Improving forecasts via subspace projections

Rob J Hyndman

Photo by Stanley Cheung on Unsplash

MONASH University

OPTiMA

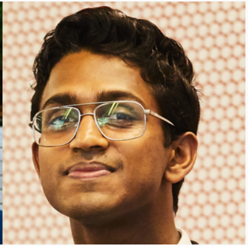
Sri Lankan PhD students and post-docs







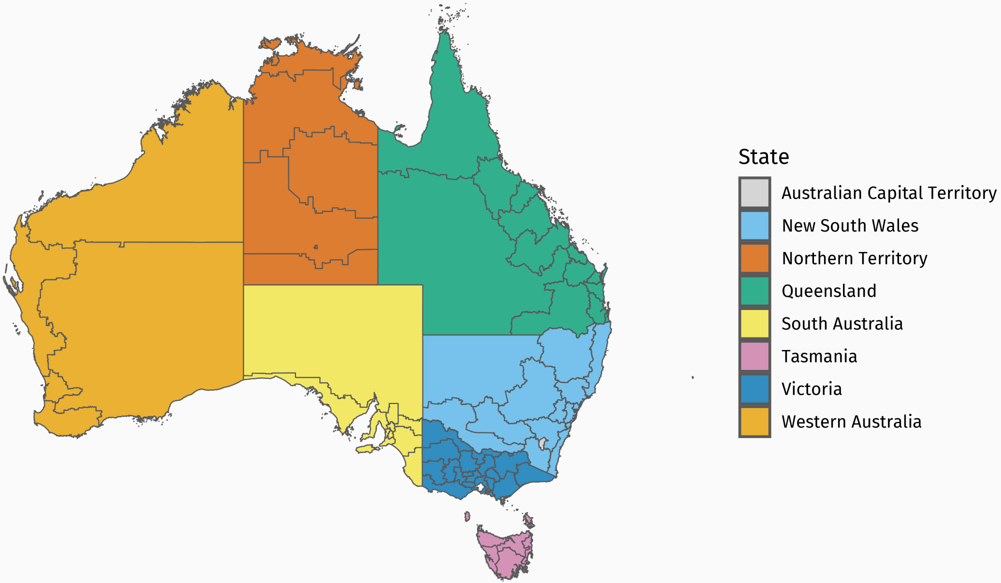




Outline

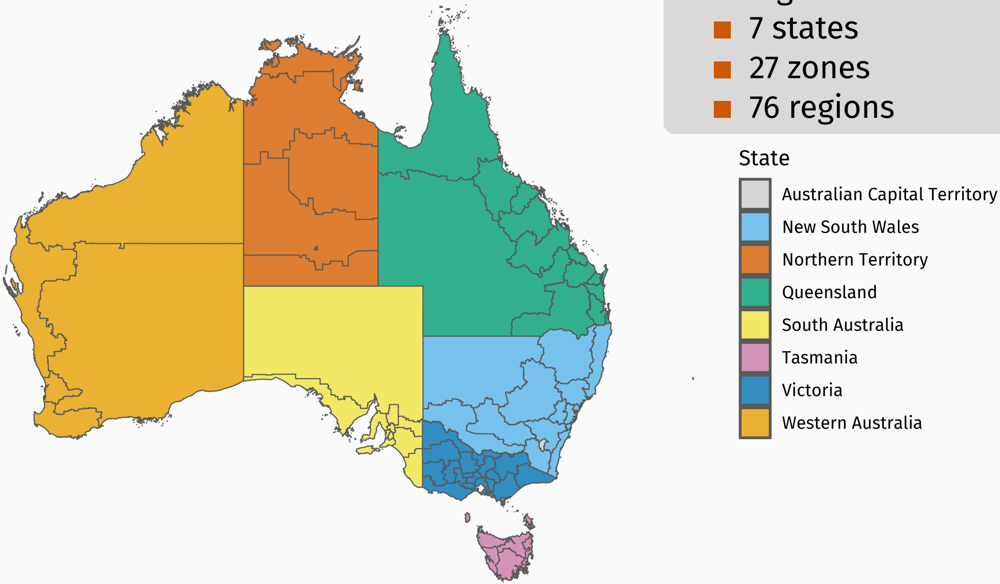
Outline

Australian tourism regions

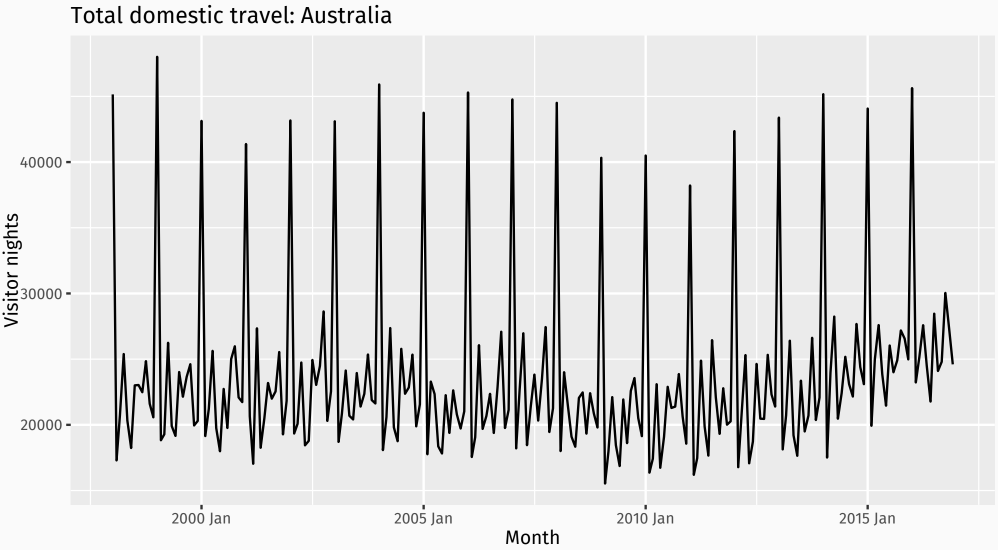


Australian tourism regions

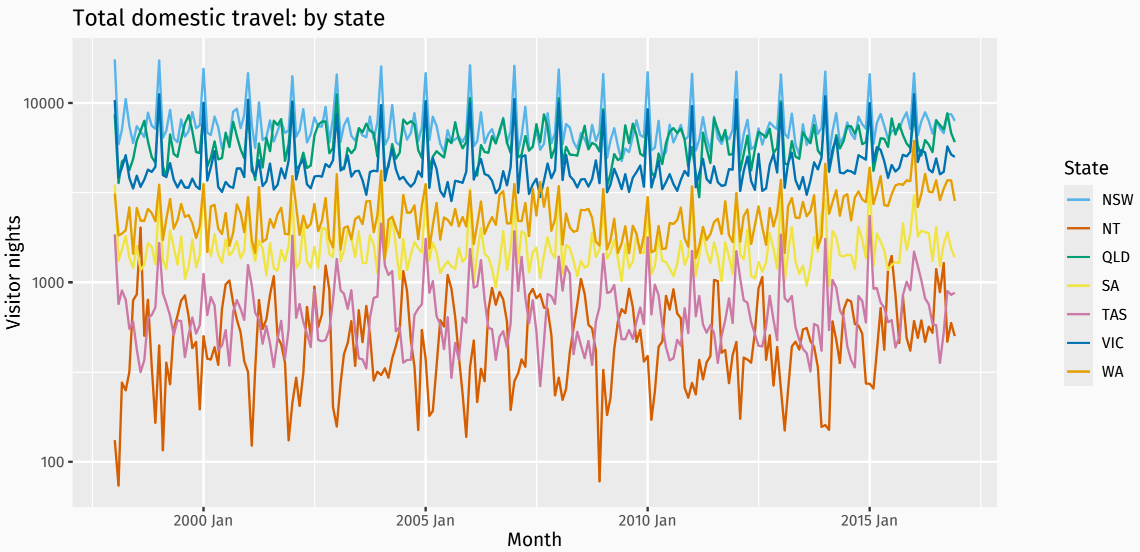
- Monthly data on visitor nights: 1998 – 2016
- 7 states
- 27 zones
- 76 regions



