Forecast reconciliation

2. Perspectives on forecast reconciliation

Rob J Hyndman

MONASH University

Photo by Edvard Alexander Rølvaag on Unsplash

robjhyndman.com/fr2023

Outline

1 Time series reconciliation
2 Reconciliation via constraints
3 Example: reconciling GDP forecasts
4 The geometry of forecast reconciliation
5 Optimization and reconcilation
6 In-built coherence
7 Time series cross-validation

Outline

1 Time series reconciliation
2 Reconciliation via constraints
3 Example: reconciling GDP forecasts
4 The geometry of forecast reconciliation
5 Optimization and reconcilation
6 In-built coherence
7 Time series cross-validation

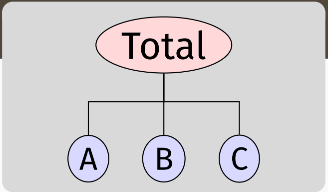
Time series reconciliation

- Stone, Champernowne, and Meade (1942): reconciling national economic accounts (disaggregated into production, income, outlay, capital transactions, etc.)
- Byron (1978): extended Stone's work using more computationally efficient methods.
- 1984: Stone wins Nobel Prize in Economics.
- Same approach used for reconciling seasonally adjusted data.
- Chow and Lin (1971): Temporal reconciliation of monthly or quarterly estimates to sum to annual estimates.
- Di Fonzo (1990): Cross-temporal reconciliation of time series data.

Outline

1. Time series reconciliation
2. Reconciliation via constraints
3. Example: reconciling GDP forecasts
4. The geometry of forecast reconciliation
5. Optimization and reconcilation
6. In-built coherence
7. Time series cross-validation

Notation reminder



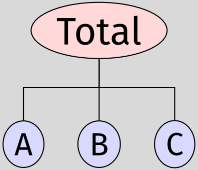
Every collection of time series with linear constraints can be written as

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

- $\boldsymbol{y}_t$ = vector of all series at time t
- $y_{\text{Total},t}$ = aggregate of all series at time t.
- $y_{X,t}$ = value of series X at time t.
- $\boldsymbol{b}_t$ = vector of most disaggregated series at time t
- $\boldsymbol{S}$ = "summing matrix" containing the linear constraints.

$$\boldsymbol{y}_t = \begin{pmatrix} y_{\text{Total},t} \\ y_{A,t} \\ y_{B,t} \\ y_{C,t} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\boldsymbol{S}} \underbrace{\begin{pmatrix} y_{A,t} \\ y_{B,t} \\ y_{C,t} \end{pmatrix}}_{\boldsymbol{b}_t}$$

Notation reminder



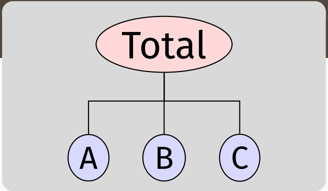
Every collection of time series with linear constraints can be written as

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

- $\boldsymbol{y}_t$ = vector of all series at time t
- $y_{\text{Total},t}$ = aggregate of all series at time t.
- $y_{X,t}$ = value of series X at time t.
- $\boldsymbol{b}_t$ = vector of most disaggregated series at time t
- $\boldsymbol{S}$ = "summing matrix" containing the linear constraints.

- Base forecasts: $\hat{\boldsymbol{y}}_{T+h|T}$
- Reconciled forecasts: $\tilde{\boldsymbol{y}}_{T+h|T} = \boldsymbol{S}\boldsymbol{G}\hat{\boldsymbol{y}}_{T+h|T}$
- MinT: $\boldsymbol{G} = (\boldsymbol{S}'\boldsymbol{W}_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{W}_h^{-1}$ where $\boldsymbol{W}_h$ is covariance matrix of base forecast errors.

Notation



Aggregation matrix

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

$$\begin{pmatrix} y_{\text{Total},t} \\ y_{A,t} \\ y_{B,t} \\ y_{C,t} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_{A,t} \\ y_{B,t} \\ y_{C,t} \end{pmatrix}$$

$$\begin{pmatrix} \boldsymbol{a}_t \\ \boldsymbol{b}_t \end{pmatrix} = \begin{pmatrix} \boldsymbol{A} \\ \boldsymbol{I}_{n_b} \end{pmatrix} \boldsymbol{b}_t$$

Constraint matrix

$$\boldsymbol{C}\boldsymbol{y}_t = \boldsymbol{0}$$

where

$$\begin{aligned} \boldsymbol{C} &= \begin{bmatrix} 1 & -1 & -1 & -1 \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{I}_{n_a} & -\boldsymbol{A} \end{bmatrix} \end{aligned}$$

Zero-constraint representation

Aggregation matrix A

$$\boldsymbol{y}_t = \begin{bmatrix} \boldsymbol{a}_t \\ \boldsymbol{b}_t \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{I}_{n_b} \end{bmatrix} \boldsymbol{b}_t = \boldsymbol{S}\boldsymbol{b}_t$$

Zero-constraint representation

Aggregation matrix $\boldsymbol{A}$

$$\boldsymbol{y}_t = \begin{bmatrix} \boldsymbol{a}_t \\ \boldsymbol{b}_t \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{I}_{n_b} \end{bmatrix} \boldsymbol{b}_t = \boldsymbol{S}\boldsymbol{b}_t$$

Constraint matrix $\boldsymbol{C}$

$$\boldsymbol{C}\boldsymbol{y}_t = \boldsymbol{0}$$

- Constraint matrix approach more general & more parsimonious.
- $\boldsymbol{C} = [\boldsymbol{I}_{n_a} \quad -\boldsymbol{A}]$.
- $\boldsymbol{S}$, $\boldsymbol{A}$ and $\boldsymbol{C}$ may contain any real values (not just 0s and 1s).

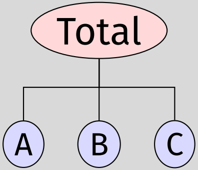
Zero-constraint representation

Assuming $\boldsymbol{C}$ is full rank

$$\tilde{\boldsymbol{y}}_{T+h|T} = \boldsymbol{M}\hat{\boldsymbol{y}}_{T+h|T}$$
$$\text{where} \qquad \boldsymbol{M} = \boldsymbol{I} - \boldsymbol{W}_h\boldsymbol{C}'(\boldsymbol{C}\boldsymbol{W}_h\boldsymbol{C}')^{-1}\boldsymbol{C}$$

- Originally proved by Byron (1978) for reconciling data.
- Re-discovered by Wickramasuriya, Athanasopoulos, and Hyndman (2019) for reconciling forecasts.
- $\boldsymbol{M} = \boldsymbol{S}\boldsymbol{G}$ (the MinT solution)
- Leads to more efficient reconciliation than using $\boldsymbol{G}$.

Zero-constraint representation



Suppose $\boldsymbol{W}_h = \boldsymbol{I}$. Then

$$\boldsymbol{M} = \boldsymbol{I} - \boldsymbol{W}_h\boldsymbol{C}'(\boldsymbol{C}\boldsymbol{W}_h\boldsymbol{C}')^{-1}\boldsymbol{C}$$
$$= \begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{pmatrix} - \begin{pmatrix}1\\-1\\-1\\-1\end{pmatrix}\frac{1}{4}\begin{pmatrix}1&-1&-1&-1\end{pmatrix}$$
$$= \begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{pmatrix} - \begin{pmatrix}1&-1&-1&-1\\-1&1&1&1\\-1&1&1&1\\-1&1&1&1\end{pmatrix}$$
$$= \begin{pmatrix}\frac{3}{4}&\frac{1}{4}&\frac{1}{4}&\frac{1}{4}\\\frac{1}{4}&\frac{3}{4}&-\frac{1}{4}&-\frac{1}{4}\\\frac{1}{4}&-\frac{1}{4}&\frac{3}{4}&-\frac{1}{4}\\\frac{1}{4}&-\frac{1}{4}&-\frac{1}{4}&\frac{3}{4}\end{pmatrix}$$

$$\boldsymbol{A} = \begin{pmatrix}1&1&1\end{pmatrix}$$
$$\boldsymbol{S} = \begin{pmatrix}\boldsymbol{A}\\\boldsymbol{I}_{n_b}\end{pmatrix} = \begin{pmatrix}1&1&1\\1&0&0\\0&1&0\\0&0&1\end{pmatrix}$$
$$\boldsymbol{C} = \begin{pmatrix}\boldsymbol{I}_{n_a} & -\boldsymbol{A}\end{pmatrix} = \begin{pmatrix}1&-1&-1&-1\end{pmatrix}$$

Outline

1 Time series reconciliation

2 Reconciliation via constraints

3 Example: reconciling GDP forecasts

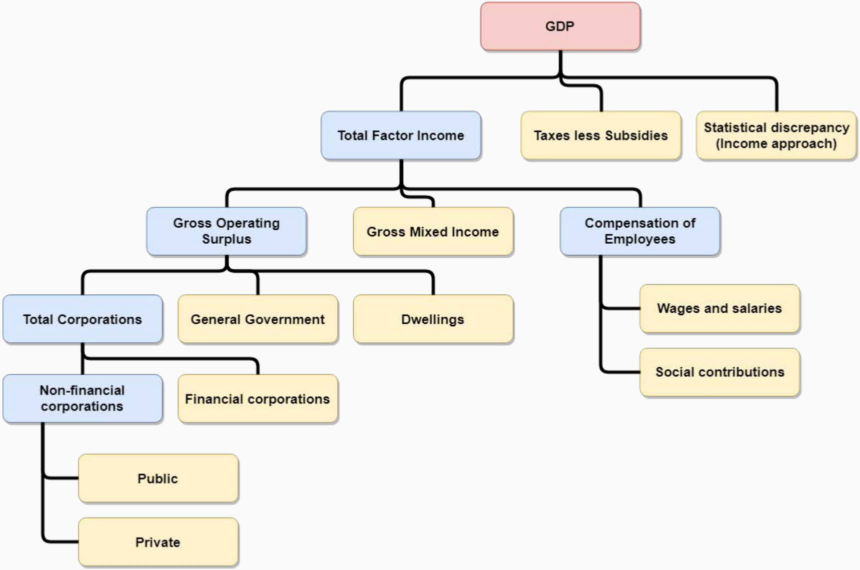
4 The geometry of forecast reconciliation

