MONASH University

MONASH BUSINESS SCHOOL

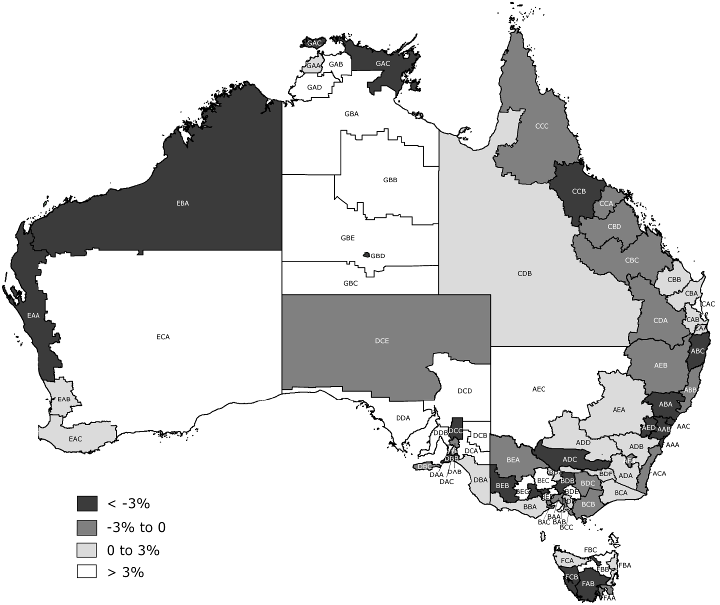
Forecasting: principles and practice

Rob J Hyndman

3.3 Hierarchical forecasting

Outline

Australian tourism demand



Australian tourism demand

- Quarterly data on visitor night from 1998:Q1 – 2013:Q4
- From: *National Visitor Survey*, based on annual interviews of 120,000 Australians aged 15+, collected by Tourism Research Australia.
- 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

> 3%

Spectacle sales

- Monthly UK sales data from 2000 – 2014
- Provided by a large spectacle manufacturer
- Split by brand (26), gender (3), price range (6), materials (4), and stores (600)
- About 1 million bottom-level series

Spectacle sales

- Monthly UK sales data from 2000 – 2014
- Provided by a large spectacle manufacturer
- Split by brand (26), gender (3), price range (6), materials (4), and stores (600)
- About 1 million bottom-level series

Spectacle sales

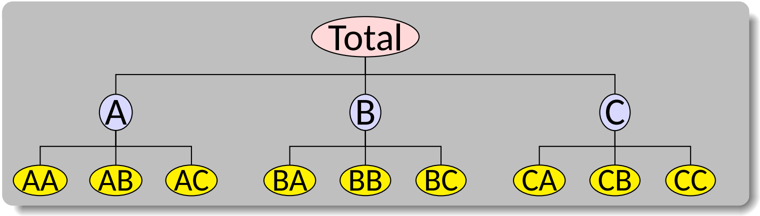
- Monthly UK sales data from 2000 – 2014
- Provided by a large spectacle manufacturer
- Split by brand (26), gender (3), price range (6), materials (4), and stores (600)
- About 1 million bottom-level series

Spectacle sales

- Monthly UK sales data from 2000 – 2014
- Provided by a large spectacle manufacturer
- Split by brand (26), gender (3), price range (6), materials (4), and stores (600)
- About 1 million bottom-level series

Hierarchical time series

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

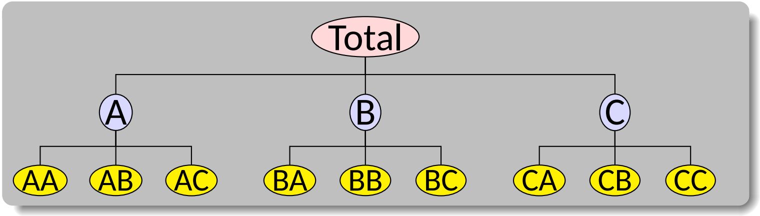


Examples

- Tourism by state and region

Hierarchical time series

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

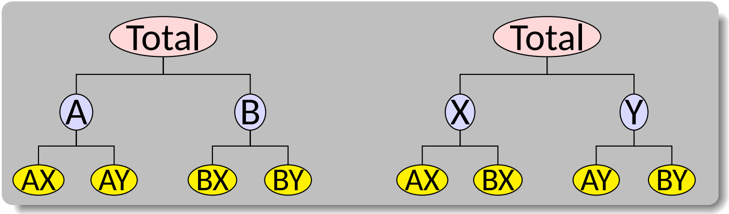


Examples

- Tourism by state and region

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.

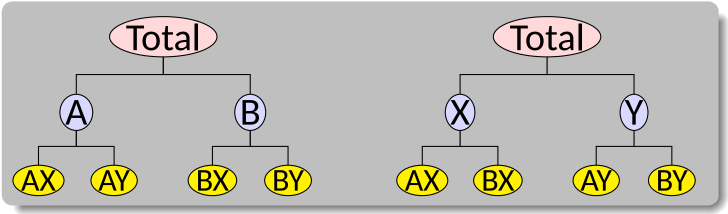


Examples

- Labour turnover by occupation and state
- Spectacle sales by brand, gender, stores, etc.

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.

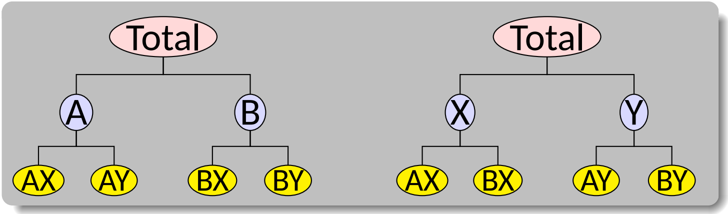


Examples

- Labour turnover by occupation and state
- Spectacle sales by brand, gender, stores, etc.

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

- Labour turnover by occupation and state
- Spectacle sales by brand, gender, stores, etc.

The problem

1. How to forecast time series at all nodes such that the forecasts add up in the same way as the original data?
2. Can we exploit relationships between the series to improve the forecasts?

The problem

1. How to forecast time series at all nodes such that the forecasts add up in the same way as the original data?
2. Can we exploit relationships between the series to improve the forecasts?

The solution

1. Forecast all series at all levels of aggregation using an automatic forecasting algorithm (e.g., `ets`, `auto.arima`, `...`)
2. Reconcile the resulting forecasts so they add up correctly using least squares optimization (i.e., find closest reconciled forecasts to the original forecasts).
3. This is all available in the **hts** package in R.

The solution

1. Forecast all series at all levels of aggregation using an automatic forecasting algorithm (e.g., `ets`, `auto.arima`, `...`)
2. Reconcile the resulting forecasts so they add up correctly using least squares optimization (i.e., find closest reconciled forecasts to the original forecasts).
3. This is all available in the **hts** package in R.