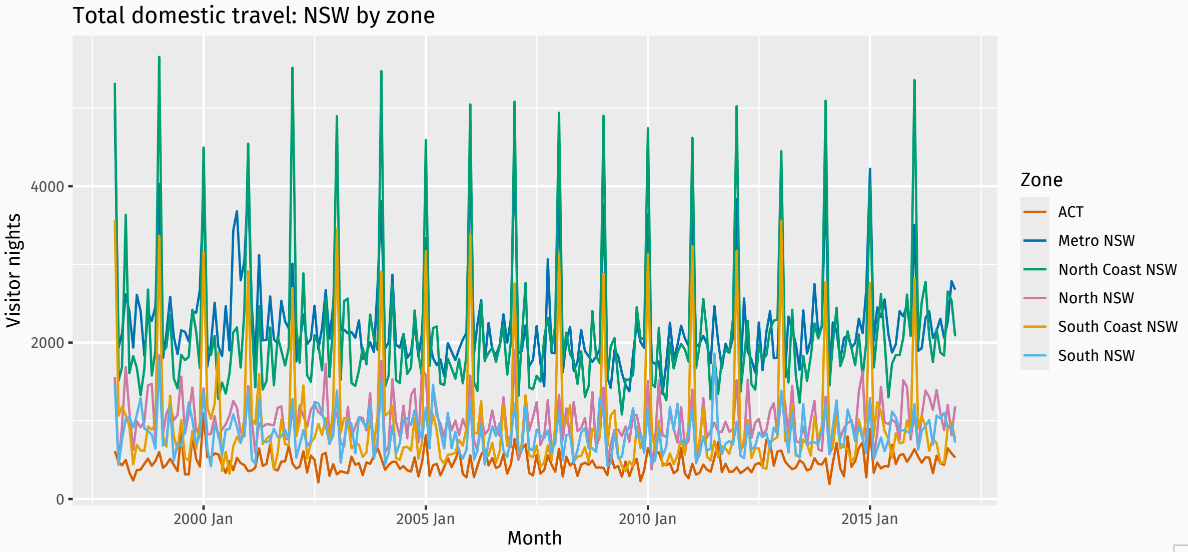
Australian tourism data



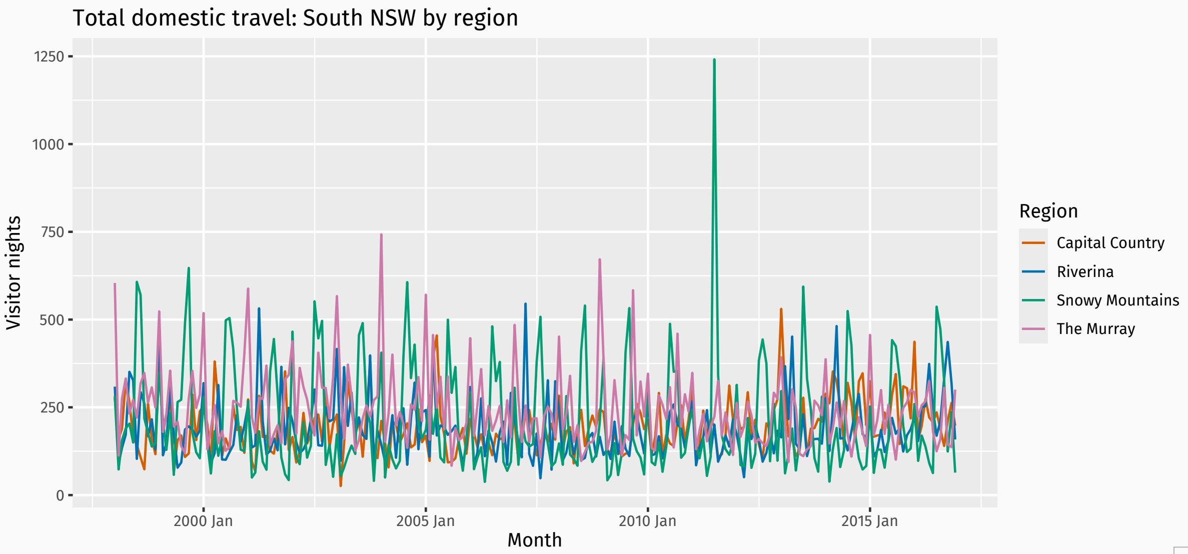
Australian tourism data



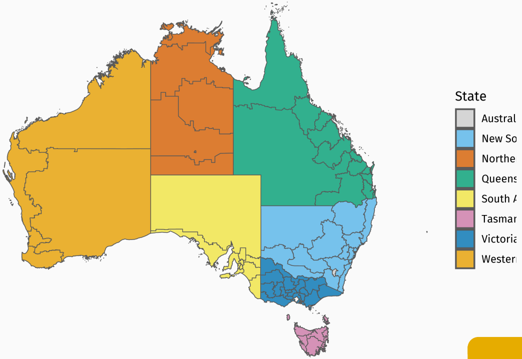
Australian tourism data



Australian tourism data



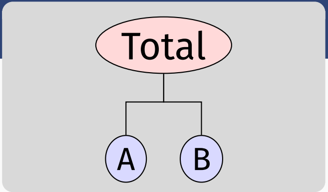
Australian tourism data



Aggregation level	# series
National	1
State	7
Zone	27
Region	76
Total	**111**

- Need forecasts at all levels of aggregation.
- Compute **base forecasts** using univariate models. These will not add up.
- Adjust base forecasts to ensure they are "coherent" giving **reconciled forecasts**.

Notation



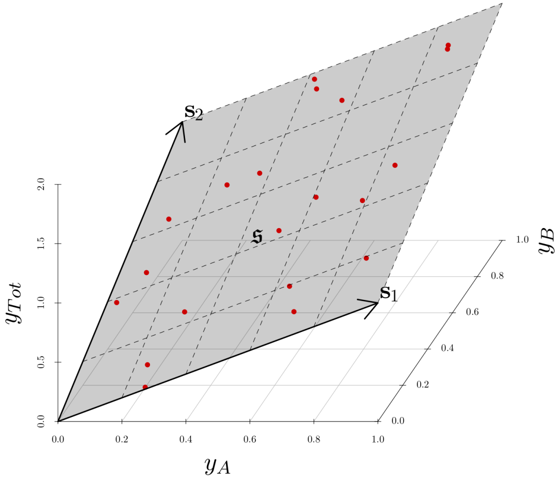
Almost all collections of time series with linear constraints can be written as

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

- $\boldsymbol{y}_t$ = 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.

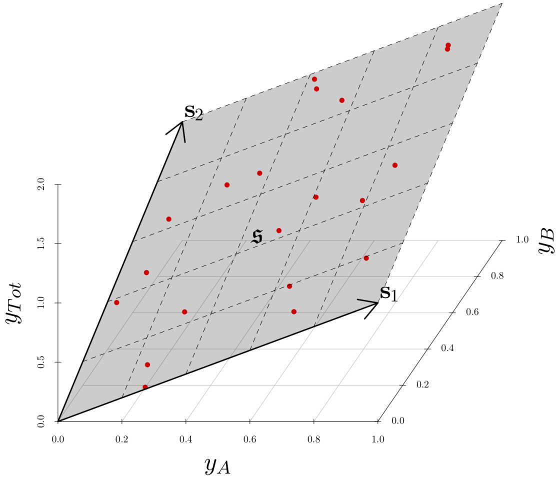
$$\boldsymbol{y}_t = \begin{pmatrix} y_{\text{Total},t} \\ y_{A,t} \\ y_{B,t} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}}_{\boldsymbol{S}} \underbrace{\begin{pmatrix} y_{A,t} \\ y_{B,t} \end{pmatrix}}_{\boldsymbol{b}_t}$$