5 Optimization and reconcilation

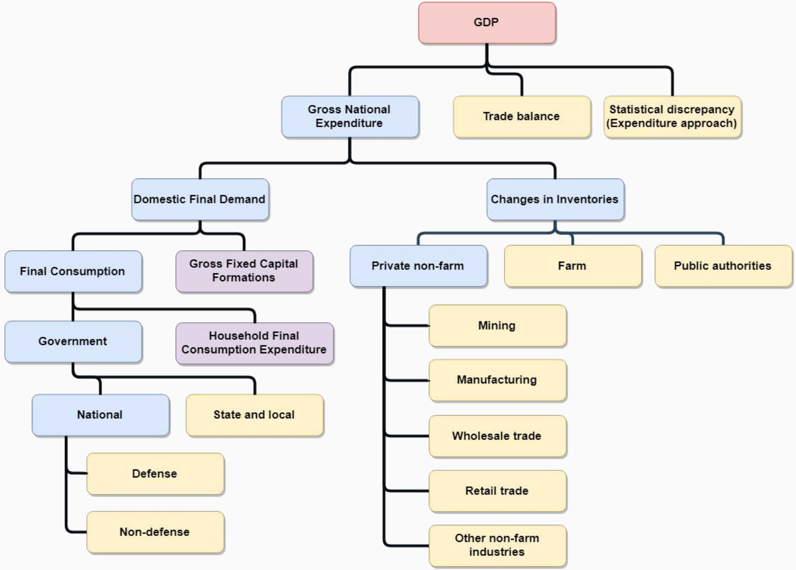
6 In-built coherence

7 Time series cross-validation

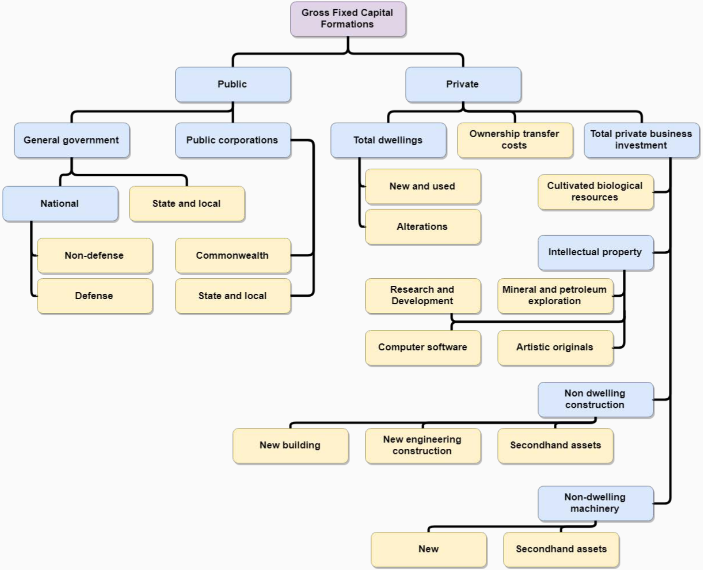
Example: reconciling GDP forecasts



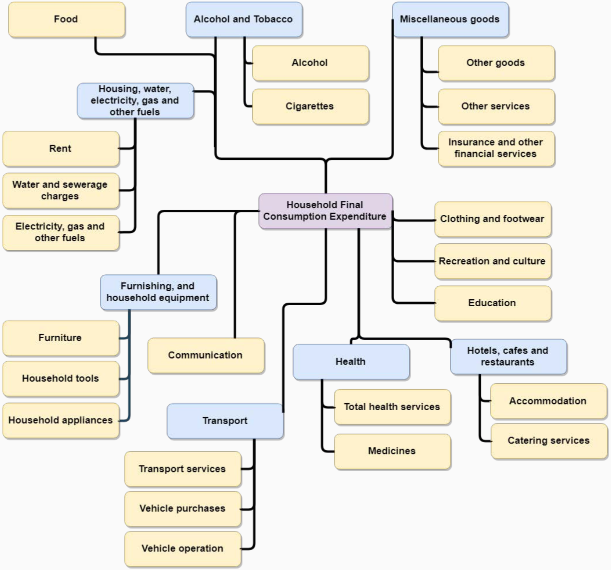
Example: reconciling GDP forecasts



Example: reconciling GDP forecasts



Example: reconciling GDP forecasts



Example: reconciling GDP forecasts

- No unique hierarchy.
- Several disaggregations with the same parent node
- Not possible to represent using structural $\boldsymbol{S}$ notation.
- Instead, we can use the constraint $\boldsymbol{C}$ notation.

Example: reconciling GDP forecasts

Using structural notation:

$$\boldsymbol{y}_t^I = \begin{bmatrix} x_t \\ \boldsymbol{a}_t^I \\ \boldsymbol{b}_t^I \end{bmatrix} = \boldsymbol{S}^I \boldsymbol{b}_t^I \qquad \boldsymbol{y}_t^E = \begin{bmatrix} x_t \\ \boldsymbol{a}_t^E \\ \boldsymbol{b}_t^E \end{bmatrix} = \boldsymbol{S}^E \boldsymbol{b}_t^E$$

where

$$\boldsymbol{S}^I = \begin{bmatrix} \boldsymbol{1}_{10}' \\ \boldsymbol{A}^I \\ \boldsymbol{I}_{10} \end{bmatrix} \qquad \boldsymbol{S}^E = \begin{bmatrix} \boldsymbol{1}_{53}' \\ \boldsymbol{A}^E \\ \boldsymbol{I}_{53} \end{bmatrix}$$

- Can reconcile both trees, but the totals won’t be equal.

Example: reconciling GDP forecasts

Using constraint notation:

$$\boldsymbol{C}\boldsymbol{y}_t = \boldsymbol{0}$$

where

$$\boldsymbol{y}_t = \begin{bmatrix} x_t \\ \boldsymbol{a}_t^I \\ \boldsymbol{b}_t^I \\ \boldsymbol{a}_t^E \\ \boldsymbol{b}_t^E \end{bmatrix} \quad \text{and} \quad \boldsymbol{C} = \begin{bmatrix} 1 & \boldsymbol{0}_5' & -\boldsymbol{1}_{10}' & \boldsymbol{0}_{26}' & \boldsymbol{0}_{53}' \\ 1 & \boldsymbol{0}_5' & \boldsymbol{0}_{10}' & \boldsymbol{0}_{26}' & -\boldsymbol{1}_{53}' \\ \boldsymbol{0}_5 & \boldsymbol{I}_5 & -\boldsymbol{A}^I & \boldsymbol{0}_{5\times 26} & \boldsymbol{0}_{5\times 53} \\ \boldsymbol{0}_{26} & \boldsymbol{0}_{26\times 5} & \boldsymbol{0}_{26\times 10} & \boldsymbol{I}_{26} & -\boldsymbol{A}^E \end{bmatrix}$$

Ref: Bisaglia, Di Fonzo, and Girolimetto (2020)

Outline

1 Time series reconciliation

2 Reconciliation via constraints

3 Example: reconciling GDP forecasts

4 The geometry of forecast reconciliation

5 Optimization and reconcilation

6 In-built coherence

7 Time series cross-validation

The coherent subspace

Coherent subspace

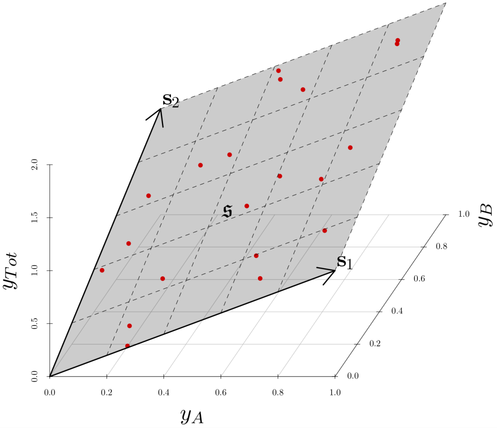
n_b-dimensional linear subspace $\mathfrak{s} \subset \chi^n$ for which linear constraints hold for all $\boldsymbol{y} \in \mathfrak{s}$.

Hierarchical time series

An n-dimensional multivariate time series such that $\boldsymbol{y}_t \in \mathfrak{s} \quad \forall t$.

Coherent point forecasts

$\tilde{\boldsymbol{y}}_{t+h|t}$ is *coherent* if $\tilde{\boldsymbol{y}}_{t+h|t} \in \mathfrak{s}$.



$y_{Tot} = y_A + y_B$

The coherent subspace

Coherent subspace

n_b-dimensional linear subspace $\mathfrak{s} \subset \chi^n$ for which linear constraints hold for all $\boldsymbol{y} \in \mathfrak{s}$.

Hierarchical time series

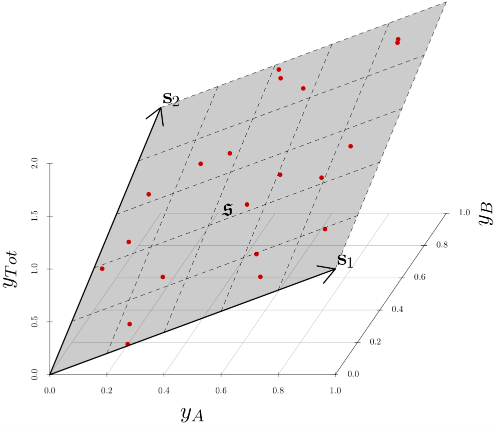
An n-dimensional multivariate time series such that $\boldsymbol{y}_t \in \mathfrak{s} \quad \forall t$.

Coherent point forecasts

$\tilde{\boldsymbol{y}}_{t+h|t}$ is *coherent* if $\tilde{\boldsymbol{y}}_{t+h|t} \in \mathfrak{s}$.

Base forecasts

Let $\hat{\boldsymbol{y}}_{t+h|t}$ be vector of *incoherent* initial h-step forecasts.



$y_{Tot} = y_A + y_B$

The coherent subspace

Coherent subspace

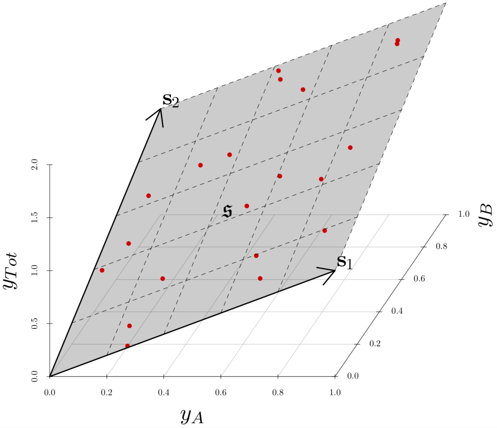
n_b-dimensional linear subspace $\mathfrak{s} \subset \chi^n$ for which linear constraints hold for all $\boldsymbol{y} \in \mathfrak{s}$.

Hierarchical time series

An n-dimensional multivariate time series such that $\boldsymbol{y}_t \in \mathfrak{s} \quad \forall t$.

Coherent point forecasts

$\tilde{\boldsymbol{y}}_{t+h|t}$ is *coherent* if $\tilde{\boldsymbol{y}}_{t+h|t} \in \mathfrak{s}$.



$y_{Tot} = y_A + y_B$

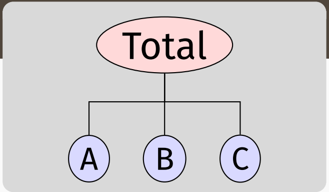
Base forecasts

Let $\hat{\boldsymbol{y}}_{t+h|t}$ be vector of *incoherent* initial h-step forecasts.

Reconciled forecasts

Let ψ be a mapping, $\psi : \chi^n \rightarrow \mathfrak{s}$.
$\tilde{\boldsymbol{y}}_{t+h|t} = \psi(\hat{\boldsymbol{y}}_{t+h|t})$ “reconciles” $\hat{\boldsymbol{y}}_{t+h|t}$.

The coherent subspace

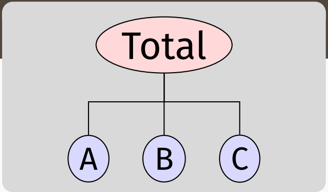


The columns of $\boldsymbol{S}$ form a basis set for $\mathfrak{s}$.

