Ten years of forecast reconciliation

Rob J Hyndman   ISF 2020

robjhyndman.com/isf2020

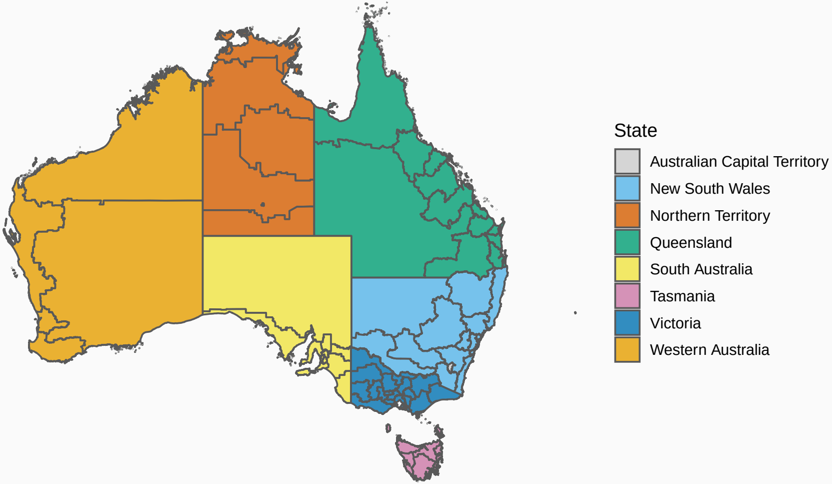
Photo by Edvard Alexander Rølvaag on Unsplash

Outline

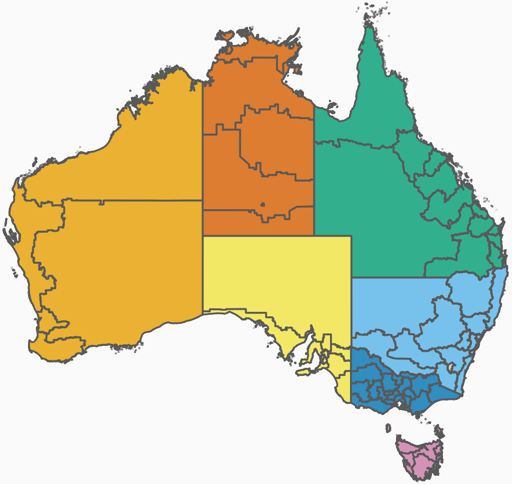
Outline

1. Hierarchical forecasting 20 years ago
2. Point forecast reconciliation
3. Example: Australian tourism
4. Probabilistic forecast reconciliation
5. Example: Australian electricity generation
6. Extensions

Australian tourism regions



Australian tourism regions

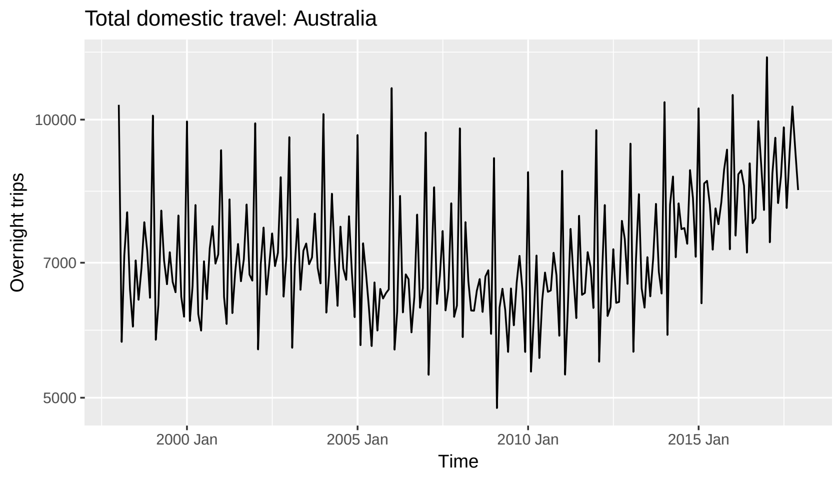


- Monthly data on visitor night from 1998 – 2017
- From *National Visitor Survey*, annual interviews of 120,000 Australians aged 15+.
- Geographical hierarchy split by
  - 7 states
  - 27 zones
  - 75 regions

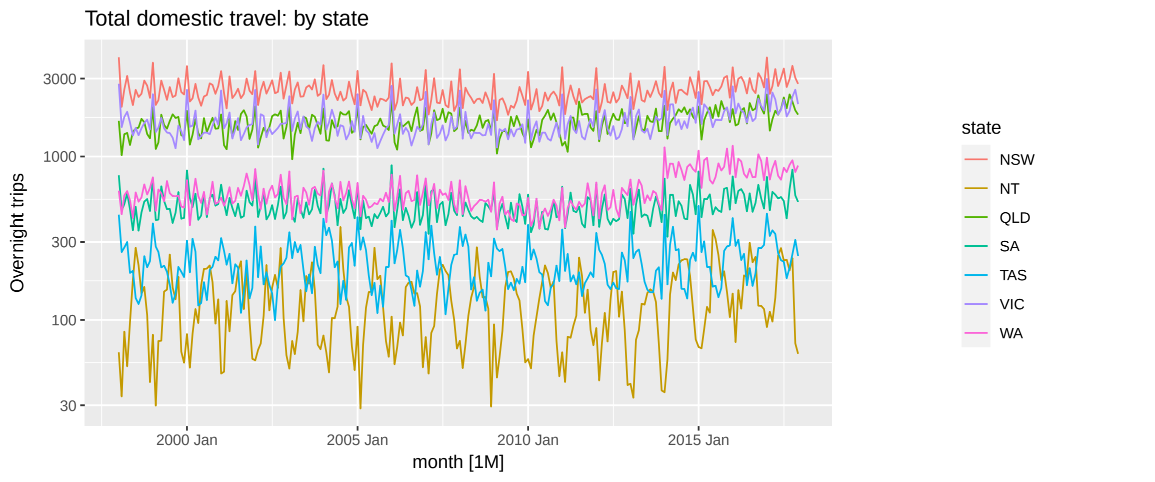
Australian tourism data

```
## # 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.
## # ... with 17,990 more rows
```