The solution

1. Forecast all series at all levels of aggregation using an automatic forecasting algorithm (e.g., `ets`, `auto.arima`, `...`)
2. Reconcile the resulting forecasts so they add up correctly using least squares optimization (i.e., find closest reconciled forecasts to the original forecasts).
3. This is all available in the **hts** package in R.

hts package for R

hts: Hierarchical and Grouped Time Series

Methods for analysing and forecasting hierarchical and grouped time series

Version:	5.0
Depends:	R ($\geq$ 3.0.2), forecast ($\geq$ 5.0), SparseM, Matrix, matrixcalc
Imports:	parallel, utils, methods, graphics, grDevices, stats
LinkingTo:	Rcpp ($\geq$ 0.11.0), RcppEigen
Suggests:	testthat
Published:	2016-04-06
Author:	Rob J Hyndman, Earo Wang, Alan Lee, Shanika Wickramasuriya
Maintainer:	Rob J Hyndman <Rob.Hyndman at monash.edu>
BugReports:	`https://github.com/robjhyndman/hts/issues`
License:	GPL ($\geq$ 2)

Example using R

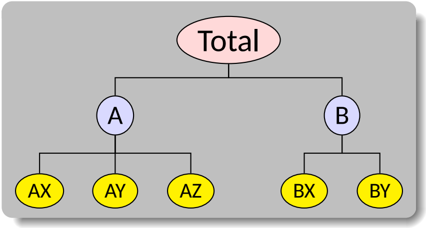
```
library(hts)

# bts is a matrix containing the bottom level time series
# nodes describes the hierarchical structure
y <- hts(bts, nodes=list(2, c(3,2)))
```

Example using R

```
library(hts)

# bts is a matrix containing the bottom level time series
# nodes describes the hierarchical structure
y <- hts(bts, nodes=list(2, c(3,2)))
```

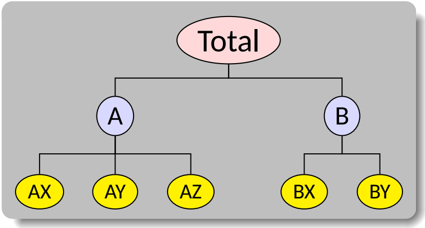


Example using R

```
library(hts)

# bts is a matrix containing the bottom level time series
# nodes describes the hierarchical structure
y <- hts(bts, nodes=list(2, c(3,2)))

# Forecast 10-step-ahead using WLS combination method
# ETS used for each series by default
fc <- forecast(y, h=10)
```



gts function

Usage

```
gts(y, characters)
```

Arguments

`y`	Multivariate time series containing the bottom level series
`characters`	Vector of integers, or list of vectors, showing how column names indicate group structure.

Example

```
bnames <-
  c("VIC1F","VIC1M","VIC2F","VIC2M","VIC3F","VIC3M",
    "NSW1F","NSW1M","NSW2F","NSW2M","NSW3F","NSW3M")
bts <- matrix(ts(rnorm(120)), ncol = 12)
colnames(bts) <- bnames
x <- gts(bts, characters = c(3, 1, 1))
```

gts function

Example 2

```
bnames <-
 c("VICMelbAA","VICMelbAB",
   "VICGeelAA","VICGeelAB",
   "VICMelbBA","VICMelbBB",
   "VICGeelBA","VICGeelBB",
   "NSWSyndAA","NSWSyndAB",
   "NSWWollAA","NSWWollAB",
   "NSWSyndBA","NSWSyndBB",
   "NSWWollBA","NSWWollBB")
bts <- matrix(ts(rnorm(160)), ncol = 16)
colnames(bts) <- bnames
x <- gts(bts, characters = list(c(3, 4), c(1, 1)))
```

forecast.gts function

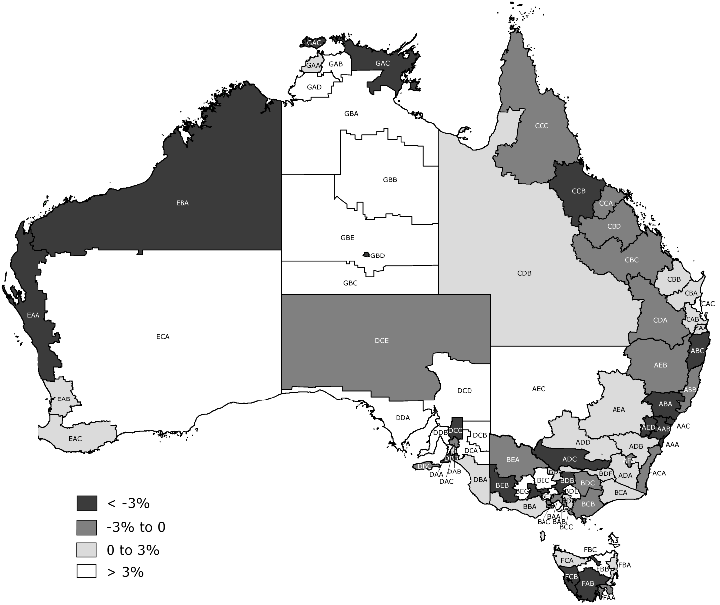
Usage

```
forecast(object, h,
  method = c("comb", "bu", "mo","tdgsa", "tdgsf", "tdfp"),
  weights = c("wls", "ols", "mint", "nseries"),
  fmethod = c("ets", "arima", "rw"),
  algorithms = c("lu", "cg", "chol", "recursive", "slm"),
  covariance = c("shr", "sam"),
  positive = FALSE,
  parallel = FALSE, num.cores = 2, ...)
```

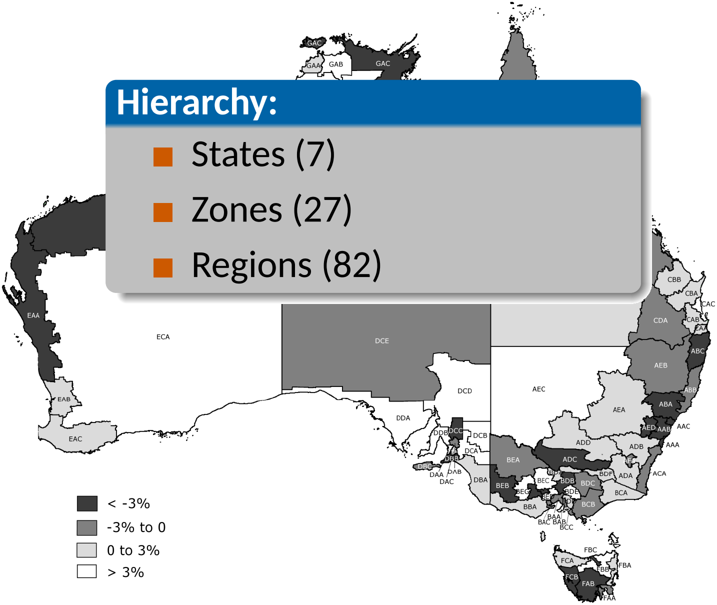
Arguments

`object`	Hierarchical time series object of class `gts`.
`h`	Forecast horizon
`method`	Method for distributing forecasts within the hierarchy.
`weights`	Weights used for "optimal combination" method. When weights = "sd", it takes account of the standard deviation of forecasts.
`fmethod`	Forecasting method to use
`algorithm`	Method for solving regression equations
`positive`	If TRUE, forecasts are forced to be strictly positive
`parallel`	If TRUE, allow parallel processing
`num.cores`	If parallel = TRUE, specify how many cores are going to be used