Projections onto the coherent subspace



$$y_{\text{Total}} = y_A + y_B$$

Projections onto the coherent subspace

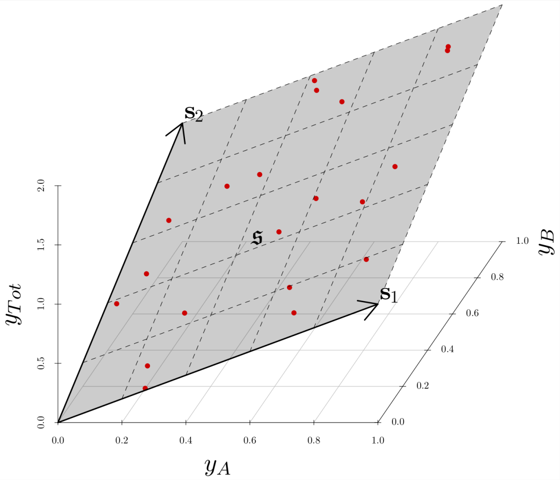


$y_{\text{Total}} = y_A + y_B$

Base forecasts

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

Projections onto the coherent subspace



$y_{\text{Total}} = y_A + y_B$

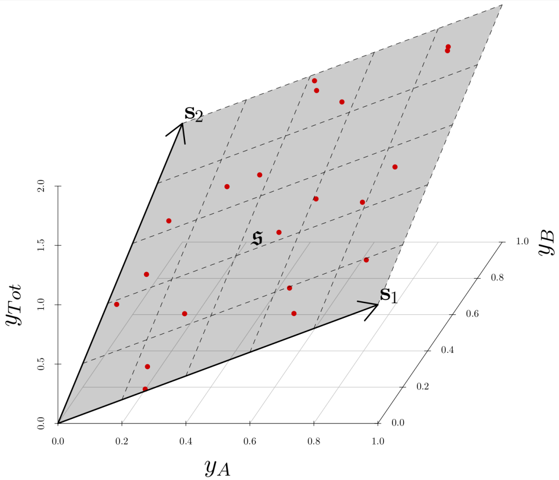
Base forecasts

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

Reconciled forecasts

Let $\boldsymbol{M}$ be a projection matrix.
$\tilde{\boldsymbol{y}}_{t+h|t} = \boldsymbol{M}\hat{\boldsymbol{y}}_{t+h|t}$ "reconciles" $\hat{\boldsymbol{y}}_{t+h|t}$.

Projections onto the coherent subspace



$y_{\text{Total}} = y_A + y_B$

Base forecasts

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

Reconciled forecasts

Let $\boldsymbol{M}$ be a projection matrix. $\tilde{\boldsymbol{y}}_{t+h|t} = \boldsymbol{M}\hat{\boldsymbol{y}}_{t+h|t}$ "reconciles" $\hat{\boldsymbol{y}}_{t+h|t}$.

- $\boldsymbol{S}$ forms a basis set for $\mathfrak{s}$
- All 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.
- How to choose the best Ψ?

Minimum trace reconciliation

Wickramasuriya et al (2019)

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

Reconciled forecasts

Base forecasts

Minimum trace reconciliation

Wickramasuriya et al (2019)

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

Reconciled forecasts   Base forecasts

- Base forecast covariance:
  $\boldsymbol{W}_h = \text{Var}[\boldsymbol{y}_{T+h} - \hat{\boldsymbol{y}}_{t+h|t} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_T]$
- Reconciled forecast covariance:
  $\boldsymbol{V}_h = \text{Var}[\boldsymbol{y}_{T+h} - \tilde{\boldsymbol{y}}_{t+h|t} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_T]) = \boldsymbol{M}\boldsymbol{W}_h\boldsymbol{M}'$

Minimum trace reconciliation

Wickramasuriya et al (2019)

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

Reconciled forecasts Base forecasts

- Base forecast covariance:
  $\boldsymbol{W}_h = \text{Var}[\boldsymbol{y}_{T+h} - \hat{\boldsymbol{y}}_{t+h|t} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_T]$
- Reconciled forecast covariance:
  $\boldsymbol{V}_h = \text{Var}[\boldsymbol{y}_{T+h} - \tilde{\boldsymbol{y}}_{t+h|t} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_T]) = \boldsymbol{M}\boldsymbol{W}_h\boldsymbol{M}'$

Minimum trace (MinT) reconciliation

If $\boldsymbol{M}$ is a projection, then trace of $\boldsymbol{V}_h$ is minimized when $\Psi = \boldsymbol{W}_h$

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

Linear projections

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

Reconciliation method	$\boldsymbol{W}_h$
OLS	$\boldsymbol{I}$
WLS(var)	$\text{diag}(\boldsymbol{W}_1)$
WLS(struct)	$\text{diag}(\boldsymbol{S}\boldsymbol{1})$
MinT(sample)	$\hat{\boldsymbol{W}}_{\text{sam}}$
MinT(shrink)	$\hat{\boldsymbol{W}}_{\text{shr}}$

- All approximate MinT by assuming $\boldsymbol{W}_h = k_h\boldsymbol{W}_1$.
- $\hat{\boldsymbol{W}}_{\text{sam}}$ is sample estimate of the residual covariance matrix
- $\hat{\boldsymbol{W}}_{\text{shr}}$ is shrinkage estimator $\tau\text{diag}(\hat{\boldsymbol{W}}_{\text{sam}}) + (1-\tau)\hat{\boldsymbol{W}}_{\text{sam}}$ where τ selected optimally.

Example: Australian tourism

```r
tourism
```

```
# A tsibble: 17,328 x 5 [1M]
# Key:       state, zone, region [76]
   state zone  region      month visitors
   <chr> <chr> <chr>       <mth>    <dbl>
 1 NSW   ACT   Canberra 1998 Jan     612.
 2 NSW   ACT   Canberra 1998 Feb     471.
 3 NSW   ACT   Canberra 1998 Mar     430.
 4 NSW   ACT   Canberra 1998 Apr     499.
 5 NSW   ACT   Canberra 1998 May     338.
 6 NSW   ACT   Canberra 1998 Jun     236.
 7 NSW   ACT   Canberra 1998 Jul     371.
 8 NSW   ACT   Canberra 1998 Aug     375.
 9 NSW   ACT   Canberra 1998 Sep     449.
10 NSW   ACT   Canberra 1998 Oct     517.
# i 17,318 more rows
```

Example: Australian tourism

```r
tourism_agg <- tourism |>
  aggregate_key(state / zone / region, visitors = sum(visitors))
```

```
# A tsibble: 25,308 x 5 [1M]
# Key:       state, zone, region [111]
      month state        zone         region       visitors
      <mth> <chr*>       <chr*>       <chr*>          <dbl>
 1 1998 Jan <aggregated> <aggregated> <aggregated>   45151.
 2 1998 Feb <aggregated> <aggregated> <aggregated>   17295.
 3 1998 Mar <aggregated> <aggregated> <aggregated>   20725.
 4 1998 Apr <aggregated> <aggregated> <aggregated>   25389.
 5 1998 May <aggregated> <aggregated> <aggregated>   20330.
 6 1998 Jun <aggregated> <aggregated> <aggregated>   18238.
 7 1998 Jul <aggregated> <aggregated> <aggregated>   23005.
 8 1998 Aug <aggregated> <aggregated> <aggregated>   23033.
 9 1998 Sep <aggregated> <aggregated> <aggregated>   22483.
10 1998 Oct <aggregated> <aggregated> <aggregated>   24845.
# i 25,298 more rows
```

Example: Australian tourism

```r
fit <- tourism_agg |>
  filter(year(month) <= 2015) |>
  model(ets = ETS(visitors))
```

```
# A mable: 111 x 4
# Key:     state, zone, region [111]
   state  zone            region                    ets
   <chr*> <chr*>          <chr*>                <model>
 1 NSW    ACT             Canberra         <ETS(M,N,A)>
 2 NSW    ACT             <aggregated>     <ETS(M,N,A)>
 3 NSW    Metro NSW       Central Coast    <ETS(M,N,A)>
 4 NSW    Metro NSW       Sydney           <ETS(M,N,A)>
 5 NSW    Metro NSW       <aggregated>     <ETS(M,N,A)>
 6 NSW    North Coast NSW Hunter           <ETS(M,N,M)>
 7 NSW    North Coast NSW North Coast NSW  <ETS(M,N,M)>
 8 NSW    North Coast NSW <aggregated>     <ETS(M,N,M)>
 9 NSW    North NSW       Blue Mountains   <ETS(M,N,M)>
10 NSW    North NSW       Central NSW      <ETS(A,N,A)>
# i 101 more rows
```

