**	**–12.6**	**–1.8**	**–9.9**

reconciliation errors across 111 series. These results are consistent with the average results mentioned earlier, indicating relatively high-quality forecasts from Subset and Elasso methods. In addition, Table E.2 summarizes the number of series selected at each level, their average RMSSE values, and the optimal tuning parameters. Here we focus on the Subset and Elasso methods since they show improved forecast results in this application. We observe that OLS-subset and WLSs-subset exclude some series at the State and Zone levels, yielding lower RMSSE values for the selected series at each level. Notably, WLSs-subset selects only the best-performing series at the State level. Despite using only 13 time series, Elasso performs exceptionally well, with reduced RMSSE values at each level for the selected series. Thus, both the Subset and Elasso methods are able to exclude series with high scaled forecast errors. In contrast, the WLSv and MinTs subset methods retain all series. Nonetheless, the inclusion of shrinkage through ridge regularization in the WLSv and MinTs subset methods still enhances the quality of reconciled forecasts.

I don't think you have interpreted t plot. XQ — Figu E1 and Table E1 ally show the sar results, but Figur E1 presents them a heatmap format and excludes the -parsim and -lass results. To keep paper concise and considering their consistency with earlier findings, I think the followi sentence is enoug for summarizing these results.

again, "high fore- cast error"? XQ Corrected to "hig scaled forecast er ror"

6 Discussion

While our proposed methods opt to leave "poor" base forecasts unused in forming reconciled forecasts, the approach by Zhang et al. (2023) is primarily focused on keeping "good" base forecasts unchanged after reconciliation. Both methods frame the problem as a quadratic programming problem to enhance reconciled forecast performance by somehow managing the impact of some time series during the process, ultimately providing reconciled forecasts for all series in the hierarchy. However, our methods differ from the approach by Zhang et al. (2023) in four main ways. First, our proposed methods alter the poor-performing base forecasts, ensuring these do not influence other nodes, whereas the approach by Zhang et al. (2023) keeps the forecasts of certain nodes immutable, which then impact

others. Second, our methods automate the selection of series unused during reconciliation, while the latter method requires an additional, yet-to-be-determined procedure to identify the immutable set of series. Third, we use only base forecasts of the selected series for reconciliation, while Zhang et al. (2023) use base forecasts of all series, even though some remain unchanged after reconciliation. Fourth, beyond preserving the unbiasedness of forecasts, we relax the unbiasedness assumption and introduce an unconstrained "in-sample" reconciliation method that leverages in-sample information for selection.

The proposed methods offer four key benefits to hierarchical forecasting process. First, they automatically exclude series with high scaled forecast errors from reconciliation and enhance computational efficiency by eliminating the need to forecast these series in future operations. Second, they generally produce promising reconciled forecasts for the whole hierarchy, particularly in cases of model misspecification. Third, they reduce disparities resulting from different estimators of $\boldsymbol{W}$ in the MinT framework. Lastly, a further practical benefit of the proposed methods is their adaptability to scenarios where practitioners wish to avoid forecasting certain series, such as those with extensive missing values, by simply imposing additional constraints to enforce the corresponding columns of $\boldsymbol{G}$ to zero.

high forecast erro
XQ — high scale
forecast errors

don't we need fo
casts for all the
series to decide
which columns t
set to zero? XQ
— Corrected. As
explained earlier,
it's for future fut
operations.

Table 8: *Computational time (in seconds) for a single hierarchy across the four experiments, considering all candidate hyperparameters and rolling test sets.*

	Section 4.1	Section 4.2	Section 5.1	Section 5.2
Number of parameters	28.0	28.0	3024	8436
Subset (NOCH = 126)				
OLS-subset	3.1	3.3	1242	2329
WLSs-subset	3.3	3.0	1304	2386
WLSv-subset	3.3	3.0	1306	2377
MinT-subset	4.4	4.1	-	-
MinTs-subset	4.4	4.0	1467	2493
Parsimonious (NOCH = 21)				
OLS-parsim	2.2	2.3	1170	1939
WLSs-parsim	2.3	2.0	1171	1968
WLSv-parsim	2.4	2.3	1171	2001
MinT-parsim	3.3	3.5	-	-
MinTs-parsim	3.7	3.7	1346	2162
Lasso (NOCH = 21)				
OLS-lasso	2.1	2.4	1167	1822
WLSs-lasso	2.1	2.4	1169	1840
WLSv-lasso	2.0	2.4	1235	1841
MinT-lasso	3.7	3.9	-	-
MinTs-lasso	3.9	3.9	1330	2059
Elasso (NOCH = 21)				
Elasso	3.0	3.2	1155	1860

Note: NOCH refers to the number of candidate hyperparameters considered. A dash (-) denotes that the method is not applicable to the experiment.

One limitation of the proposed methods is their scalability. As the number of series in a hierarchy increases, solving these problems efficiently becomes increasingly challenging, and the exact computation of the estimates remains a significant hurdle. In this study, we employed Gurobi, a widely used commercial solvers, to tackle the NP-hard MIP problems. Table 8 provides the computational time for solving the reconciliation problem for a single hierarchy in each of our experiments, considering all candidate hyperparameters and rolling test sets. In terms of computational resources we used 20 CPUs and 3 GB of memory per CPU in a high-performance computing system. Given these computational demands, we recommend applying the proposed methods primarily to hierarchies of small to moderate size. While there are ongoing efforts to develop MIP-based approaches for solving L_0-regularized

are these for all
hyperparameters
and rolling test
sets/simulations?
Explain what is d
noted by "-" in th
table. XQ — Co
rected

regression in high-dimensional settings (Mazumder et al. 2022), extending these approaches to accommodate additional constraints remains a challenge, which we leave for future research.

Other promising directions for further investigation are also worth exploring. Relaxing the assumption of unbiasedness of base forecasts, the Elasso method directly minimizes the mean squared reconciled forecast errors while incorporating regularization terms for time series selection. Unlike constrained "out-of-sample" reconciliation methods, Elasso does not use a $\boldsymbol{W}$ matrix in its optimization problem as $\boldsymbol{W}$ is introduced when deriving the loss function (Equation (2)) under the unbiasedness condition for both base and reconciled forecasts. Future research could explore including a $\boldsymbol{V}$ matrix in the Elasso method by considering the following optimization problem

$$\min_{\boldsymbol{G}} \quad \frac{1}{2T}\operatorname{Tr}\left(\left(\boldsymbol{Y}-\hat{\boldsymbol{Y}}\boldsymbol{G}'\boldsymbol{S}'\right)\boldsymbol{V}^{-1}\left(\boldsymbol{Y}-\hat{\boldsymbol{Y}}\boldsymbol{G}'\boldsymbol{S}'\right)'\right)+\lambda\sum_{j=1}^{n}w_j\left\|\boldsymbol{G}_{\cdot j}\right\|_2,$$

where $\operatorname{Tr}(\cdot)$ denotes the trace of a matrix. In this context, $\boldsymbol{V}$ can be arbitrary and is not restricted to the covariance matrix of base forecast errors. The purpose of its inclusion here is to assign different weights to reconciled forecast errors of various series in the hierarchical structure. Conversely, Elasso and other proposed methods currently treat reconciled forecast errors equally across different series. This suggests a potential new direction for research that differs from the problems addressed in this paper.

Additionally, Panagiotelis et al. (2021) highlighted the critical need for bias correction prior to forecast reconciliation when dealing with biased forecasts. Given that the unbiasedness preserving property is dropped in the Elasso method, a possible direction for future research could be to extend Elasso by including a bias correction mechanism. This can be achieved by formulating the optimization problem as follows:

$$\min_{\boldsymbol{G},\boldsymbol{d}} \quad \frac{1}{2T}\left\|\boldsymbol{Y}-\left(\boldsymbol{D}+\hat{\boldsymbol{Y}}\boldsymbol{G}'\right)\boldsymbol{S}'\right\|_F^2+\lambda\sum_{j=1}^{n}w_j\left\|\boldsymbol{G}_{\cdot j}\right\|_2,$$

where $\boldsymbol{D}\in\mathbb{R}^{T\times n_b}$ is a shift parameter, with each row equal to an n_b-dimensional $\boldsymbol{d}$. This vector $\boldsymbol{d}$ can be trained alongside $\boldsymbol{G}$ and act as a bias correction.