Example: Australian tourism



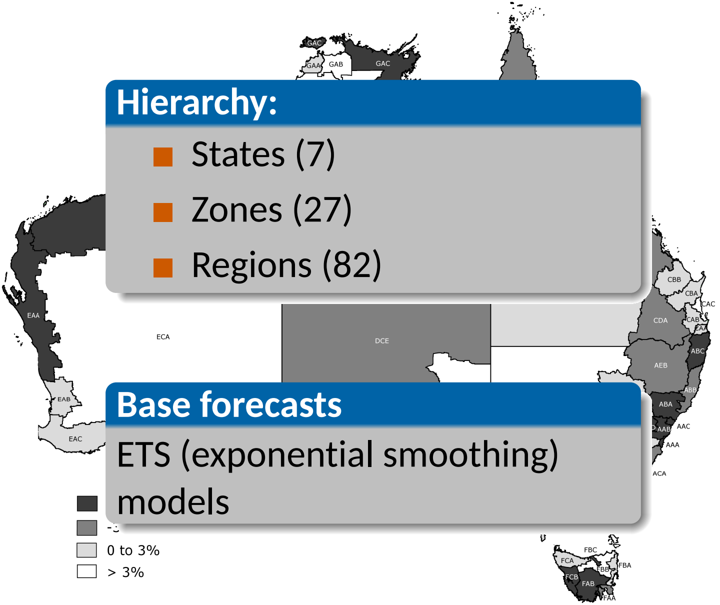
Example: Australian tourism



Hierarchy:

- States (7)
- Zones (27)
- Regions (82)

Example: Australian tourism



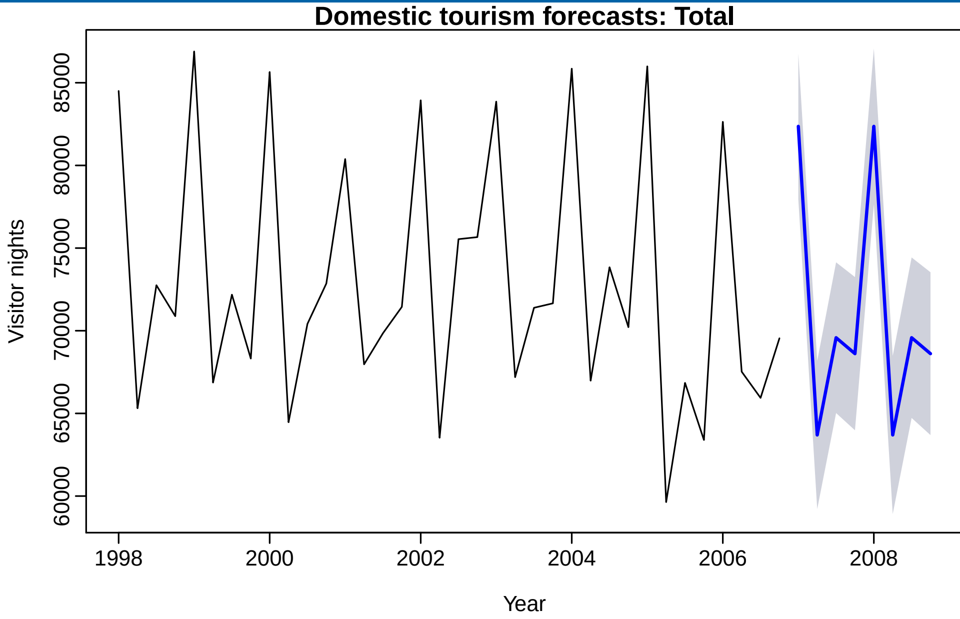
Hierarchy:

- States (7)
- Zones (27)
- Regions (82)