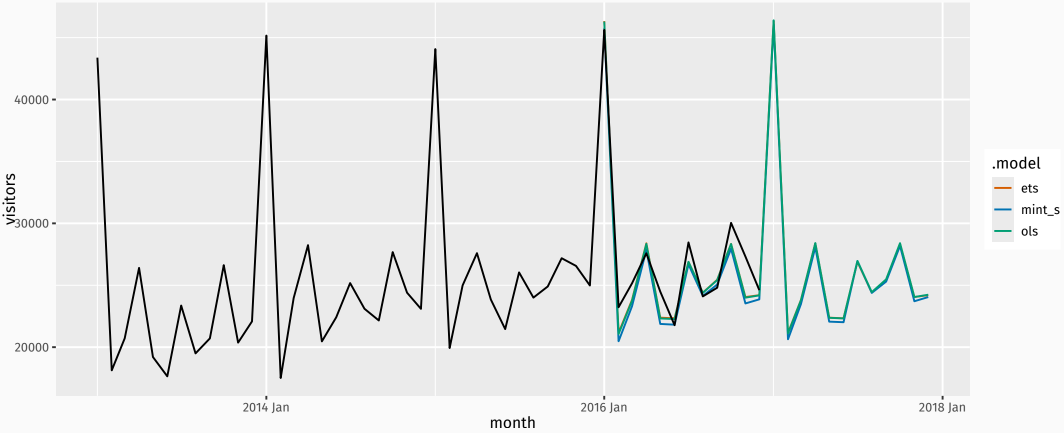
Example: Australian tourism

```
fc <- fit |>
  reconcile(
    ols = min_trace(ets, method = "ols"),
    mint_s = min_trace(ets, method = "mint_shrink"),
  ) |>
  forecast(h = "2 years")
```

```
# A fable: 7,992 x 7 [1M]
# Key:     state, zone, region, .model [333]
  state  zone   region   .model    month
  <chr*> <chr*> <chr*>   <chr>     <mth>
1 NSW    ACT    Canberra ets    2016 Jan
2 NSW    ACT    Canberra ets    2016 Feb
3 NSW    ACT    Canberra ets    2016 Mar
4 NSW    ACT    Canberra ets    2016 Apr
5 NSW    ACT    Canberra ets    2016 May
# i 7,987 more rows
# i 2 more variables: visitors <dist>, .mean <dbl>
```

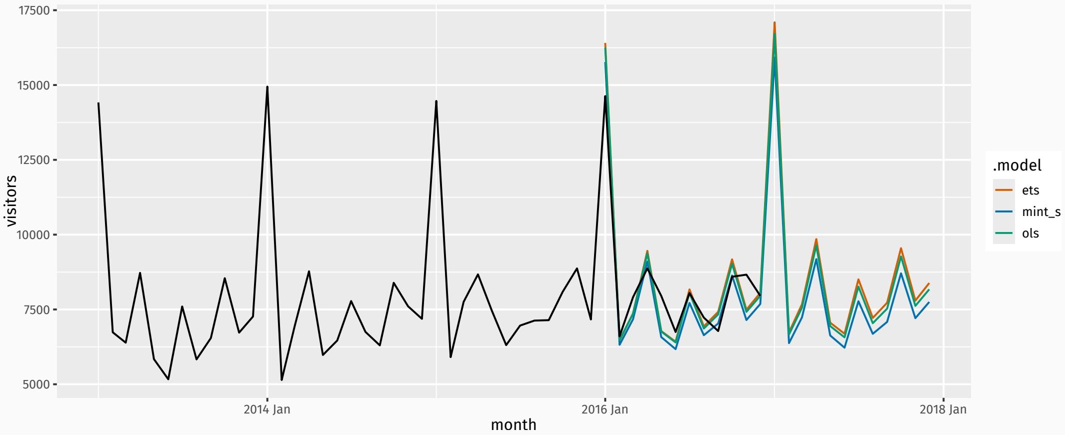
Example: Australian tourism

```r
fc |>
  filter(is_aggregated(state)) |>
  autoplot(filter(tourism_agg, year(month) > 2012), level = NULL)
```



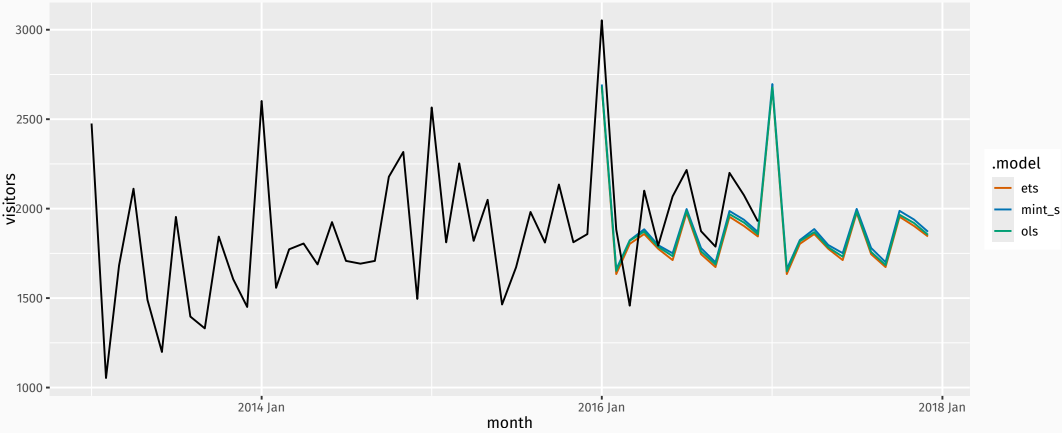
Example: Australian tourism

```r
fc |>
  filter(state == "NSW" & is_aggregated(zone)) |>
  autoplot(filter(tourism_agg, year(month) > 2012), level = NULL)
```



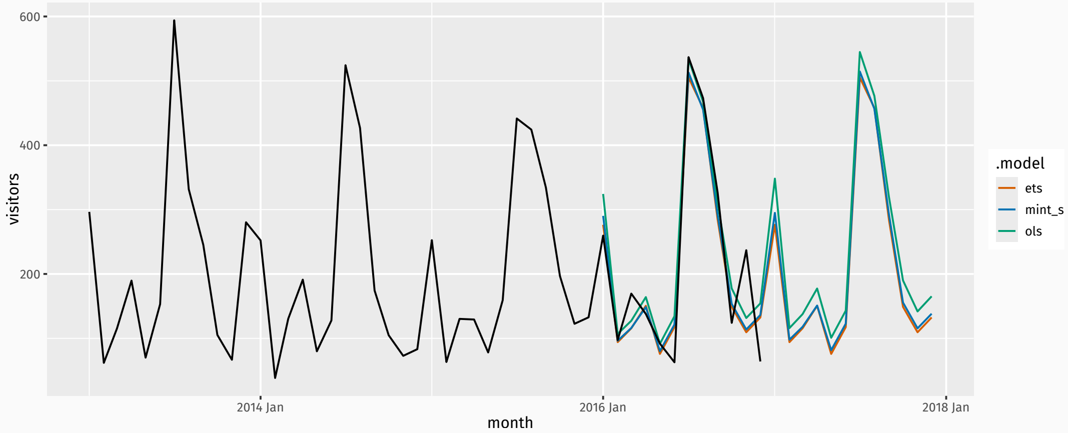
Example: Australian tourism

```
fc |>
  filter(region == "Melbourne") |>
  autoplot(filter(tourism_agg, year(month) > 2012), level = NULL)
```



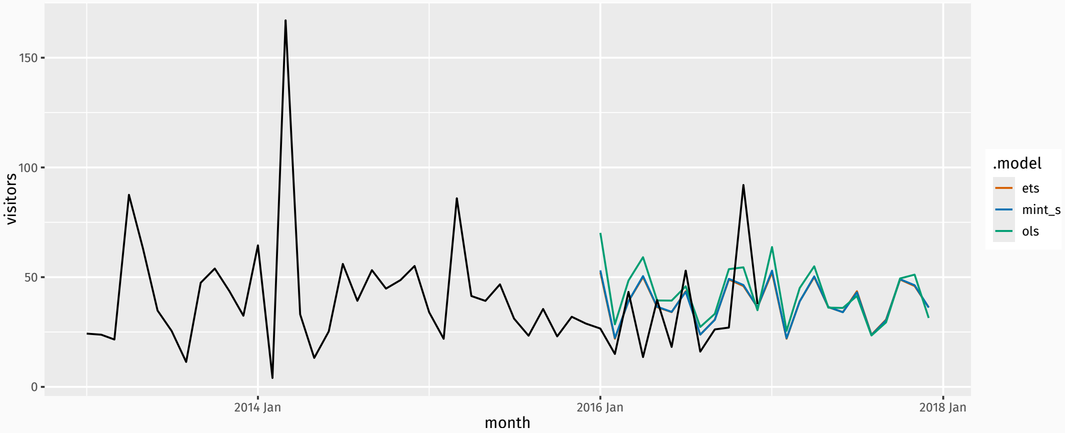
Example: Australian tourism

```r
fc |>
  filter(region == "Snowy Mountains") |>
  autoplot(filter(tourism_agg, year(month) > 2012), level = NULL)
```



