The geometry of forecast reconciliation

Rob J Hyndman

28 August 2020

Outline

1 Hierarchical and grouped time series

2 Forecast reconciliation

3 Example: Australian tourism

4 Probabilistic forecast reconciliation

Outline

1 Hierarchical and grouped time series

2 Forecast reconciliation

3 Example: Australian tourism

4 Probabilistic forecast reconciliation

Australian Pharmaceutical Benefits Scheme

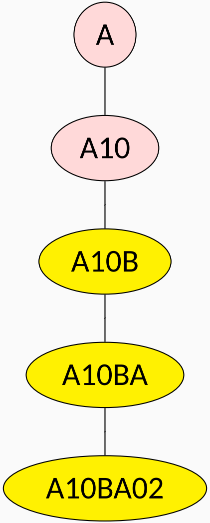
ATC drug classification

A	Alimentary tract and metabolism
B	Blood and blood forming organs
C	Cardiovascular system
D	Dermatologicals
G	Genito-urinary system and sex hormones
H	Systemic hormonal preparations, excluding sex hormones and insulins
J	Anti-infectives for systemic use
L	Antineoplastic and immunomodulating agents
M	Musculo-skeletal system
N	Nervous system
P	Antiparasitic products, insecticides and repellents
R	Respiratory system
S	Sensory organs

ATC drug classification

ATC1: 14 classes

ATC2: 84 classes



Alimentary tract and metabolism

Drugs used in diabetes

Blood glucose lowering drugs

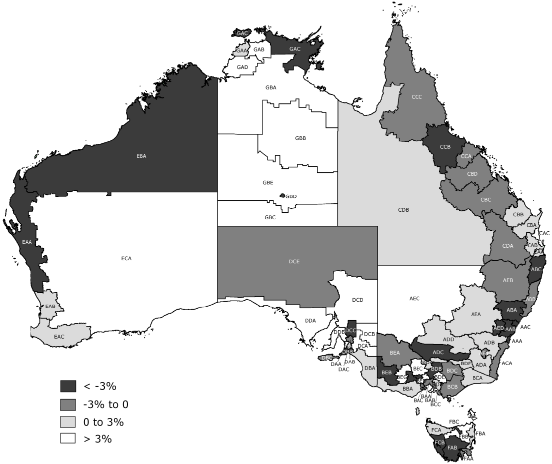
Biguanides

Metformin

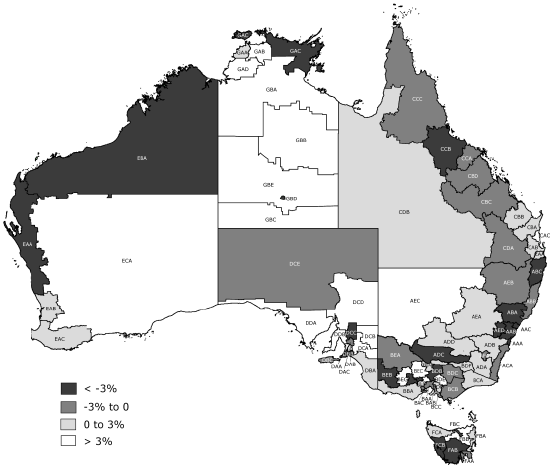
PBS sales

```
## # A tsibble: 16,688 x 5 [1M]
## # Key:       ATC1, ATC2 [84]
##       Month ATC1  ATC2  Scripts   Cost
##       <mth> <chr> <chr>   <dbl>  <dbl>
##  1 1991 Jul A     A01     22615  94351
##  2 1991 Aug A     A01     20443  87899
##  3 1991 Sep A     A01     21389  94949
##  4 1991 Oct A     A01     23746 107926
##  5 1991 Nov A     A01     23477 107032
##  6 1991 Dec A     A01     26316 123000
##  7 1992 Jan A     A01     22041 105996
##  8 1992 Feb A     A01     16393  61818
##  9 1992 Mar A     A01     17207  63010
## 10 1992 Apr A     A01     18847  69591
## # ... with 16,678 more rows
```

Australian tourism



Australian tourism



- Quarterly data on visitor night from 1998:Q1 – 2013:Q4
- From *National Visitor Survey*, annual interviews of 120,000 Australians aged 15+.
- Split by 7 states, 27 zones and 76 regions (a geographical hierarchy)
- Also split by purpose of travel
  - Holiday
  - Visiting friends and relatives (VFR)
  - Business
  - Other
- 304 bottom-level series