They are not unique.

Each corresponds to different vector of "bottom-level" series.

The coherent subspace

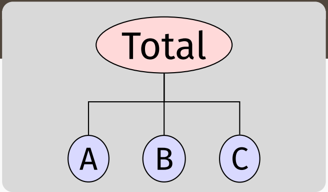


The columns of $\boldsymbol{S}$ form a basis set for $\mathfrak{s}$.

They are not unique.
Each corresponds to different vector of “bottom-level” series.

$$\boldsymbol{y} = \begin{pmatrix} \text{Total} \\ A \\ B \\ C \end{pmatrix} \qquad \boldsymbol{S} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \boldsymbol{b} = \begin{pmatrix} A \\ B \\ C \end{pmatrix}$$

The coherent subspace

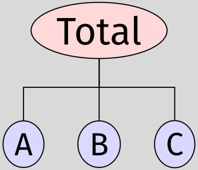


The columns of $\boldsymbol{S}$ form a basis set for $\mathfrak{s}$.

They are not unique.
Each corresponds to different vector of “bottom-level” series.

$$\boldsymbol{y} = \begin{pmatrix} \text{Total} \\ A \\ B \\ C \end{pmatrix} \qquad \boldsymbol{S} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & -1 & -1 \end{pmatrix} \qquad \boldsymbol{b} = \begin{pmatrix} \text{Total} \\ B \\ A \end{pmatrix}$$

The coherent subspace



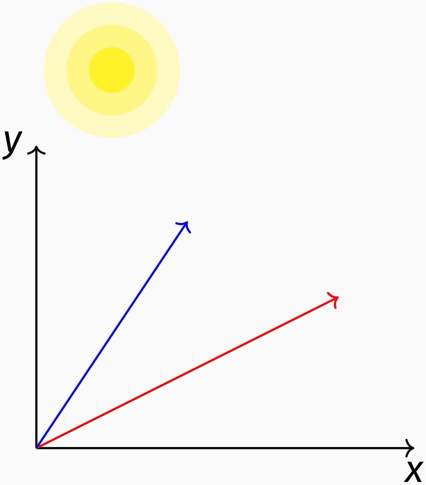
The columns of $\boldsymbol{S}$ form a basis set for $\mathfrak{s}$.

They are not unique.

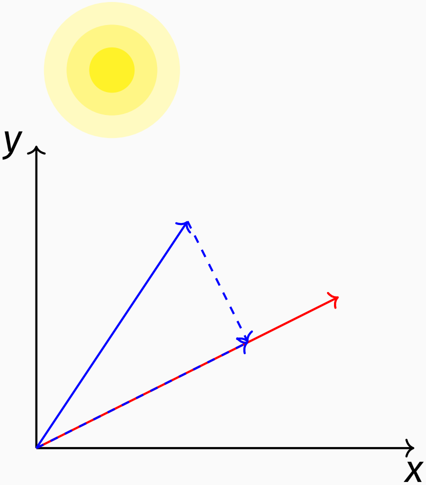
Each corresponds to different vector of "bottom-level" series.

$$\boldsymbol{y} = \begin{pmatrix} \text{Total} \\ A \\ B \\ C \end{pmatrix} \qquad \boldsymbol{S} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & -1 \\ -1 & 1 & 1 \\ 1 & -1 & 0 \end{pmatrix} \qquad \boldsymbol{b} = \begin{pmatrix} \text{Total} \\ B + A \\ C + B \end{pmatrix}$$

