Forecast Reconciliation Made Easy: The `FoReco` Package

Daniele Girolimetto
Department of Statistical Sciences, University of Padova

- danigiro.github.io
- danigiro
- daniele.girolimetto@unipd.it

FoReco

July 10th, 2024

useR! conference

useR! Salzburg 2024

Forecast reconciliation
Introduction

- Post-forecasting process aimed to improve the quality of the base forecasts (however obtained) of a linearly constrained multiple time series by exploiting **cross-sectional** (e.g., spatial) and/or **temporal** constraints of the target forecasts

Target	Base forecasts		Reconciled forecasts
$\boldsymbol{C}\boldsymbol{y}_h = \boldsymbol{0}$	$\boldsymbol{C}\widehat{\boldsymbol{y}}_h \neq \boldsymbol{0}$	$\rightarrow$	$\boldsymbol{C}\widetilde{\boldsymbol{y}}_h = \boldsymbol{0}$

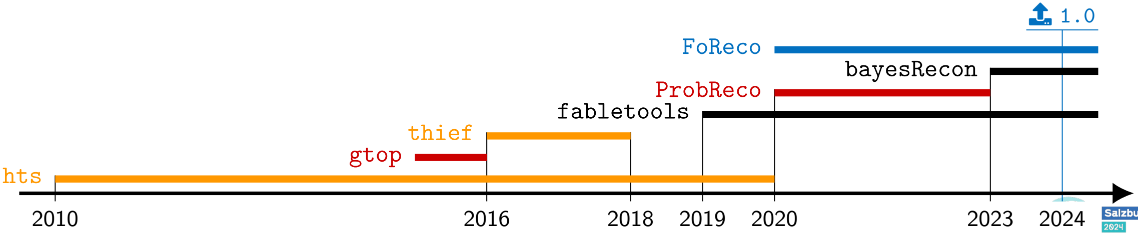
- **Forecasting examples**: Sales, Tourism, Energy demand, Healthcare, Supply chain ...
- The R package `FoReco` offers a robust set of tools for implementing forecast reconciliation with a variety of approaches accounting for different constraints
- Additional resources and examples on GitHub and on the documentation page

Present and future of reconciliation packages

in **R** (R Core Team, 2024)

	hts	gtop	thief	ProbReco	fable	bayesRecon	FoReco
Cross-sectional	✓	✓		✓	✓	✓	✓
Temporal			✓			✓	✓
Cross-temporal							✓
Probabilistic				✓	✓	✓	✓

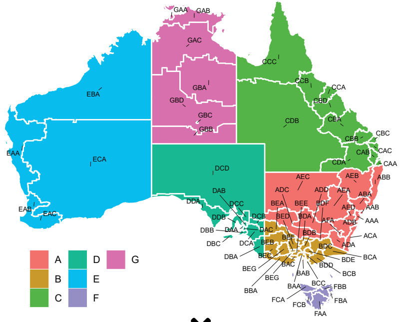
Supported Not updated Archived



Australian Tourism Demand

Today example: `data(vndata)`

Geographical division, g.d.



×

Purpose of travel, p.o.t.

Holiday, Visiting friends and relatives, Business, Other

- **Grouped ts** (geographical divisions × purpose of travel)

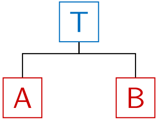
	AUS	States	Zones*	Regions	Tot
g.d.	1	7	21	76	105
p.o.t.	4	28	84	304	420
Tot	5	35	105	380	**525**

$n_a = 221$, $n_b = 304$, and $\mathbf{n} = \mathbf{525}$

- **Unique time series, no redundancy** (*6 Zones with only one Region are included in the Regions)
- **Temporal framework**, frequencies:
  - Monthly
  - Bi-Monthly
  - Quarterly
  - Four-Monthly
  - Semi-Annual
  - Annual

Cross-sectional framework

Hyndman *et al.* (2011); Panagiotelis *et al.* (2021)



2-level: bottom and upper

A cross-sectional hierarchical/grouped time series is a collection of n variables for which - at each time - **aggregation relationships** hold.