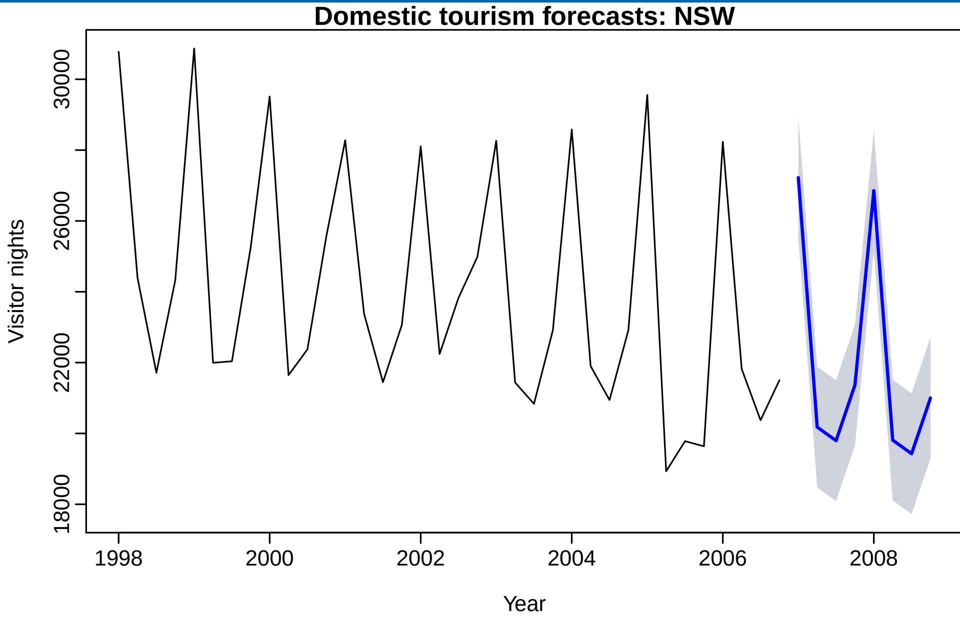
Base forecasts

ETS (exponential smoothing) models

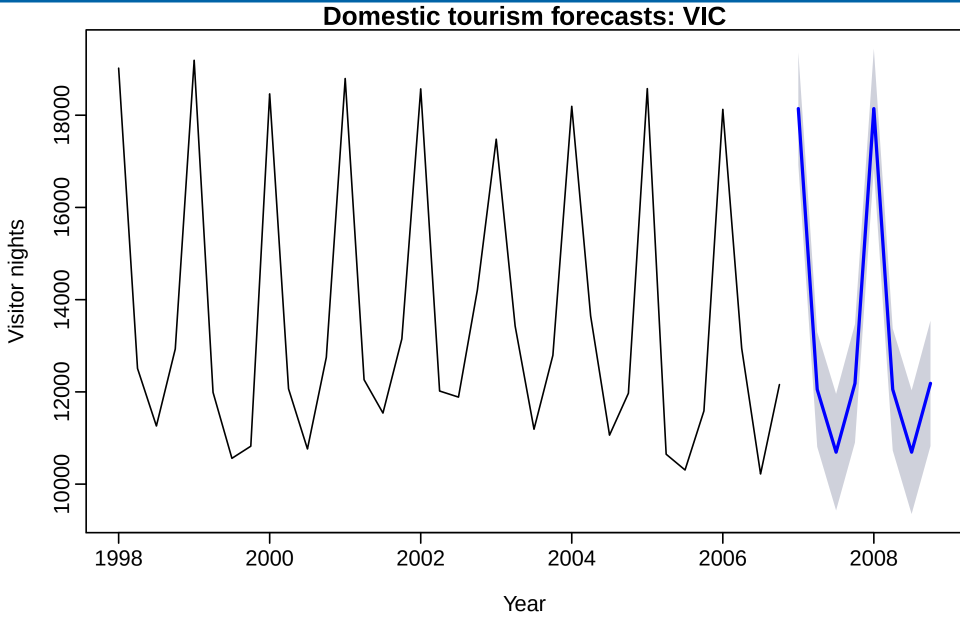
Base forecasts



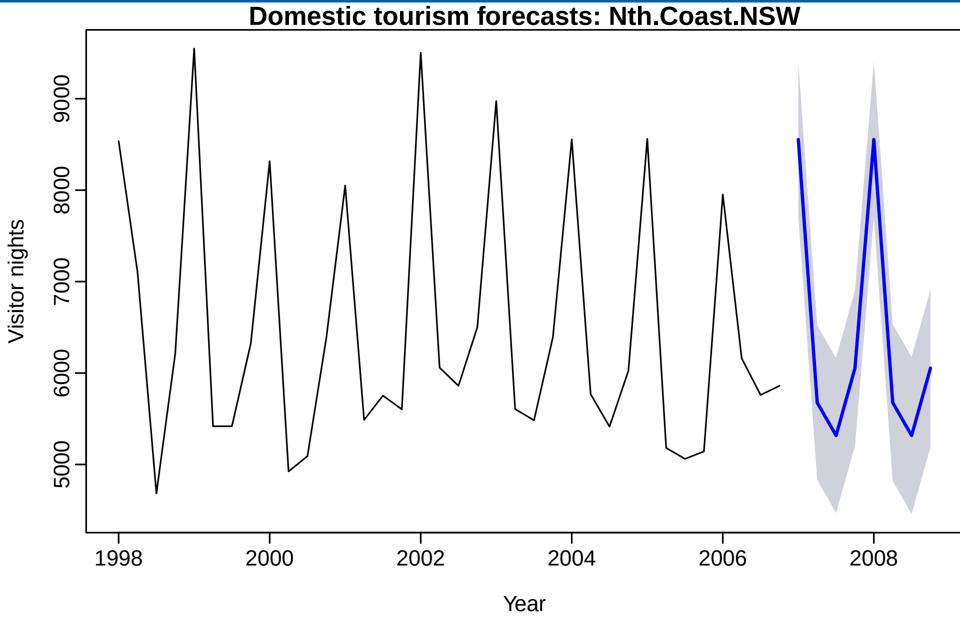
Base forecasts



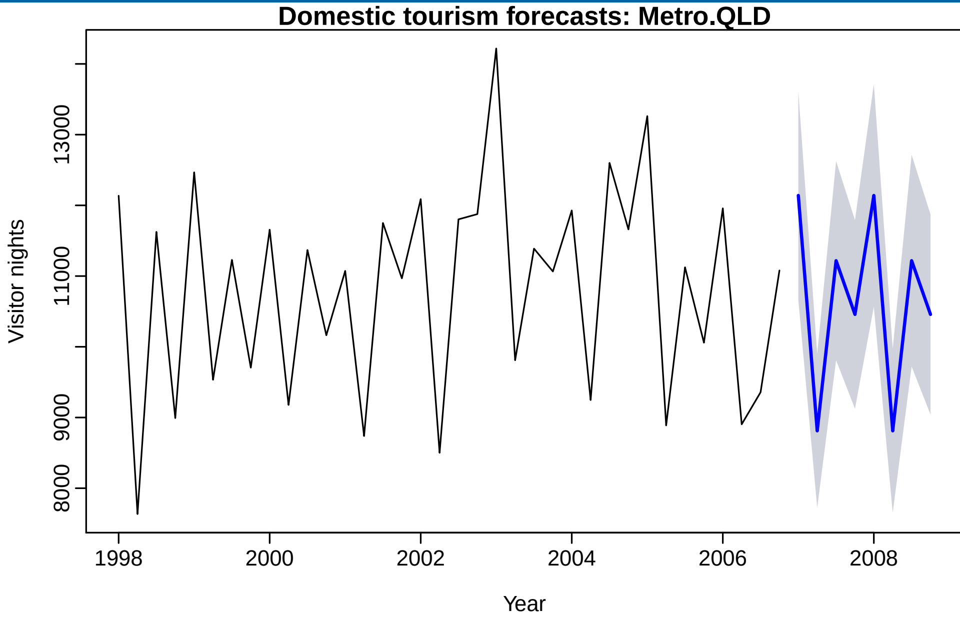
Base forecasts



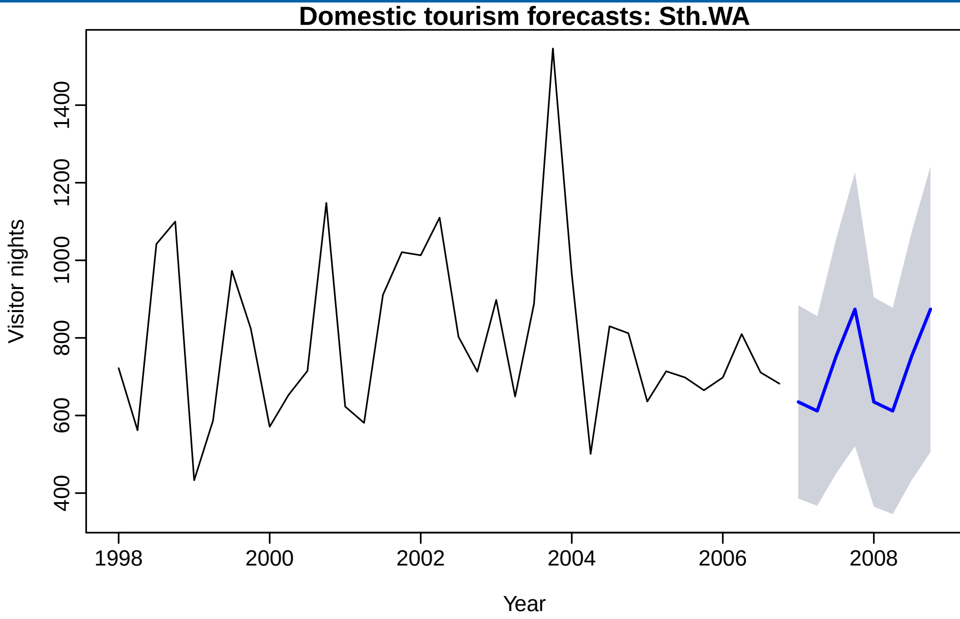
Base forecasts



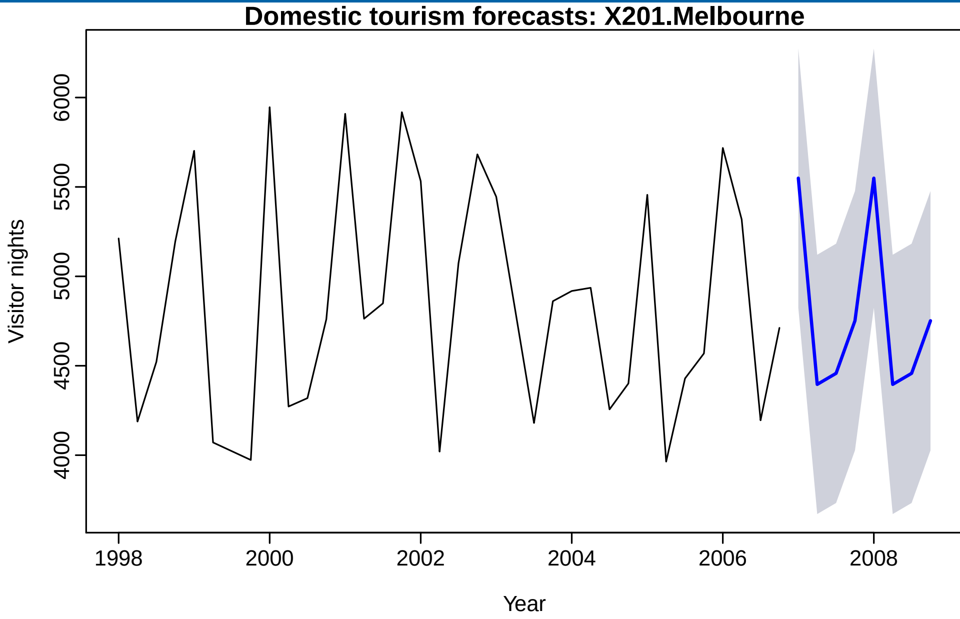
Base forecasts



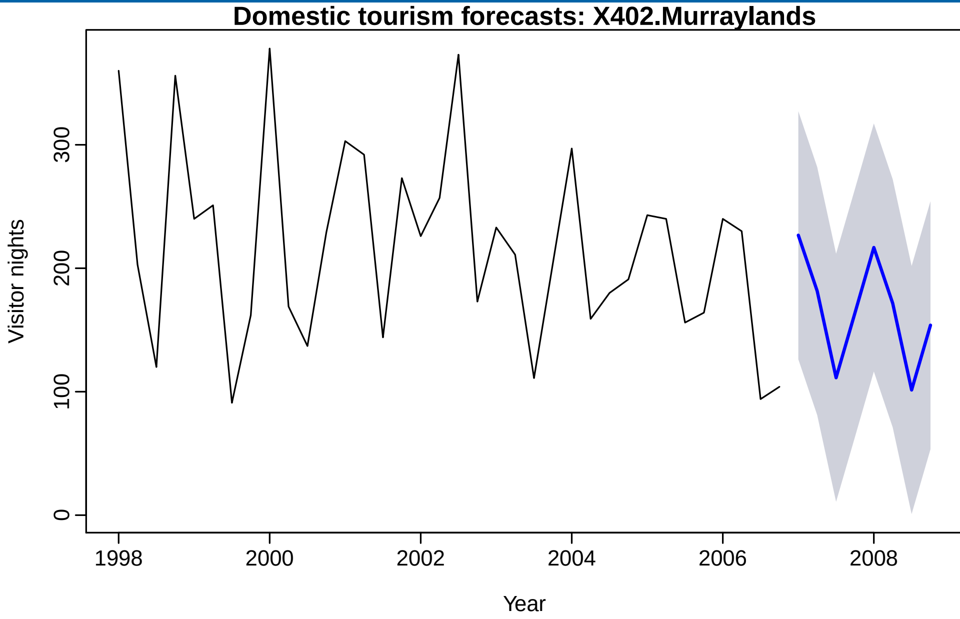