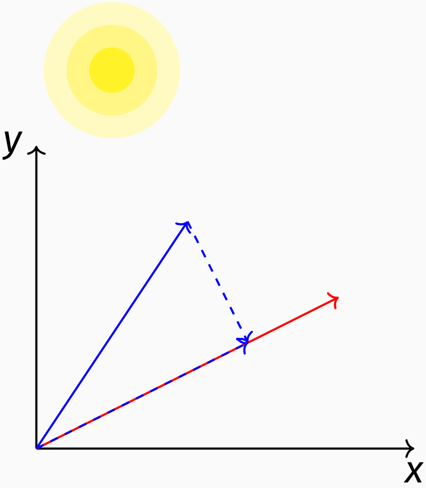
Projections in linear algebra

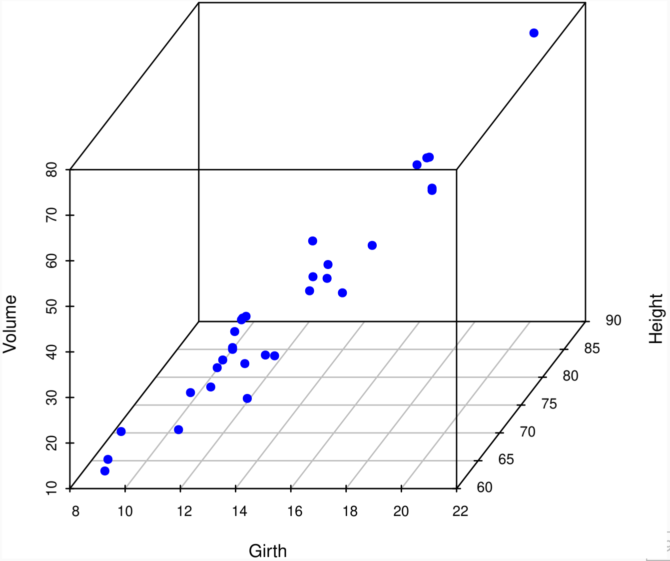


Projections in linear algebra

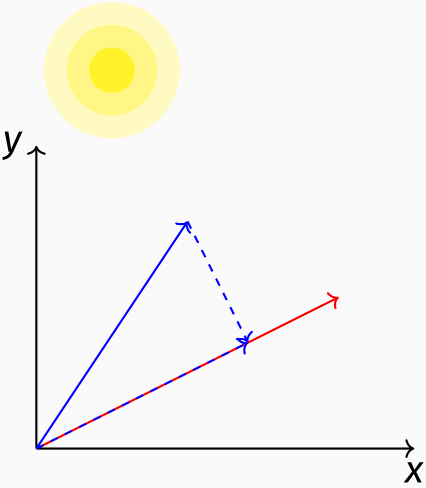


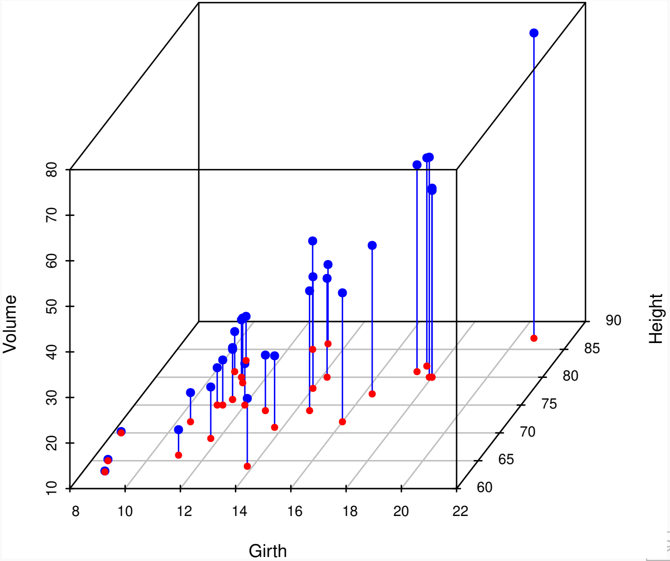
Projections in linear algebra



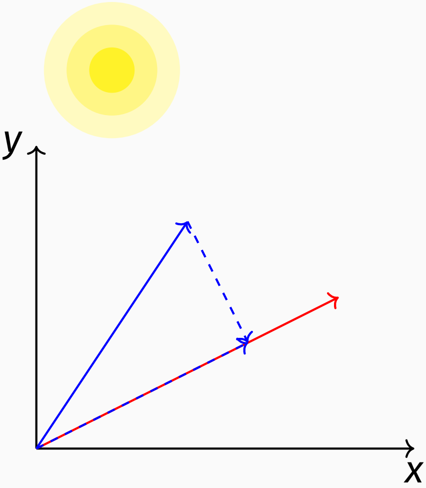


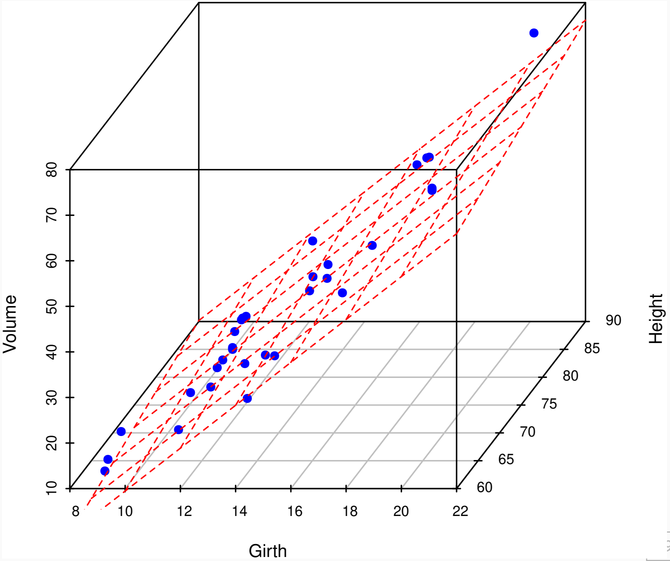
Projections in linear algebra



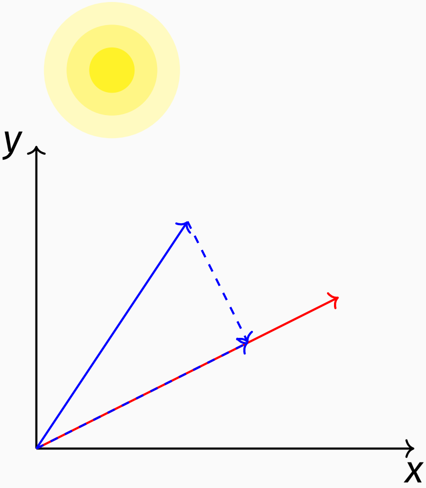


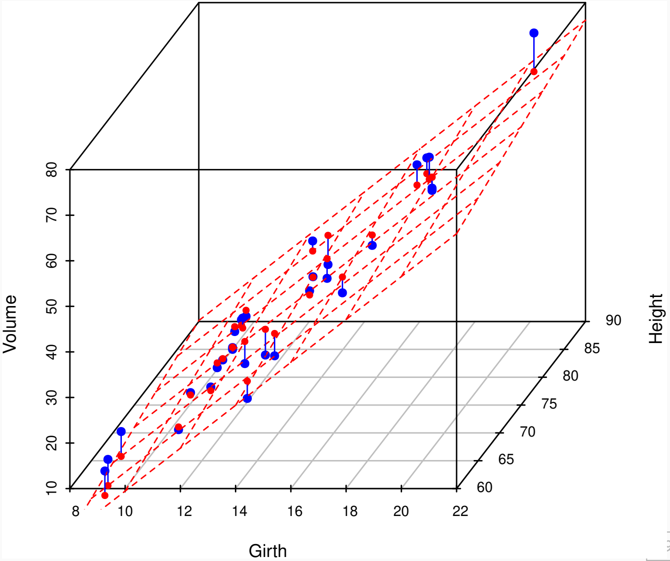
Projections in linear algebra





Projections in linear algebra



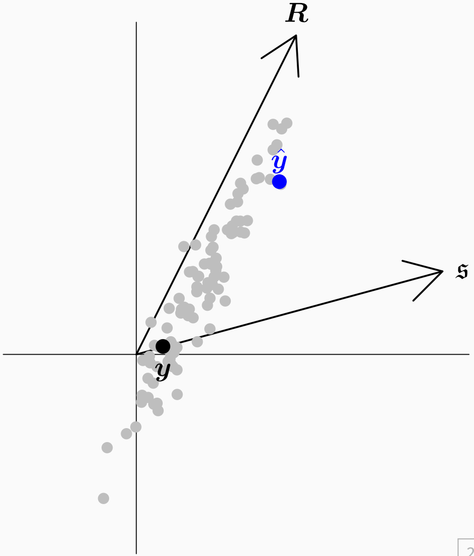


Projections in linear algebra

- A projection is a linear transformation $\boldsymbol{M}$ such that $\boldsymbol{M}^2 = \boldsymbol{M}$.
- i.e., M is idempotent: it leaves its image unchanged.
- $\boldsymbol{M}$ projects onto $\mathfrak{s}$ if $\boldsymbol{M}\boldsymbol{y} = \boldsymbol{y}$ for all $\boldsymbol{y} \in \mathfrak{s}$.
- All eigenvalues of $\boldsymbol{M}$ are either 0 or 1.
- All singular values of $\boldsymbol{M}$ are greater than or equal to 1 (with equality iff $\boldsymbol{M}$ is orthogonal).
- A projection is *orthogonal* if $\boldsymbol{M}' = \boldsymbol{M}$.
- If a projection is not orthogonal, it is called *oblique*.
- In regression, OLS is an orthogonal projection onto space spanned by predictors.

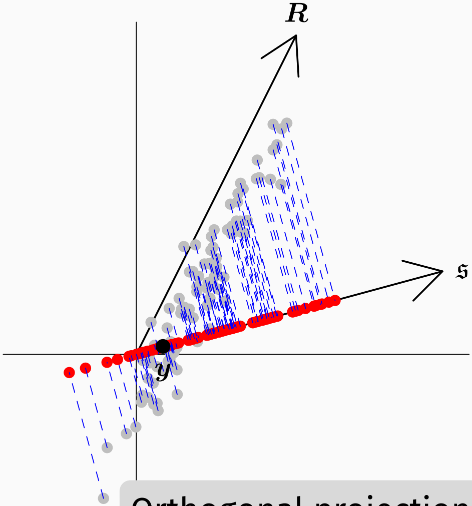
Linear projection reconciliation

- R is the most likely direction of deviations from $\mathfrak{s}$.
- Grey: potential base forecasts



Linear projection reconciliation

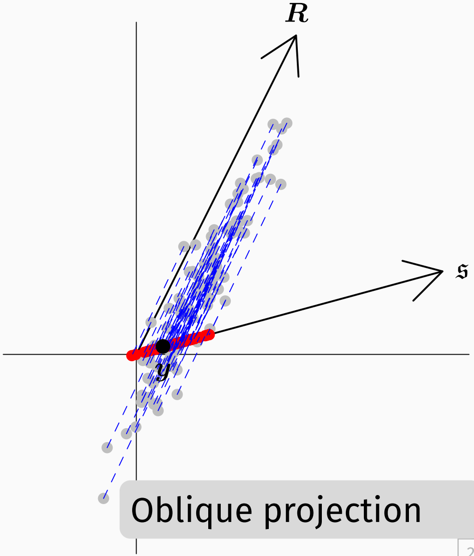
- R is the most likely direction of deviations from $\mathfrak{s}$.
- Grey: potential base forecasts
- Red: reconciled forecasts
- Orthogonal projections (i.e., OLS) lead to smallest possible adjustments of base forecasts.



Orthogonal projection

Linear projection reconciliation

- R is the most likely direction of deviations from $\mathfrak{s}$.
- Grey: potential base forecasts
- Red: reconciled forecasts
- Orthogonal projections (i.e., OLS) lead to smallest possible adjustments of base forecasts.
- Oblique projections (i.e., MinT) give reconciled forecasts with smallest variance.



Oblique projection

Linear projection reconciliation

$$\tilde{\boldsymbol{y}}_{t+h|t} = \psi(\hat{\boldsymbol{y}}_{t+h|t}) = \boldsymbol{M}\hat{\boldsymbol{y}}_{t+h|t}$$

- $\boldsymbol{M}$ is a projection onto $\mathfrak{s}$ if and only if $\boldsymbol{M}\boldsymbol{y} = \boldsymbol{y}$ for all $\boldsymbol{y} \in \mathfrak{s}$.
- Coherent base forecasts are unchanged since $\boldsymbol{M}\hat{\boldsymbol{y}} = \hat{\boldsymbol{y}}$
- If $\hat{\boldsymbol{y}}$ is unbiased, then $\tilde{\boldsymbol{y}}$ is also unbiased since

  $$\text{E}(\tilde{\boldsymbol{y}}_{t+h|t}) = \text{E}(\boldsymbol{M}\hat{\boldsymbol{y}}_{t+h|t}) = \boldsymbol{M}\text{E}(\hat{\boldsymbol{y}}_{t+h|t}) = \text{E}(\hat{\boldsymbol{y}}_{t+h|t}),$$

  and unbiased estimates must lie on $\mathfrak{s}$.
- The projection is orthogonal if and only if $\boldsymbol{M}' = \boldsymbol{M}$.
- If $\boldsymbol{S}$ forms a basis set for $\mathfrak{s}$, then projections are of the form $\boldsymbol{M} = \boldsymbol{S}(\boldsymbol{S}'\Psi\boldsymbol{S})^{-1}\boldsymbol{S}'\Psi$ where Ψ is a positive definite matrix.

Linear projection reconciliation

$$\tilde{\boldsymbol{y}}_{t+h|t} = \psi(\hat{\boldsymbol{y}}_{t+h|t}) = \boldsymbol{M}\hat{\boldsymbol{y}}_{t+h|t}, \qquad \text{where} \quad \boldsymbol{M} = \boldsymbol{S}(\boldsymbol{S}'\Psi\boldsymbol{S})^{-1}\boldsymbol{S}'\Psi$$

$$\begin{array}{llll} \text{OLS:} & \Psi = \boldsymbol{I} & \boldsymbol{M} = \boldsymbol{S}(\boldsymbol{S}'\boldsymbol{S})^{-1}\boldsymbol{S}' & = \boldsymbol{I} - \boldsymbol{C}'(\boldsymbol{C}\boldsymbol{C}')^{-1}\boldsymbol{C} \\ \text{MinT:} & \Psi = \boldsymbol{W}_h & \boldsymbol{M} = \boldsymbol{S}(\boldsymbol{S}'\boldsymbol{W}_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{W}_h^{-1} & = \boldsymbol{I} - \boldsymbol{W}_h\boldsymbol{C}'(\boldsymbol{C}\boldsymbol{W}_h\boldsymbol{C}')^{-1}\boldsymbol{C} \end{array}$$

- $\boldsymbol{M}$ is orthogonal iff $\Psi = \boldsymbol{I}$.
- $\boldsymbol{W}_h = \text{Var}[\boldsymbol{y}_{T+h} - \hat{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_T]$ is the covariance matrix of the base forecast errors.
- $\boldsymbol{V}_h = \text{Var}[\boldsymbol{y}_{T+h} - \tilde{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_T] = \boldsymbol{M}\boldsymbol{W}_h\boldsymbol{M}'$ is minimized when $\Psi = \boldsymbol{W}_h$.

Mean square error bounds

Panagiotelis et al. (2021)

Distance reducing property

Let $\|\boldsymbol{u}\|_\Psi = \boldsymbol{u}'\Psi\boldsymbol{u}$. Then

$$\|\boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}_{t+h|t}\|_\Psi \le \|\boldsymbol{y}_{t+h} - \hat{\boldsymbol{y}}_{t+h|t}\|_\Psi$$

- Ψ-projection is guaranteed to improve forecast accuracy over base forecasts *using this distance measure*.
- Distance reduction holds for any realisation and any forecast.
- OLS reconciliation minimizes Euclidean distance.
- Other measures of forecast accuracy may be worse.

Mean square error bounds

Wickramasuriya (2021)

$$\begin{aligned}\|\boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}_{t+h}\|_2^2 &= \|\boldsymbol{M}(\boldsymbol{y}_{t+h} - \hat{\boldsymbol{y}}_{t+h})\|_2^2 \\ &\leq \|\boldsymbol{M}\|_2^2 \|\boldsymbol{y}_{t+h} - \hat{\boldsymbol{y}}_{t+h}\|_2^2 \\ &= \sigma_{\text{max}}^2 \|\boldsymbol{y}_{t+h} - \hat{\boldsymbol{y}}_{t+h}\|_2^2\end{aligned}$$

- σ_{max} is the largest eigenvalue of $\boldsymbol{M}$
- $\sigma_{\text{max}} \geq 1$ as $\boldsymbol{M}$ is a projection matrix.
- Every projection reconciliation is better than base forecasts using Euclidean distance.

Mean square error bounds

Wickramasuriya (2021)

$$
\begin{aligned}
& \operatorname{tr}\Big(\mathrm{E}[\boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}^{\text{MinT}}_{t+h|t}]'[\boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}^{\text{MinT}}_{t+h|t}]\Big) \\
\leq\ & \operatorname{tr}\Big(\mathrm{E}[\boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}^{\text{OLS}}_{t+h|t}]'[\boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}^{\text{OLS}}_{t+h|t}]\Big) \\
\leq\ & \operatorname{tr}\Big(\mathrm{E}[\boldsymbol{y}_{t+h} - \hat{\boldsymbol{y}}_{t+h|t}]'[\boldsymbol{y}_{t+h} - \hat{\boldsymbol{y}}_{t+h|t}]\Big)
\end{aligned}
$$

Using sums of variances:

- MinT reconciliation is better than OLS reconciliation
- OLS reconciliation is better than base forecasts

Outline

1 Time series reconciliation
2 Reconciliation via constraints
3 Example: reconciling GDP forecasts
4 The geometry of forecast reconciliation
5 Optimization and reconcilation
6 In-built coherence
7 Time series cross-validation

Minimum trace reconciliation

Minimum trace (MinT) reconciliation

If $\boldsymbol{SG}$ is a projection, then the trace of $\boldsymbol{V}_h = \text{Var}(\tilde{\boldsymbol{y}}_{t+h|t} - \boldsymbol{y}_{t+h})$ is **minimized** when

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

$$\tilde{\boldsymbol{y}}_{T+h|T} = \boldsymbol{S}(\boldsymbol{S}'\boldsymbol{W}_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{W}_h^{-1}\hat{\boldsymbol{y}}_{T+h|T}$$

Reconciled forecasts

Base forecasts

- Trace of $\boldsymbol{V}_h$ is sum of forecast variances.
- MinT solution is L_2 **optimal** amongst linear unbiased forecasts.