Example: Australian tourism

```r
fc |>
  filter(region == "Barossa") |>
  autoplot(filter(tourism_agg, year(month) > 2012), level = NULL)
```



Performance evaluation

$$\text{RMSSE} = \sqrt{\text{mean}(q_j^2)}$$

$$q_j^2 = \frac{e_j^2}{\frac{1}{T-m}\sum_{t=m+1}^{T}(y_t - y_{t-m})^2}$$

- y_t = observation for period t
- e_j = forecast error for forecast horizon j
- T = size of training set
- m = 12

Example: Australian tourism

```r
accuracy(fc, tourism_agg, measures = list(rmsse = RMSSE))
```

```
# A tibble: 333 x 7
   .model state        zone         region       .type rmsse level
   <chr>  <chr*>       <chr*>       <chr*>       <chr> <dbl> <fct>
 1 ets    <aggregated> <aggregated> <aggregated> Test  0.755 National
 2 mint_s <aggregated> <aggregated> <aggregated> Test  0.896 National
 3 ols    <aggregated> <aggregated> <aggregated> Test  0.768 National
 4 ets    NSW          <aggregated> <aggregated> Test  0.921 State
 5 mint_s NSW          <aggregated> <aggregated> Test  0.893 State
 6 ols    NSW          <aggregated> <aggregated> Test  0.881 State
 7 ets    NT           <aggregated> <aggregated> Test  1.24  State
 8 mint_s NT           <aggregated> <aggregated> Test  1.22  State
 9 ols    NT           <aggregated> <aggregated> Test  1.18  State
10 ets    QLD          <aggregated> <aggregated> Test  0.860 State
# i 323 more rows
```

Example: Australian tourism

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

```
# A tibble: 3 x 2
  .model rmsse
  <chr>  <dbl>
1 ols    0.863
2 mint_s 0.866
3 ets    0.880
```

Example: Australian tourism

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

```
# A tibble: 3 x 2
  .model rmsse
  <chr>  <dbl>
1 ols    0.863
2 mint_s 0.866
3 ets    0.880
```

- Overall, both reconciliation methods are more accurate than the base ETS forecasts.

Example: Australian tourism

```r
accuracy(fc, tourism_agg, measures = list(rmsse = RMSSE)) |>
  summarise(rmsse = sqrt(mean(rmsse^2)), .by = c(.model, level)) |>
```

```
# A tibble: 12 x 3
   .model level    rmsse
   <chr>  <fct>    <dbl>
 1 ets    National 0.755
 2 ols    National 0.768
 3 mint_s National 0.896
 4 ols    State    0.905
 5 ets    State    0.919
 6 mint_s State    0.953
 7 ols    Zone     0.912
 8 mint_s Zone     0.914
 9 ets    Zone     0.935
10 mint_s Region   0.839
11 ols    Region   0.843
12 ets    Region   0.858
```

Example: Australian tourism

```
accuracy(fc, tourism_agg, measures = list(rmsse = RMSSE)) |>
  summarise(rmsse = sqrt(mean(rmsse^2)), .by = c(.model, level)) |>
```

```
# A tibble: 12 x 3
   .model level    rmsse
   <chr>  <fct>    <dbl>
 1 ets    National 0.755
 2 ols    National 0.768
 3 mint_s National 0.896
 4 ols    State    0.905
 5 ets    State    0.919
 6 mint_s State    0.953
 7 ols    Zone     0.912
 8 mint_s Zone     0.914
 9 ets    Zone     0.935
10 mint_s Region   0.839
11 ols    Region   0.843
12 ets    Region   0.858
```

- Reconciliation is best for all levels except national.
- Improvements due to reconciliation are greater at lower levels.

Mean square error bounds

Panagiotelis, Gamakumara, Athanasopoulos & Hyndman (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.

Mean square error bounds

Wickramasuriya (2021)

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

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

Mean square error bounds

Wickramasuriya (2021)

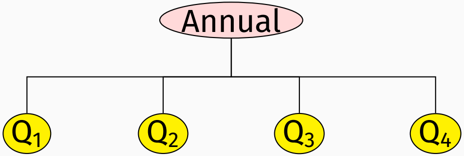
$$
\begin{aligned}
&\operatorname{tr}\Big(\mathrm{E}[\boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}_{t+h|t}{}^{\text{MinT}}]'[\boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}_{t+h|t}{}^{\text{MinT}}]\Big) \\
\le\; &\operatorname{tr}\Big(\mathrm{E}[\boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}_{t+h|t}{}^{\text{OLS}}]'[\boldsymbol{y}_{t+h}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

Using sums of variances:

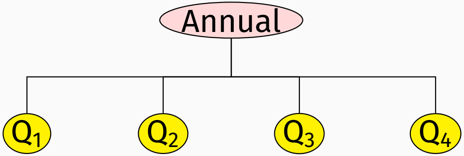
- MinT reconciliation is better than OLS reconciliation
- OLS reconciliation is better than base forecasts

Outline

Temporal reconciliation: quarterly data

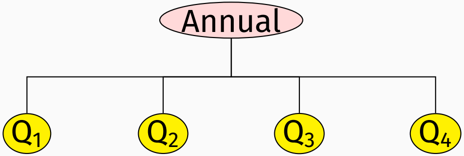


Temporal reconciliation: quarterly data



- Forecast series at each available frequency.
- Optimally combine forecasts within the same year.

Temporal reconciliation: quarterly data

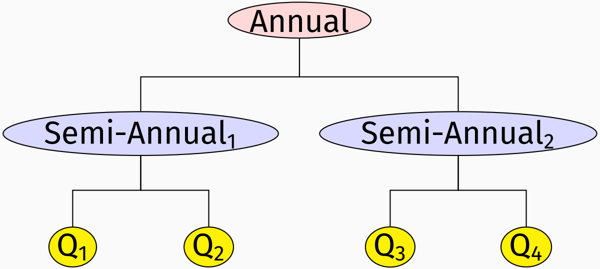


$$\boldsymbol{y}_\tau = \begin{bmatrix} x_\tau^{[4]} \\ x_{\tau,1}^{[1]} \\ x_{\tau,2}^{[1]} \\ x_{\tau,3}^{[1]} \\ x_{\tau,4}^{[1]} \end{bmatrix} \qquad \boldsymbol{S} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Forecast series at each available frequency.
- Optimally combine forecasts within the same year.

τ = year

Temporal reconciliation: quarterly data

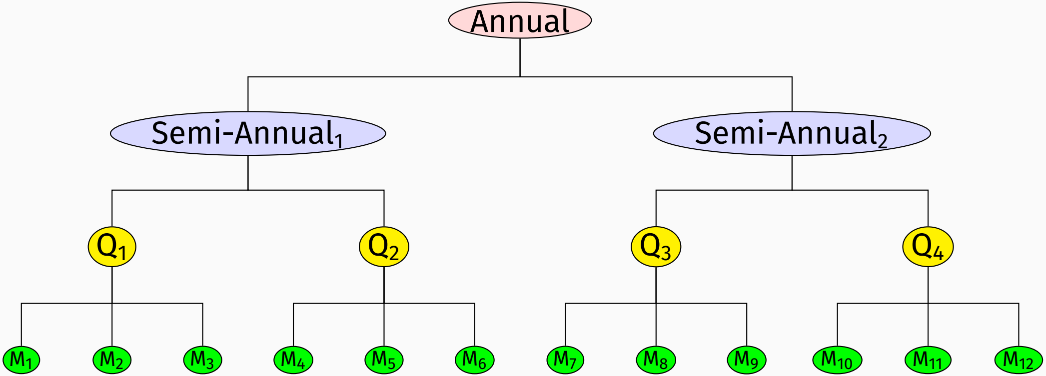


- Forecast series at each available frequency.
- Optimally combine forecasts within the same year.

$$\boldsymbol{y}_\tau = \begin{bmatrix} x_\tau^{[4]} \\ x_{\tau,1}^{[2]} \\ x_{\tau,2}^{[2]} \\ x_{\tau,1}^{[1]} \\ x_{\tau,2}^{[1]} \\ x_{\tau,3}^{[1]} \\ x_{\tau,4}^{[1]} \end{bmatrix} \qquad \boldsymbol{S} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

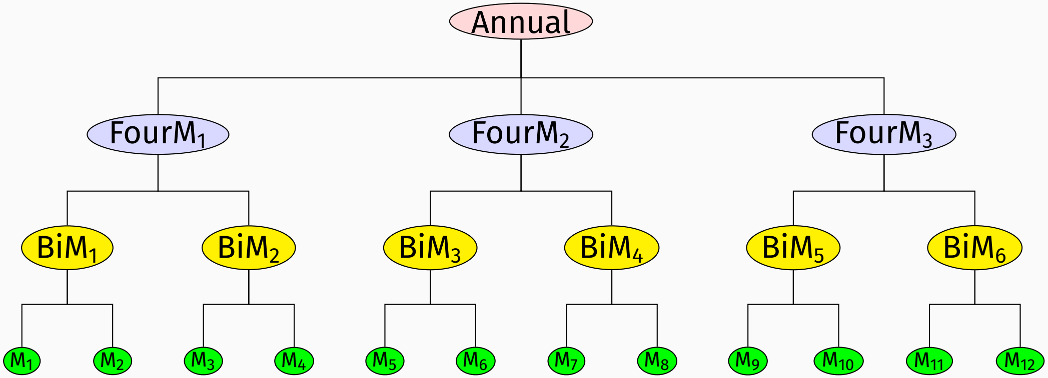
τ = index of largest temporal aggregation level.