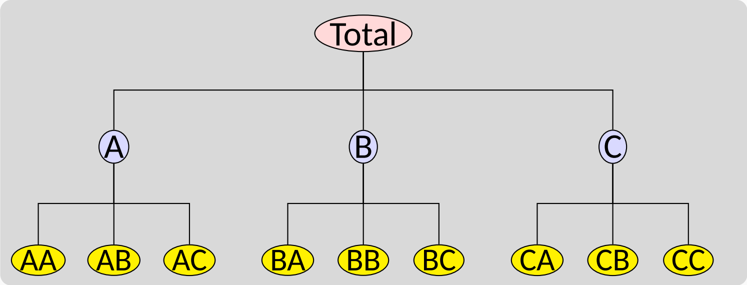
Australian tourism

```
tourism
```

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:       Region, State, Purpose [304]
##    Quarter Region   State           Purpose  Trips
##      <qtr> <chr>    <chr>           <chr>    <dbl>
##  1 1998 Q1 Adelaide South Australia Business  135.
##  2 1998 Q2 Adelaide South Australia Business  110.
##  3 1998 Q3 Adelaide South Australia Business  166.
##  4 1998 Q4 Adelaide South Australia Business  127.
##  5 1999 Q1 Adelaide South Australia Business  137.
##  6 1999 Q2 Adelaide South Australia Business  200.
##  7 1999 Q3 Adelaide South Australia Business  169.
##  8 1999 Q4 Adelaide South Australia Business  134.
##  9 2000 Q1 Adelaide South Australia Business  154.
```

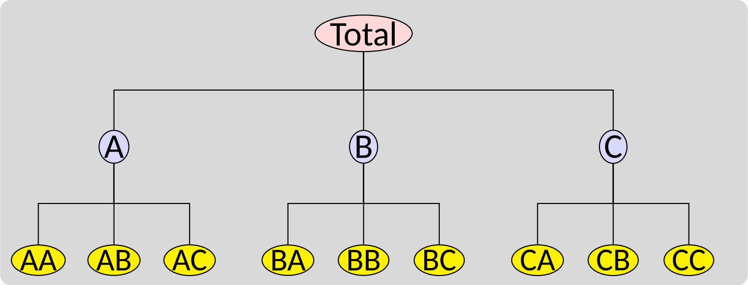
Hierarchical time series

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



Hierarchical time series

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

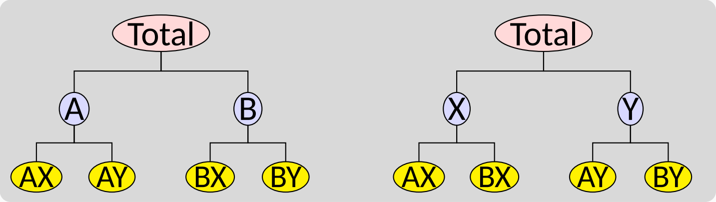


Examples

- PBS sales by ATC groups
- Tourism demand by states, zones, regions

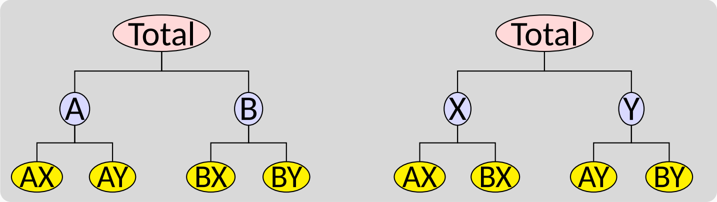
Grouped time series

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

Outline

1 Hierarchical and grouped time series

2 Forecast reconciliation

3 Example: Australian tourism

4 Probabilistic forecast reconciliation

The problem

How to produce **coherent** forecasts at all nodes?

The problem

How to produce **coherent** forecasts at all nodes?

Old approaches (pre 2009)

- Bottom-up forecasting
- Top-down forecasting
- Middle-out forecasting

The problem

How to produce **coherent** forecasts at all nodes?

Old approaches (pre 2009)

- Bottom-up forecasting
- Top-down forecasting
- Middle-out forecasting

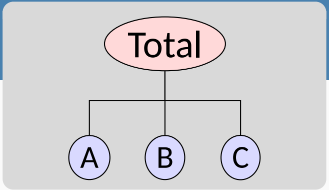
Forecast reconcilation approach

1. Forecast all series at all levels of aggregation using an automatic forecasting algorithm. (e.g., `ETS`, `ARIMA`, . . .)
2. Reconcile the resulting forecasts so they are coherent using least squares optimization (i.e., find closest reconciled forecasts to the original forecasts).

Key forecast reconciliation papers

- Hyndman, Ahmed, Athanasopoulos, Shang (2011 *CSDA*) Optimal combination forecasts for hierarchical time series.
- Athanasopoulos, Ahmed, Hyndman (2009 *IJF*) Hierarchical forecasts for Australian domestic tourism.
- 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.
- Panagiotelis, Gamakumara, Athanasopoulos, Hyndman (2020) Probabilistic forecast reconciliation: properties, evaluation and score optimisation.

Hierarchical and grouped time series



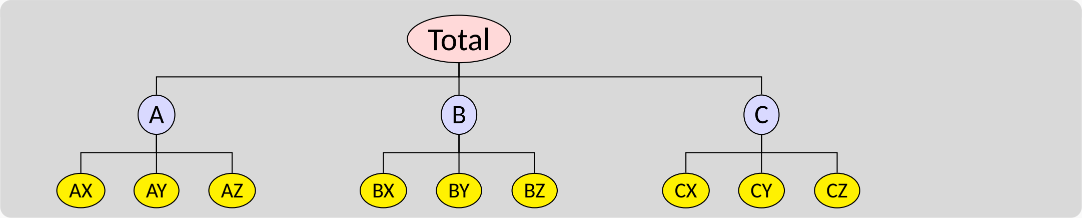
Every collection of time series with linear constraints can be written as

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

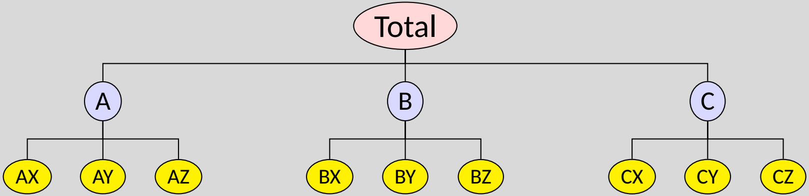
- $\boldsymbol{y}_t$ = vector of all series at time t
- y_t = aggregate of all series at time t.
- $y_{X,t}$ = value of series X at time t.
- $\boldsymbol{b}_t$ = vector of most disaggregated series at time t
- $\boldsymbol{S}$ = "summing matrix" containing the linear constraints.

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

Hierarchical time series

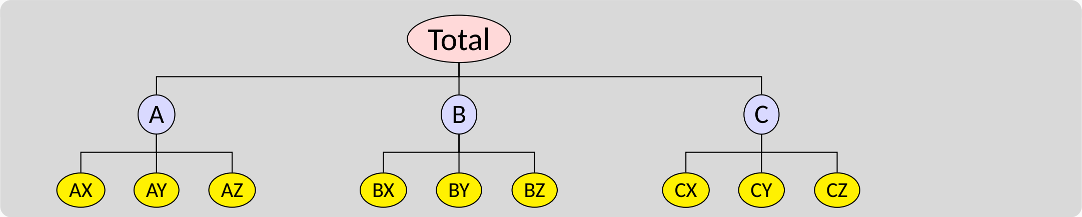


Hierarchical time series



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

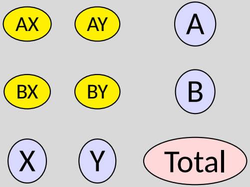
Hierarchical time series