Base forecasts



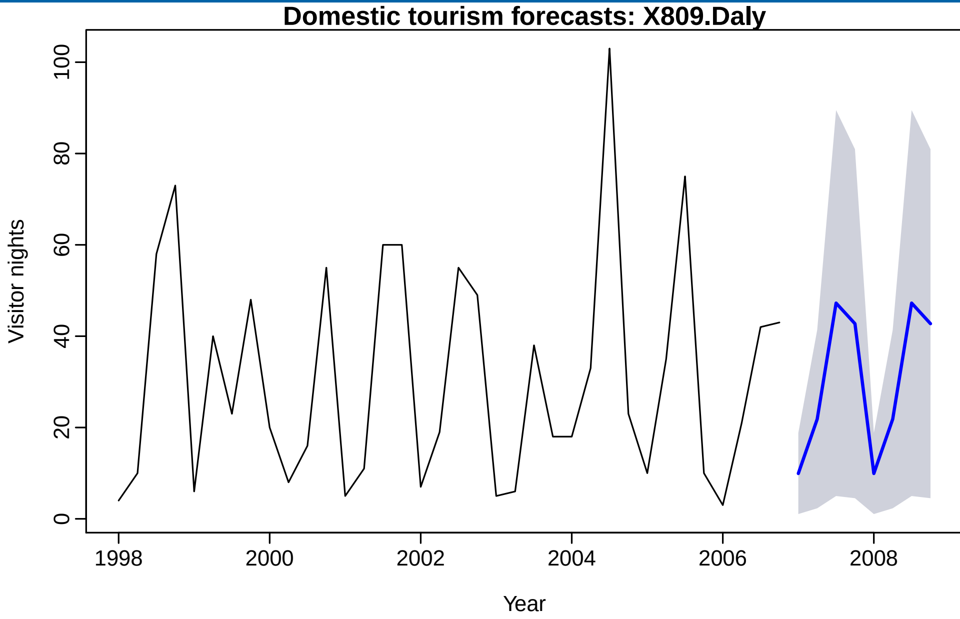
Base forecasts



Base forecasts



Base forecasts

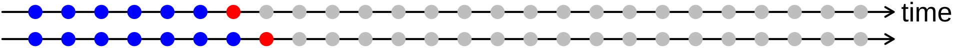


Forecast evaluation

Training sets **Test sets $h = 1$**

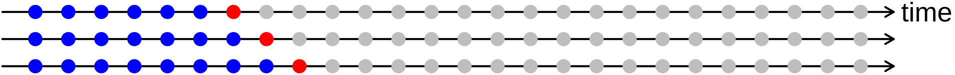
Forecast evaluation

Training sets **Test sets $h = 1$**



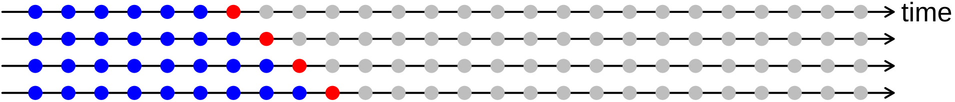
Forecast evaluation

Training sets **Test sets $h = 1$**

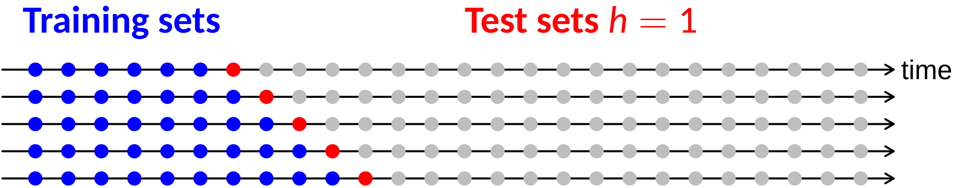