Temporal reconciliation: monthly data



- Forecast series at each available frequency.
- Optimally combine forecasts within the same year.

Temporal reconciliation: monthly data



- Forecast series at each available frequency.
- Optimally combine forecasts within the same year.

Temporal reconciliation: monthly data

$$
\boldsymbol{y}_\tau = \begin{bmatrix} x_\tau^{[12]} \\ \boldsymbol{x}_\tau^{[6]} \\ \boldsymbol{x}_\tau^{[4]} \\ \boldsymbol{x}_\tau^{[3]} \\ \boldsymbol{x}_\tau^{[2]} \\ \boldsymbol{x}_\tau^{[1]} \end{bmatrix}
\qquad
\boldsymbol{S} = \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 & & & & & & \vdots & & & & & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 & & & & & & \vdots & & & & & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
 & & & & & & \boldsymbol{I}_{12} & & & & &
\end{bmatrix}
$$

Temporal reconciliation

For a time series $y_1, \dots, y_T$, observed at frequency m:

$$x_j^{[k]} = \sum_{t=(j-1)k+1}^{jk} y_t \qquad \text{for } j = 1, \dots, \lfloor T/k \rfloor$$

- $k \in \mathcal{K} = \{k_1, \dots, k_p\}$ denote the p factors of m in ascending order, where $k_1 = 1$ and $k_p = m$
- $x_j^{[1]} = y_t$
- A single unique hierarchy is only possible when there are no coprime pairs in $\mathcal{K}$.
- $M_k = m/k$ is seasonal period of aggregated series.

Temporal reconciliation

$$\boldsymbol{x}_\tau = \boldsymbol{S}\boldsymbol{x}_\tau^{[1]}, \qquad \boldsymbol{S} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{I} \end{bmatrix}$$

where

$$\boldsymbol{x}_\tau = \begin{bmatrix} \boldsymbol{x}_\tau^{[k_p]} \\ \boldsymbol{x}_\tau^{[k_{p-1}]} \\ \vdots \\ \boldsymbol{x}_\tau^{[k_1]} \end{bmatrix} \qquad \boldsymbol{x}_\tau^{[k]} = \begin{bmatrix} x_{M_k(\tau-1)+1}^{[k]} \\ x_{M_k(\tau-1)+2}^{[k]} \\ \vdots \\ x_{M_k\tau}^{[k]} \end{bmatrix} \qquad \boldsymbol{A} = \begin{bmatrix} \boldsymbol{1}'_m \\ \boldsymbol{I}_{m/k_{p-1}} \otimes \boldsymbol{1}'_{k_{p-1}} \\ \vdots \\ \boldsymbol{I}_{m/k_2} \otimes \boldsymbol{1}'_{k_2} \end{bmatrix}$$

τ is time index for most aggregated series,

$k \in \mathcal{K} = \{k_1, \dots, k_p\}, \quad k_1 = 1, \quad k_p = m, \quad \tau = 1, \dots, T/m.$

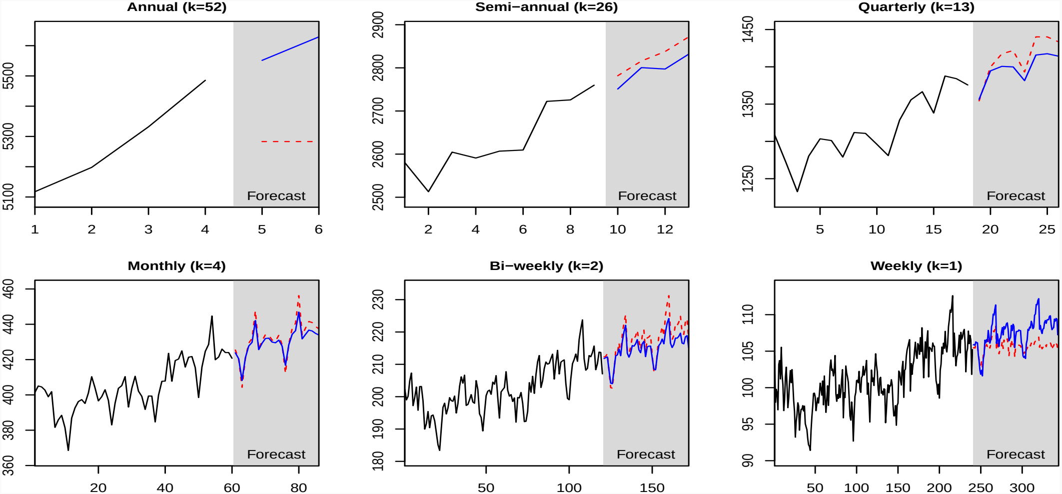
Example: Accident & emergency services demand

Weekly A&E demand data: 7 November 2010 to 7 June 2015.

Type 1 Departments — Major A&E
Type 2 Departments — Single Specialty
Type 3 Departments — Other A&E/Minor Injury Unit
Total Attendances
Type 1 Departments — Major A&E $>$ 2 hours
Type 2 Departments — Single Specialty $>$ 2 hours
Type 3 Departments — Other A&E/Minor Injury Unit $>$ 2 hours
Total Attendances $>$ 2 hours
Emergency Admissions via Type 1 A&E
Total Emergency Admissions via A&E
Other Emergency Admissions (i.e not via A&E)
Total Emergency Admissions
Number of patients spending $>$ 2 hours from decision to admit to admission

Example: Accident & emergency services demand

Total emergency admissions via A&E



Example: Accident & emergency services demand

Test set: last 52 weeks

MASE comparison (ARIMA models)

Aggregation Level	h	Base	Reconciled	Change
Annual	1	3.4	1.9	-42.9%
Weekly	1-52	2.0	1.9	-5.0%
Weekly	13	2.3	1.9	-16.2%
Weekly	4	1.9	1.5	-18.6%
Weekly	1	1.6	1.3	-17.2%

Temporal reconciliation: M3 monthly series

- Apply temporal reconciliation to all 1428 monthly series from M3 competition
- Forecast horizon $h = 18$ months
- ETS and ARIMA models
- Measure percentage difference to base forecasts
- Reconciliation methods:
  - WLS_H (diagonal)
  - WLS_V (diagonal with common variances for same frequency)
  - WLS_S (diagonal/structural)

Temporal reconciliation: M3 monthly series

Improvement in MASE relative to base forecasts

		ETS				ARIMA			
Aggregation level	*h*	BU	WLS_H	WLS_V	WLS_S	BU	WLS_H	WLS_V	WLS_S
Annual	1	-12.1	-17.9	-17.8	**-18.5**	-25.4	-29.9	-29.9	**-30.2**
Semi-annual	3	0.0	-6.3	-6.0	**-6.9**	-2.9	-8.1	-8.2	**-9.4**
Four-monthly	4	3.1	-3.2	-3.0	**-3.4**	-1.8	-6.2	-6.5	**-7.1**
Quarterly	6	3.2	-2.8	-2.7	**-3.4**	-2.6	-6.9	-7.4	**-8.1**
Bi-monthly	9	2.7	-2.9	-3.0	**-3.7**	-1.3	-5.0	-5.5	**-6.3**
Monthly	18	0.0	-3.7	-4.6	**-5.0**	0.0	-1.9	-3.2	**-3.7**
Average	NA	-0.5	-6.1	-6.2	**-6.8**	-5.7	-9.7	-10.1	**-10.8**