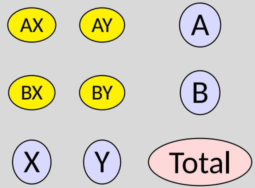
$$\boldsymbol{y}_t = \begin{pmatrix} y_t \\ y_{A,t} \\ y_{B,t} \\ y_{C,t} \\ y_{AX,t} \\ y_{AY,t} \\ y_{AZ,t} \\ y_{BX,t} \\ y_{BY,t} \\ y_{BZ,t} \\ y_{CX,t} \\ y_{CY,t} \\ y_{CZ,t} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1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

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

Grouped data

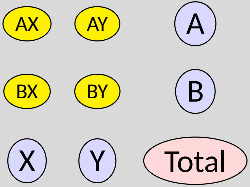


Grouped data



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

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

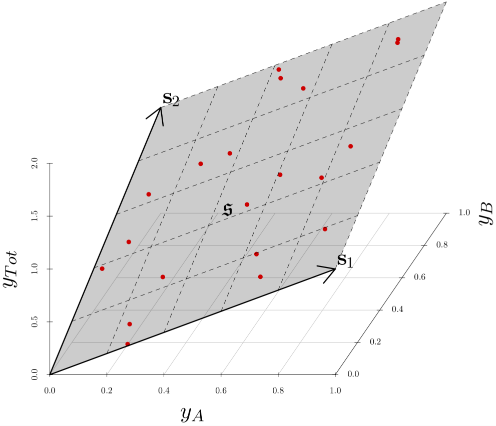
m-dimensional linear subspace $\mathfrak{s} \subset \mathbb{R}^n$ for which linear constraints hold for all $\boldsymbol{y} \in \mathfrak{s}$.

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

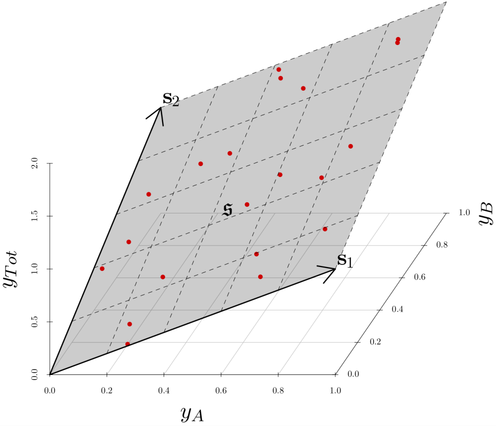
An n-dimensional multivariate time series such that $\boldsymbol{y}_t \in \mathfrak{s} \quad \forall t$.

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

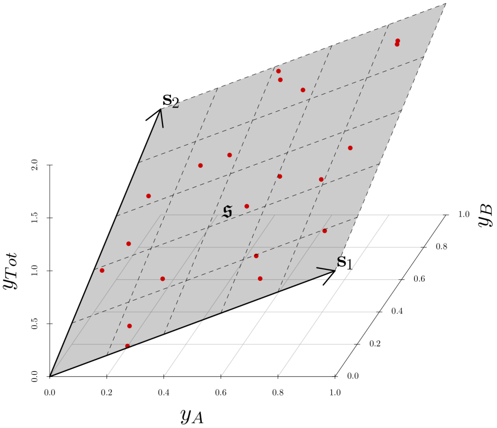
m-dimensional linear subspace $\mathfrak{s} \subset \mathbb{R}^n$ for which linear constraints hold for all $\boldsymbol{y} \in \mathfrak{s}$.

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 and $\boldsymbol{G}$ is some matrix, then

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

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

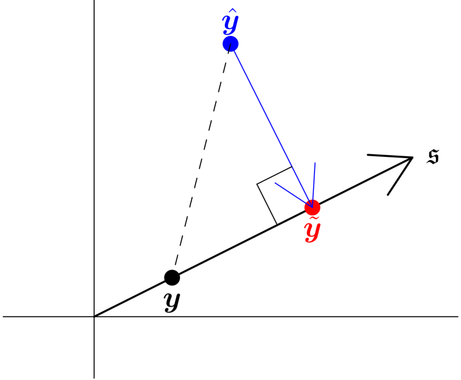
Linear reconciliation

If ψ is a linear function and $\boldsymbol{G}$ is some matrix, then

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

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

e.g., OLS reconciliation: $\boldsymbol{G} = (\boldsymbol{S}'\boldsymbol{S})^{-1}\boldsymbol{S}'$



Linear reconciliation

If ψ is a linear function and $\boldsymbol{G}$ is some matrix, then

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

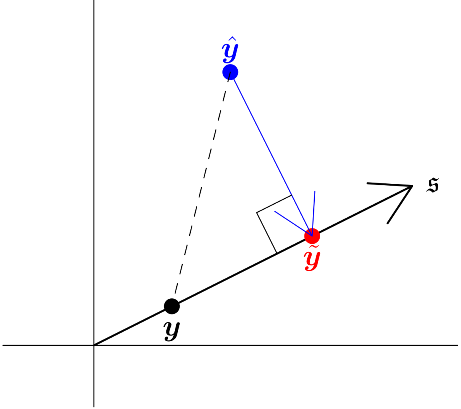
- $\boldsymbol{G}$ extracts and combines base forecasts $\hat{\boldsymbol{y}}_{T+h|T}$ to get bottom-level forecasts.
- $\boldsymbol{S}$ creates linear combinations.

e.g., OLS reconciliation: $\boldsymbol{G} = (\boldsymbol{S}'\boldsymbol{S})^{-1}\boldsymbol{S}'$

Projections

Suppose $\boldsymbol{S}\boldsymbol{G}$ is a projection onto $\mathfrak{s}$, then

- Coherent base forecasts are unchanged.
- Unbiased base forecasts remain unbiased.



- Orthogonal projections lead to smallest possible adjustments of base forecasts.

Linear reconciliation

If ψ is a linear function and $\boldsymbol{G}$ is some matrix, then

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

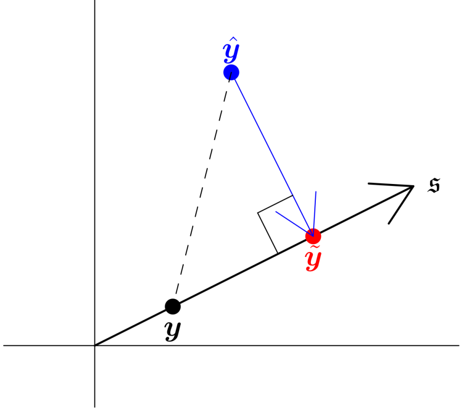
- $\boldsymbol{G}$ extracts and combines base forecasts $\hat{\boldsymbol{y}}_{T+h|T}$ to get bottom-level forecasts.
- $\boldsymbol{S}$ creates linear combinations.

e.g., OLS reconciliation: $\boldsymbol{G} = (\boldsymbol{S}'\boldsymbol{S})^{-1}\boldsymbol{S}'$

Distance reducing property

If $\boldsymbol{S}\boldsymbol{G}$ is an orthogonal projection onto $\mathfrak{s}$ then:

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



- Distance reduction holds for any realisation and any forecast.
- Other measures of forecast accuracy may be worse.
- Not necessarily the optimal reconciliation.

Oblique projections

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

Variance

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

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