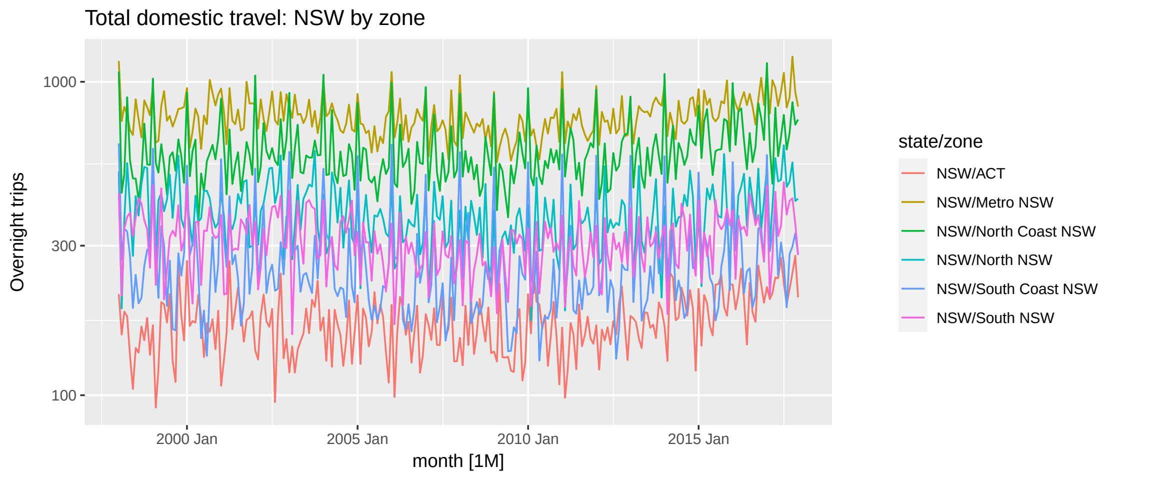
Australian tourism data



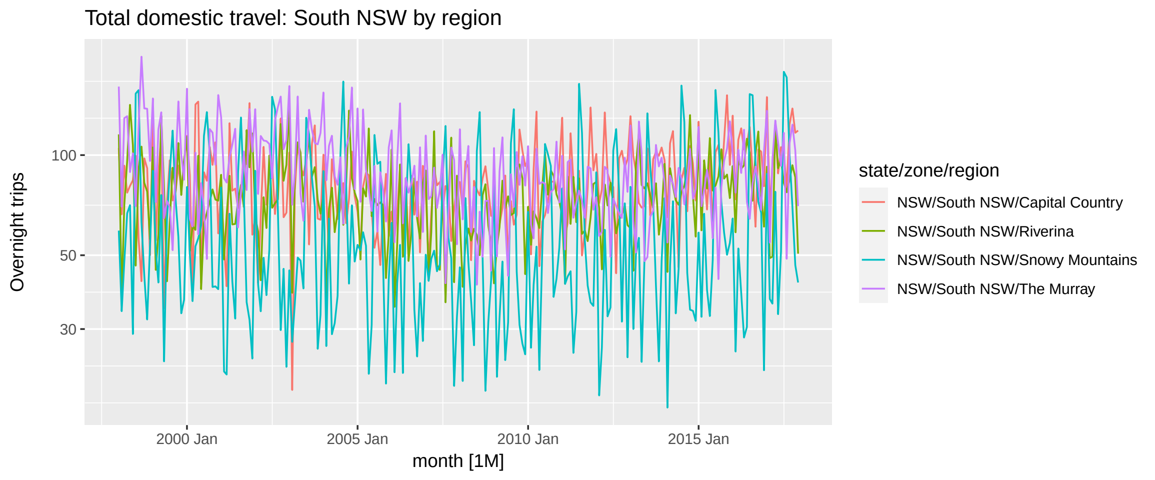
Australian tourism data



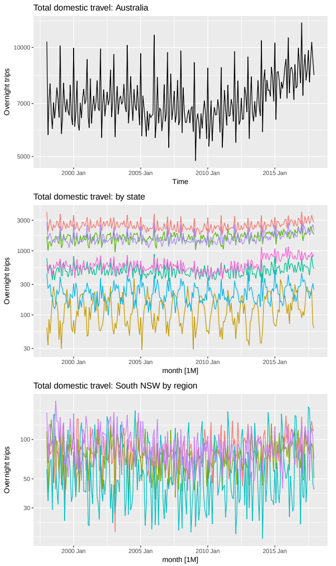
Australian tourism data

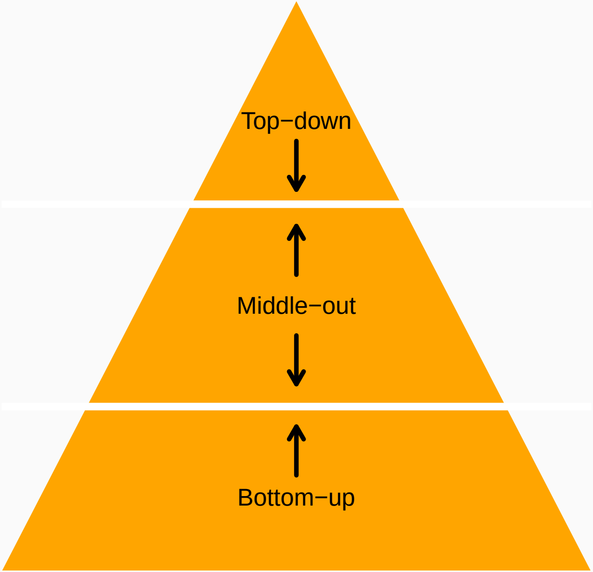


Australian tourism data



Hierarchical forecasting 20 years ago





Forecast reconciliation

- Forecast all series at all levels of aggregation.
- Reconcile forecasts using least squares optimization.

History

2001: Idea to use all available series to forecast Australia's labour market by occupation.

2004: PhD student Roman Ahmed begins, co-supervised with George Athanasopoulos.

2006: Presentation at ISF, Santander.

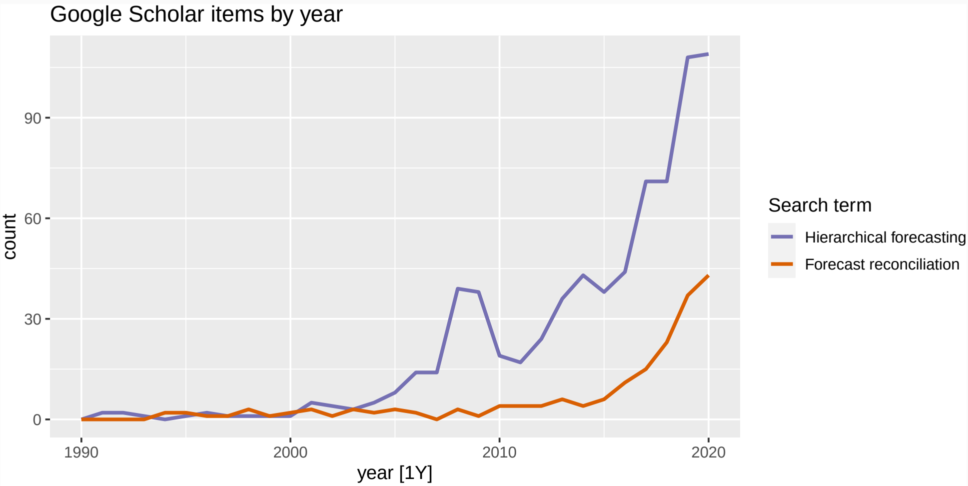
2007: Pre-print of "Optimal combination forecasts for hierarchical time series".

2009: Application to Australian tourism published in IJF.

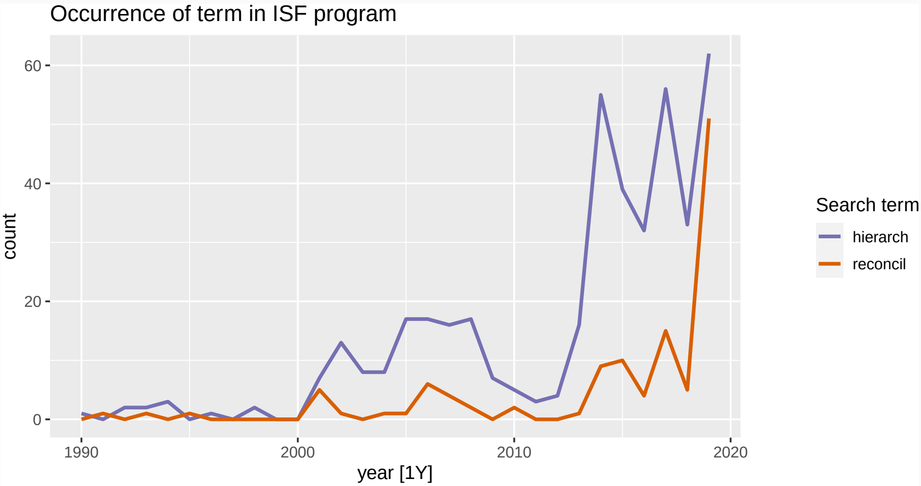
2010: First version of hts package on CRAN.

2011: "Optimal combination forecasts for hierarchical time series" appears in CSDA.

Forecast reconciliation research



Forecast reconciliation research



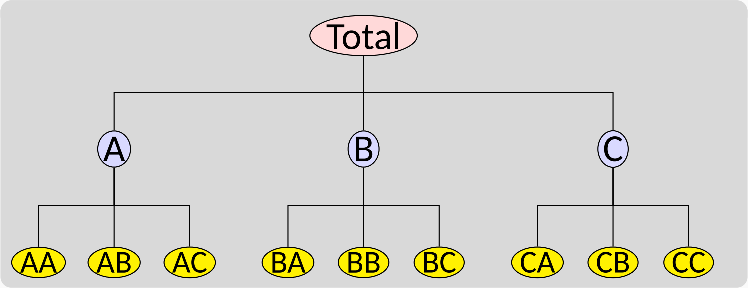
Outline

Point forecast reconciliation papers

- Hyndman, Ahmed, Athanasopoulos, Shang (2011 *CSDA*) Optimal combination forecasts for hierarchical time series.
- Hyndman, Lee, Wang (2016 *CSDA*) Fast computation of reconciled forecasts for hierarchical and grouped time series.
- Wickramasuriya, Athanasopoulos, Hyndman (2019 *JASA*) Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization.
- Panagiotelis, Gamakumara, Athanasopoulos, Hyndman (2020 *IJF*) Forecast reconciliation: A geometric view with new insights on bias correction.