A game theoretic perspective

van Erven and Cugliari (2015)

Find the solution to the minimax problem

$$V = \min_{\tilde{\boldsymbol{y}} \in \mathfrak{s}} \max_{\boldsymbol{y} \in \mathfrak{s}} \left\{ \ell(\boldsymbol{y}, \tilde{\boldsymbol{y}}) - \ell(\boldsymbol{y}, \hat{\boldsymbol{y}}) \right\},$$

where ℓ is a loss function, and $\mathfrak{s}$ is the coherent subspace.

- $V \leq 0$: reconciliation guaranteed to reduce loss.
- If $\ell(\boldsymbol{y}, \tilde{\boldsymbol{y}}) = \|\boldsymbol{y} - \tilde{\boldsymbol{y}}\|_\Psi = (\boldsymbol{y} - \tilde{\boldsymbol{y}})'\Psi(\boldsymbol{y} - \tilde{\boldsymbol{y}})$, where Ψ is any symmetric pd matrix, then:

  1. $\tilde{\boldsymbol{y}} = \boldsymbol{S}(\boldsymbol{S}'\Psi\boldsymbol{S})^{-1}\boldsymbol{S}'\Psi\hat{\boldsymbol{y}}$ will always improve upon the base forecasts;
  2. The MinT solution $\tilde{\boldsymbol{y}} = \boldsymbol{S}(\boldsymbol{S}'\boldsymbol{W}_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{W}_h^{-1}\hat{\boldsymbol{y}}$ will optimise loss in expectation over any choice of Ψ.

Biased reconciliation

Ben Taieb and Koo (2019)

Regularized empirical risk minimization problem:

$$\min_{\boldsymbol{G}} \frac{1}{Nn}\|\boldsymbol{Y} - \hat{\boldsymbol{Y}}\boldsymbol{G}'\boldsymbol{S}'\|_F + \lambda\|\text{vec}\boldsymbol{G}\|_1,$$

- $N = T - T_1 - h + 1, \quad T_1$ is minimum training sample size
- $\|\cdot\|_F$ is the Frobenius norm
- $\boldsymbol{Y} = [\boldsymbol{y}_{T_1+h}, \dots, \boldsymbol{y}_T]'$
- $\hat{\boldsymbol{Y}} = [\hat{\boldsymbol{y}}_{T_1+h|T_1}, \dots, \hat{\boldsymbol{y}}_{T|T-h}]'$
- λ is a regularization parameter

When $\lambda = 0$: $\quad \hat{\boldsymbol{G}} = \boldsymbol{B}'\hat{\boldsymbol{Y}}(\hat{\boldsymbol{Y}}'\hat{\boldsymbol{Y}})^{-1}$ where $\boldsymbol{B} = [\boldsymbol{b}_{T_1+h}, \dots, \boldsymbol{b}_T]'$.

Reference: Ben Taieb and Koo (2019)

MinT expressed as a regression

Since $\tilde{\bm{b}}_{t+h|t} = (\bm{S}'\bm{W}_h^{-1}\bm{S})^{-1}\bm{S}'\bm{W}_h^{-1}\hat{\bm{y}}_{t+h|t}$, we can write the MinT solution as a regression problem:

$$\begin{aligned}
\tilde{\bm{b}}_{t+h|t} &= \arg\min_{\bm{b}} \left[\hat{\bm{y}}_{t+h|t} - \bm{S}\bm{b}\right]'\bm{W}_h^{-1}\left[\hat{\bm{y}}_{t+h|t} - \bm{S}\bm{b}\right] \\
&= \arg\min_{\bm{b}} \left[\bm{b}'\bm{S}'\bm{W}_h^{-1}\bm{S}\bm{b} - 2\bm{b}'\bm{S}'\bm{W}_h^{-1}\hat{\bm{y}}_{t+h|t} + \hat{\bm{y}}'_{t+h|t}\bm{W}_h^{-1}\hat{\bm{y}}_{t+h|t}\right] \\
&= \arg\min_{\bm{b}} \left[\bm{b}'\bm{S}'\bm{W}_h^{-1}\bm{S}\bm{b} - 2\bm{b}'\bm{S}'\bm{W}_h^{-1}\hat{\bm{y}}_{t+h|t}\right]
\end{aligned}$$

- MinT solution is equivalent to a GLS regression of $\hat{\bm{y}}_{t+h|t}$ on $\bm{S}$ with covariance weights $\bm{W}_h^{-1}$.
- The estimated coefficients are the forecasts of the bottom level series.

Non-negative forecasts

$$\min_{\boldsymbol{G}_h} \text{tr}\Big(\text{E}[\boldsymbol{y}_{t+h} - \boldsymbol{S}\boldsymbol{G}_h\hat{\boldsymbol{y}}_{t+h|t}]'[\boldsymbol{y}_{t+h} - \boldsymbol{S}\boldsymbol{G}_h\hat{\boldsymbol{y}}_{t+h|t}]\Big)$$
$$\text{such that } \boldsymbol{b}_{t+h|t} = \boldsymbol{G}_h\hat{\boldsymbol{y}}_{t+h|t} \geq 0$$

Non-negative forecasts

$$\min_{\boldsymbol{G}_h} \text{tr}\Big(\text{E}[\boldsymbol{y}_{t+h} - \boldsymbol{S}\boldsymbol{G}_h\hat{\boldsymbol{y}}_{t+h|t}]'[\boldsymbol{y}_{t+h} - \boldsymbol{S}\boldsymbol{G}_h\hat{\boldsymbol{y}}_{t+h|t}]\Big)$$

$$\text{such that } \boldsymbol{b}_{t+h|t} = \boldsymbol{G}_h\hat{\boldsymbol{y}}_{t+h|t} \geq 0$$

Solve via quadratic programming:

$$\min_{\boldsymbol{b}}\Big[\boldsymbol{b}'\boldsymbol{S}'\boldsymbol{W}_h^{-1}\boldsymbol{S}\boldsymbol{b} - 2\boldsymbol{b}'\boldsymbol{S}'\boldsymbol{W}_h^{-1}\hat{\boldsymbol{y}}_{T+h|T}\Big] \quad \text{s.t. } \boldsymbol{b} \geq 0$$

(Wickramasuriya, Turlach, and Hyndman, 2020)

Non-negative forecasts

$$\min_{\boldsymbol{G}_h} \text{tr}\Big(\text{E}[\boldsymbol{y}_{t+h} - \boldsymbol{S}\boldsymbol{G}_h\hat{\boldsymbol{y}}_{t+h|t}]'[\boldsymbol{y}_{t+h} - \boldsymbol{S}\boldsymbol{G}_h\hat{\boldsymbol{y}}_{t+h|t}]\Big)$$

$$\text{such that } \boldsymbol{b}_{t+h|t} = \boldsymbol{G}_h\hat{\boldsymbol{y}}_{t+h|t} \geq 0$$

Solve via quadratic programming:

$$\min_{\boldsymbol{b}}\left[\boldsymbol{b}'\boldsymbol{S}'\boldsymbol{W}_h^{-1}\boldsymbol{S}\boldsymbol{b} - 2\boldsymbol{b}'\boldsymbol{S}'\boldsymbol{W}_h^{-1}\hat{\boldsymbol{y}}_{T+h|T}\right] \quad \text{s.t. } \boldsymbol{b} \geq 0$$

(Wickramasuriya, Turlach, and Hyndman, 2020)

Set-negative-to-zero heuristic solution

- Negative reconciled forecasts at bottom level set to zero
- Remaining forecasts computed via aggregation

(Di Fonzo and Girolimetto, 2023)

Immutable forecasts

Zhang et al. (2022)

$$\hat{\boldsymbol{y}}_{t+h|t} = \begin{bmatrix}\hat{\boldsymbol{a}}_{t+h|t}\\ \hat{\boldsymbol{v}}_{t+h|t}\\ \hat{\boldsymbol{u}}_{t+h|t}\end{bmatrix} = \begin{bmatrix}\boldsymbol{A}_1 & \boldsymbol{A}_2\\ \boldsymbol{I}_{n_b-k} & \boldsymbol{O}\\ \boldsymbol{O} & \boldsymbol{I}_k\end{bmatrix}\begin{bmatrix}\hat{\boldsymbol{v}}_{t+h|t}\\ \hat{\boldsymbol{u}}_{t+h|t}\end{bmatrix}$$

Suppose $\hat{\boldsymbol{u}}_{t+h|t}$ are fixed and let $\hat{\boldsymbol{w}}_{t+h|t} = \begin{bmatrix}\hat{\boldsymbol{a}}_{t+h|t} - \boldsymbol{A}_2\hat{\boldsymbol{u}}_{t+h|t}\\ \hat{\boldsymbol{v}}_{t+h|t}\end{bmatrix}$.

Optimization problem

$$\min_{\boldsymbol{v}}\left[\hat{\boldsymbol{w}}_{t+h|t} - \boldsymbol{A}_3\boldsymbol{v}\right]'\boldsymbol{W}_{\boldsymbol{v}}^{-1}\left[\hat{\boldsymbol{w}}_{t+h|t} - \boldsymbol{A}_3\boldsymbol{v}\right] \qquad \text{where} \qquad \boldsymbol{A}_3 = \begin{bmatrix}\boldsymbol{A}_1\\ \boldsymbol{I}_{n_b-k}\end{bmatrix}$$

and $\boldsymbol{W}_{\boldsymbol{v}}$ contains elements of $\boldsymbol{W}_h$ corresponding to $\hat{\boldsymbol{v}}_{t+h|t}$.

Immutable forecasts

Zhang et al. (2022)

$$\hat{\boldsymbol{y}}_{t+h|t} = \begin{bmatrix}\hat{\boldsymbol{a}}_{t+h|t}\\ \hat{\boldsymbol{v}}_{t+h|t}\\ \hat{\boldsymbol{u}}_{t+h|t}\end{bmatrix} = \begin{bmatrix}\boldsymbol{A}_1 & \boldsymbol{A}_2\\ \boldsymbol{I}_{n_b-k} & \boldsymbol{O}\\ \boldsymbol{O} & \boldsymbol{I}_k\end{bmatrix} \begin{bmatrix}\hat{\boldsymbol{v}}_{t+h|t}\\ \hat{\boldsymbol{u}}_{t+h|t}\end{bmatrix}$$

Suppose $\hat{\boldsymbol{u}}_{t+h|t}$ are fixed and let $\hat{\boldsymbol{w}}_{t+h|t} = \begin{bmatrix}\hat{\boldsymbol{a}}_{t+h|t} - \boldsymbol{A}_2\hat{\boldsymbol{u}}_{t+h|t}\\ \hat{\boldsymbol{v}}_{t+h|t}\end{bmatrix}$.