Finally, the concept of bi-level variable selection can be applied to enhance the methodology's capabilities. Such methodologies are well-documented in the literature, including the sparse group lasso (Simon et al. 2013), hierarchical lasso (Zhou & Zhu 2010), and group bridge (Huang et al. 2009). By treating each column of $\boldsymbol{G}$ as a group and its elements as individuals, we have introduced group-wise sparsity in our methods through column-wise shrinkage towards zero. In this case, within-group sparsity can be achieved by simply including an additional lasso penalty to shrink individual elements, as suggested in Simon et al. (2013). This bi-level selection mechanism could potentially offer deeper insights into the contributions of individual base forecasts, particularly regarding their significance when mapped to bottom-level disaggregated forecasts.

7 Conclusion

In the existing forecast reconciliation literature, base forecasts are mapped into bottom-level disaggregated forecasts, which are then summed to yield coherent forecasts for the entire structure. The mapping step can be conceptually

regarded as a forecast combination. However, poor performance in some base forecasts can diminish overall reconciliation effectiveness. To address this issue, we incorporate time series selection to enhance forecast reconciliation, while ensuring coherent forecasts for all series.

Recognizing that reconciled forecasts $\tilde{\boldsymbol{y}}$ are simply linear combinations of base forecasts $\hat{\boldsymbol{y}}$ via $\tilde{\boldsymbol{y}} = \boldsymbol{SG}\hat{\boldsymbol{y}}$, setting columns of $\boldsymbol{G}$ to zero can eliminate heavily misspecified forecasts from the combination. Under the assumption of unbiased base forecasts, we developed three constrained "out-of-sample" reconciliation methods with selection mechanisms. These methods use various penalty functions to shrink the columns of the weighting matrix $\boldsymbol{G}$ towards zero. We proved that the selected time series can effectively reconstruct the whole hierarchy. Additionally, we relaxed the unbiasedness assumption by introducing the unconstrained Elasso method, which selects time series based on in-sample observations and fitted values. Elasso may use fewer time series than the number of bottom-level series and, in extreme cases, resemble a top-down approach.

Simulation experiments and two empirical applications demonstrated the superiority of the proposed methods over existing reconciliation methods. When model misspecification was introduced in some series within the hierarchy, our methods effectively excluded these "problematic" series, ensuring good quality coherent forecasts. In both empirical applications, where no apparent model misspecification was present, the Subset and Elasso methods demonstrated promising results. In contrast, Parsimonious and Lasso generally produced results similar to their benchmarks, as they tend to deliver dense estimates.

While this study provides an important advancement in hierarchical forecasting, several promising avenues for further research remain. These include improving the scalability of the methods, extending the unconstrained in-sample approach to account for varying weights for forecast errors across series, integrating bias correction mechanisms during the reconciliation to address biased forecasts, and exploring bi-level variable selection strategies.

prior to reconcili
ation or during th
reconciliation, lo
at the equation o
page 27. XQ —
should be "during
the reconciliation
Now corrected.

Supplementary materials

Appendix: Summary of proofs, and additional results obtained in Section 4 and Section 5.

Acknowledgments

Rob J Hyndman was supported by the Australian Research Council Industrial Transformation Training Centre in Optimisation Technologies, Integrated Methodologies, and Applications (OPTIMA), Project ID IC200100009.

References

Athanasopoulos, G., Ahmed, R. A. & Hyndman, R. J. (2009), 'Hierarchical forecasts for australian domestic tourism', *International J Forecasting* **25**(1), 146–166.

Athanasopoulos, G., Hyndman, R. J., Kourentzes, N. & Panagiotelis, A. (2024), 'Forecast reconciliation: A review', *International J Forecasting* **40**(2), 430–456.

Athanasopoulos, G., Hyndman, R. J., Kourentzes, N. & Petropoulos, F. (2017), 'Forecasting with temporal hierarchies', *European J Operational Research* **262**(1), 60–74.

Ben Taieb, S. & Koo, B. (2019), Regularized regression for hierarchical forecasting without unbiasedness conditions, *in* 'Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining', KDD '19, Association for Computing Machinery, New York, NY, USA, pp. 1337–1347.

Ben Taieb, S., Taylor, J. W. & Hyndman, R. J. (2021), 'Hierarchical probabilistic forecasting of electricity demand with smart meter data', *J American Statistical Association* **116**(533), 27–43.

Bertsimas, D., King, A. & Mazumder, R. (2016), 'Best subset selection via a modern optimization lens', *The Annals of Statistics* **44**(2), 813–852.

Di Fonzo, T. & Girolimetto, D. (2023), 'Cross-temporal forecast reconciliation: Optimal combination method and heuristic alternatives', *International J Forecasting* **39**(1), 39–57.

Gleason, J. L. (2020), Forecasting hierarchical time series with a regularized embedding space, *in* '6th Workshop on Mining and Learning from Time Series', San Diego, CA, USA.

Greenshtein, E. (2006), 'Best subset selection, persistence in high-dimensional statistical learning and optimization under ℓ_0 constraint', *The Annals of Statistics* **34**(5), 2367–2386.

Hazimeh, H. & Mazumder, R. (2020), 'Fast best subset selection: Coordinate descent and local combinatorial optimization algorithms', *Operations Research* **68**(5), 1517–1537.

Hazimeh, H., Mazumder, R. & Radchenko, P. (2023), 'Grouped variable selection with discrete optimization: Computational and statistical perspectives', *The Annals of Statistics* **51**(1), 1–32.

Hazimeh, H., Mazumder, R. & Saab, A. (2022), 'Sparse regression at scale: branch-and-bound rooted in first-order optimization', *Mathematical Programming* **196**(1), 347–388.

Hollyman, R., Petropoulos, F. & Tipping, M. E. (2021), 'Understanding forecast reconciliation', *European J Operational Research* **294**(1), 149–160.

Huang, J., Ma, S., Xie, H. & Zhang, C.-H. (2009), 'A group bridge approach for variable selection', *Biometrika* **96**(2), 339–355.

Hyndman, R. J., Ahmed, R. A., Athanasopoulos, G. & Shang, H. L. (2011), 'Optimal combination forecasts for hierarchical time series', *Computational Statistics & Data Analysis* **55**(9), 2579–2589.

Hyndman, R. J. & Athanasopoulos, G. (2021), *Forecasting: principles and practice*, 3rd edn, OTexts, Melbourne, Australia. https://OTexts.com/fpp3.

Hyndman, R. J., Athanasopoulos, G., Bergmeir, C., Caceres, G., Chhay, L., O'Hara-Wild, M., Petropoulos, F., Razbash, S., Wang, E. & Yasmeen, F. (2023), *forecast: Forecasting functions for time series and linear models*. R package version 8.21.1. https://pkg.robjhyndman.com/forecast/.

Hyndman, R. J. & Khandakar, Y. (2008), 'Automatic time series forecasting: the forecast package for R', *J Statistical Software* **26**(3), 1–22.

Hyndman, R. J., Lee, A. J. & Wang, E. (2016), 'Fast computation of reconciled forecasts for hierarchical and grouped time series', *Computational Statistics & Data Analysis* **97**, 16–32.

Lounici, K., Pontil, M., Van De Geer, S. & Tsybakov, A. B. (2011), 'Oracle inequalities and optimal inference under group sparsity', *The Annals of Statistics* **39**(4), 2164–2204.

Mazumder, R., Radchenko, P. & Dedieu, A. (2022), 'Subset selection with shrinkage: Sparse linear modeling when the SNR is low', *Operations Research* **71**(1), 129–147.