Forecast evaluation

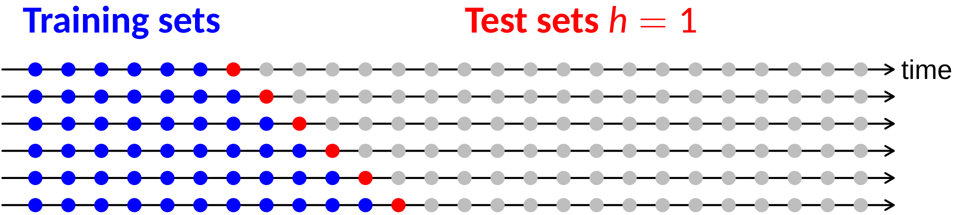
Training sets **Test sets $h = 1$**



Forecast evaluation

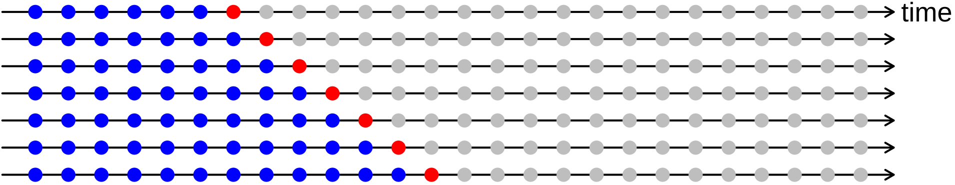


Forecast evaluation

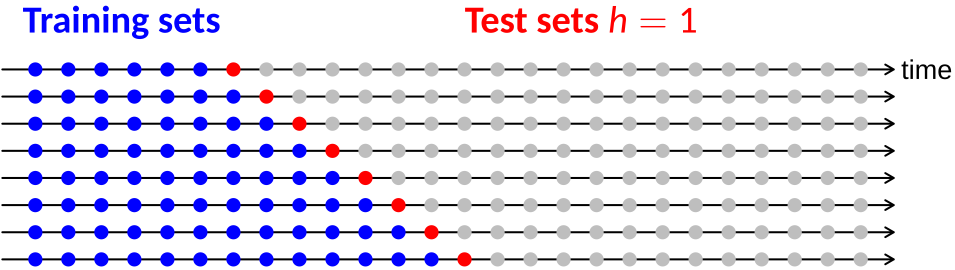


Forecast evaluation

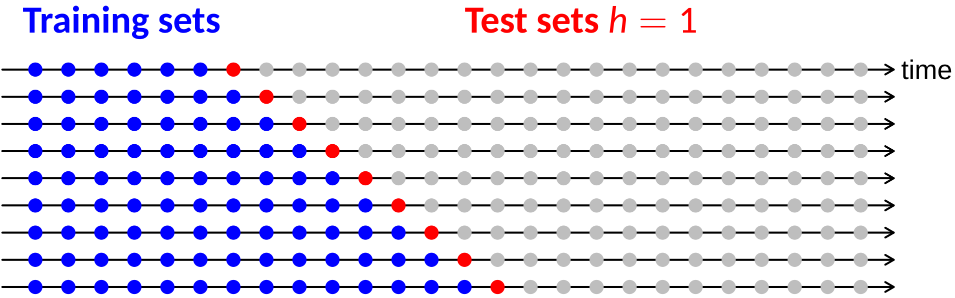
Training sets **Test sets $h = 1$**



Forecast evaluation

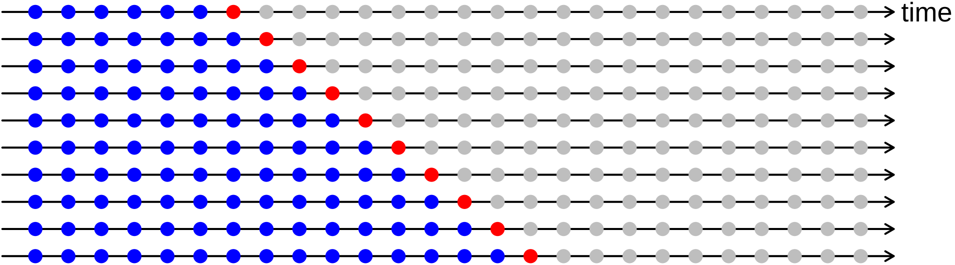


Forecast evaluation