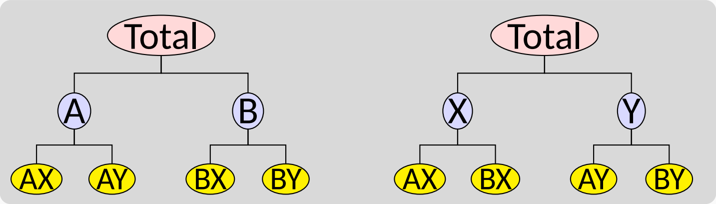
Hierarchical time series

A **hierarchical time series** is a collection of several time series that are linked together in a hierarchical structure.



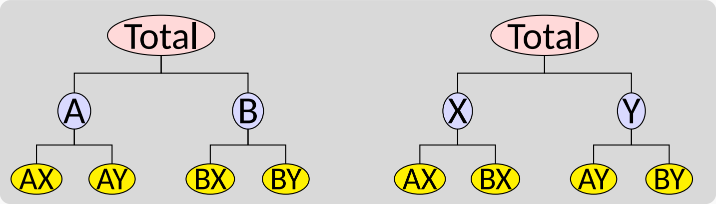
Grouped time series

A **grouped time series** is a collection of time series that can be grouped together in a number of non-hierarchical ways.



Grouped time series

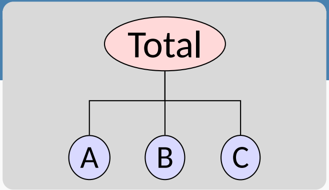
A **grouped time series** is a collection of time series that can be grouped together in a number of non-hierarchical ways.



Examples

- Tourism by state and purpose of travel
- Retail sales by product groups/sub groups, and by countries/regions

Hierarchical and grouped time series



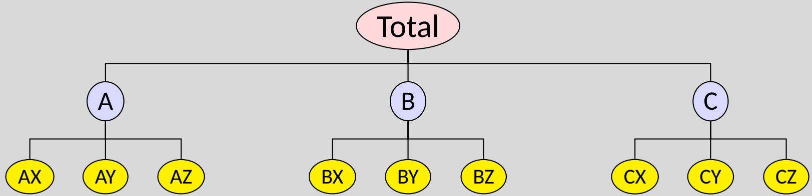
Every collection of time series with linear constraints can be written as

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

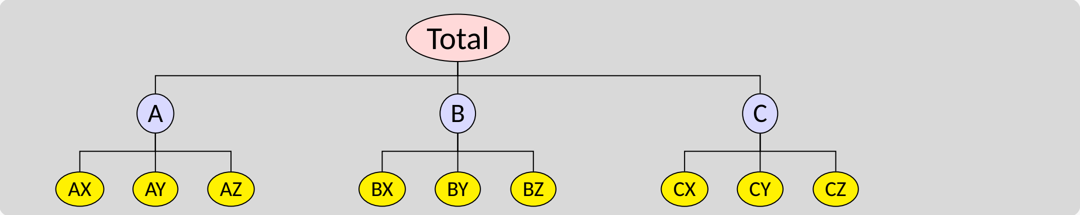
- $\boldsymbol{y}_t$ = vector of all series at time t
- y_t = aggregate of all series at time t.
- $y_{X,t}$ = value of series X at time t.
- $\boldsymbol{b}_t$ = vector of most disaggregated series at time t
- $\boldsymbol{S}$ = "summing matrix" containing the linear constraints.

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

Hierarchical time series

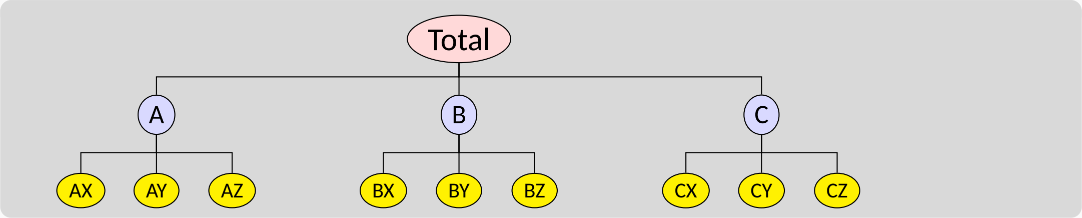


Hierarchical time series



$$\boldsymbol{y}_t = \begin{pmatrix} y_t \\ y_{A,t} \\ y_{B,t} \\ y_{C,t} \\ y_{AX,t} \\ y_{AY,t} \\ y_{AZ,t} \\ y_{BX,t} \\ y_{BY,t} \\ y_{BZ,t} \\ y_{CX,t} \\ y_{CY,t} \\ y_{CZ,t} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_{AX,t} \\ y_{AY,t} \\ y_{AZ,t} \\ y_{BX,t} \\ y_{BY,t} \\ y_{BZ,t} \\ y_{CX,t} \\ y_{CY,t} \\ y_{CZ,t} \end{pmatrix}$$

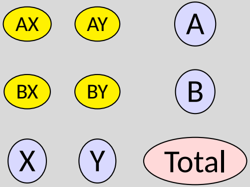
Hierarchical time series



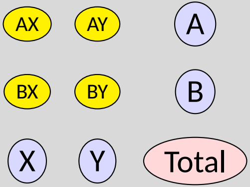
$$\boldsymbol{y}_t = \begin{pmatrix} y_t \\ y_{A,t} \\ y_{B,t} \\ y_{C,t} \\ y_{AX,t} \\ y_{AY,t} \\ y_{AZ,t} \\ y_{BX,t} \\ y_{BY,t} \\ y_{BZ,t} \\ y_{CX,t} \\ y_{CY,t} \\ y_{CZ,t} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_{AX,t} \\ y_{AY,t} \\ y_{AZ,t} \\ y_{BX,t} \\ y_{BY,t} \\ y_{BZ,t} \\ y_{CX,t} \\ y_{CY,t} \\ y_{CZ,t} \end{pmatrix}$$

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

Grouped data

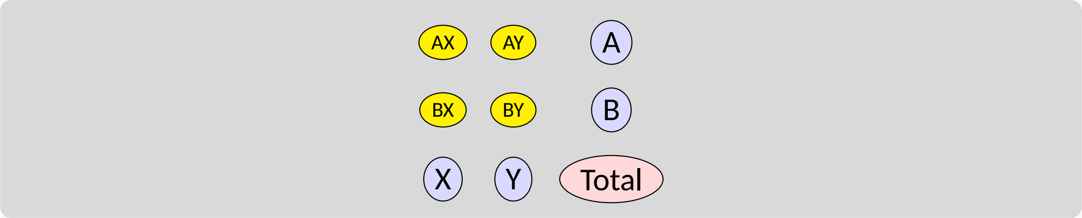


Grouped data



$$\boldsymbol{y}_t = \begin{pmatrix} y_t \\ y_{A,t} \\ y_{B,t} \\ y_{X,t} \\ y_{Y,t} \\ y_{AX,t} \\ y_{AY,t} \\ y_{BX,t} \\ y_{BY,t} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_{AX,t} \\ y_{AY,t} \\ y_{BX,t} \\ y_{BY,t} \end{pmatrix}$$

Grouped data



$$\boldsymbol{y}_t = \begin{pmatrix} y_t \\ y_{A,t} \\ y_{B,t} \\ y_{X,t} \\ y_{Y,t} \\ y_{AX,t} \\ y_{AY,t} \\ y_{BX,t} \\ y_{BY,t} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_{AX,t} \\ y_{AY,t} \\ y_{BX,t} \\ y_{BY,t} \end{pmatrix}$$

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

Definitions

Coherent subspace

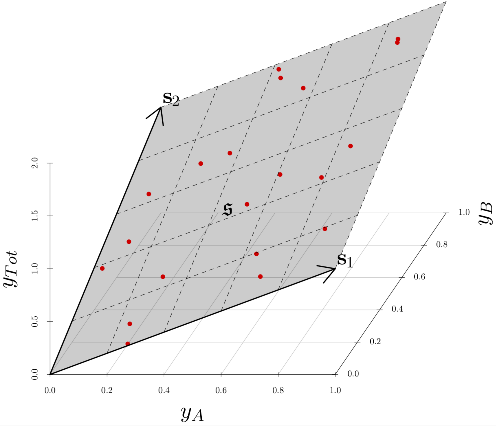
m-dimensional linear subspace $\mathfrak{s} \subset \mathbb{R}^n$ for which linear constraints hold for all $\boldsymbol{y} \in \mathfrak{s}$.