Solve with non-negativity constraint

$$\min_{\boldsymbol{v}} \left[\hat{\boldsymbol{w}}_{t+h|t} - \boldsymbol{A}_3\boldsymbol{v}\right]' \boldsymbol{W}_{\boldsymbol{v}}^{-1} \left[\hat{\boldsymbol{w}}_{t+h|t} - \boldsymbol{A}_3\boldsymbol{v}\right] \qquad \text{where} \qquad \boldsymbol{A}_3 = \begin{bmatrix}\boldsymbol{A}_1\\ \boldsymbol{I}_{n_b-k}\end{bmatrix}$$

$$\text{such that} \qquad \boldsymbol{A}_3\boldsymbol{v} \geq \begin{bmatrix}-\boldsymbol{A}_2\hat{\boldsymbol{u}}_{t+h|t}\\ \boldsymbol{0}\end{bmatrix}$$

Reconciliation and regularization

Mishchenko, Montgomery, and Vaggi (2019):
Optimize all forecasts with an incoherence penalty

$$\min_{\hat{\boldsymbol{y}}_t} \sum_{t=1}^{T} \|\boldsymbol{y}_t - \hat{\boldsymbol{y}}_t\|_2 + \lambda \sum_{t=1}^{T} \|\hat{\boldsymbol{y}}_t - \boldsymbol{S}_t \hat{\boldsymbol{b}}_t\|_2$$

Reconciliation and regularization

Mishchenko, Montgomery, and Vaggi (2019):
Optimize all forecasts with an incoherence penalty

$$\min_{\hat{\boldsymbol{y}}_t} \sum_{t=1}^{T} \|\boldsymbol{y}_t - \hat{\boldsymbol{y}}_t\|_2 + \lambda \sum_{t=1}^{T} \|\hat{\boldsymbol{y}}_t - \boldsymbol{S}_t \hat{\boldsymbol{b}}_t\|_2$$

Shiratori, Kobayashi, and Takano (2020):
Optimize bottom level forecasts with an incoherence penalty

$$\min_{\hat{\boldsymbol{b}}_t} \sum_{t=1}^{T} \|\hat{\boldsymbol{b}}_t - \boldsymbol{b}_t\|_2 + \sum_{t=1}^{T} \Lambda \|\boldsymbol{a}_t - \boldsymbol{A}_t \hat{\boldsymbol{b}}_t\|_2$$

Outline

1 Time series reconciliation
2 Reconciliation via constraints
3 Example: reconciling GDP forecasts
4 The geometry of forecast reconciliation
5 Optimization and reconcilation
6 In-built coherence
7 Time series cross-validation

In-built coherence

Two-step approach: compute base forecasts $\hat{\boldsymbol{y}}_h$, and then reconcile them to produce $\tilde{\boldsymbol{y}}_h$.

One-step approaches: compute coherent $\tilde{\boldsymbol{y}}_h$ directly.

- Ashouri, Hyndman, and Shmueli (2022): linear regression models
- Pennings and Dalen (2017): state space models
- Villegas and Pedregal (2018): state space models

In-built coherence using linear models

Suppose $\hat{y}_{t,i} = \hat{\beta}_i'\boldsymbol{x}_{t,i}$ with $\boldsymbol{x}_{t,i} = (1, x_{t,1,i}, \dots, x_{t,p,i})$ & $\hat{\boldsymbol{y}}_i = (\hat{y}_{1,i}, \dots, \hat{y}_{T,i})$.

In-built coherence using linear models

Suppose $\hat{y}_{t,i} = \hat{\beta}_i' \boldsymbol{x}_{t,i}$ with $\boldsymbol{x}_{t,i} = (1, x_{t,1,i}, \dots, x_{t,p,i})$ & $\hat{\boldsymbol{y}}_i = (\hat{y}_{1,i}, \dots, \hat{y}_{T,i})$.

$$\begin{pmatrix} \hat{\boldsymbol{y}}_1 \\ \hat{\boldsymbol{y}}_2 \\ \vdots \\ \hat{\boldsymbol{y}}_n \end{pmatrix} = \begin{pmatrix} \boldsymbol{X}_1 & 0 & \dots & 0 \\ 0 & \boldsymbol{X}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \boldsymbol{X}_n \end{pmatrix} \begin{pmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \vdots \\ \hat{\beta}_n \end{pmatrix}, \qquad \boldsymbol{X}_i = \begin{pmatrix} 1 & x_{1,i,1} & x_{1,i,2} & \dots & x_{1,i,p} \\ 1 & x_{2,i,1} & x_{2,i,2} & \dots & x_{2,i,p} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{T,i,1} & x_{T,i,2} & \dots & x_{T,i,p} \end{pmatrix}$$

In-built coherence using linear models

Suppose $\hat{y}_{t,i} = \hat{\beta}_i'\boldsymbol{x}_{t,i}$ with $\boldsymbol{x}_{t,i} = (1, x_{t,1,i}, \dots, x_{t,p,i})$ & $\hat{\boldsymbol{y}}_i = (\hat{y}_{1,i}, \dots, \hat{y}_{T,i})$.

$$\begin{pmatrix} \hat{\boldsymbol{y}}_1 \\ \hat{\boldsymbol{y}}_2 \\ \vdots \\ \hat{\boldsymbol{y}}_n \end{pmatrix} = \begin{pmatrix} \boldsymbol{X}_1 & 0 & \dots & 0 \\ 0 & \boldsymbol{X}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \boldsymbol{X}_n \end{pmatrix} \begin{pmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \vdots \\ \hat{\beta}_n \end{pmatrix}, \qquad \boldsymbol{X}_i = \begin{pmatrix} 1 & x_{1,i,1} & x_{1,i,2} & \dots & x_{1,i,p} \\ 1 & x_{2,i,1} & x_{2,i,2} & \dots & x_{2,i,p} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{T,i,1} & x_{T,i,2} & \dots & x_{T,i,p} \end{pmatrix}$$

$$\hat{\boldsymbol{B}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y}$$

In-built coherence using linear models

Suppose $\hat{y}_{t,i} = \hat{\beta}_i' \boldsymbol{x}_{t,i}$ with $\boldsymbol{x}_{t,i} = (1, x_{t,1,i}, \dots, x_{t,p,i})$ & $\hat{\boldsymbol{y}}_i = (\hat{y}_{1,i}, \dots, \hat{y}_{T,i})$.

$$\begin{pmatrix} \hat{\boldsymbol{y}}_1 \\ \hat{\boldsymbol{y}}_2 \\ \vdots \\ \hat{\boldsymbol{y}}_n \end{pmatrix} = \begin{pmatrix} \boldsymbol{X}_1 & 0 & \dots & 0 \\ 0 & \boldsymbol{X}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \boldsymbol{X}_n \end{pmatrix} \begin{pmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \vdots \\ \hat{\beta}_n \end{pmatrix}, \qquad \boldsymbol{X}_i = \begin{pmatrix} 1 & x_{1,i,1} & x_{1,i,2} & \dots & x_{1,i,p} \\ 1 & x_{2,i,1} & x_{2,i,2} & \dots & x_{2,i,p} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{T,i,1} & x_{T,i,2} & \dots & x_{T,i,p} \end{pmatrix}$$

$$\hat{\boldsymbol{B}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y} \qquad \hat{\boldsymbol{y}}_{t+h} = \boldsymbol{X}^*_{t+h}\hat{\boldsymbol{B}}$$

In-built coherence using linear models

Suppose $\hat{y}_{t,i} = \hat{\beta}_i' \boldsymbol{x}_{t,i}$ with $\boldsymbol{x}_{t,i} = (1, x_{t,1,i}, \dots, x_{t,p,i})$ & $\hat{\boldsymbol{y}}_i = (\hat{y}_{1,i}, \dots, \hat{y}_{T,i})$.

$$\begin{pmatrix} \hat{\boldsymbol{y}}_1 \\ \hat{\boldsymbol{y}}_2 \\ \vdots \\ \hat{\boldsymbol{y}}_n \end{pmatrix} = \begin{pmatrix} \boldsymbol{X}_1 & 0 & \dots & 0 \\ 0 & \boldsymbol{X}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \boldsymbol{X}_n \end{pmatrix} \begin{pmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \vdots \\ \hat{\beta}_n \end{pmatrix}, \qquad \boldsymbol{X}_i = \begin{pmatrix} 1 & x_{1,i,1} & x_{1,i,2} & \dots & x_{1,i,p} \\ 1 & x_{2,i,1} & x_{2,i,2} & \dots & x_{2,i,p} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{T,i,1} & x_{T,i,2} & \dots & x_{T,i,p} \end{pmatrix}$$

$$\hat{\boldsymbol{B}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y} \qquad \hat{\boldsymbol{y}}_{t+h} = \boldsymbol{X}^*_{t+h}\hat{\boldsymbol{B}} \qquad \boldsymbol{X}^*_{t+h} = \text{diag}(\boldsymbol{x}'_{t+h,i}, \dots, \boldsymbol{x}'_{t+h,n})$$

In-built coherence using linear models

Suppose $\hat{y}_{t,i} = \hat{\beta}_i' \boldsymbol{x}_{t,i}$ with $\boldsymbol{x}_{t,i} = (1, x_{t,1,i}, \dots, x_{t,p,i})$ & $\hat{\boldsymbol{y}}_i = (\hat{y}_{1,i}, \dots, \hat{y}_{T,i})$.

$$\begin{pmatrix} \hat{\boldsymbol{y}}_1 \\ \hat{\boldsymbol{y}}_2 \\ \vdots \\ \hat{\boldsymbol{y}}_n \end{pmatrix} = \begin{pmatrix} \boldsymbol{X}_1 & 0 & \dots & 0 \\ 0 & \boldsymbol{X}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \boldsymbol{X}_n \end{pmatrix} \begin{pmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \vdots \\ \hat{\beta}_n \end{pmatrix}, \qquad \boldsymbol{X}_i = \begin{pmatrix} 1 & x_{1,i,1} & x_{1,i,2} & \dots & x_{1,i,p} \\ 1 & x_{2,i,1} & x_{2,i,2} & \dots & x_{2,i,p} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{T,i,1} & x_{T,i,2} & \dots & x_{T,i,p} \end{pmatrix}$$

$$\hat{\boldsymbol{B}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y} \qquad \hat{\boldsymbol{y}}_{t+h} = \boldsymbol{X}^*_{t+h}\hat{\boldsymbol{B}} \qquad \boldsymbol{X}^*_{t+h} = \text{diag}(\boldsymbol{x}'_{t+h,i}, \dots, \boldsymbol{x}'_{t+h,n})$$

$$\tilde{\boldsymbol{y}}_{t+h} = \boldsymbol{S}(\boldsymbol{S}'\boldsymbol{W}_h\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{W}_h\hat{\boldsymbol{y}}_{t+h} = \boldsymbol{S}(\boldsymbol{S}'\boldsymbol{W}_h\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{W}_h\boldsymbol{X}^*_{t+h}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y}$$

$$\boldsymbol{V}_h = \sigma^2\boldsymbol{S}(\boldsymbol{S}'\boldsymbol{W}_h\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{W}_h\left[1 + \boldsymbol{X}^*_{T+h}(\boldsymbol{X}'\boldsymbol{X})^{-1}(\boldsymbol{X}^*_{T+h})'\right]\boldsymbol{W}_h\boldsymbol{S}'(\boldsymbol{S}'\boldsymbol{W}_h\boldsymbol{S})^{-1}\boldsymbol{S}'$$

Reference: Ashouri, Hyndman, and Shmueli (2022)