Minimum trace (MinT) reconciliation

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

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

Oblique projections

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

Variance

$$\mathbf{V}_h = \text{Var}[\mathbf{y}_{T+h} - \tilde{\mathbf{y}}_{T+h|T} \mid \mathbf{y}_1, \dots, \mathbf{y}_n] = \mathbf{S}\mathbf{G}\mathbf{W}_h\mathbf{G}'\mathbf{S}'$$

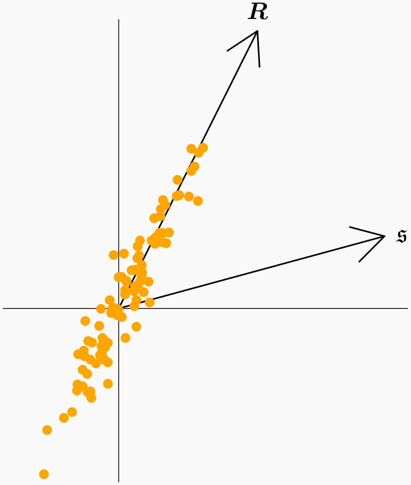
where $\mathbf{W}_h = \text{Var}[\mathbf{y}_{T+h} - \hat{\mathbf{y}}_{T+h|T} \mid \mathbf{y}_1, \dots, \mathbf{y}_n]$.

Minimum trace (MinT) reconciliation

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

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

- R is the most likely direction of deviations from $\mathfrak{s}$.
- Orange: in-sample errors



Oblique projections

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

Variance

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

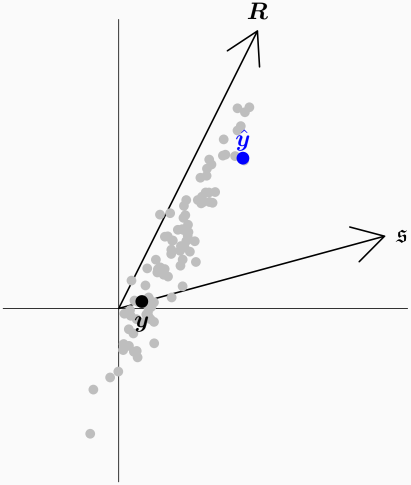
where $\boldsymbol{W}_h = \text{Var}[\boldsymbol{y}_{T+h} - \hat{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n]$.

Minimum trace (MinT) reconciliation

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

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

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



Oblique projections

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

Variance

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

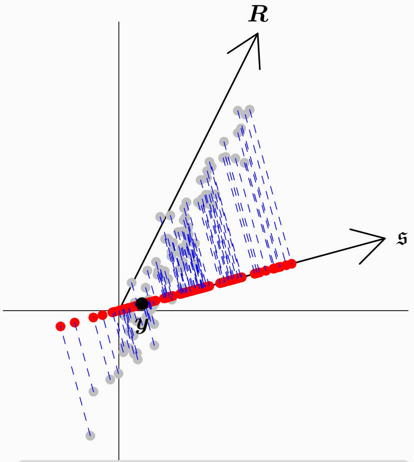
where $\boldsymbol{W}_h = \text{Var}[\boldsymbol{y}_{T+h} - \hat{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n]$.

Minimum trace (MinT) reconciliation

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

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

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



Orthogonal projection

Oblique projections

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

Variance

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

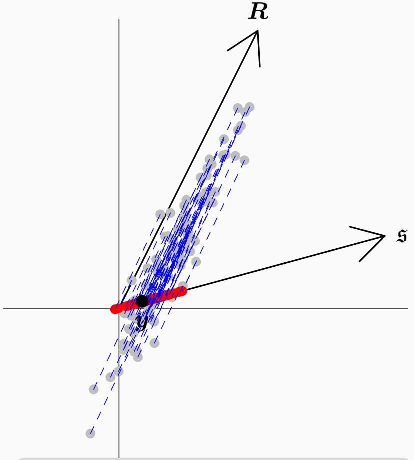
where $\boldsymbol{W}_h = \text{Var}[\boldsymbol{y}_{T+h} - \hat{\boldsymbol{y}}_{T+h|T} \mid \boldsymbol{y}_1, \dots, \boldsymbol{y}_n]$.

Minimum trace (MinT) reconciliation

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

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

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



Oblique projection

Linear projections

Reconciliation method	$\boldsymbol{G}$
OLS	$(\boldsymbol{S}'\boldsymbol{S})^{-1}\boldsymbol{S}'$
WLS	$(\boldsymbol{S}'\boldsymbol{\Lambda}\boldsymbol{S})^{-1}\boldsymbol{S}'\boldsymbol{\Lambda}$
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 all assume that $\boldsymbol{W}_h = k_h\boldsymbol{W}_1$ to simplify computations.

- $\boldsymbol{\Lambda}$ is diagonal matrix
- $\hat{\boldsymbol{W}}_{\text{sam}}$ 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 $\tau = \frac{\sum_{i\neq j}\hat{\text{Var}}(\hat{\sigma}_{ij})}{\sum_{i\neq j}\hat{\sigma}_{ij}^2}$ and σ_{ij} denotes the (i,j)th element of $\hat{\boldsymbol{W}}_{\text{sam}}$.

To add from geometry paper

- Tourism example from geometry paper (not bias section)

Outline

1 Hierarchical and grouped time series

2 Forecast reconciliation

3 Example: Australian tourism

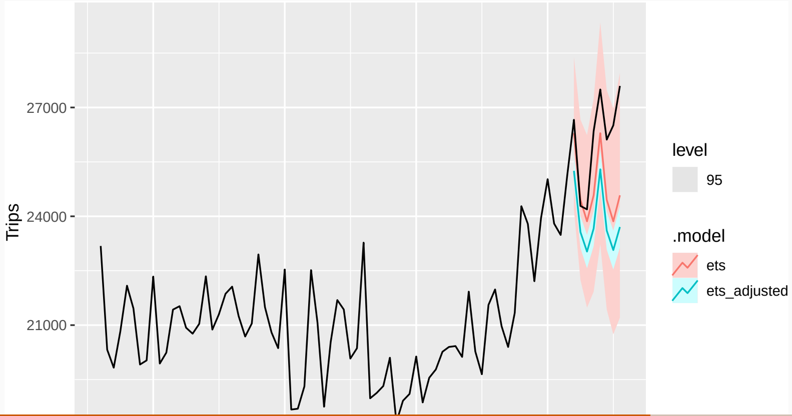
4 Probabilistic forecast reconciliation

Example: Australian tourism