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



$$y_{Tot} = y_A + y_B$$

Definitions

Coherent subspace

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

Hierarchical time series

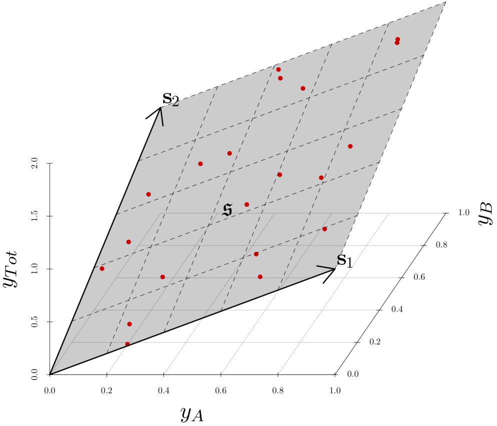
An n-dimensional multivariate time series such that $\boldsymbol{y}_t \in \mathfrak{s} \quad \forall t$.

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.



$y_{Tot} = y_A + y_B$

Definitions

Coherent subspace

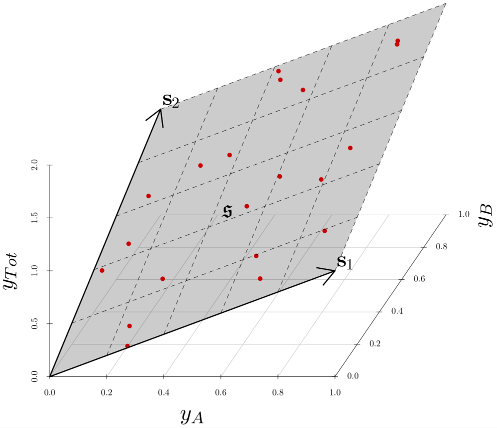
m-dimensional linear subspace $\mathfrak{s} \subset \mathbb{R}^n$ for which linear constraints hold for all $\boldsymbol{y} \in \mathfrak{s}$.

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, $\psi : \mathbb{R}^n \rightarrow \mathfrak{s}$.

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

Linear reconciliation

If ψ is a linear function, then $\tilde{\boldsymbol{y}}_{t+h|t} = \boldsymbol{S}\boldsymbol{G}\hat{\boldsymbol{y}}_{t+h|t}$

- $\boldsymbol{G}$ combines base forecasts $\hat{\boldsymbol{y}}_{T+h|T}$ to get bottom-level forecasts.
- $\boldsymbol{S}$ creates linear combinations.

Linear reconciliation

If ψ is a linear function, then $\tilde{\boldsymbol{y}}_{t+h|t} = \boldsymbol{S}\boldsymbol{G}\hat{\boldsymbol{y}}_{t+h|t}$

- $\boldsymbol{G}$ combines base forecasts $\hat{\boldsymbol{y}}_{T+h|T}$ to get bottom-level forecasts.
- $\boldsymbol{S}$ creates linear combinations.

Mean

$$\mathrm{E}[\tilde{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1,\dots,\boldsymbol{y}_n] = \mathrm{E}[\boldsymbol{y}_{T+h|T} \mid \boldsymbol{y}_1,\dots,\boldsymbol{y}_n]$$

provided $\boldsymbol{S}\boldsymbol{G}\boldsymbol{S}' = \boldsymbol{S}$ and

$$\mathrm{E}[\hat{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1,\dots,\boldsymbol{y}_n] = \mathrm{E}[\boldsymbol{y}_{T+h|T} \mid \boldsymbol{y}_1,\dots,\boldsymbol{y}_n]$$

i.e., reconciled forecasts are unbiased if base forecasts are unbiased and $\boldsymbol{S}\boldsymbol{G}$ is a projection.

Linear reconciliation

If ψ is a linear function, then $\tilde{\boldsymbol{y}}_{t+h|t} = \boldsymbol{S}\boldsymbol{G}\hat{\boldsymbol{y}}_{t+h|t}$

- $\boldsymbol{G}$ combines base forecasts $\hat{\boldsymbol{y}}_{T+h|T}$ to get bottom-level forecasts.
- $\boldsymbol{S}$ creates linear combinations.

Mean

$$\mathrm{E}[\tilde{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n] = \mathrm{E}[\boldsymbol{y}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n]$$

provided $\boldsymbol{S}\boldsymbol{G}\boldsymbol{S}' = \boldsymbol{S}$ and

$$\mathrm{E}[\hat{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n] = \mathrm{E}[\boldsymbol{y}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n]$$

i.e., reconciled forecasts are unbiased if base forecasts are unbiased and $\boldsymbol{S}\boldsymbol{G}$ is a projection.

Variance

$$\begin{aligned}\boldsymbol{V}_h &= \mathrm{Var}[\boldsymbol{y}_{T+h} - \tilde{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n] \\ &= \boldsymbol{S}\boldsymbol{G}\boldsymbol{W}_h\boldsymbol{G}'\boldsymbol{S}'\end{aligned}$$

where

$$\boldsymbol{W}_h = \mathrm{Var}[\boldsymbol{y}_{T+h} - \hat{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n]$$

Linear reconciliation

If ψ is a linear function, then $\tilde{\boldsymbol{y}}_{t+h|t} = \boldsymbol{S}\boldsymbol{G}\hat{\boldsymbol{y}}_{t+h|t}$

- $\boldsymbol{G}$ combines base forecasts $\hat{\boldsymbol{y}}_{T+h|T}$ to get bottom-level forecasts.
- $\boldsymbol{S}$ creates linear combinations.

Mean

$$\mathrm{E}[\tilde{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n] = \mathrm{E}[\boldsymbol{y}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n]$$

provided $\boldsymbol{S}\boldsymbol{G}\boldsymbol{S}' = \boldsymbol{S}$ and

$$\mathrm{E}[\hat{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n] = \mathrm{E}[\boldsymbol{y}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n]$$

i.e., reconciled forecasts are unbiased if base forecasts are unbiased and $\boldsymbol{S}\boldsymbol{G}$ is a projection.

Variance

$$\begin{aligned}\boldsymbol{V}_h &= \mathrm{Var}[\boldsymbol{y}_{T+h} - \tilde{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n] \\ &= \boldsymbol{S}\boldsymbol{G}\boldsymbol{W}_h\boldsymbol{G}'\boldsymbol{S}'\end{aligned}$$

where

$$\boldsymbol{W}_h = \mathrm{Var}[\boldsymbol{y}_{T+h} - \hat{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n]$$

Minimum trace (MinT) reconciliation

If $\boldsymbol{S}\boldsymbol{G}$ is a projection, then the trace of $\boldsymbol{V}_h$ is minimized when

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

Linear projections

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

Reconciliation method	$\boldsymbol{G}$
OLS	$(\boldsymbol{S}'\boldsymbol{S})^{-1}\boldsymbol{S}'$
WLS(var)	$(\boldsymbol{S}'\boldsymbol{\Lambda}_v\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{\Lambda}_v$
WLS(struct)	$(\boldsymbol{S}'\boldsymbol{\Lambda}_s\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{\Lambda}_s$
MinT(sample)	$(\boldsymbol{S}'\hat{\boldsymbol{W}}_{\text{sam}}^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\hat{\boldsymbol{W}}_{\text{sam}}^{-1}$
MinT(shrink)	$(\boldsymbol{S}'\hat{\boldsymbol{W}}_{\text{shr}}^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\hat{\boldsymbol{W}}_{\text{shr}}^{-1}$

These approximate MinT by assuming $\boldsymbol{W}_h = k_h\boldsymbol{W}_1$.

- $\boldsymbol{\Lambda}_v = \text{diag}(\boldsymbol{W}_1)^{-1}$
- $\boldsymbol{\Lambda}_s = \text{diag}(\boldsymbol{S}\boldsymbol{1})^{-1}$
- $\hat{\boldsymbol{W}}_{\text{sam}}$ 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.