In-built coherence using state space models

Pennings and Dalen (2017) propose the state space model

$$\begin{aligned} \boldsymbol{y}_t &= \boldsymbol{S}\boldsymbol{\mu}_t + \boldsymbol{Z}_t\beta + \varepsilon_t, && \varepsilon_t \sim N(\boldsymbol{0}, \Sigma_\varepsilon), \\ \boldsymbol{\mu}_t &= \boldsymbol{\mu}_{t-1} + \eta_t, && \eta_t \sim N(\boldsymbol{0}, \Sigma_\eta). \end{aligned}$$

- Coherent forecasts arise naturally using the Kalman filter
- Covariance matrices difficult to estimate except for small hierarchies.
- Requires the same model for all series
- A related approach proposed by Villegas and Pedregal (2018)

Outline

1 Time series reconciliation

2 Reconciliation via constraints

3 Example: reconciling GDP forecasts

4 The geometry of forecast reconciliation

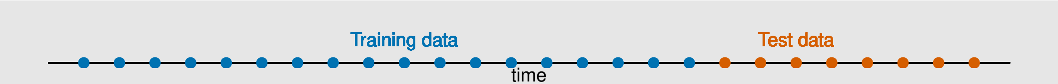
5 Optimization and reconcilation

6 In-built coherence

7 Time series cross-validation

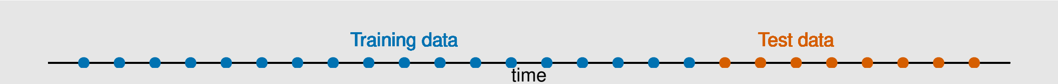
Time series cross-validation

Traditional evaluation

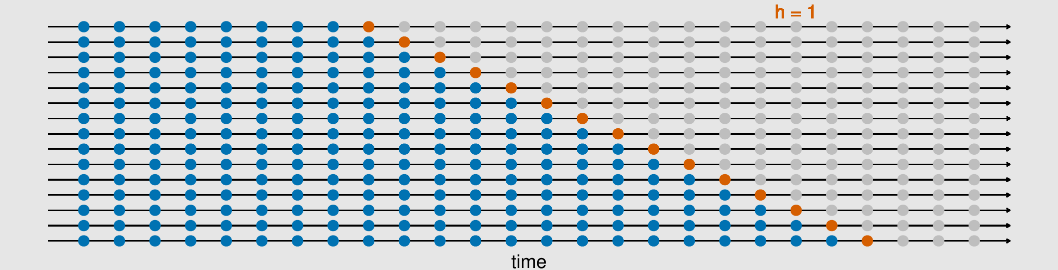


Time series cross-validation

Traditional evaluation

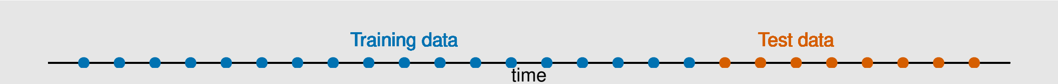


Time series cross-validation

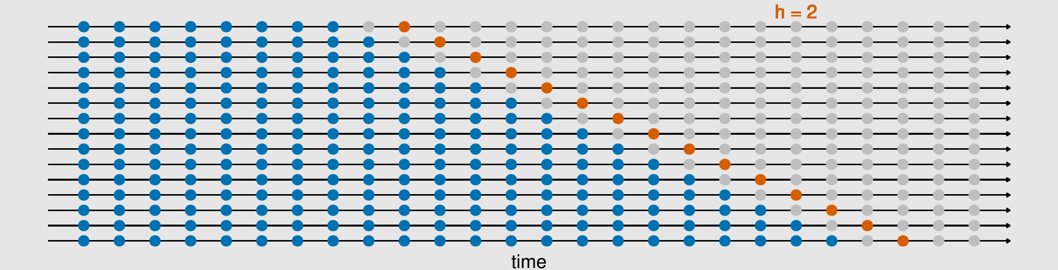


Time series cross-validation

Traditional evaluation

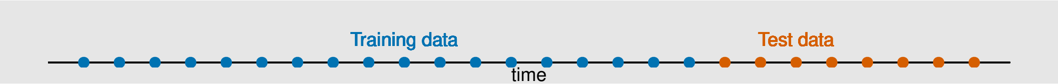


Time series cross-validation

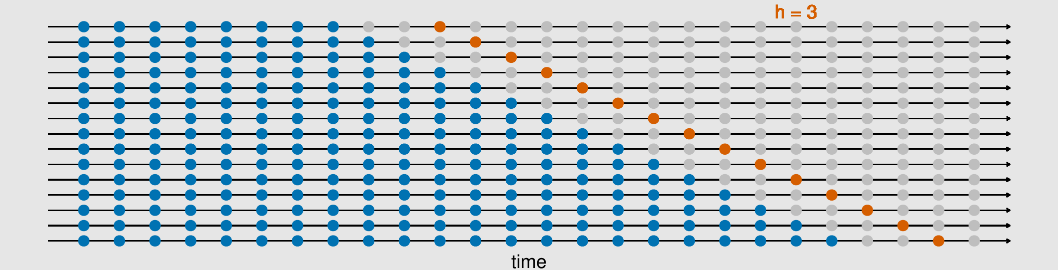


Time series cross-validation

Traditional evaluation

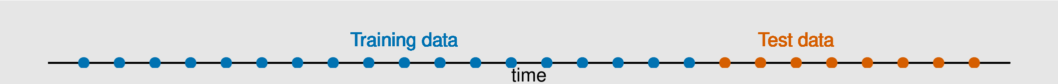


Time series cross-validation

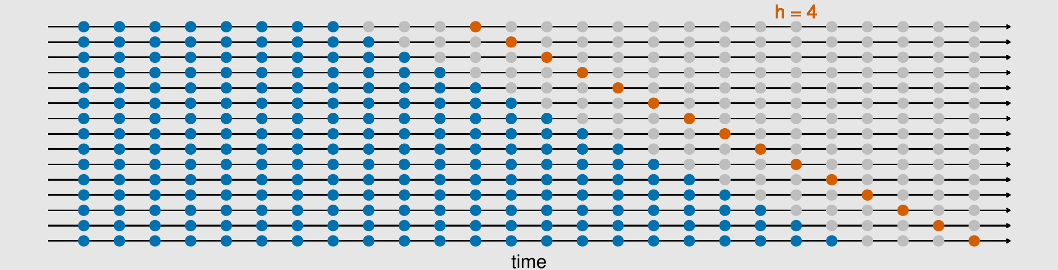


Time series cross-validation

Traditional evaluation

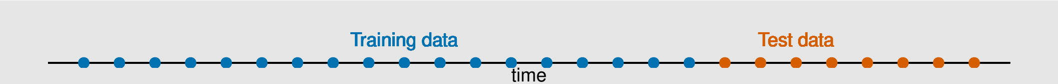


Time series cross-validation

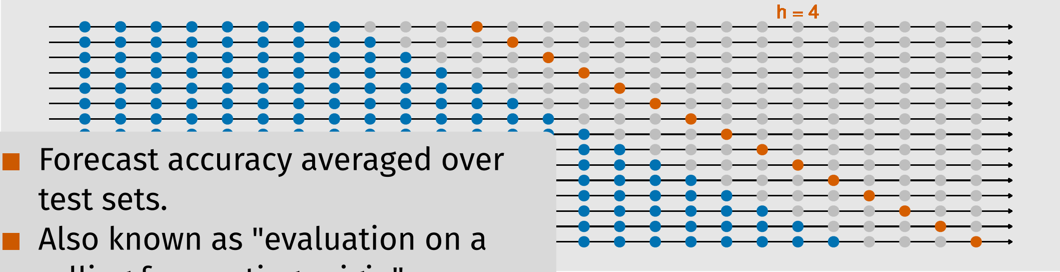


Time series cross-validation

Traditional evaluation



Time series cross-validation



- Forecast accuracy averaged over test sets.
- Also known as "evaluation on a rolling forecasting origin"

Example: Australian tourism

```
# A tsibble: 18,000 x 5 [1M]
# Key:       state, zone, region [75]
      month state zone      region visitors
      <mth> <chr> <chr>     <chr>     <dbl>
 1 1998 Jan NSW   Metro NSW Sydney     926.
 2 1998 Feb NSW   Metro NSW Sydney     647.
 3 1998 Mar NSW   Metro NSW Sydney     716.
 4 1998 Apr NSW   Metro NSW Sydney     621.
 5 1998 May NSW   Metro NSW Sydney     598.
 6 1998 Jun NSW   Metro NSW Sydney     601.
 7 1998 Jul NSW   Metro NSW Sydney     720.
 8 1998 Aug NSW   Metro NSW Sydney     645.
 9 1998 Sep NSW   Metro NSW Sydney     633.
10 1998 Oct NSW   Metro NSW Sydney     771.
# i 17,990 more rows
```

Example: Australian tourism

```
tourism_agg <- tourism |>
  aggregate_key(state/zone/region, visitors = sum(visitors))
tourism_stretch <- tourism_agg |>
  stretch_tsibble(.init = 48, .step=20*12)#.step = 1)
```

```
# A tsibble: 5,280 x 6 [1M]
# Key:       .id, state, zone, region [110]
      month state  zone   region   visitors   .id
      <mth> <chr*> <chr*> <chr*>      <dbl> <int>
 1 1998 Jan NSW    ACT    Canberra     210.     1
 2 1998 Feb NSW    ACT    Canberra     156.     1
 3 1998 Mar NSW    ACT    Canberra     185.     1
 4 1998 Apr NSW    ACT    Canberra     178.     1
 5 1998 May NSW    ACT    Canberra     134.     1
 6 1998 Jun NSW    ACT    Canberra     105.     1
 7 1998 Jul NSW    ACT    Canberra     142.     1
 8 1998 Aug NSW    ACT    Canberra     137.     1
 9 1998 Sep NSW    ACT    Canberra     166.     1
```

Example: Australian tourism

```
fit <- tourism_stretch |>
  model(
    ets = ETS(visitors),
    arima = ARIMA(visitors)
  ) |>
  mutate(comb = (ets+arima)/2)
```

```
# A mable: 110 x 7
# Key:     .id, state, zone, region [110]
      .id state  zone             region                  ets
    <int> <chr*> <chr*>           <chr*>              <model>
 1      1 NSW    ACT              Canberra        <ETS(A,N,N)>
 2      1 NSW    ACT              <aggregated>    <ETS(A,N,N)>
 3      1 NSW    Metro NSW        Central Coast   <ETS(A,N,A)>
 4      1 NSW    Metro NSW        Sydney          <ETS(A,N,N)>
 5      1 NSW    Metro NSW        <aggregated>    <ETS(M,N,M)>
 6      1 NSW    North Coast NSW  Hunter          <ETS(A,N,N)>
 7      1 NSW    North Coast NSW  North Coast NSW <ETS(M,N,M)>
```