```
tourism_agg <- tourism %>%
  aggregate_key(Purpose * (State / Region),
    Trips = sum(Trips)
  )
fc <- tourism_agg %>%
  filter(Quarter <= yearquarter("2015 Q4")) %>%
  model(ets = ETS(Trips)) %>%
  reconcile(ets_adjusted = min_trace(ets)) %>%
  forecast(h = "2 years")
```

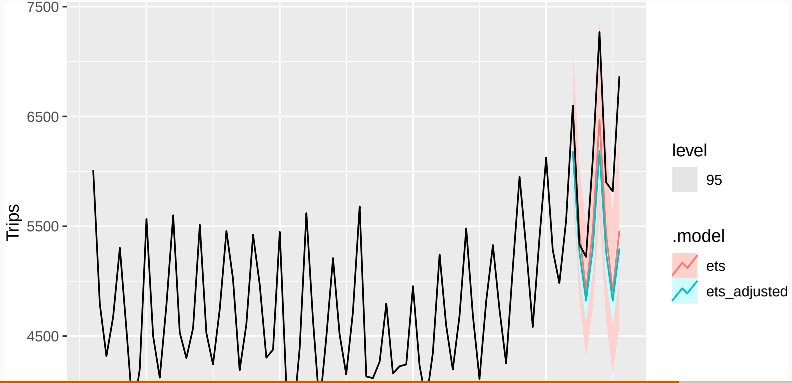
Example: Australian tourism