Outline

Example: Australian tourism

```
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.
```

Example: Australian tourism

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

```
## # A tsibble: 26,400 x 5 [1M]
## # Key:       state, zone, region [110]
##       month state        zone         region       visitors
##       <mth> <chr>        <chr>        <chr>           <dbl>
##  1 1998 Jan <aggregated> <aggregated> <aggregated>   10376.
##  2 1998 Feb <aggregated> <aggregated> <aggregated>    5746.
##  3 1998 Mar <aggregated> <aggregated> <aggregated>    7129.
##  4 1998 Apr <aggregated> <aggregated> <aggregated>    7939.
##  5 1998 May <aggregated> <aggregated> <aggregated>    6552.
##  6 1998 Jun <aggregated> <aggregated> <aggregated>    5969.
##  7 1998 Jul <aggregated> <aggregated> <aggregated>    7041.
##  8 1998 Aug <aggregated> <aggregated> <aggregated>    6382.
##  9 1998 Sep <aggregated> <aggregated> <aggregated>    6907.
```

Example: Australian tourism

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

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

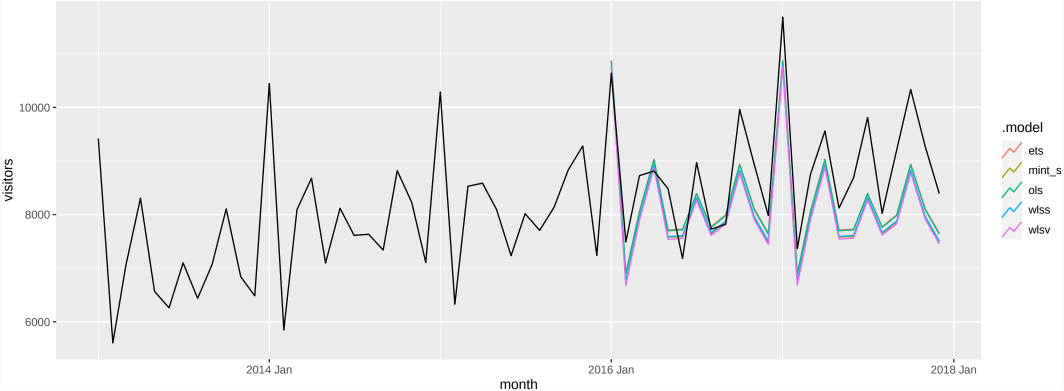
Example: Australian tourism

```
fc <- fit %>%
  reconcile(
    ols = min_trace(ets, method="ols"),
    wlsv = min_trace(ets, method="wls_var"),
    wlss = min_trace(ets, method="wls_struct"),
    #mint_c = min_trace(ets, method="mint_cov"),
    mint_s = min_trace(ets, method="mint_shrink"),
  ) %>%
  forecast(h = "2 years")
```

```
## # A fable: 5,280 x 7 [1M]
## # Key:     state, zone, region, .model [220]
##    state zone         region       .model    month        visitors .mean
##    <chr> <chr>        <chr>        <chr>     <mth>          <dist> <dbl>
##  1 NSW   <aggregated> <aggregated> ets    2016 Jan N(3679, 71136) 3679.
##  2 NSW   <aggregated> <aggregated> ets    2016 Feb N(2241, 27912) 2241.
```

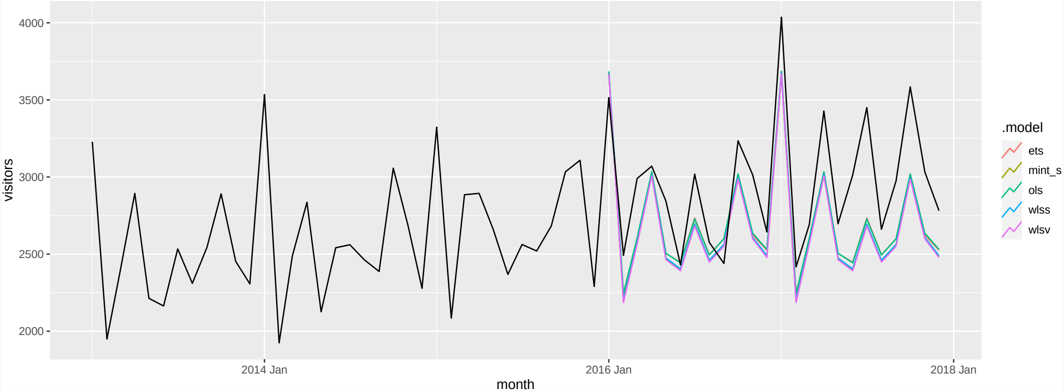
Example: Australian tourism

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



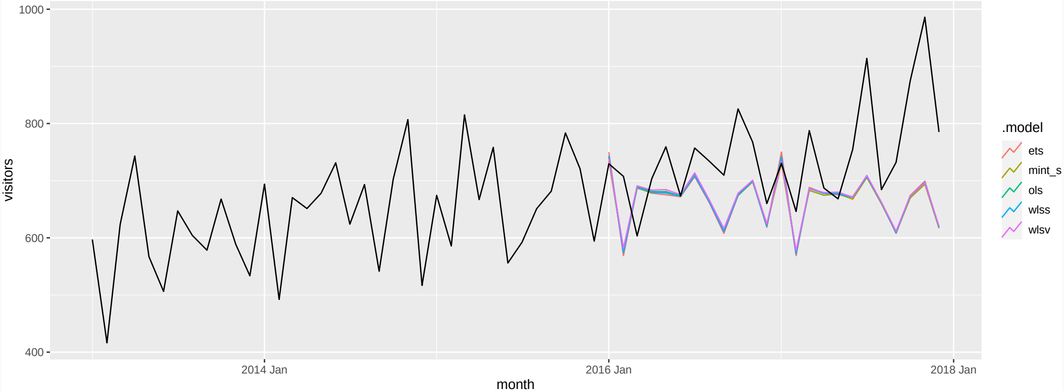
Example: Australian tourism

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



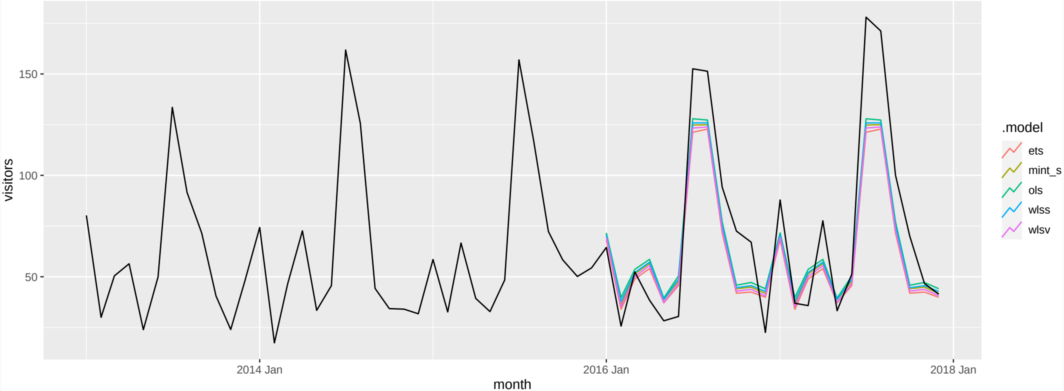
Example: Australian tourism

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



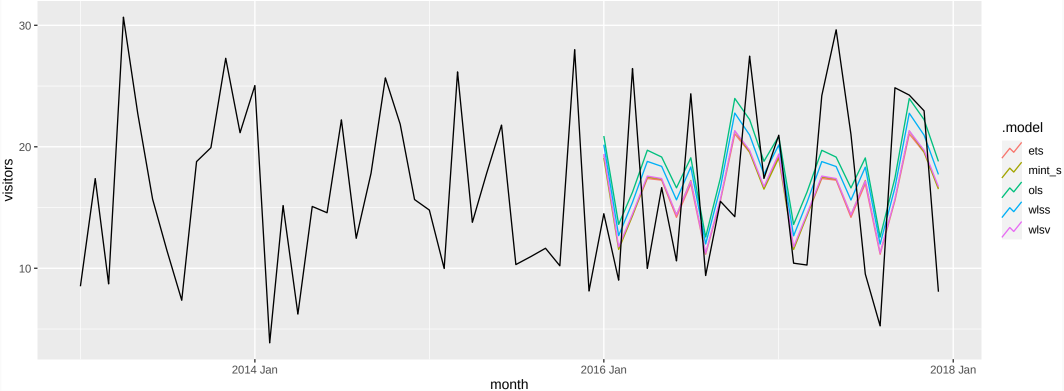
Example: Australian tourism

```
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)
```