Mishchenko, K., Montgomery, M. & Vaggi, F. (2019), A self-supervised approach to hierarchical forecasting with applications to groupwise synthetic controls. arXiv:1906.10586.

Nardi, Y. & Rinaldo, A. (2008), 'On the asymptotic properties of the group lasso estimator for linear models', *Electronic J Statistics* **2**, 605–633.

Nystrup, P., Lindström, E., Møller, J. K. & Madsen, H. (2021), 'Dimensionality reduction in forecasting with temporal hierarchies', *International J Forecasting* **37**(3), 1127–1146.

Nystrup, P., Lindström, E., Pinson, P. & Madsen, H. (2020), 'Temporal hierarchies with autocorrelation for load forecasting', *European J Operational Research* **280**(3), 876–888.

Panagiotelis, A., Athanasopoulos, G., Gamakumara, P. & Hyndman, R. J. (2021), 'Forecast reconciliation: A geometric view with new insights on bias correction', *International J Forecasting* **37**(1), 343–359.

Pang, Y., Zhou, X., Zhang, J., Sun, Q. & Zheng, J. (2022), 'Hierarchical electricity time series prediction with cluster analysis and sparse penalty', *Pattern Recognition* **126**, 108555.

Simon, N., Friedman, J., Hastie, T. & Tibshirani, R. (2013), 'A sparse-group lasso', *J Computational and Graphical Statistics* **22**(2), 231–245.

Wickramasuriya, S. L. (2021), Properties of point forecast reconciliation approaches. arXiv:2103.11129.

Wickramasuriya, S. L., Athanasopoulos, G. & Hyndman, R. J. (2019), 'Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization', *J American Statistical Association* **114**(526), 804–819.

Yuan, M. & Lin, Y. (2006), 'Model selection and estimation in regression with grouped variables', *J Royal Statistical Society. Series B, Statistical Methodology* **68**(1), 49–67.

Zhang, B., Kang, Y., Panagiotelis, A. & Li, F. (2023), 'Optimal reconciliation with immutable forecasts', *European J Operational Research* **308**(2), 650–660.

Zhang, C.-H. & Zhang, T. (2012), 'A general theory of concave regularization for high-dimensional sparse estimation problems', *Statistical Science* **27**(4), 576–593.

Zhou, N. & Zhu, J. (2010), 'Group variable selection via a hierarchical lasso and its oracle property', *arXiv preprint arXiv:1006.2871* .

Appendix

A Proofs of Propositions

A.1 Proof of Proposition 1

Proof. Let $\boldsymbol{X}_{\cdot\mathbb{S}} \in \mathbb{R}^{r\times|\mathbb{S}|}$ denote the submatrix of the $r \times c$ matrix $\boldsymbol{X}$ with the columns indexed by the set $\mathbb{S}$, where $|\mathbb{S}|$ is the cardinality of the set $\mathbb{S}$. Similarly, let $\boldsymbol{X}_{\mathbb{S}\cdot} \in \mathbb{R}^{|\mathbb{S}|\times c}$ denote the submatrix of $\boldsymbol{X}$ with the rows indexed by $\mathbb{S}$. If $\mathbb{S}$ is the set of indices of nonzero columns in the solution $\hat{\boldsymbol{G}}$ to Equation (5), then the following equations hold:

$$\hat{\boldsymbol{G}}\boldsymbol{S} = \hat{\boldsymbol{G}}_{\cdot\mathbb{S}}\boldsymbol{S}_{\mathbb{S}\cdot} = \boldsymbol{I}_{n_b}, \qquad \text{and} \qquad \min\left(\text{rank}(\hat{\boldsymbol{G}}_{\cdot\mathbb{S}}), \text{rank}(\boldsymbol{S}_{\mathbb{S}\cdot})\right) \geq \text{rank}(\boldsymbol{I}_{n_b}) = n_b.$$

This indicates that the number of nonzero columns of $\hat{\boldsymbol{G}}$ should be no less than n_b, i.e., $|\mathbb{S}| \geq n_b$.

Moreover, we have $\text{rank}(\boldsymbol{S}_{\mathbb{S}\cdot}) = n_b$ because $\text{rank}(\boldsymbol{S}_{\mathbb{S}\cdot}) \leq n_b$, given that $\boldsymbol{S}$ has n_b columns. If the solution to Equation (5) yields a $\hat{\boldsymbol{G}}$ with exactly n_b nonzero columns (i.e., $|\mathbb{S}| = n_b$), then $\boldsymbol{S}_{\mathbb{S}\cdot}$ is a full rank square matrix and thus invertible. Applying Theorem 2 in Zhang et al. (2023), $\boldsymbol{y}_{\mathbb{S}\cdot}$ is valid for constructing the full hierarchy using nothing but the information embedded in the aggregation constraints. If the solution yields a $\hat{\boldsymbol{G}}$ with more than n_b nonzero columns, we should be able to identify more than one subset $\mathbb{S}^* \subset \mathbb{S}$ with $|\mathbb{S}^*| = n_b$ to construct an invertible square matrix $\boldsymbol{S}_{\mathbb{S}^*\cdot}$ and thereby restore the full hierarchy using the valid $\boldsymbol{y}_{\mathbb{S}^*\cdot}$. Therefore, the constraint $\boldsymbol{G}\boldsymbol{S} = \boldsymbol{I}$ ensures that the selected columns of $\hat{\boldsymbol{G}}$ correspond to variables that can restore the full hierarchy. □

A.2 Proof of Proposition 2

Proof. We have

$$\boldsymbol{S}\boldsymbol{G}\hat{\boldsymbol{y}} = \text{vec}\left(\boldsymbol{S}\boldsymbol{G}\hat{\boldsymbol{y}}\right) = \left(\hat{\boldsymbol{y}}' \otimes \boldsymbol{S}\right)\text{vec}(\boldsymbol{G}),$$

$$\text{vec}\left(\boldsymbol{G}\boldsymbol{S}\right) = \text{vec}\left(\boldsymbol{I}_{n_b}\boldsymbol{G}\boldsymbol{S}\right) = \left(\boldsymbol{S}' \otimes \boldsymbol{I}_{n_b}\right)\text{vec}(\boldsymbol{G}).$$

Substituting these into Equation (5), the previous problem now takes the form of a regression problem with an additional regularization term and an equality constraint on the coefficients, as shown in Equation (6). □

A.3 Proof of Proposition 3

Proof. Denote $\beta = \text{vec}(\boldsymbol{G})$, and the first term in the objective of Equation (10) as $L\left(\beta \mid \boldsymbol{D}\right)$, where $\boldsymbol{D}$ is the working data $\{\hat{\boldsymbol{y}}, \hat{\boldsymbol{y}}' \otimes \boldsymbol{S}\}$. Relaxing the constraint $\boldsymbol{G_h}\boldsymbol{S} = \boldsymbol{I}_{n_b}$, we define λ^1 as the smallest λ value such that all predictors in the group lasso problem have zero coefficients, i.e., the solution at λ^1 is $\hat{\beta}^1 = \boldsymbol{0}$. (Note that there is no intercept in our problem.) Under the Karush-Kuhn-Tucker conditions, we have

$$\lambda^1 = \max_{j=1,\ldots,n} \left\| \left[\nabla L(\hat{\beta}^1 \mid \boldsymbol{D})\right]^{(j)} \right\|_2 / w_j = \max_{j=1,\ldots,n} \left\| -\left((\hat{\boldsymbol{y}}' \otimes \boldsymbol{S})_{\cdot j^*}\right)' \boldsymbol{W}^{-1}\hat{\boldsymbol{y}} \right\|_2 / w_j.$$