Example: Australian tourism

```r
fc <- fit |>
  reconcile(
    ols = min_trace(comb, method = "ols"),
    wlsv = min_trace(comb, method = "wls_var"),
    wlss = min_trace(comb, method = "wls_struct"),
    mint_s = min_trace(comb, method = "mint_shrink"),
  ) |>
  forecast(h = "2 years")
```

```
# A fable: 18,480 x 8 [1M]
# Key:     .id, state, zone, region, .model [770]
     .id state  zone   region   .model    month      visitors .mean
   <int> <chr*> <chr*> <chr*>   <chr>     <mth>        <dist> <dbl>
 1     1 NSW    ACT    Canberra ets    2002 Jan N(169, 1553)  169.
 2     1 NSW    ACT    Canberra ets    2002 Feb N(169, 1553)  169.
 3     1 NSW    ACT    Canberra ets    2002 Mar N(169, 1553)  169.
 4     1 NSW    ACT    Canberra ets    2002 Apr N(169, 1553)  169.
 5     1 NSW    ACT    Canberra ets    2002 May N(169, 1553)  169.
```

Example: Australian tourism

```r
fc |>
  accuracy(tourism_agg, measures = list(mase = MASE, rmsse = RMSSE)) |>
  group_by(.model) |>
  summarise(mase = mean(mase), rmsse = sqrt(mean(rmsse^2))) |>
  arrange(rmsse)
```

```
# A tibble: 7 x 3
  .model  mase rmsse
  <chr>  <dbl> <dbl>
1 mint_s 0.798 0.823
2 wlsv   0.818 0.840
3 wlss   0.820 0.843
4 comb   0.833 0.857
5 ols    0.830 0.858
6 ets    0.834 0.864
7 arima  0.876 0.895
```

Example: Australian tourism

```
fc |>
  accuracy(tourism_agg, measures = list(mase = MASE, rmsse = RMSSE)) |>
  group_by(.model) |>
  summarise(mase = mean(mase), rmsse = sqrt(mean(rmsse^2))) |>
  arrange(rmsse)
```

```
# A tibble: 7 x 3
  .model  mase rmsse
  <chr>  <dbl> <dbl>
1 mint_s 0.798 0.823
2 wlsv   0.818 0.840
3 wlss   0.820 0.843
4 comb   0.833 0.857
5 ols    0.830 0.858
6 ets    0.834 0.864
7 arima  0.876 0.895
```

- Combining improves forecast accuracy.
- Reconciling improves further.
- MinT-Shrink is almost always the best reconciliation method.

Example: Australian tourism

```
# A tibble: 28 x 4
# Groups:   .model [7]
   .model level     mase rmsse
   <chr>  <fct>    <dbl> <dbl>
 1 ets    National 0.835 0.823
 2 mint_s National 0.837 0.839
 3 ols    National 0.846 0.848
 4 comb   National 0.849 0.852
 5 wlss   National 0.872 0.873
 6 wlsv   National 0.873 0.876
 7 arima  National 0.945 0.948
 8 mint_s State    0.801 0.827
 9 wlsv   State    0.820 0.849
10 wlss   State    0.819 0.850
11 ols    State    0.818 0.860
12 ets    State    0.807 0.863
13 comb   State    0.827 0.871
14 arima  State    0.899 0.941
15 mint_s Zone     0.801 0.828
16 wlss   Zone     0.820 0.836
17 wlsv   Zone     0.821 0.843
18 ols    Zone     0.828 0.850
19 ets    Zone     0.835 0.863
20 comb   Zone     0.841 0.865
21 arima  Zone     0.905 0.926
22 mint_s Region   0.796 0.821
23 wlsv   Region   0.815 0.837
```

Example: Australian tourism

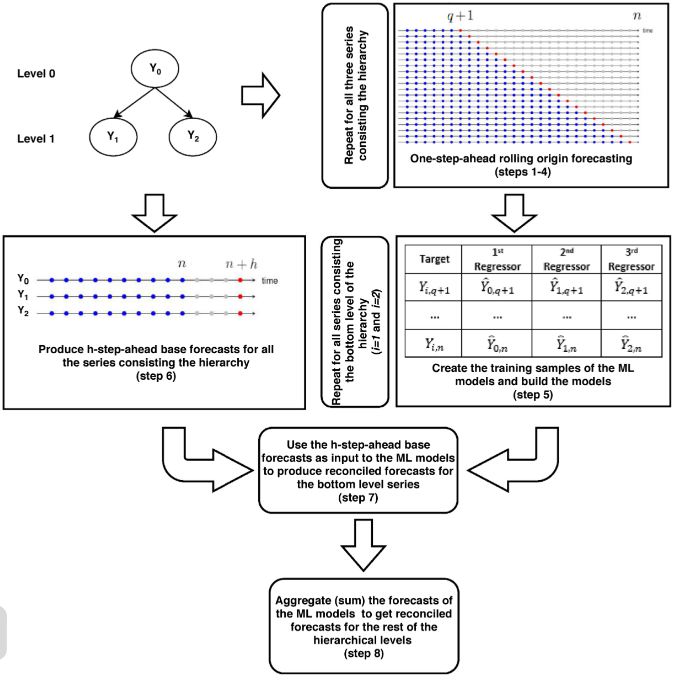
```
# A tibble: 28 x 4
# Groups:   .model [7]
   .model level     mase rmsse
   <chr>  <fct>    <dbl> <dbl>
 1 ets    National 0.835 0.823
 2 mint_s National 0.837 0.839
 3 ols    National 0.846 0.848
 4 comb   National 0.849 0.852
 5 wlss   National 0.872 0.873
 6 wlsv   National 0.873 0.876
 7 arima  National 0.945 0.948
 8 mint_s State    0.801 0.827
 9 wlsv   State    0.820 0.849
10 wlss   State    0.819 0.850
11 ols    State    0.818 0.860
12 ets    State    0.807 0.863
13 comb   State    0.827 0.871
14 arima  State    0.899 0.941
15 mint_s Zone     0.801 0.828
16 wlss   Zone     0.820 0.836
17 wlsv   Zone     0.821 0.843
18 ols    Zone     0.828 0.850
19 ets    Zone     0.835 0.863
20 comb   Zone     0.841 0.865
21 arima  Zone     0.905 0.926
22 mint_s Region   0.796 0.821
23 wlsv   Region   0.815 0.837
```

- MinT-Shrink is almost always the best reconciliation method.

ML reconciliation

1. Split all series using time series cross-validation
2. For each training set, compute one-step-ahead forecasts for all series
3. For each bottom-level series, use RF or XGB to predict values using forecasts of all series as inputs
4. Forecast all series
5. For each bottom-level series, apply ML model to improve forecasts
6. Aggregate bottom-level forecasts to obtain forecasts for other series.

Spiliotis et al. (ASC, 2021)



ML reconciliation: tourism data

Method	Total	States	Zones	Regions	Average
			MASE		
MinT-Struct	1.094	0.968	0.887	0.843	0.948
MinT-Shrink	1.047	**0.956**	0.872	0.824	0.925
ML-RF	1.045	0.964	0.859	0.812	0.920
ML-XGB	**1.043**	0.965	**0.859**	**0.812**	**0.920**
			RMSSE		
MinT-Struct	1.308	1.225	1.137	1.109	1.195
MinT-Shrink	1.265	1.214	1.120	1.086	1.171
ML-RF	1.261	**1.208**	1.104	1.066	1.159
ML-XGB	1.255	1.208	**1.101**	**1.064**	**1.157**
			AMSE		
MinT-Struct	0.988	0.611	0.426	0.349	0.593
MinT-Shrink	0.935	0.599	0.417	0.337	0.572
ML-RF	0.780	**0.526**	0.366	0.319	0.498
ML-XGB	**0.779**	0.526	**0.365**	**0.317**	**0.497**

- ML methods not significantly different.
- MinT methods significantly different from each other and from ML methods.

References

- Ashouri, M, RJ Hyndman, and G Shmueli (2022). Fast forecast reconciliation using linear models. *J Computational & Graphical Statistics* **31**(1), 263–282.
- Ben Taieb, S and B Koo (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. Anchorage, AK, USA: Association for Computing Machinery, pp.1337–1347.
- Bisaglia, L, T Di Fonzo, and D Girolimetto (2020). Fully reconciled GDP forecasts from income and expenditure sides. Ed. by A Pollice, N Salvati, and F Schirripa Spagnolo, 951–956.
- Byron, RP (1978). The estimation of large social account matrices. *Journal of the Royal Statistical Society, Series A* **141**(3), 359–367.
- Chow, GC and Al Lin (1971). Best linear unbiased interpolation, distribution, and extrapolation of time series by related series. *The Review of Economics and Statistics* **53**(4), 372–375.

References

- Di Fonzo, T (1990). The estimation of *M* disaggregate time series when contemporaneous and temporal aggregates are known. *The Review of Economics and Statistics* **72**(1), 178–182.
- Di Fonzo, T and D Girolimetto (2022). Forecast combination-based forecast reconciliation: Insights and extensions. *International Journal of Forecasting* **forthcoming**.
- Di Fonzo, T and D Girolimetto (2023). Spatio-temporal reconciliation of solar forecasts. *Solar Energy* **251**, 13–29.
- Mishchenko, K, M Montgomery, and F Vaggi (2019). A self-supervised approach to hierarchical forecasting with applications to groupwise synthetic controls. In: *Proceedings of the Time Series Workshop at the 36th International Conference on Machine Learning, Long Beach, California*. PMLR 97.
- Panagiotelis, A, P Gamakumara, G Athanasopoulos, and RJ Hyndman (2021). Forecast reconciliation: A geometric view with new insights on bias correction. *International J Forecasting* **37**(1), 343–359.

References

- Pennings, CL and J van Dalen (2017). Integrated hierarchical forecasting. *European Journal of Operational Research* **263**(2), 412–418.
- Shiratori, T, K Kobayashi, and Y Takano (2020). Prediction of hierarchical time series using structured regularization and its application to artificial neural networks. *PLOS ONE* **15**(11), e0242099.
- Spiliotis, E, M Abolghasemi, RJ Hyndman, F Petropoulos, and V Assimakopoulos (2021). Hierarchical forecast reconciliation with machine learning. *Applied Soft Computing* **112**, 107756.
- Stone, R, DG Champernowne, and JE Meade (1942). The precision of national income estimates. *The Review of Economic Studies* **9**(2), 111–125.
- van Erven, T and J Cugliari (2015). "Game-theoretically optimal reconciliation of contemporaneous hierarchical time series forecasts". In: *Modeling and Stochastic Learning for Forecasting in High Dimension*. Ed. by A Antoniadis, JM Poggi, and X Brossat. Cham: Springer International Publishing, pp.297–317.

References

- Villegas, MA and DJ Pedregal (2018). Supply chain decision support systems based on a novel hierarchical forecasting approach. *Decision Support Systems* **114**, 29–36.
- Wickramasuriya, SL (2021). Properties of point forecast reconciliation approaches. *arXiv preprint arXiv:2103.11129*.
- Wickramasuriya, SL, G Athanasopoulos, and RJ Hyndman (2019). Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization. *J American Statistical Association* **114**(526), 804–819.
- Wickramasuriya, SL, BA Turlach, and RJ Hyndman (2020). Optimal non-negative forecast reconciliation. *Statistics & Computing* **30**(5), 1167–1182.
- Zhang, B, Y Kang, A Panagiotelis, and F Li (2022). Optimal reconciliation with immutable forecasts. *European Journal of Operational Research* **forthcoming**.