$\Downarrow$

multiple time series with exact **linear constraints**.

- Two equivalent representations (Girolimetto and Di Fonzo, 2024)

Zero-constrained

$$\boldsymbol{C}_{cs}\boldsymbol{y} = \boldsymbol{0}_{n_a}$$

$$\boldsymbol{C}_{cs} = \begin{bmatrix} \boldsymbol{I} & -\boldsymbol{A} \end{bmatrix}$$

not unique

$$\begin{bmatrix} 1 & -1 & -1 \end{bmatrix}$$

$$\boldsymbol{A} : \quad \boldsymbol{a} = \boldsymbol{A}\boldsymbol{b}$$

Linear combination (or aggregation) matrix

$$\begin{bmatrix} 1 & 1 \end{bmatrix}$$

Structural

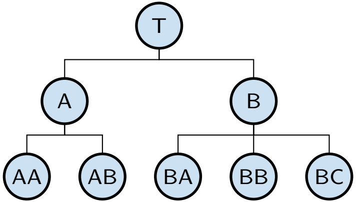
$$\boldsymbol{y} = \boldsymbol{S}_{cs}\boldsymbol{b}$$

$$\boldsymbol{S}_{cs} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{I}_{n_b} \end{bmatrix}$$

$$\begin{bmatrix} 1 \quad 1 \\ \boldsymbol{I}_2 \end{bmatrix}$$

Hierarchical, grouped and linearly constrained time series

Genuine hierarchical time series



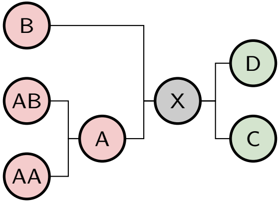
Constraints

$T = A + B$

$A = AA + AB$

$B = BA + BB + BC$

General linearly constrained time series



Constraints

$X = A + B$

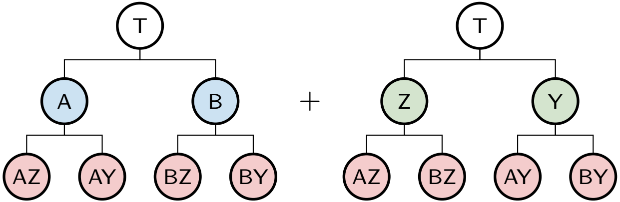
$X = C + D$

$A = AA + AB$

Grouped time series: two or more genuine hierarchies sharing the same top and bottom variables

T	A	B
Z	AZ	BZ
Y	AY	BY

=



Constraints

$T = A + B$

$A = AZ + AY$

$B = BZ + BY$

$T = Z + Y$

$Z = AZ + BZ$

$Y = AY + BY$

Optimal combination forecast reconciliation

Wickramasuriya *et al.* (2019); Panagiotelis *et al.* (2021)

1. Forecast **all series at all levels** of aggregation $\rightarrow$ base forecasts
2. Make the base forecasts **coherent** using least squares $\rightarrow$ reconciled forecasts

Two equivalent point forecast reconciliation formulae

Structural reconciliation approach

`approach = "strc"`

$$\widehat{\boldsymbol{y}} = \boldsymbol{S}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$
$$\Downarrow$$
$$\widetilde{\boldsymbol{y}} = \boldsymbol{S}\left(\boldsymbol{S}'\boldsymbol{W}^{-1}\boldsymbol{S}\right)^{-1}\boldsymbol{S}'\boldsymbol{W}^{-1}\widehat{\boldsymbol{y}} = \boldsymbol{S}\boldsymbol{G}\widehat{\boldsymbol{y}}$$