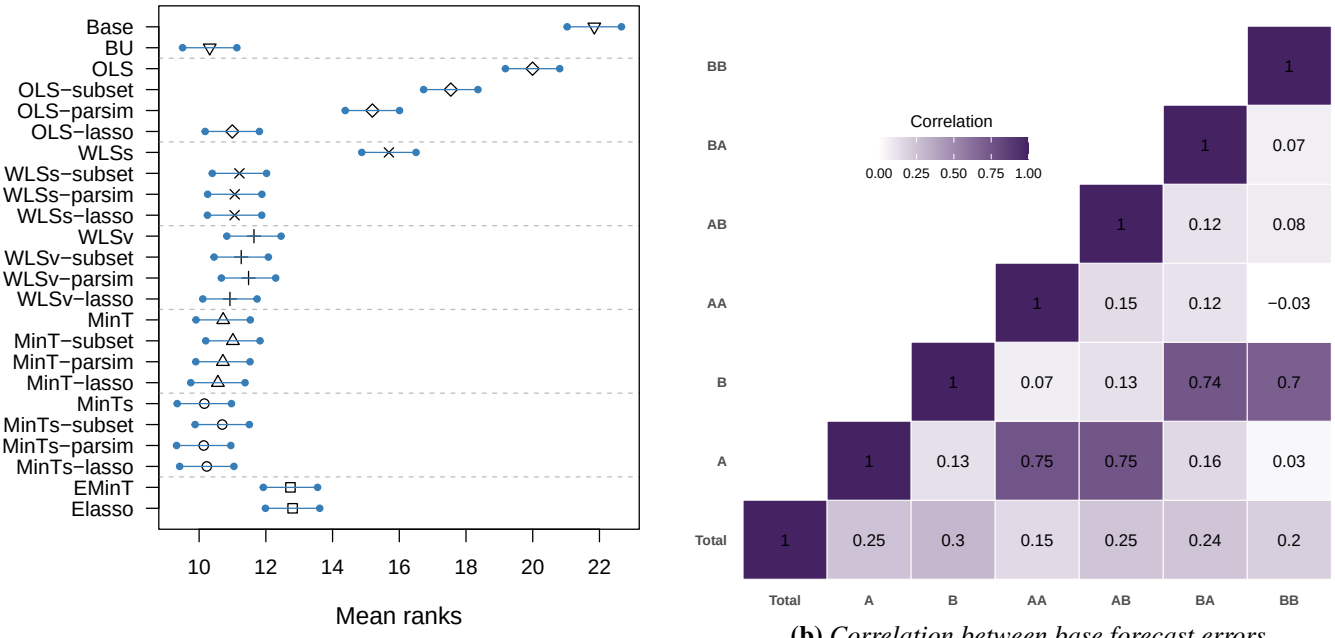
□

B Results from simulation Setup 1

The section provides additional results obtained in Section 4.1.

Table B.1: *Performance of proposed (gray-shaded) and benchmark methods for simulation Setup 1, with model misspecification in series Total. The Base row shows average RMSE of base forecasts, while entries below show RMSE percentage decrease (negative) or increase (positive) for reconciliation methods. Blue entries highlight the best-performing methods; bold entries indicate proposed methods outperforming bechmarks.*

	Top				Middle				Bottom				Average			
Method	h=1	1–4	1–8	1–16	h=1	1–4	1–8	1–16	h=1	1–4	1–8	1–16	h=1	1–4	1–8	1–16
Base	25.0	30.3	30.9	32.3	6.3	7.3	8.6	10.8	4.2	4.9	5.9	7.5	7.8	9.2	10.3	12.0
BU	–62.0	**–64.4**	**–59.0**	–51.5	**–0.3**	**0.0**	**0.1**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**–28.5**	**–30.2**	**–25.3**	**–19.8**
OLS	–34.8	–35.5	–33.5	–30.1	45.3	50.6	37.7	25.1	27.7	29.9	21.2	13.7	3.1	3.8	1.6	–0.2
OLS-subset	**–35.3**	**–41.9**	**–39.2**	**–35.0**	**43.9**	**39.5**	**29.5**	**19.6**	**27.1**	**23.6**	**16.8**	**10.9**	**2.4**	**–3.5**	**–4.2**	**–4.5**
OLS-parsim	**–41.2**	**–49.2**	**–45.5**	**–40.0**	**35.1**	**26.8**	**20.3**	**13.7**	**21.9**	**15.9**	**11.5**	**7.6**	**–4.0**	**–12.2**	**–10.9**	**–9.1**
OLS-lasso	**–61.8**	**–63.6**	**–58.1**	**–50.9**	**0.4**	**1.3**	**1.3**	**0.7**	**0.3**	**0.8**	**0.6**	**0.4**	**–28.2**	**–29.3**	**–24.5**	**–19.2**
WLSs	–50.9	–52.4	–48.7	–43.3	17.6	20.0	14.5	9.3	9.6	11.3	7.7	4.9	–16.3	–16.7	–14.9	–12.5
WLSs-subset	**–61.8**	**–63.6**	**–58.1**	**–50.7**	**0.3**	**1.4**	**1.4**	**0.9**	**0.3**	**0.9**	**0.7**	**0.6**	**–28.2**	**–29.3**	**–24.4**	**–19.0**
WLSs-parsim	**–61.8**	**–63.8**	**–58.3**	**–50.9**	**0.0**	**1.0**	**1.0**	**0.7**	**0.3**	**0.7**	**0.6**	**0.5**	**–28.3**	**–29.5**	**–24.6**	**–19.2**
WLSs-lasso	**–61.7**	**–63.5**	**–58.0**	**–50.7**	**0.5**	**1.5**	**1.4**	**0.9**	**0.3**	**0.9**	**0.7**	**0.5**	**–28.1**	**–29.2**	**–24.4**	**–19.1**
WLSv	–61.1	–63.4	–58.1	–50.8	1.0	1.7	1.3	0.8	0.7	1.0	0.6	0.4	–27.6	–29.1	–24.5	–19.2
WLSv-subset	**–61.9**	**–63.6**	**–58.2**	**–50.9**	**0.2**	**1.3**	**1.2**	0.8	**0.1**	**0.8**	0.6	0.5	**–28.3**	**–29.3**	–24.5	–19.2
WLSv-parsim	**–61.8**	**–63.8**	**–58.3**	**–51.0**	**0.0**	**1.1**	**1.1**	**0.6**	**0.1**	**0.6**	**0.5**	0.4	**–28.4**	**–29.5**	**–24.7**	**–19.3**
WLSv-lasso	**–61.8**	**–63.9**	**–58.4**	**–51.1**	**0.2**	**0.9**	**0.9**	**0.5**	**0.1**	**0.5**	**0.4**	**0.3**	**–28.3**	**–29.6**	**–24.8**	**–19.4**
MinT	–62.1	–64.3	–58.9	**–51.6**	–0.2	0.6	0.5	0.2	0.8	0.5	0.3	0.1	–28.3	–29.9	–25.1	**–19.8**
MinT-subset	–61.8	–63.7	–58.2	–50.9	0.4	1.2	1.3	0.8	0.8	1.0	0.7	0.5	–28.0	–29.3	–24.5	–19.2
MinT-parsim	–62.1	–64.3	–58.9	**–51.6**	–0.2	0.6	0.5	0.2	0.8	0.5	0.3	0.1	–28.3	–29.9	–25.1	**–19.8**
MinT-lasso	–62.1	**–64.4**	–58.9	–51.5	**–0.3**	**0.3**	**0.4**	**0.1**	**0.6**	**0.3**	**0.1**	0.1	**–28.4**	**–30.1**	**–25.2**	**–19.8**
MinTs	**–62.2**	**–64.4**	**–59.0**	**–51.6**	**–0.3**	0.3	0.4	0.1	0.4	0.3	0.1	**0.0**	**–28.5**	–30.1	–25.2	**–19.8**
MinTs-subset	–62.0	–63.8	–58.4	–51.1	0.4	1.1	1.2	0.7	0.5	0.9	0.7	0.5	–28.2	–29.5	–24.6	–19.3
MinTs-parsim	**–62.2**	**–64.4**	**–59.0**	**–51.6**	**–0.3**	0.3	0.4	0.1	0.4	0.3	0.1	**0.0**	**–28.5**	–30.1	–25.2	**–19.8**
MinTs-lasso	**–62.2**	**–64.4**	–58.9	–51.5	–0.2	0.3	0.4	0.1	**0.2**	**0.2**	0.1	**0.0**	**–28.5**	–30.1	–25.2	**–19.8**
EMinT	–60.7	–63.5	–58.2	–51.0	2.5	2.9	2.3	1.3	3.6	2.9	2.0	1.1	–26.2	–28.3	–23.8	–18.9
Elasso	**–60.9**	**–63.6**	–58.2	**–51.1**	**2.3**	**2.8**	2.3	1.3	**3.1**	3.1	2.1	1.2	**–26.5**	–28.3	–23.8	–18.9