Outline

Forecast Linear Augmented Projection (FLAP)

- We want to forecast multivariate series $\boldsymbol{y}_t$.
- Linear combinations of $\boldsymbol{y}_t$ may have better signal-noise ratio
- Construct many linear combinations $\boldsymbol{c}_t = \Phi\boldsymbol{y}_t$ (e.g., principal components or random combinations)
- Produce univariate forecasts of all series $\hat{\boldsymbol{y}}_{t+h|t}$ and all linear combinations $\hat{\boldsymbol{c}}_{t+h|t}$.
- Reconcile forecasts so they are coherent ($\tilde{\boldsymbol{c}}_{t+h|t} = \Phi\tilde{\boldsymbol{y}}_{t+h|t}$)

Forecast Linear Augmented Projection (FLAP)

- We want to forecast multivariate series $\boldsymbol{y}_t$.
- Linear combinations of $\boldsymbol{y}_t$ may have better signal-noise ratio
- Construct many linear combinations $\boldsymbol{c}_t = \Phi\boldsymbol{y}_t$ (e.g., principal components or random combinations)
- Produce univariate forecasts of all series $\hat{\boldsymbol{y}}_{t+h|t}$ and all linear combinations $\hat{\boldsymbol{c}}_{t+h|t}$.
- Reconcile forecasts so they are coherent ($\tilde{\boldsymbol{c}}_{t+h|t} = \Phi\tilde{\boldsymbol{y}}_{t+h|t}$)

$$\boldsymbol{z}_t = \begin{bmatrix}\boldsymbol{y}_t \\ \boldsymbol{c}_t\end{bmatrix} \qquad \tilde{\boldsymbol{z}}_{t+h} = \boldsymbol{M}\hat{\boldsymbol{z}}_{t+h|t}$$

Forecast error variance reduction

If we know the covariance matrix $\boldsymbol{W}_h = \text{Var}(\boldsymbol{z}_{t+h} - \hat{\boldsymbol{z}}_{t+h|t})$, then

1. The forecast error variance is reduced:
   - $\text{Var}(\boldsymbol{y}_{t+h} - \hat{\boldsymbol{y}}_{t+h|t}) - \text{Var}(\boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}_{t+h|t})$ is positive semi-definite.
2. The forecast error variance of each series monotonically decreases with increasing number of components.

Forecast error variance reduction

If we know the covariance matrix $\boldsymbol{W}_h = \text{Var}(\boldsymbol{z}_{t+h} - \hat{\boldsymbol{z}}_{t+h|t})$, then

1. The forecast error variance is reduced:
   - $\text{Var}(\boldsymbol{y}_{t+h} - \hat{\boldsymbol{y}}_{t+h|t}) - \text{Var}(\boldsymbol{y}_{t+h} - \tilde{\boldsymbol{y}}_{t+h|t})$ is positive semi-definite.
2. The forecast error variance of each series monotonically decreases with increasing number of components.

In practice, we need to:

- Estimate $\boldsymbol{W}_h$ (use the MinT shrinkage estimator).
- Construct the components, Φ.

Construction of Φ

Principal component analysis (PCA)

Finding the weights matrix Φ so that the resulting components **maximise variance**

Simulation

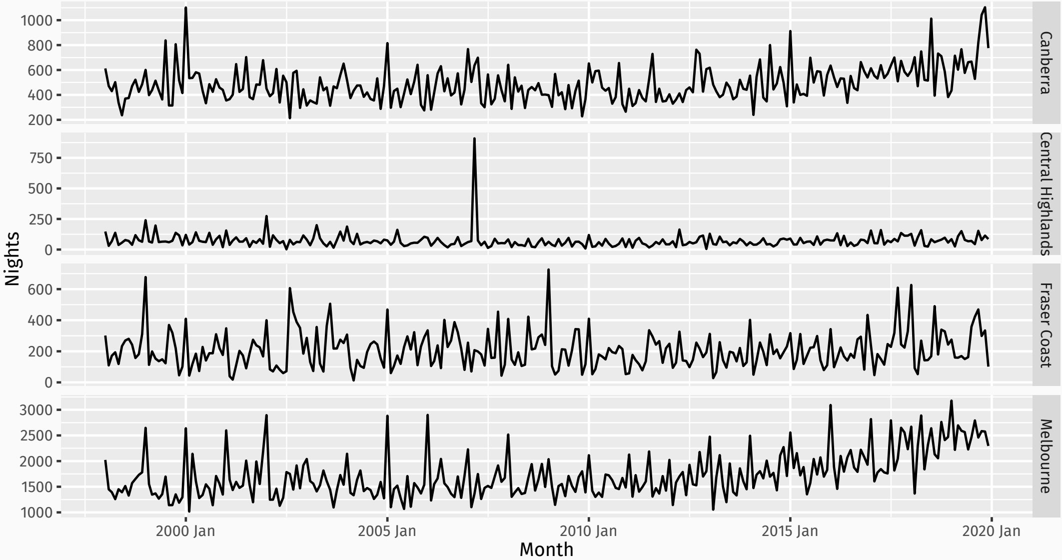
Generating values of Φ from a random distribution and normalising them to unit vectors

- Normal distribution
- Uniform distribution
- Orthonormal matrix

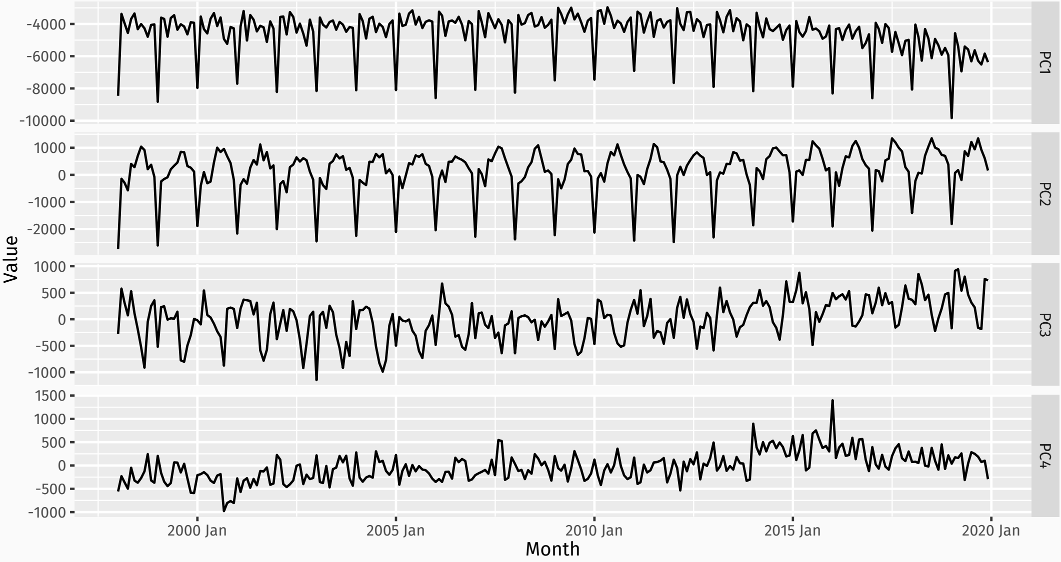
Monthly Australian regional tourism

- Monthly Australian tourism data set aggregated by region giving 77 series, from Jan 1998 to Dec 2019.
- Use expanding window time series cross-validation with $T = 84$ observations in first training set, and forecast horizons $h = 1, 2, \ldots, 12$.
- Fit univariate ETS models to each series.