```r
fc %>%
  filter(is_aggregated(Purpose) & is_aggregated(State)) %>%
  autoplot(tourism_agg, level = 95)
```



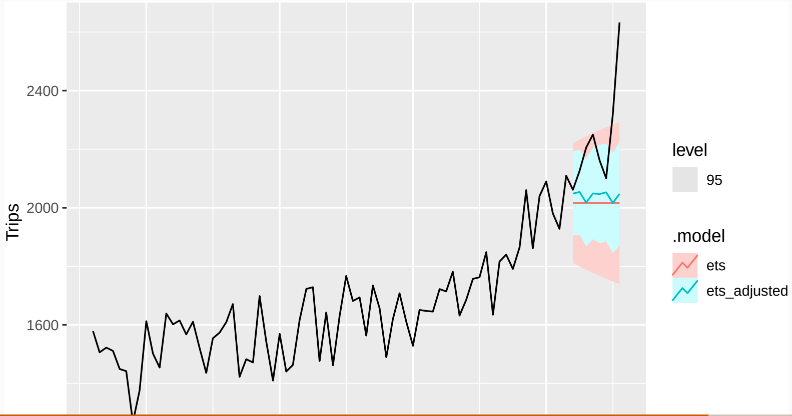
Example: Australian tourism

```
fc %>%
  filter(is_aggregated(Purpose) & State == "Victoria" &
    is_aggregated(Region)) %>%
  autoplot(tourism_agg, level = 95)
```



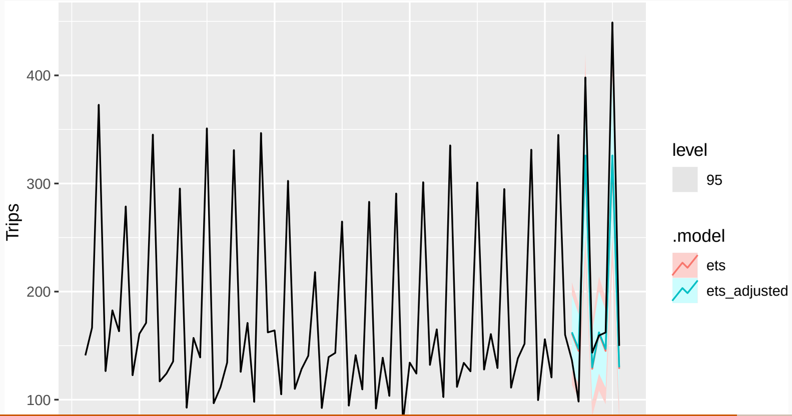
Example: Australian tourism

```
fc %>%
  filter(is_aggregated(Purpose) & Region == "Melbourne") %>%
  autoplot(tourism_agg, level = 95)
```



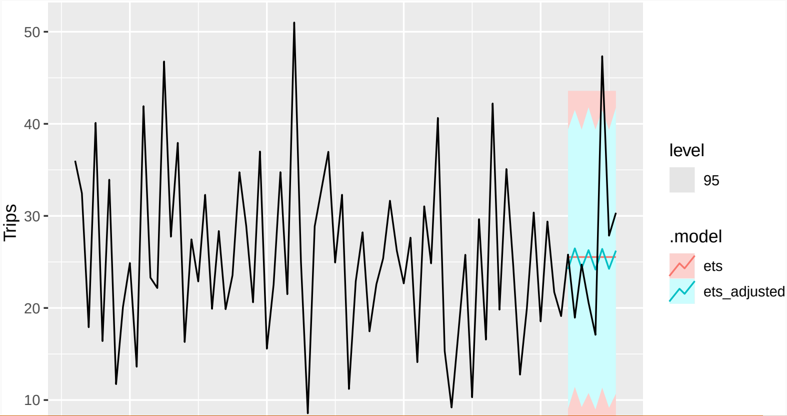
Example: Australian tourism