(a) *MCB test*

(b) *Correlation between base forecast errors*

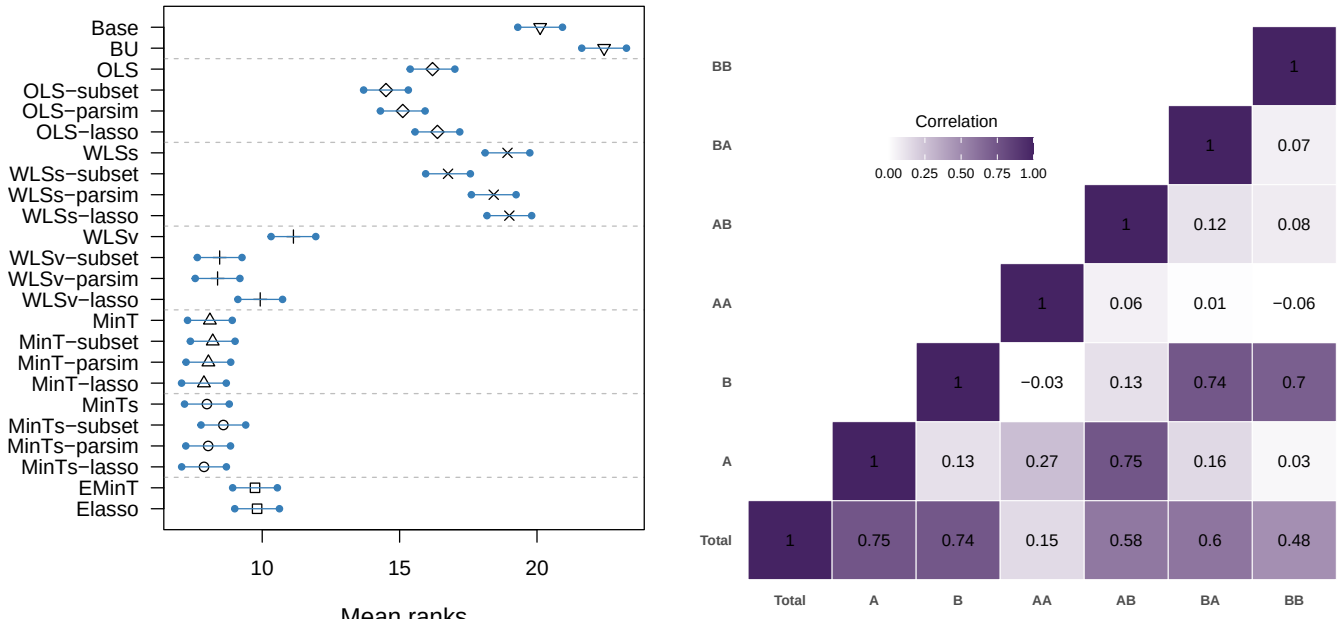
Figure B.1: *MCB test result and correlation matrix heatmap for simulation Setup 1, with model misspecification in series Total.*

Table B.2: *Proportion of time series being selected using proposed reconciliation methods for simulation Setup 1, with model misspecification in series Total. RMSSE values of base forecasts for each series are in parentheses. The last column displays a stacked barplot of the total selected series from 500 hierarchies, with darker sub-bars indicating higher counts.*

(RMSSE)	Total (3.08)	A (1.52)	B (1.55)	AA (1.57)	AB (1.62)	BA (1.62)	BB (1.57)	Summary
OLS-subset	0.75	0.45	0.44	0.82	0.79	0.83	0.80	
OLS-parsim	0.47	0.70	0.69	0.86	0.92	0.90	0.89	
OLS-lasso	0.38	0.01	0.01	1.00	1.00	1.00	1.00	
WLSs-subset	0.08	0.42	0.41	0.87	0.85	0.84	0.89	
WLSs-parsim	0.06	0.55	0.50	0.66	0.87	0.69	0.88	
WLSs-lasso	0.35	0.03	0.03	1.00	1.00	1.00	1.00	
WLSv-subset	0.31	0.67	0.65	0.88	0.90	0.91	0.90	
WLSv-parsim	0.34	0.63	0.60	0.80	0.89	0.84	0.87	
WLSv-lasso	0.45	0.35	0.36	1.00	1.00	1.00	1.00	
MinT-subset	0.69	0.78	0.80	0.91	0.91	0.91	0.91	
MinT-parsim	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinT-lasso	0.75	0.89	0.86	0.97	0.97	0.97	0.97	
MinTs-subset	0.67	0.74	0.76	0.90	0.89	0.88	0.91	
MinTs-parsim	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinTs-lasso	0.77	0.72	0.73	1.00	1.00	1.00	1.00	
Elasso	0.95	0.64	0.64	1.00	1.00	1.00	1.00	

Table B.3: *Performance of proposed (gray-shaded) and benchmark methods for simulation Setup 1, with model misspecification in series AA. The Base row shows average RMSE of base forecasts, while entries below show RMSE percentage decrease (negative) or increase (positive) for reconciliation methods. Blue entries highlight the best-performing methods; bold entries indicate proposed methods outperforming bechmarks.*