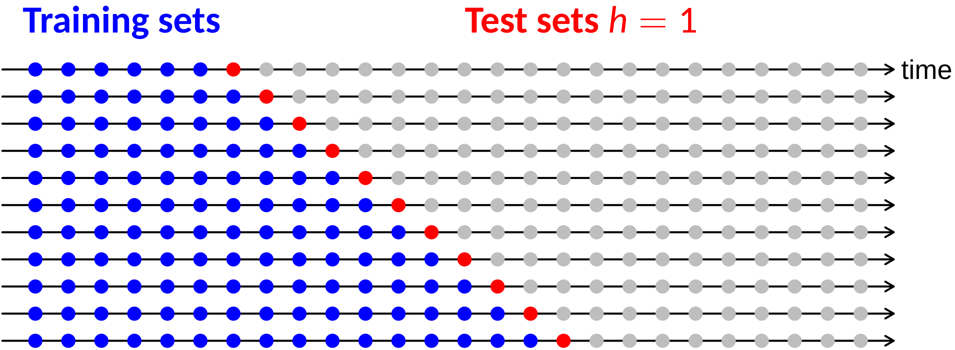


Forecast evaluation

Training sets **Test sets $h = 1$**

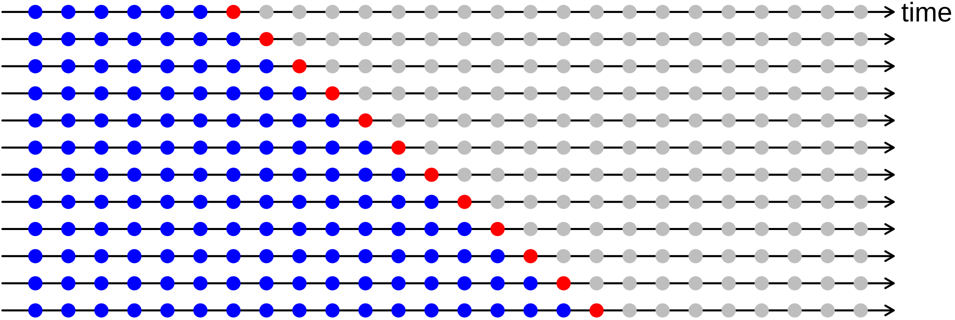


Forecast evaluation



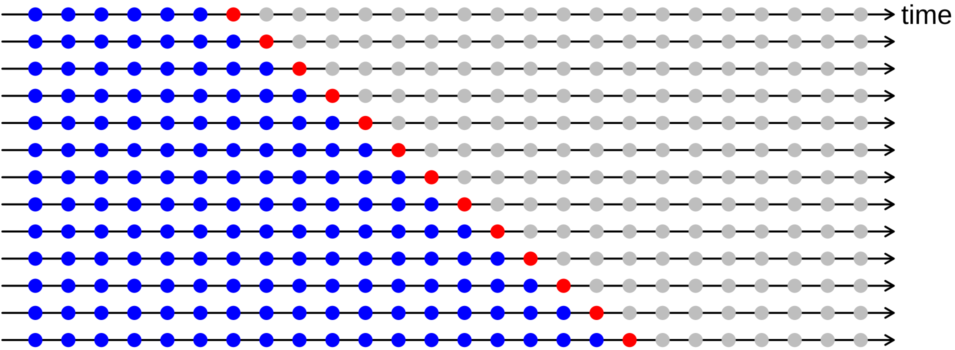
Forecast evaluation

Training sets **Test sets $h = 1$**



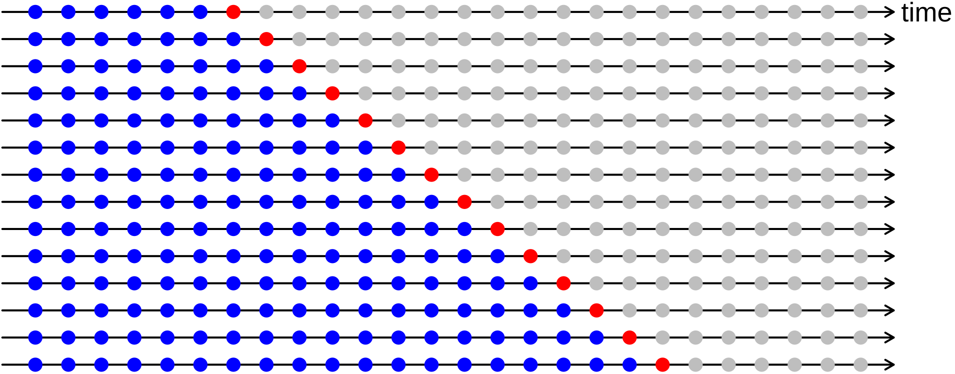
Forecast evaluation

Training sets **Test sets $h = 1$**

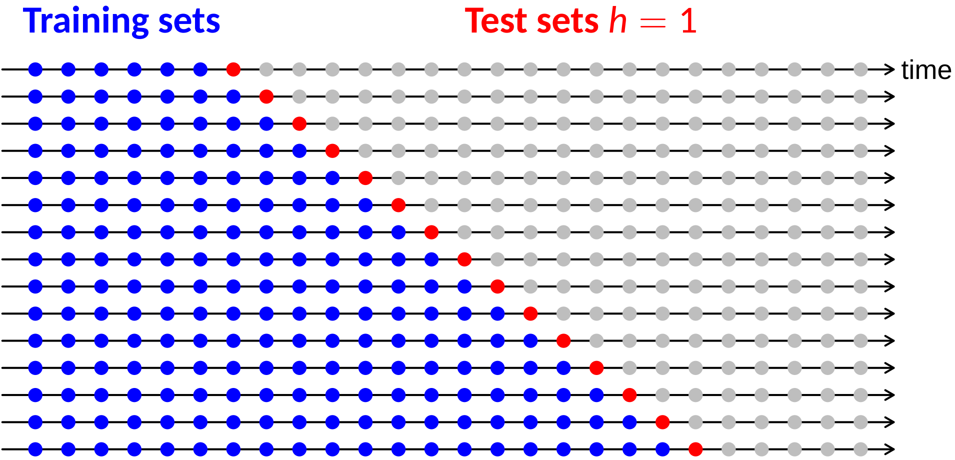


Forecast evaluation