Projection reconciliation approach

`approach = "proj"`

$$\widehat{\boldsymbol{y}} = \boldsymbol{y} + \boldsymbol{\varepsilon}, \quad \text{s.t.} \quad \boldsymbol{C}\boldsymbol{y} = 0$$
$$\Downarrow$$
$$\widetilde{\boldsymbol{y}} = \left[\boldsymbol{I} - \boldsymbol{W}\boldsymbol{C}'\left(\boldsymbol{C}\boldsymbol{W}\boldsymbol{C}'\right)^{-1}\boldsymbol{C}\right]\widehat{\boldsymbol{y}} = \boldsymbol{M}\widehat{\boldsymbol{y}}$$

- The formulation of $\boldsymbol{W} = \mathrm{E}(\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}')$ is conceptually complex; in practice, approximate forms are used, possibly using in-sample residuals

Projection vs structural approach: performance index

Cross-sectional wls reconciliation - *time* in seconds - median of 100 replications

'wls' reconciliation

Number of bottom time series \ Number of upper time series	1	5	25	50	100	250	500	750	1000	1500	2500
2500	82.6	81.53	77.13	75.95	73.56	68.79	65.23	60.95	53.19	37.2	1.9
1500	72.15	70.89	68.74	68.81	66.89	65.46	52.09	40.52	27.1	2.21	40.63
1000	53.13	52.58	49.77	41.33	50.45	37.03	30.93	15.8	3.95	35.44	56.62
750	32.86	34.18	33.12	39.1	31.31	19.85	9.42	1.7	20.22	48.7	68.58
500	22.83	26.2	27.8	26.78	22.77	13.03	8.12	17.66	33.38	58.39	74.01
250	13.16	15.3	17.26	12.63	8.2	1.66	10.05	19.69	38.51	63.72	80.62
100	$\boxed{0}$	$\boxed{0}$	9.67	$\boxed{0}$	7.08	14.09	30.35	24.14	53.29	67.53	82.23
50	$\boxed{0}$	$\boxed{0}$	$\boxed{0}$	13.63	18.3	16.8	33.63	27	51.84	69.56	82.37
25	$\boxed{0}$	$\boxed{0}$	$\boxed{0}$	21.17	28.61	21.29	32.48	24.87	54.17	71.38	82.87
5	$\boxed{0}$	$\boxed{0}$	$\boxed{0}$	$\boxed{0}$	32.17	30.79	32.06	24.89	56.58	72.24	82.49

‘strc‘ < ‘proj‘ ‘strc‘ > ‘proj‘ |‘strc‘−‘proj‘| < 0.001

Two main factors:

- dimensions

  cs: n_a, n_b

  te: $\mathcal{K}$ (set of temporal aggregation orders)

  ct: n_a, n_b, $\mathcal{K}$
- computational cost of $\boldsymbol{W}^{-1}$

$$\mathbf{PI} = \left| \frac{time_{\text{strc}} - time_{\text{proj}}}{\max\left(time_{\text{strc}}, time_{\text{proj}}\right)} \right| 100$$

FoReco at work </>

Cross-sectional optimal forecast reconciliation

Input:

`base`	(12×525) monthly base forecasts
`comb`	A string specifying the reconciliation method (e.g., ols, str, wls, shr, sam)
`res`	(228×525) residuals possibly used to compute the covariance matrix
`agg_mat`	(221×304) cross-sectional aggregation matrix $\boldsymbol{A}$
`cons_mat`	(221×525) zero constraints cross-sectional matrix $\boldsymbol{C}$

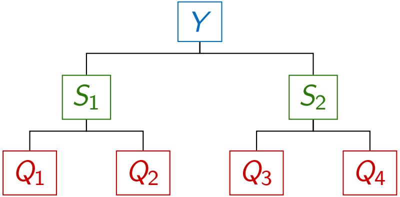
FoReco

```r
# Using the aggregation matrix A
rf_opt <- csrec(base = base, agg_mat = vnaggmat, res = res, comb = "shr")
str(rf_opt, give.attr = FALSE)
#>  num [1:12, 1:525] 49160 21622 24815 29433 23260 ...

# Using the zero constraints matrix C: vnconsmat <- cbind(diag(221), -vnaggmat)
csrec(base = base, cons_mat = vnconsmat, res = res, comb = "shr")
```