	Top				Middle				Bottom				Average			
Method	h=1	1–4	1–8	1–16	h=1	1–4	1–8	1–16	h=1	1–4	1–8	1–16	h=1	1–4	1–8	1–16
Base	9.6	10.7	12.6	15.6	6.3	7.3	8.6	10.8	6.4	7.5	8.3	9.8	6.8	7.9	9.0	10.9
BU	57.8	68.5	53.7	38.9	58.2	61.8	48.1	34.4	0.0	0.0	0.0	0.0	27.0	29.6	23.8	17.7
OLS	0.6	2.2	1.8	1.4	7.1	6.4	4.6	3.1	–7.6	–8.6	–8.2	–7.3	–2.1	–2.5	–2.7	–2.6
OLS-subset	0.6	**1.8**	**1.5**	**1.3**	7.2	**5.2**	**3.8**	**2.6**	**–8.3**	**–12.9**	**–11.6**	**–9.9**	**–2.4**	**–5.2**	**–4.8**	**–4.1**
OLS-parsim	0.8	2.6	2.1	1.8	7.5	**6.1**	**4.4**	**3.0**	**–9.0**	**–12.8**	**–11.6**	**–9.9**	**–2.7**	**–4.8**	**–4.5**	**–3.8**
OLS-lasso	0.6	2.2	1.8	1.6	7.4	6.7	4.8	3.2	–7.6	–8.5	–8.1	–7.2	–2.0	–2.4	–2.6	–2.5
WLSs	7.3	10.6	8.1	5.9	15.6	16.0	11.8	8.0	–6.9	–7.8	–7.4	–6.4	1.9	2.0	1.0	0.2
WLSs-subset	**5.0**	**5.7**	**4.6**	**3.6**	**12.3**	**10.0**	**7.5**	**5.2**	**–7.6**	**–10.5**	**–9.6**	**–8.2**	**0.2**	**–2.0**	**–2.1**	**–2.0**
WLSs-parsim	**7.1**	**9.2**	**7.1**	**5.2**	16.5	**15.5**	**11.5**	**7.9**	–6.8	**–9.2**	**–8.4**	**–7.3**	2.1	**0.9**	**0.1**	**–0.4**
WLSs-lasso	7.3	**10.3**	**8.0**	5.9	15.7	16.1	11.8	8.1	**–7.0**	–7.8	–7.3	–6.4	1.9	2.0	1.0	0.2
WLSv	1.0	2.9	2.3	1.9	4.5	4.3	3.2	2.1	–25.8	–26.4	–22.7	–18.3	–12.4	–12.6	–10.7	–8.4
WLSv-subset	**–1.0**	**0.3**	**0.4**	**0.5**	**0.6**	**0.6**	**0.5**	**0.3**	**–32.3**	**–32.2**	**–27.3**	**–21.7**	**–17.3**	**–17.3**	**–14.2**	**–10.9**
WLSv-parsim	**–0.5**	**0.2**	**0.3**	**0.5**	**0.9**	**0.7**	**0.5**	**0.3**	**–32.3**	**–32.3**	**–27.4**	**–21.7**	**–17.1**	**–17.3**	**–14.2**	**–10.9**
WLSv-lasso	**0.4**	**1.5**	**1.5**	**1.4**	**3.0**	**2.5**	**2.0**	**1.3**	**–28.5**	**–29.2**	**–24.9**	**–19.9**	**–14.4**	**–14.9**	**–12.3**	**–9.5**
MinT	–0.4	0.7	0.9	0.6	0.7	0.7	0.6	0.3	–32.9	–33.4	–28.3	–22.5	–17.5	–17.8	–14.6	–11.3
MinT-subset	**–0.6**	0.7	**0.8**	0.7	**0.6**	0.8	0.6	0.3	**–33.0**	–33.1	–28.0	–22.3	**–17.6**	–17.6	–14.5	–11.2
MinT-parsim	–0.4	0.7	0.9	0.6	0.7	0.7	0.6	0.3	–32.9	–33.4	–28.3	–22.5	–17.5	–17.8	–14.6	–11.3
MinT-lasso	**–0.7**	**0.3**	**0.6**	**0.4**	**0.3**	**0.4**	**0.4**	**0.1**	**–33.2**	**–33.7**	**–28.5**	**–22.6**	**–17.8**	**–18.1**	**–14.8**	**–11.4**
MinTs	–0.9	0.6	0.7	0.5	0.6	0.6	0.5	0.2	–32.9	–33.5	–28.3	–22.5	–17.6	–17.9	–14.6	–11.3
MinTs-subset	–0.7	0.9	1.1	1.0	0.7	0.8	0.7	0.4	**–33.0**	–33.1	–27.9	–22.2	–17.6	–17.5	–14.3	–11.0
MinTs-parsim	–0.9	0.6	0.7	0.5	0.6	0.6	0.5	0.2	–32.9	–33.5	–28.3	–22.5	–17.6	–17.9	–14.6	–11.3
MinTs-lasso	–0.9	**0.4**	**0.6**	0.5	0.6	**0.4**	**0.4**	**0.1**	**–33.2**	**–33.6**	**–28.4**	**–22.6**	**–17.7**	**–18.0**	**–14.8**	**–11.4**
EMinT	2.2	2.9	2.5	1.7	2.5	2.9	2.3	1.3	–31.9	–32.3	–27.5	–22.0	–15.9	–16.2	–13.4	–10.5
Elasso	**1.5**	**2.8**	**2.4**	1.7	**2.1**	**2.8**	2.3	1.3	**–32.1**	–32.2	–27.4	–21.9	**–16.3**	–16.2	–13.3	–10.5



(a) *MCB test*

(b) *Correlation between base forecast errors*

Figure B.2: *MCB test result and correlation matrix heatmap for simulation Setup 1, with model misspecification in series AA.*

Table B.4: *Proportion of time series being selected using proposed reconciliation methods for simulation Setup 1, with model misspecification in series AA. RMSSE values of base forecasts for each series are in parentheses. The last column displays a stacked barplot of the total selected series from 500 hierarchies, with darker sub-bars indicating higher counts.*

(RMSSE)	Total (1.48)	A (1.52)	B (1.55)	AA (3.45)	AB (1.62)	BA (1.62)	BB (1.57)	Summary
OLS-subset	0.52	0.79	0.57	0.79	1	0.91	0.85	
OLS-parsim	0.80	0.90	0.81	0.80	1	0.85	0.86	
OLS-lasso	0.90	1.00	0.68	1.00	1	1.00	1.00	
WLSs-subset	0.85	0.91	0.86	0.90	1	0.97	0.97	
WLSs-parsim	0.92	0.95	0.67	0.92	1	0.92	0.95	
WLSs-lasso	0.72	1.00	0.72	1.00	1	1.00	1.00	
WLSv-subset	0.50	0.62	0.42	0.19	1	0.81	0.87	
WLSv-parsim	0.59	0.55	0.49	0.17	1	0.76	0.86	
WLSv-lasso	0.40	1.00	0.41	0.77	1	1.00	1.00	
MinT-subset	0.66	0.90	0.61	0.72	1	0.91	0.93	
MinT-parsim	1.00	1.00	1.00	1.00	1	1.00	1.00	
MinT-lasso	0.80	0.96	0.84	0.72	1	0.98	0.97	
MinTs-subset	0.57	0.88	0.52	0.67	1	0.89	0.92	
MinTs-parsim	1.00	1.00	1.00	1.00	1	1.00	1.00	
MinTs-lasso	0.68	1.00	0.66	0.74	1	1.00	1.00	
Elasso	0.82	0.63	0.69	1.00	1	1.00	1.00	

C Results from simulation Setup 2

The section provides additional results obtained in Section 4.2.

Table C.1: *Proportion of time series selected using proposed reconciliation methods for simulation Setup 2, with error correlation being -0.4. RMSSE values of base forecasts for each series are in parentheses. The last column displays a stacked barplot of the total selected series from 500 hierarchies, with darker sub-bars indicating higher counts.*

(RMSSE)	Total (0.88)	A (0.84)	B (0.74)	AA (0.77)	AB (0.81)	BA (0.87)	BB (0.79)	Summary
OLS-subset	0.35	0.38	0.96	0.97	0.99	0.73	0.98	
OLS-parsim	0.59	0.62	0.93	0.92	1.00	0.61	0.96	
OLS-lasso	0.65	0.41	0.50	1.00	1.00	1.00	1.00	
WLSs-subset	0.35	0.43	0.99	1.00	1.00	0.73	0.99	
WLSs-parsim	0.56	0.54	0.97	0.98	1.00	0.70	0.98	
WLSs-lasso	0.62	0.66	0.82	1.00	1.00	1.00	1.00	
WLSv-subset	0.34	0.44	1.00	1.00	1.00	0.68	0.99	
WLSv-parsim	0.55	0.52	0.98	1.00	1.00	0.62	0.97	
WLSv-lasso	0.63	0.76	0.92	1.00	1.00	1.00	1.00	
MinT-subset	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinT-parsim	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinT-lasso	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinTs-subset	0.95	0.93	1.00	1.00	1.00	0.93	1.00	
MinTs-parsim	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinTs-lasso	0.96	0.94	1.00	1.00	1.00	0.96	1.00	
Elasso	0.93	0.73	0.98	1.00	0.99	1.00	1.00	

Table C.2: *Proportion of time series selected using proposed reconciliation methods for simulation Setup 2, with error correlation being 0. RMSSE values of base forecasts for each series are in parentheses. The last column displays a stacked barplot of the total selected series from 500 hierarchies, with darker sub-bars indicating higher counts.*