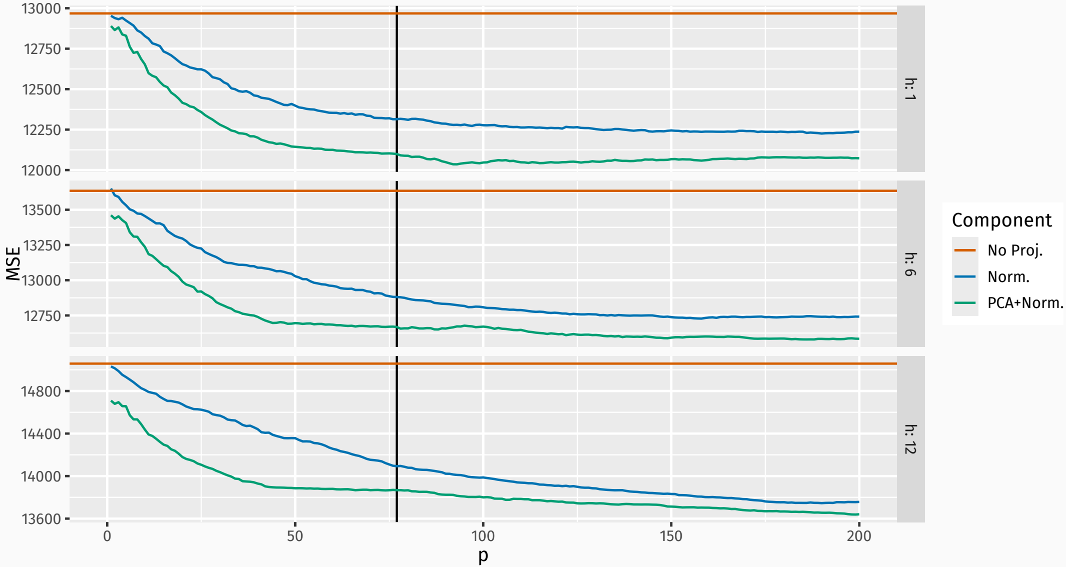
Monthly Australian regional tourism



Monthly Australian regional tourism



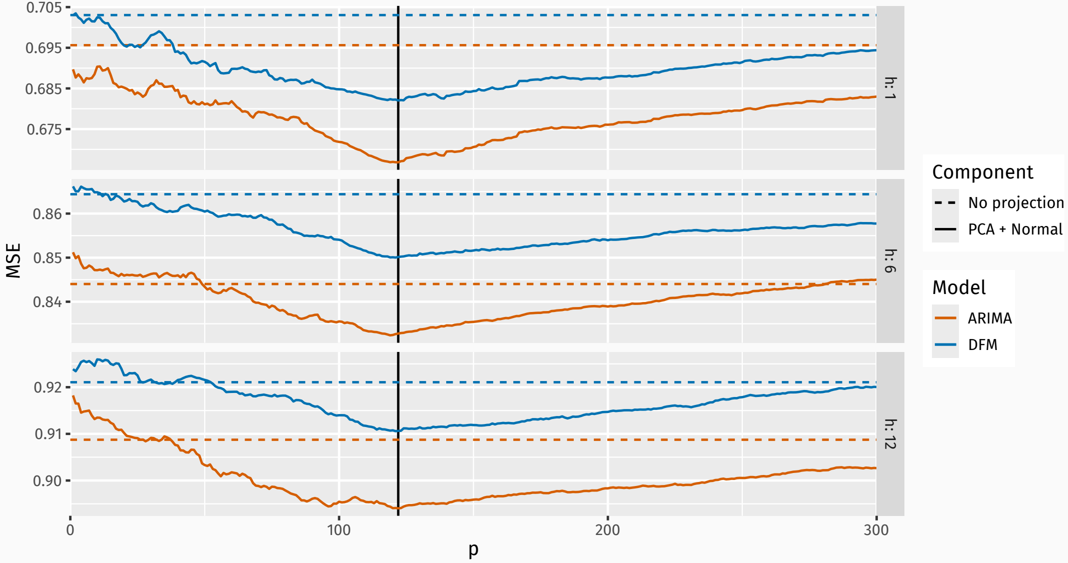
Monthly Australian regional tourism



FRED-MD

- Monthly data of macroeconomic variables (McCracken and Ng, 2016).
- Data from Jan 1959 – Sep 2023. 777 observations on 122 series.
- Same cleaning process as per McCracken and Ng (2016).
- All series scaled to have mean 0 and variance 1.
- Expanding time series cross-validation with initial size of 25 years and forecast horizon 12 months.

FRED-MD

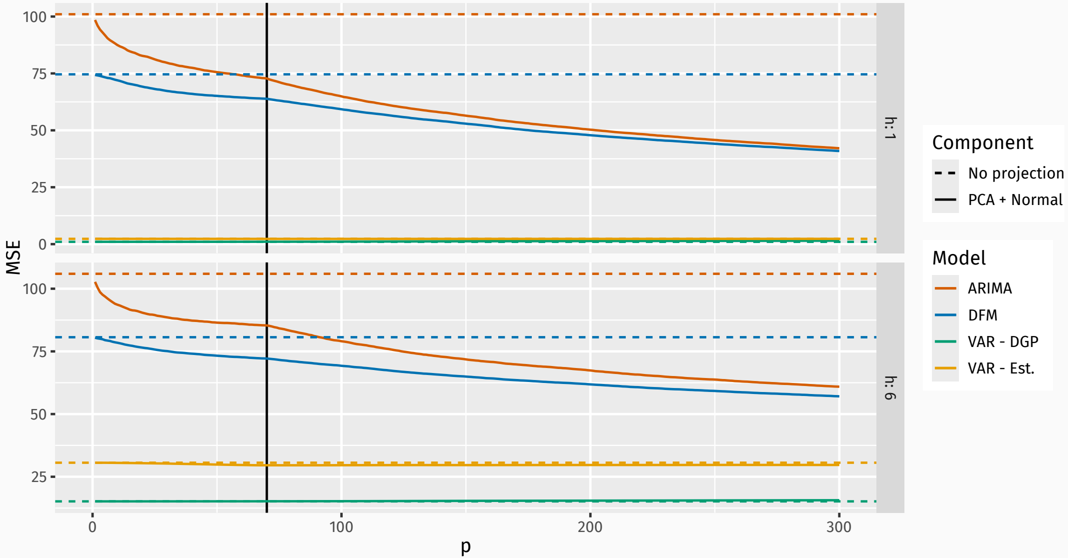


0.705
0.695
0.685
0.675
0.86
0.85
0.84
0.92
0.91
0.90
MSE
0
100
200
300
p
h: 1
h: 6
h: 12
Component
No projection
PCA + Normal
Model
ARIMA
DFM

Simulation

- Data generating process: VAR(3) with 70 variables
- Sample size: $T = 400$
- Number of repeated samples: 220
- Base models:
  - automatic ARIMA (based on AICc)
  - DFM (structure chosen using BIC, different model for each horizon)

Simulation



MSE
100
75
50
25
0
h: 1
100
75
50
25
h: 6
0
100
200
300
p
Component
No projection
PCA + Normal
Model
ARIMA
DFM
VAR - DGP
VAR - Est.

Outline

Software

Package	Language	Cross-sectional	Temporal	Cross-temporal	Probabilistic	Multivariate
`hts`	R	✓				
`thief`	R		✓			
`fable`	R	✓			✓	
`FoReco`	R	✓	✓	✓	✓	
`flap`	R					✓
`pyhts`	Python	✓	✓			
`hierarchicalforecast`	Python	✓			✓	

- `hts`, `thief`, and `FoReco` use `ts` objects
- `fable` uses `tsibble` objects
- `flap` uses matrices of base forecasts
- `fable` has plans to implement temporal and cross-temporal reconciliation

Thanks!

More information

robjhyndman.com/srilanka2024

- Athanasopoulos, G, RJ Hyndman, N Kourentzes & A Panagiotelis (2024). Forecast reconciliation: a review. *International J Forecasting* **40**(2), 430–456.
- Athanasopoulos, G, RJ Hyndman, N Kourentzes & F Petropoulos (2017). Forecasting with temporal hierarchies. *European J Operational Research* **262**(1), 60–74.
- Di Fonzo, T & D Girolimetto (2023). Cross-temporal forecast reconciliation: Optimal combination method and heuristic alternatives. *International J Forecasting* **39**(1), 39–57.
- Girolimetto, D, G Athanasopoulos, T Di Fonzo & RJ Hyndman (2024). Cross-temporal probabilistic forecast reconciliation: methodological and practical issues. *International J Forecasting* **40**(3), 1134–1151.
- Hyndman, RJ, RA Ahmed, G Athanasopoulos & HL Shang (2011). Optimal combination forecasts for hierarchical time series. *Computational Statistics & Data Analysis* **55**(9), 2579–2589.

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

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

Wickramasuriya, SL (2021). Properties of point forecast reconciliation approaches. *arXiv:2103.11129*.

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

Yang, YF, G Athanasopoulos, RJ Hyndman & A Panagiotelis (2024). Forecast Linear Augmented Projection (FLAP): A free lunch to reduce forecast error variance. *arXiv:2407.01868*.