Temporal framework

Athanasopoulos *et al.* (2017)



Quarterly hierarchy:
quarterly, semi-annual and annual series

Temporal hierarchy → non-overlapping aggregation of the observations of a time series (y) at regular intervals

$$x_j^{[k]} = \sum_{t=(j-1)k+1}^{jk} y_t$$

- Unlike cross-sectional hierarchies (n **variables at the same time index** are considered), in temporal hierarchies we have **one variable observed at different frequencies**
- Structural representation $\left(\boldsymbol{x}_\tau = \boldsymbol{S}_{te}\boldsymbol{x}_\tau^{[1]}\right)$ and zero-constrained representation $\left(\boldsymbol{C}_{te}\boldsymbol{x}_\tau = \boldsymbol{0}_{(k^* \times 1)}\right)$ still hold, and may be alternatively used for reconciliation

FoReco at work </>

Cross-sectional optimal forecast reconciliation

Input:

`base`	(28×1) base forecasts ordered as $\left[x^{[12]}\ \boldsymbol{x}^{[6]}\ \ldots\ \boldsymbol{x}^{[1]}\right]'$
`comb`	A string specifying the reconciliation method (e.g., ols, str, wlsv, ...)
`res`	(228×1) residuals possibly used to compute the covariance matrix
`agg_order`	max. order of temporal aggregation, $m = 12$

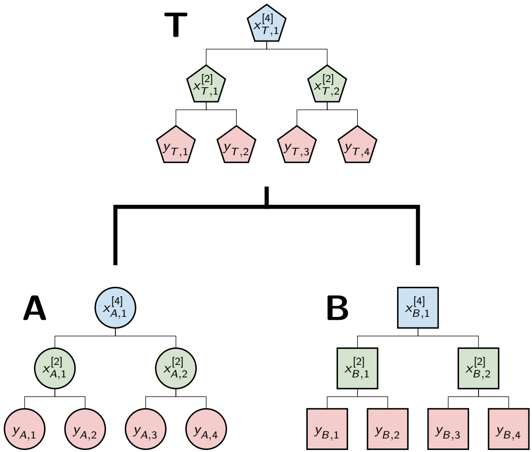
FoReco

```r
rf_opt <- terec(base = base, agg_order = m, res = res, comb = "sar1")
str(rf_opt, give.attr = FALSE)
#>  Named num [1:28] 1.329 0.665 0.665 0.443 0.443 ...
```

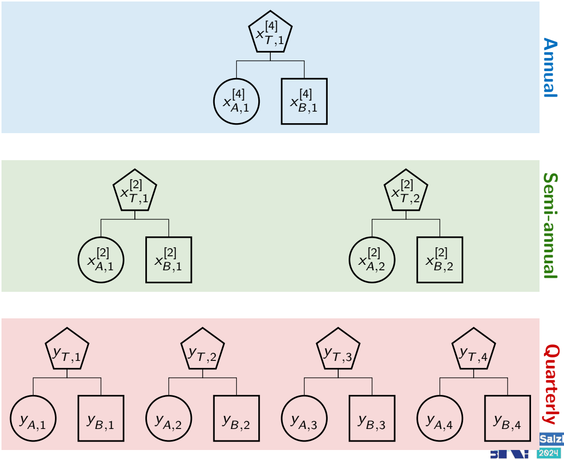
Cross-sectional + Temporal = Cross-temporal