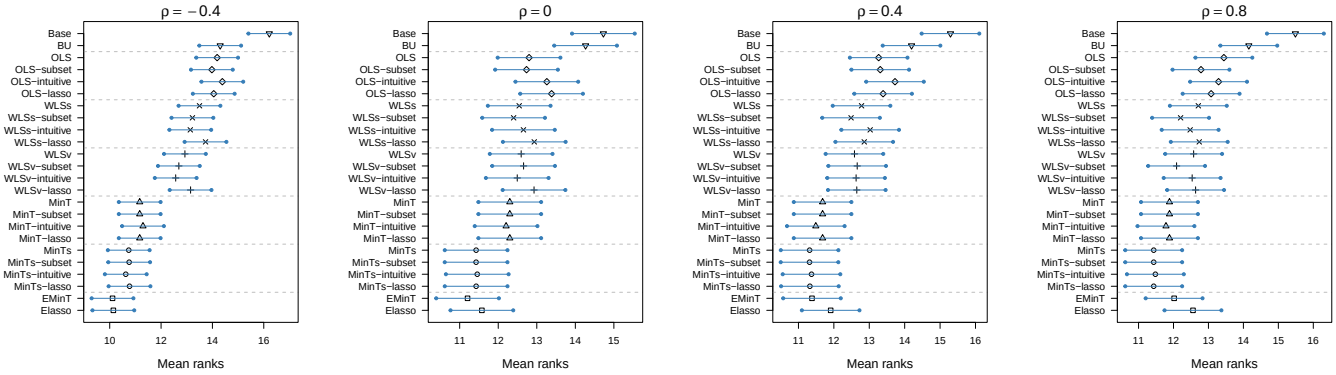
(RMSSE)	Total (0.90)	A (0.85)	B (0.76)	AA (0.86)	AB (0.83)	BA (0.74)	BB (0.80)	Summary
OLS-subset	0.37	0.49	0.95	0.94	0.99	0.73	0.95	
OLS-parsim	0.55	0.74	0.92	0.88	0.98	0.67	0.92	
OLS-lasso	0.70	0.57	0.61	1.00	1.00	1.00	1.00	
WLSs-subset	0.37	0.54	0.99	0.98	1.00	0.77	0.98	
WLSs-parsim	0.60	0.60	0.98	0.96	1.00	0.71	0.96	
WLSs-lasso	0.74	0.80	0.90	1.00	1.00	1.00	1.00	
WLSv-subset	0.35	0.55	1.00	0.99	1.00	0.78	0.95	
WLSv-parsim	0.60	0.58	0.99	1.00	1.00	0.86	0.84	
WLSv-lasso	0.76	0.82	0.95	1.00	1.00	1.00	1.00	
MinT-subset	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinT-parsim	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinT-lasso	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinTs-subset	1.00	0.99	1.00	1.00	1.00	0.99	1.00	
MinTs-parsim	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinTs-lasso	1.00	0.99	1.00	1.00	1.00	1.00	1.00	
Elasso	0.92	0.80	0.98	1.00	0.97	1.00	1.00	

Table C.3: *Proportion of time series selected using proposed reconciliation methods for simulation Setup 2, with error correlation being 0.4. RMSSE values of base forecasts for each series are in parentheses. The last column displays a stacked barplot of the total selected series from 500 hierarchies, with darker sub-bars indicating higher counts.*

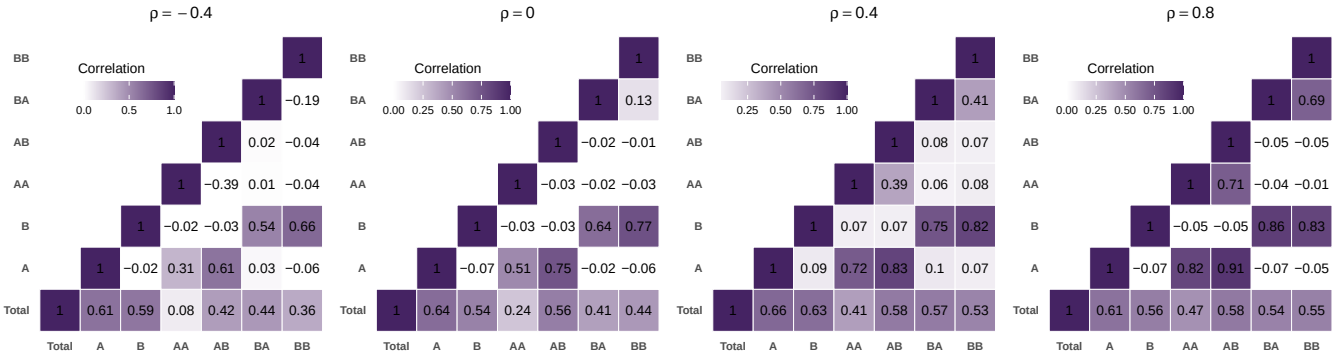
(RMSSE)	Total (0.99)	A (0.87)	B (0.78)	AA (0.83)	AB (0.77)	BA (0.81)	BB (0.81)	Summary
OLS-subset	0.37	0.53	0.98	0.92	0.98	0.81	0.90	
OLS-parsim	0.58	0.78	0.96	0.88	0.97	0.78	0.89	
OLS-lasso	0.73	0.59	0.64	1.00	1.00	1.00	1.00	
WLSs-subset	0.35	0.57	1.00	0.99	1.00	0.86	0.92	
WLSs-parsim	0.62	0.61	0.99	0.96	1.00	0.78	0.88	
WLSs-lasso	0.79	0.81	0.93	1.00	1.00	1.00	1.00	
WLSv-subset	0.35	0.58	1.00	1.00	1.00	0.89	0.85	
WLSv-parsim	0.61	0.53	1.00	1.00	0.99	0.96	0.73	
WLSv-lasso	0.83	0.89	0.98	1.00	1.00	1.00	1.00	
MinT-subset	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinT-parsim	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinT-lasso	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinTs-subset	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinTs-parsim	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinTs-lasso	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
Elasso	0.89	0.76	0.99	0.99	0.91	1.00	0.99	

Table C.4: *Proportion of time series selected using proposed reconciliation methods for simulation Setup 2, with error correlation being 0.8. RMSSE values of base forecasts for each series are in parentheses. The last column displays a stacked barplot of the total selected series from 500 hierarchies, with darker sub-bars indicating higher counts.*

(RMSSE)	Total (0.89)	A (0.85)	B (0.76)	AA (0.78)	AB (0.72)	BA (0.77)	BB (0.75)	Summary
OLS-subset	0.33	0.52	0.96	0.95	0.98	0.96	0.78	
OLS-parsim	0.54	0.77	0.93	0.89	0.97	0.83	0.85	
OLS-lasso	0.69	0.53	0.60	1.00	1.00	1.00	1.00	
WLSs-subset	0.29	0.60	1.00	1.00	1.00	0.98	0.86	
WLSs-parsim	0.63	0.67	0.99	0.98	1.00	0.93	0.86	
WLSs-lasso	0.69	0.76	0.91	1.00	1.00	1.00	1.00	
WLSv-subset	0.32	0.55	1.00	1.00	1.00	0.99	0.76	
WLSv-parsim	0.58	0.56	1.00	1.00	0.98	1.00	0.75	
WLSv-lasso	0.77	0.84	0.99	1.00	1.00	1.00	1.00	
MinT-subset	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinT-parsim	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinT-lasso	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinTs-subset	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinTs-parsim	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
MinTs-lasso	1.00	1.00	1.00	1.00	1.00	1.00	1.00	
Elasso	0.73	0.65	0.98	0.98	0.86	1.00	0.99	



(a) *MCB test*



(b) *Correlation between base forecast errors*

Figure C.1: *MCB test result and correlation matrix heatmap for simulation Setup 2, with error correlation being -0.4, 0, 0.4, 0.8, respectively.*

D Results from Australian labour force data

Table D.1: *Out-of-sample forecast results on a single test set (from August 2022 to July 2023) for Australian labour force data. The Base row shows average RMSE of base forecasts, while entries below show RMSE percentage decrease (negative) or increase (positive) for reconciliation methods. Blue entries highlight the best-performing methods; bold entries indicate proposed methods outperforming bechmarks.*