Example: Australian tourism

```
fc %>%
  accuracy(data = tourism_agg,
           measures = list(rmsse = RMSSE))
```

```
## # A tibble: 550 x 6
##    .model state zone            region          .type rmsse
##    <chr>  <chr> <chr>           <chr>           <chr> <dbl>
##  1 ets    NSW   ACT             Canberra        Test  0.835
##  2 ets    NSW   ACT             <aggregated>    Test  0.835
##  3 ets    NSW   Metro NSW       Central Coast   Test  0.747
##  4 ets    NSW   Metro NSW       Sydney          Test  1.16
##  5 ets    NSW   Metro NSW       <aggregated>    Test  1.18
##  6 ets    NSW   North Coast NSW Hunter          Test  1.21
##  7 ets    NSW   North Coast NSW North Coast NSW Test  0.884
##  8 ets    NSW   North Coast NSW <aggregated>    Test  1.02
##  9 ets    NSW   North NSW       Blue Mountains  Test  1.02
```

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: 5 x 3
##   .model  mase rmsse
##   <chr>  <dbl> <dbl>
## 1 ols    0.930 0.926
## 2 wlss   0.949 0.948
## 3 mint_s 0.953 0.954
## 4 wlsv   0.964 0.965
## 5 ets    0.968 0.968
```

Example: Australian tourism