A cross-temporal hierarchy of three quarterly time series ($T = A + B$)

cross-sectional $\longrightarrow$ temporal



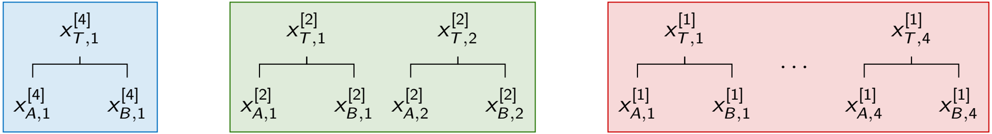
temporal $\longrightarrow$ cross-sectional



Cross-temporal framework

Di Fonzo and Girolimetto (2023a)

Annual: $j = 1$ **Semi-annual:** $j = 1, 2$ **Quarterly:** $j = 1, \dots, 4$



$$\left.\begin{array}{l} \boldsymbol{x}_{i,\tau} = \begin{bmatrix} x^{[4]}_{i,1} & x^{[2]}_{i,1} & x^{[2]}_{2} & x^{[1]}_{i,1} & \dots & x^{[1]}_{i,4} \end{bmatrix}' \\ \text{for } i = T, A, B \end{array}\right\} \Rightarrow \quad \boldsymbol{X}_\tau = \begin{bmatrix} \boldsymbol{x}'_{T,\tau} \\ \boldsymbol{x}'_{A,\tau} \\ \boldsymbol{x}'_{B,\tau} \end{bmatrix}, \quad \boldsymbol{x}_\tau = \text{vec}\left(\boldsymbol{X}'_\tau\right)$$

- Two dimensions to capture the complete nature of a multiple time series
- Any cross-temporal matrix may be constructed from the one-dimensional counterparts

  $\boldsymbol{C}_{ct} \rightarrow$ easy to compute as a function of $\boldsymbol{A}_{cs}$ and m

  $\boldsymbol{S}_{ct} \rightarrow$ fast to compute as $\boldsymbol{S}_{cs} \otimes \boldsymbol{S}_{te}$

FoReco at work </>

Cross-temporal optimal forecast reconciliation

Input:

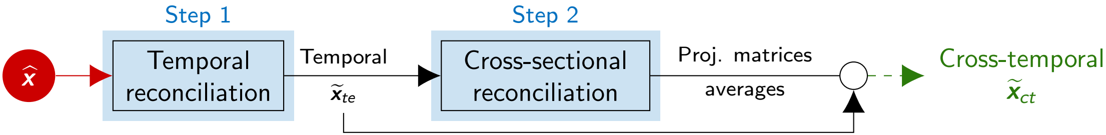
`base`	(525×28) base forecasts matrix
`comb`	A string specifying the reconciliation method (e.g., ols, wlsv, bdshr, ...)
`res`	(525×532) residuals possibly used to compute the covariance matrix
`agg_order`	max. order of temporal aggregation, $m = 12$
`agg_mat`	(221×304) cross-sectional aggregation matrix $\boldsymbol{A}$
`cons_mat`	(221×525) zero constraints cross-sectional matrix $\boldsymbol{C}$

FoReco

```r
rf_opt <- ctrec(base = base, agg_mat = vnaggmat, # or cons_mat = vnconsmat,
                agg_order = m, res = res, comb = "wlsv", approach = "strc")
str(rf_opt, give.attr=FALSE)
#>  num [1:525, 1:28] 314297 97383 62631 77793 19695 ...
```

FoReco at work

Two-step cross-temporal reconciliation, Kourentzes and Athanasopoulos (2019)



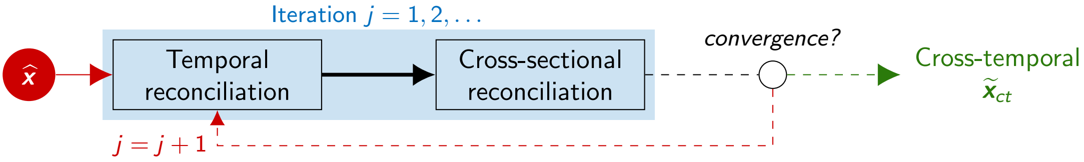
- The final temporally **and** cross-sectionally coherent reconciled forecasts are a transformation of the step 1 forecasts through the step 2 reconciliation matrices
- `tcsrec( )` → *first-temporal-then-cross-sectional reconciliation*
  `cstrec( )` → *first-cross-sectional-then-temporal reconciliation*