Method	Top	Duration	STT	Duration x STT	Average
Base	28.3	16.9	6.3	2.9	5.1
BU	–45.0	–10.1	–13.5	0.0	–9.3
OLS	–10.4	–0.6	0.3	4.9	1.1
OLS-subset	**–38.4**	**–2.7**	**–7.3**	**3.7**	**–3.8**
OLS-parsim	**–32.1**	0.3	13.2	11.6	4.4
OLS-lasso	–10.4	–0.6	0.3	4.9	1.1
WLSs	–38.6	–5.9	–8.0	1.6	–5.9
WLSs-subset	**–50.4**	**–7.8**	**–12.8**	5.9	**–6.4**
WLSs-parsim	–38.6	–5.9	–8.0	1.6	–5.9
WLSs-lasso	–38.6	–5.9	–8.0	1.6	–5.9
WLSv	–36.6	–4.8	–7.0	3.1	–4.6
WLSv-subset	–29.6	**–10.9**	**–7.3**	**1.8**	**–6.5**
WLSv-parsim	–36.6	–4.8	–7.0	3.1	–4.6
WLSv-lasso	–36.6	–4.8	–7.0	3.1	–4.6
MinTs	–21.8	–4.6	–7.6	3.4	–3.1
MinTs-subset	**–45.1**	**–6.8**	**–9.3**	3.6	**–6.1**
MinTs-parsim	–21.8	–4.6	–7.6	3.4	–3.1
MinTs-lasso	–21.8	–4.6	–7.6	3.4	–3.1
EMinT	–29.9	–6.7	**–21.0**	10.1	–3.7
Elasso	–8.0	**–25.3**	–14.6	**–4.9**	**–13.2**

Table D.2: *Number of time series selected by proposed methods and the optimal parameters identified in the labour application, considering a single test set (from August 2022 to July 2023). The Base row shows the original number of series in the structure. The numbers in parentheses show RMSSE values of base forecasts for selected series across different levels.*

	Number of time series retained					Optimal parameters		
	Top	Duration	STT	Duration x STT	Total	λ	λ_0	λ_2
Base	1 (0.39)	6 (1.07)	8 (0.57)	48 (0.90)	63	-	-	-
OLS-subset	0 (0.00)	5 (1.01)	1 (0.42)	48 (0.90)	54	-	4.16	1
WLSs-subset	0 (0.00)	5 (1.01)	1 (0.70)	46 (0.90)	52	-	0.38	0.1
WLSv-subset	1 (0.39)	5 (1.03)	7 (0.54)	48 (0.90)	61	-	0.51	1
MinTs-subset	0 (0.00)	1 (0.75)	1 (0.42)	47 (0.90)	49	-	0.03	0.01
Elasso	1 (0.39)	5 (1.20)	2 (0.57)	3 (0.88)	11	213.59	-	-

"None" or "Base" in Table D.2. XC — Now use Base instead.

E Results from Australian domestic tourism data

Table E.1: *Out-of-sample forecast results on a single test set (from January 2017 to December 2017) for Australian domestic tourism data. The Base row shows average RMSE of base forecasts, while entries below show RMSE percentage decrease (negative) or increase (positive) for reconciliation methods. Blue entries highlight the best-performing methods; bold entries indicate proposed methods outperforming bechmarks.*

Method	Top	State	Zone	Region	Average
Base	1907.6	424.8	179.7	94.1	152.1
BU	42.0	19.7	1.0	0.0	8.5
OLS	1.4	–1.5	–7.5	–3.2	–3.6
OLS-subset	**–14.1**	**–7.9**	**–8.7**	–2.3	**–6.5**
OLS-parsim	1.4	–1.5	–7.5	–3.2	–3.6
OLS-lasso	1.4	–1.5	–7.5	–3.2	–3.6
WLSs	19.1	6.2	–5.0	–2.2	0.9
WLSs-subset	**–7.8**	**–6.2**	**–8.5**	**–2.5**	**–5.5**
WLSs-parsim	19.1	6.2	–5.0	–2.2	0.9
WLSs-lasso	19.1	6.2	–5.0	–2.2	0.9
WLSv	25.6	9.9	–3.2	–2.1	2.8
WLSv-subset	**8.5**	**1.4**	**–6.7**	**–3.4**	**–2.2**
WLSv-parsim	25.6	9.9	–3.2	–2.1	2.8
WLSv-lasso	25.6	9.9	–3.2	–2.1	2.8
MinTs	19.0	6.3	–5.3	–3.1	0.4
MinTs-subset	**5.6**	**–0.1**	**–7.8**	**–3.9**	**–3.3**
MinTs-parsim	19.0	6.3	–5.3	–3.1	0.4
MinTs-lasso	19.0	6.3	–5.3	–3.1	0.4
EMinT	–13.7	47.5	54.2	78.2	55.5
Elasso	**–16.4**	**–11.4**	**–10.4**	**0.3**	**–6.7**

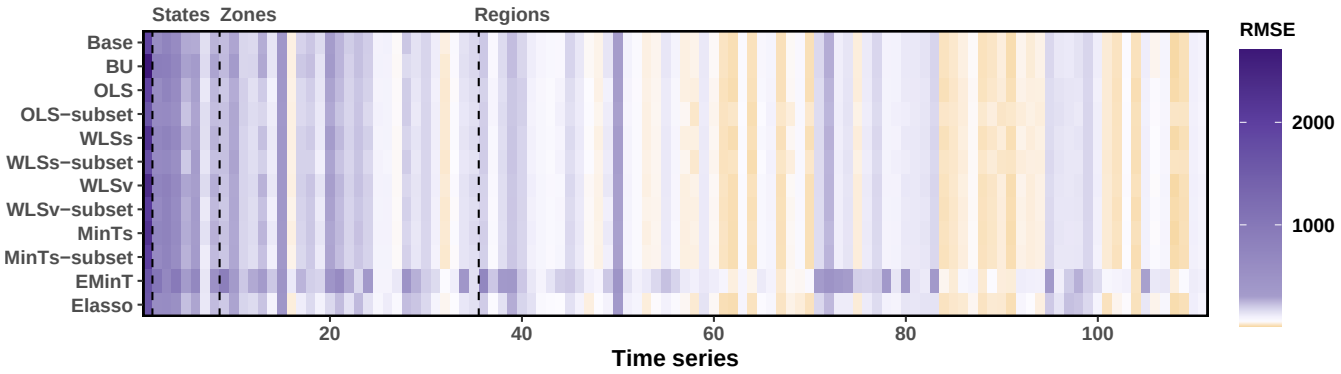


Figure E.1: *Average out-of-sample forecasting performance, measured in terms of RMSE (from 1- to 12-step-ahead), for each series across different reconciliation methods. Time series are arranged along the horizontal axis.*

Table E.2: *Number of time series selected by proposed methods and the optimal parameters identified in the tourism application, considering a single test set (from January 2017 to December 2017). The Base row shows the original number of series in the structure. The numbers in parentheses show RMSSE values of base forecasts for selected series across different levels.*

	Number of time series retained					Optimal parameters		
	Top	State	Zone	Region	Total	λ	λ_0	λ_2
Base	1 (1.19)	7 (1.09)	27 (1.13)	76 (1.17)	111	-	-	-
OLS-subset	1 (1.19)	2 (0.77)	13 (1.09)	76 (1.17)	92	-	27.98	10
WLSs-subset	1 (1.19)	1 (0.62)	15 (1.08)	76 (1.17)	93	-	18.73	10
WLSv-subset	1 (1.19)	7 (1.09)	27 (1.13)	76 (1.17)	111	-	0.03	0.01
MinTs-subset	1 (1.19)	7 (1.09)	27 (1.13)	76 (1.17)	111	-	0.05	0.01
Elasso	1 (1.19)	4 (1.04)	0 (0.00)	8 (0.86)	13	71759.21	-	-

"None" or "Base" in Table E.2. XQ — Now use Base instead