```r
fc %>%
  filter(is_aggregated(Purpose) & Region == "Snowy Mountains") %>%
  autoplot(tourism_agg, level = 95)
```



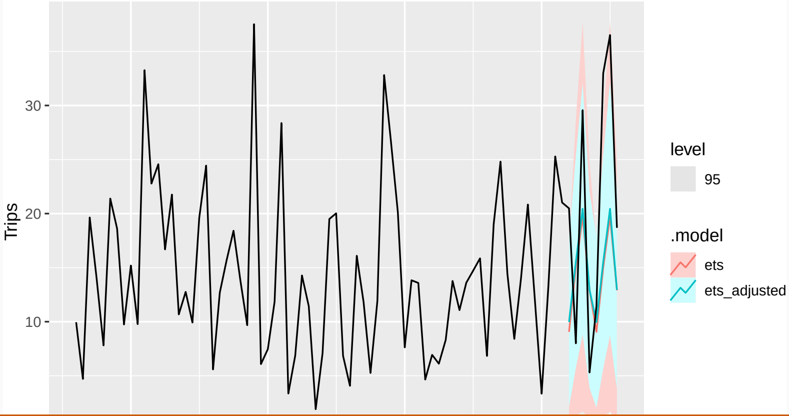
Example: Australian tourism

```
fc %>%
  filter(Purpose == "Holiday" & Region == "Barossa") %>%
  autoplot(tourism_agg, level = 95)
```



Example: Australian tourism

```
fc %>%
  filter(is_aggregated(Purpose) & Region == "MacDonnell") %>%
  autoplot(tourism_agg, level = 95)
```



Example: Australian tourism

```
fc <- tourism_agg %>%
  filter(Quarter <= yearquarter("2015 Q4")) %>%
  model(
    ets = ETS(Trips),
    arima = ARIMA(Trips)
  ) %>%
  mutate(
    comb = (ets + arima) / 2
  ) %>%
  reconcile(
    ets_adj = min_trace(ets),
    arima_adj = min_trace(arima),
    comb_adj = min_trace(comb)
  ) %>%
```

Forecast evaluation

```r
fc %>% accuracy(tourism_agg)
```

```
## # A tibble: 2,550 x 12
##    .model Purpose  State       Region      .type    ME
##    <chr>  <chr>    <chr>       <chr>       <chr> <dbl>
##  1 arima  Business <aggregat~ <aggregat~ Test   685.
##  2 arima  Business South Aus~ <aggregat~ Test    49.9
##  3 arima  Business Northern ~ <aggregat~ Test    22.2
##  4 arima  Business Western A~ <aggregat~ Test  -138.
##  5 arima  Business Victoria ~ <aggregat~ Test   232.
##  6 arima  Business New South~ <aggregat~ Test   153.
##  7 arima  Business Queenslan~ <aggregat~ Test    81.8
##  8 arima  Business ACT       ~ <aggregat~ Test    35.9
##  9 arima  Business Tasmania ~ <aggregat~ Test    28.8
## 10 arima  Business South Aus~ Adelaide   Test    20.8
## # ... with 2,540 more rows, and 6 more variables:
```

Forecast evaluation

```
fc %>%
  accuracy(tourism_agg) %>%
  group_by(.model) %>%
  summarise(MASE = mean(MASE)) %>%
  arrange(MASE)
```

```
## # A tibble: 6 x 2
##   .model      MASE
##   <chr>      <dbl>
## 1 ets_adj     1.02
## 2 comb_adj    1.02
## 3 ets         1.04
## 4 comb        1.04
## 5 arima_adj   1.07
```

Creating aggregates

```r
PBS %>%
  aggregate_key(ATC1 / ATC2, Scripts = sum(Scripts)) %>%
  filter(Month == yearmonth("1991 Jul")) %>%
  print(n = 18)
```

```
## # A tsibble: 98 x 4 [1M]
## # Key:       ATC1, ATC2 [98]
##       Month ATC1         ATC2         Scripts
##       <mth> <chr>        <chr>          <dbl>
##  1 1991 Jul <aggregated> <aggregated> 8090395
##  2 1991 Jul A            <aggregated>  799025
##  3 1991 Jul B            <aggregated>  109227
##  4 1991 Jul C            <aggregated> 1794995
##  5 1991 Jul D            <aggregated>  299779
##  6 1991 Jul G            <aggregated>  300931
##  7 1991 Jul H            <aggregated>  112114
##  8 1991 Jul J            <aggregated> 1151681
##  9 1991 Jul L            <aggregated>   24580
## 10 1991 Jul M            <aggregated>  562956
## 11 1991 Jul N            <aggregated> 1546023
## 12 1991 Jul P            <aggregated>   47661
## 13 1991 Jul R            <aggregated>  859273
## 14 1991 Jul S            <aggregated>  391639
## 15 1991 Jul V            <aggregated>   38705
## 16 1991 Jul Z            <aggregated>   51806
```

Creating aggregates

```r
tourism %>%
  aggregate_key(Purpose * (State / Region), Trips = sum(Trips)) %>%
  filter(Quarter == yearquarter("1998 Q1")) %>%
  print(n = 15)
```