FoReco

```r
tcsrec(base = base, res = res,
       cslist = list(agg_mat = vnaggmat, comb = "shr"),
       telist = list(agg_order = m, comb = "wlsv"))
```

FoReco at work </>

Iterative cross-temporal reconciliation, Di Fonzo and Girolimetto (2023a)



- Each iteration consists in the first two steps of the heuristic KA procedure, until a convergence criterion is met

FoReco

```r
iterec(base = base, res = res,
       type = "tcs", # → first-temporal-then-cross-sectional reconciliation
       # or type = "cst", → first-cross-sectional-then-temporal reconciliation
       cslist = list(agg_mat = vnaggmat, comb = "shr"),
       telist = list(agg_order = m, comb = "wlsv"))
```

Additional reconciliation approaches

Classical and LCC approaches for cross-sectional, temporal and cross-temporal frameworks

- **Classical reconciliation** (Dunn *et al.*, 1976; Gross and Sohl, 1990; Athanasopoulos *et al.*, 2009)

	Cross-sectional	*Temporal*	*Cross-Temporal*
Top-down, *`td( )`	`cstd( )`	`tetd( )`	`cttd( )`
Bottom-up, *`bu( )`	`csbu( )`	`tebu( )`	`ctbu( )`
Middle-out, *`mo( )`	`csmo( )`	`temo( )`	`ctmo( )`

- **Level conditional coherent reconciliation** (Hollyman *et al.*, 2021; Di Fonzo and Girolimetto, 2024)

	Cross-sectional	*Temporal*	*Cross-Temporal*
LCC, *`lcc( )`	`cslcc( )`	`telcc( )`	`ctlcc( )`

Practical challenges and other features

- **Non-negative forecast reconciliation** (Wickramasuriya *et al.*, 2020)
  - `nn = "osqp"`: quadratic programming optimization (Stellato *et al.*, 2020)
  - `nn = "sntz"`: heuristic "set-negative-to-zero" (Di Fonzo and Girolimetto, 2023b)
- **Probabilistic forecast reconciliation** (Panagiotelis *et al.*, 2023; Girolimetto *et al.*, 2024)
  - Non-parametric approach: `*boot( )` + `*rec( )`
  - Parametric approach (samples): `MASS::mvrnorm( )` + `*rec( )`
- Reconciliation with immutable forecasts (Zhang *et al.*, 2023) using the `immutable` argument
- Using a subset of temporal aggregation orders, e.g. `agg_order = c(12, 6, 1)`

Conclusions

- FoReco is a valuable tool for researchers and practitioners in the field of time series forecasting
- It provides state-of-the-art classical (bottom-up and top-down) as well as modern (optimal and heuristic combination) reconciliation approaches for cross-sectional, temporal, and cross-temporal frameworks
- Its flexible and comprehensive design makes it a versatile solution for a wide range of forecasting applications (Sales, Tourism, Energy demand, Healthcare, Supply chain ...)
- **Links:**
  - github.com/daniGiro/FoReco
  - danigiro.github.io/FoReco
  - cran.r-project.org/package=FoReco

Awesome Forecast Reconciliation - Github repository

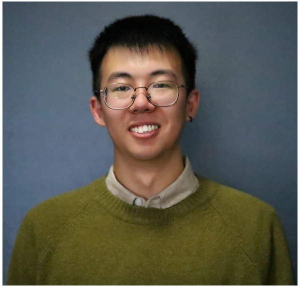
Resources about forecast reconciliation: `danigiro/awesome-forecast-reconciliation`

Awesome Forecast Reconciliation

This repository serves as a curated reference for the domain of forecast reconciliation. It aims to contain an extensive collection of academic papers, articles, software tools, and educational resources. Ideal for researchers, analysts, and practitioners seeking to improve the consistency and precision of forecasting methodologies.