```
## # A tibble: 20 x 4
## # Groups:   .model [5]
##    .model level     mase rmsse
##    <chr>  <fct>    <dbl> <dbl>
##  1 ets    National 1.44  1.27
##  2 ols    National 1.46  1.29
##  3 wlss   National 1.61  1.43
##  4 mint_s National 1.64  1.45
##  5 wlsv   National 1.69  1.49
##  6 ols    State    1.07  1.08
##  7 ets    State    1.10  1.11
##  8 wlss   State    1.13  1.14
##  9 mint_s State    1.15  1.15
## 10 wlsv   State    1.18  1.17
## 11 ols    Zone     0.954 0.948
## 12 wlss   Zone     0.987 0.980
## 13 mint_s Zone     0.995 0.988
## 14 ets    Zone     1.01  0.999
## 15 wlsv   Zone     1.01  1.00
## 16 ols    Region   0.901 0.895
## 17 wlss   Region   0.910 0.907
## 18 mint_s Region   0.911 0.911
## 19 wlsv   Region   0.917 0.919
## 20 ets    Region   0.935 0.938
```

- Overall, every reconciliation method is better than the base ETS forecasts.
- OLS is best for all levels except national.
- Improvements due to reconciliation are greater at lower levels.

Outline

Probabilistic forecast reconciliation

Key papers

- Ben Taieb, Taylor, Hyndman (*ICML*, 2017)
- Jeon, Panagiotelis, Petropoulos (*EJOR*, 2019)
- Ben Taieb, Taylor, Hyndman (*JASA*, 2020)
- Panagiotelis, Gamakumara, Athanasopoulos, Hyndman (2020).

  `robjhyndman.com/publications/coherentprob/`

Probabilistic forecast reconciliation

Key papers

- Ben Taieb, Taylor, Hyndman (*ICML*, 2017)
- Jeon, Panagiotelis, Petropoulos (*EJOR*, 2019)
- Ben Taieb, Taylor, Hyndman (*JASA*, 2020)
- Panagiotelis, Gamakumara, Athanasopoulos, Hyndman (2020). `robjhyndman.com/publications/coherentprob/`

- The reconciled density must lie on the coherent subspace.
- The univariate density at each node is a convolution of the densities of its children.

Construction of reconciled distributions

Reconciled density of bottom-level

Density of bottom-level series under reconciled distribution is

$$\tilde{f}_{\boldsymbol{b}}(\boldsymbol{b}) = |\boldsymbol{G}^*| \int \hat{f}(\boldsymbol{G}^- \boldsymbol{b} + \boldsymbol{G}_\perp \boldsymbol{a}) d\boldsymbol{a}$$

- $\hat{f}$ is density of incoherent base probabilistic forecast
- $\boldsymbol{G}^-$ is $n \times m$ generalised inverse of $\boldsymbol{G}$ st $\boldsymbol{G}\boldsymbol{G}^- = \boldsymbol{I}$
- $\boldsymbol{G}_\perp$ is $n \times (n-m)$ orthogonal complement to $\boldsymbol{G}$ st $\boldsymbol{G}\boldsymbol{G}_\perp = \boldsymbol{0}$
- $\boldsymbol{G}^* = \left(\boldsymbol{G}^- \vdots \boldsymbol{G}_\perp\right)$, and $\boldsymbol{b}$ and $\boldsymbol{a}$ are obtained via

  the change of variables $\boldsymbol{y} = \boldsymbol{G}^* \begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{a} \end{pmatrix}$

Construction of reconciled distributions

Reconciled density of full hierarchy

Density of full hierarchy under reconciled distribution is

$$\tilde{f}_{\boldsymbol{y}}(\boldsymbol{y}) = |\boldsymbol{S}^*|\tilde{f}_{\boldsymbol{b}}(\boldsymbol{S}^-\boldsymbol{y})\mathbb{1}\{\boldsymbol{y} \in \mathfrak{s}\}$$

- $\boldsymbol{S}^* = \begin{pmatrix}\boldsymbol{S}^{-\prime} & \boldsymbol{S}_\perp\end{pmatrix}'$
- $\boldsymbol{S}^-$ is $m \times n$ generalised inverse of $\boldsymbol{S}$ such that $\boldsymbol{S}^-\boldsymbol{S} = \boldsymbol{I}$,
- $\boldsymbol{S}_\perp$ is $n \times (n-m)$ orthogonal complement to $\boldsymbol{S}$ such that $\boldsymbol{S}_\perp'\boldsymbol{S} = \boldsymbol{0}$.

Gaussian reconciliation

If the incoherent base forecasts are $\mathrm{N}(\hat{\mu}, \hat{\Sigma})$, then the reconciled density is $\mathrm{N}(\boldsymbol{S}\boldsymbol{G}\hat{\mu}, \boldsymbol{S}\boldsymbol{G}\hat{\Sigma}\boldsymbol{G}'\boldsymbol{S}')$.

Bootstrap reconciliation

Reconciling sample paths from incoherent distributions works.

Evaluating probabilistic forecasts

Proper scoring rule

optimized when true forecast distribution is used.

Evaluating probabilistic forecasts

Proper scoring rule

optimized when true forecast distribution is used.

Scoring Rule	Coherent v Incoherent	Coherent v Coherent
Log Score	Not proper	• Ordering preserved if compared using bottom-level only
Energy Score	Proper	• Full hierarchy should be used. • Rankings may change otherwise.

Score optimal reconciliation

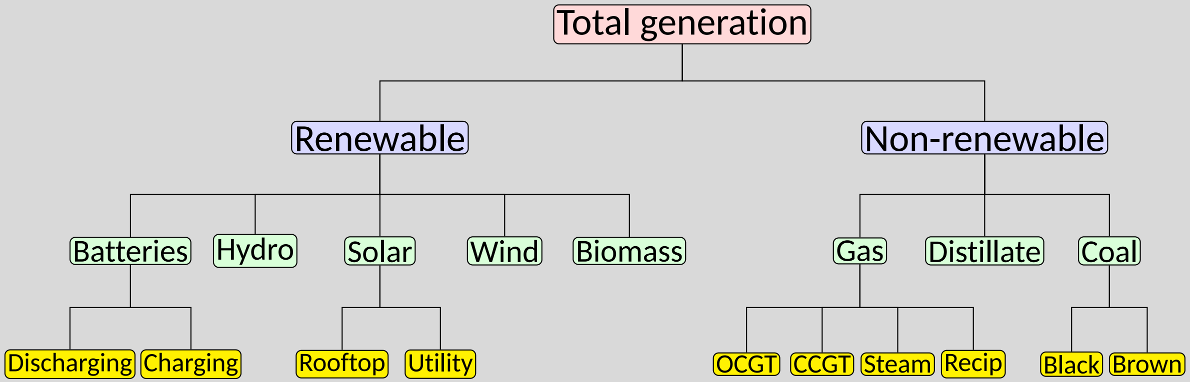
Algorithm proposed by Panagiotelis et al (2020) for optimizing $\boldsymbol{G}$ using stochastic gradient descent to optimize Energy Score.

1. Compute base forecasts over a test set.
2. Compute OLS reconciliation: $\boldsymbol{G} = (\boldsymbol{S}'\boldsymbol{S})^{-1}\boldsymbol{S}'$
3. Iteratively update $\boldsymbol{G}$ using SGD with Adam method and ES objective over a test set

Outline

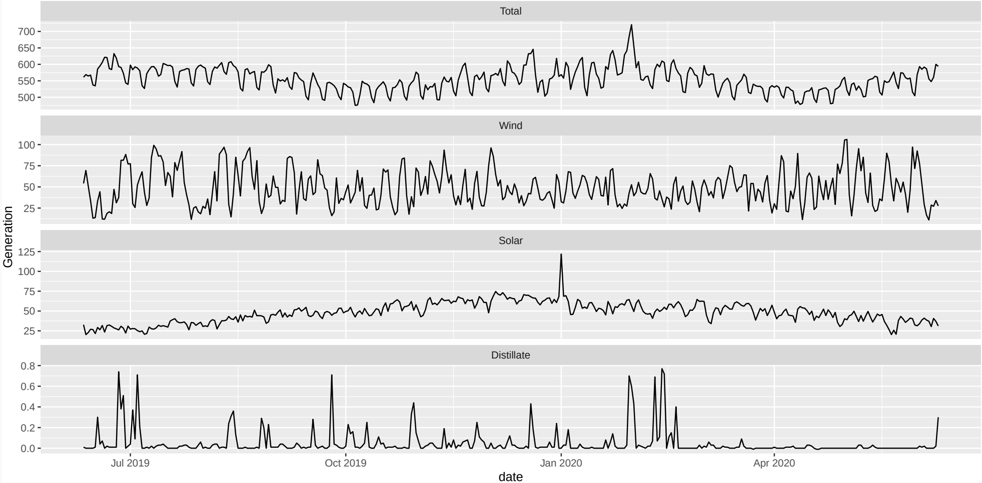
Example: Australian electricity generation

Daily time series from `opennem.org.au`



n = 23 series m = 15 bottom-level series

Example: Australian electricity generation

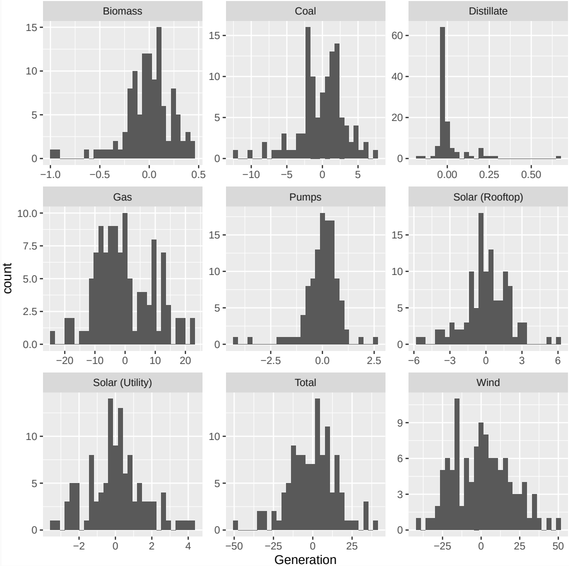


Example: Australian electricity generation

Forecast evaluation

- Rolling window of 140 days training data, and one-step-forecasts for 170 days test data.
- One-layer feed-forward neural network with up to 28 lags of target variable as inputs.
- Implemented using `NNETAR()` function in `fable` package.