```
## # A tsibble: 425 x 5 [1Q]
## # Key:       Purpose, State, Region [425]
##    Quarter Purpose      State              Region        Trips
##      <qtr> <chr>        <chr>              <chr>         <dbl>
##  1 1998 Q1 <aggregated> <aggregated>       <aggregated> 23182.
##  2 1998 Q1 Business     <aggregated>       <aggregated>  3599.
##  3 1998 Q1 Holiday      <aggregated>       <aggregated> 11806.
##  4 1998 Q1 Other        <aggregated>       <aggregated>   680.
##  5 1998 Q1 Visiting     <aggregated>       <aggregated>  7098.
##  6 1998 Q1 <aggregated> ACT               ~ <aggregated>   551.
##  7 1998 Q1 <aggregated> New South Wales~ <aggregated>  8040.
##  8 1998 Q1 <aggregated> Northern Territ~ <aggregated>   181.
##  9 1998 Q1 <aggregated> Queensland      ~ <aggregated>  4041.
## 10 1998 Q1 <aggregated> South Australia~ <aggregated>  1735.
## 11 1998 Q1 <aggregated> Tasmania        ~ <aggregated>   982.
## 12 1998 Q1 <aggregated> Victoria        ~ <aggregated>  6010.
## 13 1998 Q1 <aggregated> Western Austral~ <aggregated>  1641.
```

Creating aggregates

- Similar to `summarise()` but using the key structure
- A grouped structure is specified using `grp1 * grp2`
- A nested structure is specified via `parent / child`.
- Groups and nesting can be mixed:

```
(country/region/city) * (brand/product)
```

- All possible aggregates are produced.
- These are useful when forecasting at different levels of aggregation.

Forecast reconciliation

```r
tourism %>%
  aggregate_key(Purpose * (State / Region), Trips = sum(Trips)) %>%
  model(ets = ETS(Trips)) %>%
  reconcile(ets_adjusted = min_trace(ets)) %>%
  forecast(h = 2)
```

```
## # A fable: 1,700 x 7 [1Q]
## # Key:     Purpose, State, Region, .model [850]
##    Purpose    State        Region       .model Quarter
##    <chr>      <chr>        <chr>        <chr>    <qtr>
##  1 Business   <aggregat~ <aggregat~ ets    2018 Q1
##  2 Business   <aggregat~ <aggregat~ ets    2018 Q2
##  3 Business   <aggregat~ <aggregat~ ets_a~ 2018 Q1
##  4 Business   <aggregat~ <aggregat~ ets_a~ 2018 Q2
##  5 Business   South Aus~ <aggregat~ ets    2018 Q1
##  6 Business   South Aus~ <aggregat~ ets    2018 Q2
##  7 Business   South Aus~ <aggregat~ ets_a~ 2018 Q1
##  8 Business   South Aus~ <aggregat~ ets_a~ 2018 Q2
```

Outline

1 Hierarchical and grouped time series

2 Forecast reconciliation

3 Example: Australian tourism

4 Probabilistic forecast reconciliation

Coherent probabilistic forecasts

Coherent probabilistic forecasts

Given the triple $(\mathbb{R}^m, \mathscr{F}_{\mathbb{R}^m}, \nu)$, a coherent probability triple $(\mathfrak{s}, \mathscr{F}_{\mathfrak{s}}, \breve{\nu})$ is such that

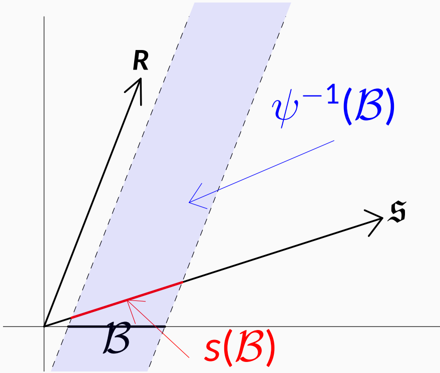
$$\breve{\nu}(s(\mathcal{B})) = \nu(\mathcal{B}) \quad \forall \mathcal{B} \in \mathscr{F}_{\mathbb{R}^m}.$$

Probabilistic forecast reconciliation

The reconciled probability measure of $\hat{\nu}$ wrt $\psi(.)$ is such that

$$\tilde{\nu}(\mathcal{B}) = \hat{\nu}(\psi^{-1}(\mathcal{B})) \qquad \forall \mathcal{B} \in \mathscr{F}_{\mathfrak{s}}\,,$$

where $\psi^{-1}(\mathcal{B}) := \{\mathbf{y} \in \mathbb{R}^n : \psi(\mathbf{y}) \in \mathcal{B}\}$ is the pre-image of $\mathcal{B}$, that is the set of all points in $\mathbb{R}^n$ that $\psi(.)$ maps to a point in $\mathcal{B}$.



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

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

- $\boldsymbol{S}^* = \begin{pmatrix} \boldsymbol{S}^- \\ \boldsymbol{S}'_\perp \end{pmatrix}$
- $\boldsymbol{S}^-$ 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 $\boldsymbol{S}'_\perp\boldsymbol{S} = \boldsymbol{0}$.

To add from probabilistic paper

- Theorem 3.5
- Table 2
- Score optimal reconciliation
- Electricity example