We wish to express our deep appreciation to the authors of the paper "Forecast reconciliation: A review" - George Athanasopoulos, Rob J Hyndman, Nikolaos Kourentzes, and Anastasios Panagiotelis - for providing their BibTeX file, which served as the cornerstone of this repository. Their paper serves as an invaluable resource with its comprehensive and insightful analysis of the forecast reconciliation field, providing a thorough overview of the existing literature and highlighting key advancements and research trends.

⚠️ The list is still incomplete and unorganized. We are in the process of reorganizing the various items.



Yangzhuoran Fin Yang
Monash University

Github repository

THANK YOU!

Daniele Girolimetto
Department of Statistical Sciences, University of Padova
danigiro.github.io
github.com/daniGiro
daniele.girolimetto@unipd.it

useR! conference

Salzburg, Austria - July 10^{th}, 2024

References I

Athanasopoulos, G., Ahmed, R. A. and Hyndman, R. J. (2009) Hierarchical forecasts for Australian domestic tourism. *International Journal of Forecasting* **25**(1), 146–166

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

Di Fonzo, T. and Girolimetto, D. (2023a) Cross-temporal forecast reconciliation: Optimal combination method and heuristic alternatives. *International Journal of Forecasting* **39**(1), 39–57

Di Fonzo, T. and Girolimetto, D. (2023b) Spatio-temporal reconciliation of solar forecasts. *Solar Energy* **251**, 13–29

Di Fonzo, T. and Girolimetto, D. (2024) Forecast combination-based forecast reconciliation: Insights and extensions. *International Journal of Forecasting* **40**(2), 490–514

Dunn, D. M., Williams, W. H. and Dechaine, T. L. (1976) Aggregate versus Subaggregate Models in Local Area Forecasting. *Journal of the American Statistical Association* **71**(353), 68–71

Girolimetto, D., Athanasopoulos, G., Di Fonzo, T. and Hyndman, R. J. (2024) Cross-temporal probabilistic forecast reconciliation: Methodological and practical issues. *International Journal of Forecasting* **40**(3), 1134–1151

Girolimetto, D. and Di Fonzo, T. (2024) Point and probabilistic forecast reconciliation for general linearly constrained multiple time series. *Statistical Methods & Applications* **33**, 581–607

Gross, C. W. and Sohl, J. E. (1990) Disaggregation methods to expedite product line forecasting. *Journal of Forecasting* **9**(3), 233–254

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

References II

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

Kourentzes, N. and Athanasopoulos, G. (2019) Cross-temporal coherent forecasts for Australian tourism. *Annals of Tourism Research* **75**, 393–409

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

Panagiotelis, A., Gamakumara, P., Athanasopoulos, G. and Hyndman, R. J. (2023) Probabilistic forecast reconciliation: Properties, evaluation and score optimisation. *European Journal of Operational Research* **306**(2), 693–706

R Core Team (2024) *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria **URL:** `https://www.R-project.org/`

Stellato, B., Banjac, G., Goulart, P., Bemporad, A. and Boyd, S. (2020) OSQP: an operator splitting solver for quadratic programs. *Mathematical Programming Computation* **12**(4), 637–672

Wickramasuriya, S. L., Athanasopoulos, G. and Hyndman, R. J. (2019) Optimal Forecast Reconciliation for Hierarchical and Grouped Time Series Through Trace Minimization. *Journal of the American Statistical Association* **114**(526), 804–819

Wickramasuriya, S. L., Turlach, B. A. and Hyndman, R. J. (2020) Optimal non-negative forecast reconciliation. *Statistics and Computing* **30**(5), 1167–1182

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