- Model could be improved with temperature predictor.

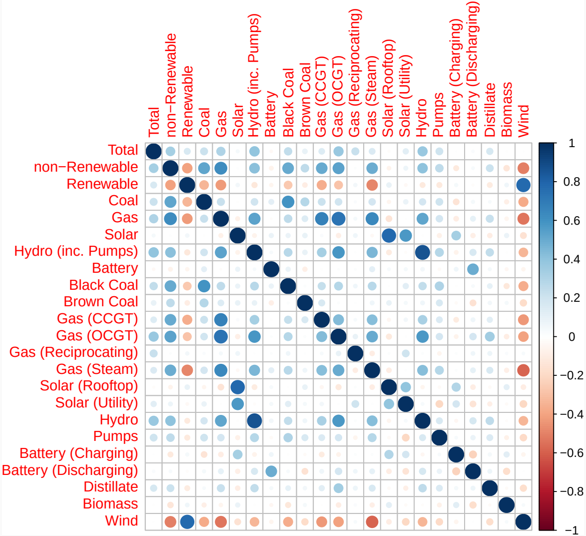
Example: Australian electricity generation



Histogram of residuals:
2 Oct 2019 – 21 Jan 2020

Clearly non-Gaussian

Example: Australian electricity generation



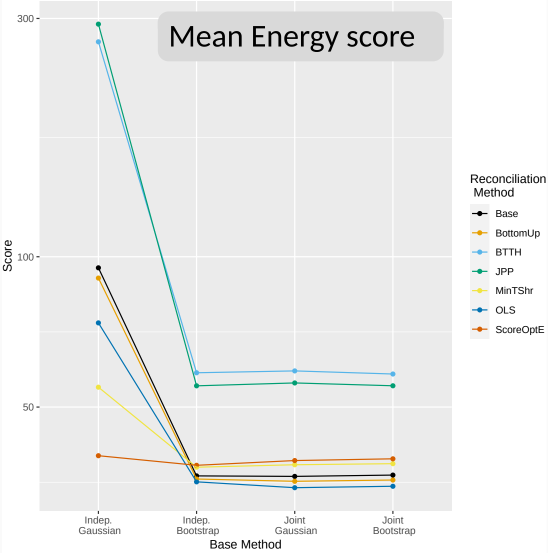
Correlations of residuals: 2 Oct 2019 – 21 Jan 2020

Blue = positive correlation.

Red = negative correlation.

Large = stronger correlations.

Example: Australian electricity generation



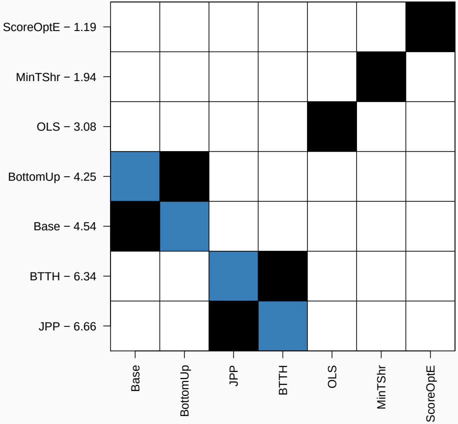
Base residual assumptions

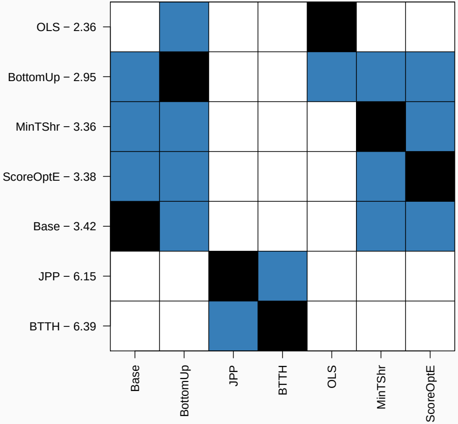
- Gaussian independent
- Gaussian dependent
- Non-Gaussian independent
- Non-Gaussian dependent

Reconciliation methods

- Base
- BottomUp
- BTTH: Ben Taieb, Taylor, Hyndman
- JPP: Jeon, Panagiotelis, Petropoulos
- OLS
- MinT(Shrink)
- Score Optimal Reconciliation

Example: Australian electricity generation





Nemenyi test for different scores

Base forecasts are independent and Gaussian.

Nemenyi test for different scores

Base forecasts are obtained by jointly bootstrapping residuals.

Outline

1. Hierarchical forecasting 20 years ago
2. Point forecast reconciliation
3. Example: Australian tourism
4. Probabilistic forecast reconciliation
5. Example: Australian electricity generation
6. Extensions

Reconciled linear regression forecasts

If the base forecasts are from a linear regression model, then we can produce coherent forecasts in one step:

$$\tilde{\boldsymbol{y}}_{T+h} = \boldsymbol{S}(\boldsymbol{S}'\Lambda_s\boldsymbol{S})^{-1}\boldsymbol{S}'\Lambda_s\boldsymbol{X}^*_{T+h}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y}$$

- $\boldsymbol{X}$ is matrix of predictors for training set
- $\boldsymbol{X}^*_{T+h}$ is vector of predictors for time $T+h$

Reconciled linear regression forecasts

If the base forecasts are from a linear regression model, then we can produce coherent forecasts in one step:

$$\tilde{\boldsymbol{y}}_{T+h} = \boldsymbol{S}(\boldsymbol{S}'\boldsymbol{\Lambda}_s\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{\Lambda}_s\boldsymbol{X}^*_{T+h}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y}$$

- $\boldsymbol{X}$ is matrix of predictors for training set
- $\boldsymbol{X}^*_{T+h}$ is vector of predictors for time $T+h$

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

- σ^2 is variance of base model residuals.

Reconciled linear regression forecasts

If the base forecasts are from a linear regression model, then we can produce coherent forecasts in one step:

$$\tilde{\boldsymbol{y}}_{T+h} = \boldsymbol{S}(\boldsymbol{S}'\boldsymbol{\Lambda}_s\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{\Lambda}_s\boldsymbol{X}^*_{T+h}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y}$$

- $\boldsymbol{X}$ is matrix of predictors for training set
- $\boldsymbol{X}^*_{T+h}$ is vector of predictors for time $T+h$

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

- σ^2 is variance of base model residuals.

Reference: Ashouri, Hyndman, and Shmueli (2019).
`robjhyndman.com/publications/lhf/`

Non-negative forecast reconciliation

Minimum trace (MinT) reconciliation

The trace of $\boldsymbol{V}_h$ is minimized when

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

subject to unbiasedness preservation ($\boldsymbol{SGS} = \boldsymbol{S}$).

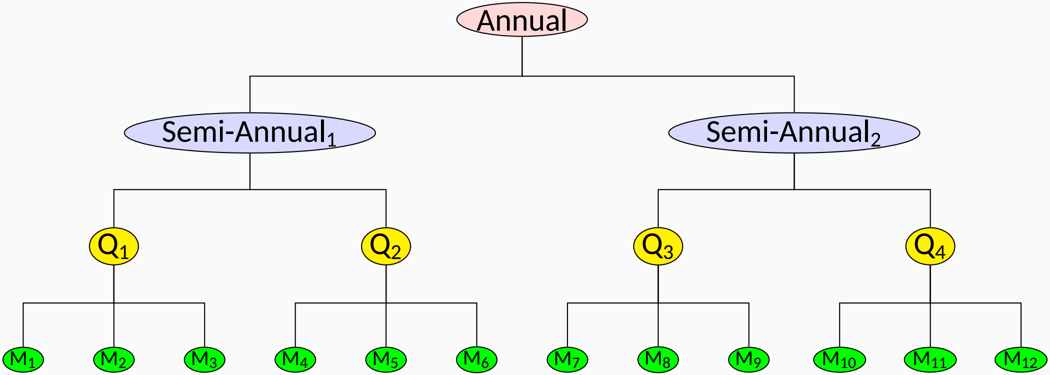
Wickramasuriya, Turlach and Hyndman (*S&C*, 2020) replace the unbiased constraint by a non-negative constraint:

$$\tilde{\boldsymbol{b}}_{T+h|T} \geq 0$$

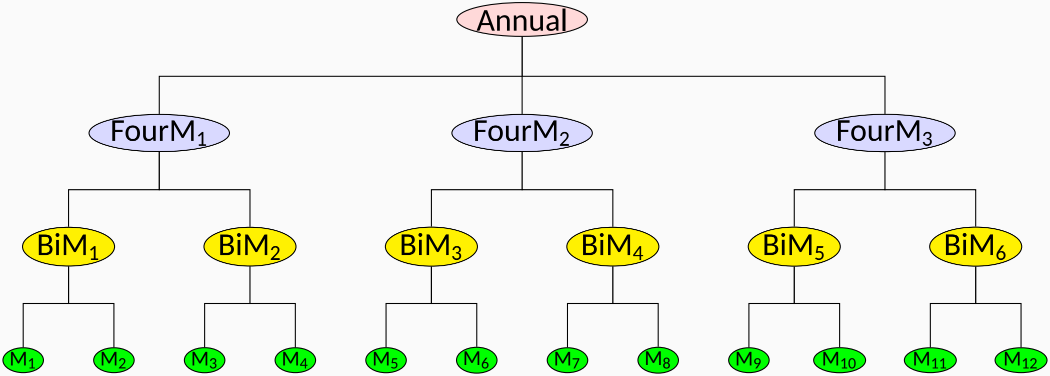
and show that it can be solved via quadratic programming:

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

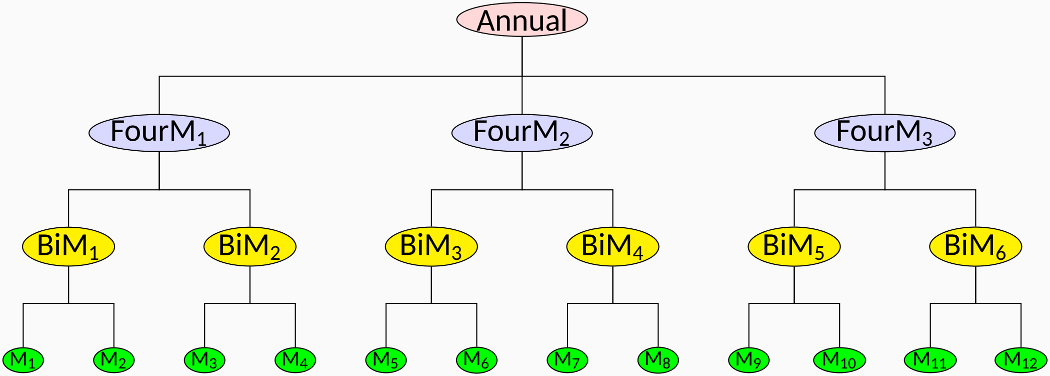
Temporal reconciliation



Temporal reconciliation



Temporal reconciliation



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

Temporal reconciliation

For a time series $y_1, \dots, y_T$, observed at frequency m, we generate aggregate series

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

- $k \in F(m) = \{\text{factors of } m\}$.
- A single unique hierarchy is only possible when there are no coprime pairs in $F(m)$.
- $M_k = m/k$ is seasonal period of aggregated series.
- Proposed by Athanasopoulos, Hyndman, Kourentzes, Petropoulos (*EJOR*, 2017)

Cross-temporal reconciliation

- Kourentzes, Athanasopoulos (*ATR*, 2019)
- Punia, Singh, Madaan (*C&IE*, 2020)
- Di Fonzo, Girolimetto (2020)

Bayesian forecast reconciliation

- Park, Nassar (*ICML*, 2014)
- Novak, McGarvie, and Garcia (2017)
- Eckert, Hyndman, Panagiotelis (*EJOR*, 2020)

ML and regularization

- Qiao, Huang (*ICIS*, 2018)
- Yang, Hu, Wang (*ICANN*, 2019)
- Abolghasemi, Hyndman, Tarr, Bergmeir (2019)
- Punia, Singh, and Madaan (*CIE*, 2020)
- Spiliotis, Abolghasemi, Hyndman, Petropoulos, Assimakopoulos (2020)

Linear combinations

- Shang, Hyndman (*JCGS*, 2017)
- Athanasopoulos, Gamakumara, Panagiotelis, Hyndman, Affan (Springer, 2020)

Thanks!

More information

- Slides and papers: **robjhyndman.com**
- Packages: **tidyverts.org**
- Forecasting textbook using fable package: **OTexts.com/fpp3**

Find me at ...

@robjhyndman

@robjhyndman

robjhyndman.com

rob.hyndman@monash.edu