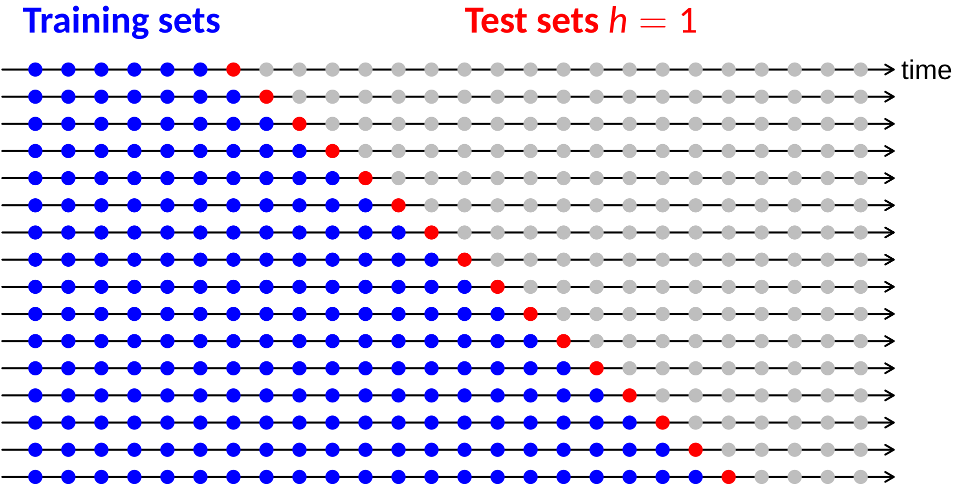
Training sets **Test sets $h = 1$**



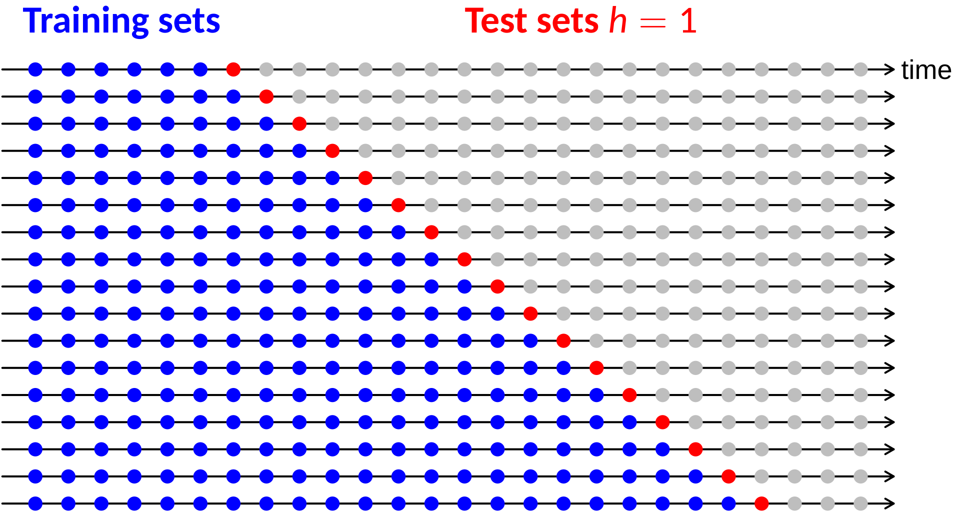
Forecast evaluation



Forecast evaluation

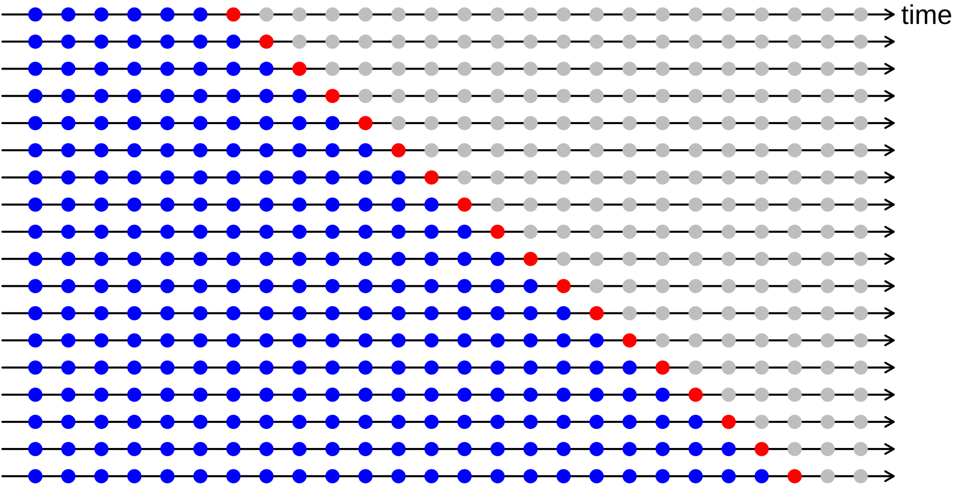


Forecast evaluation



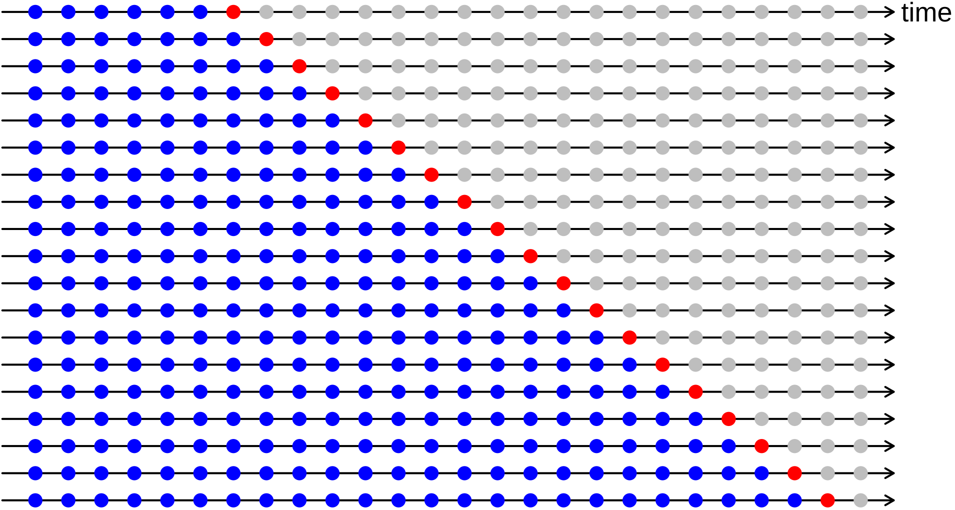
Forecast evaluation

Training sets **Test sets $h = 1$**



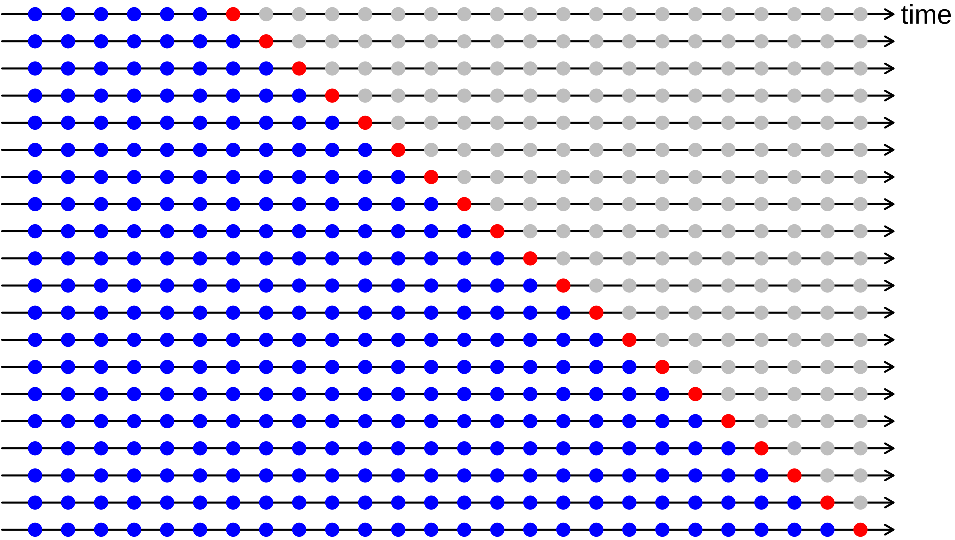
Forecast evaluation

Training sets **Test sets $h = 1$**



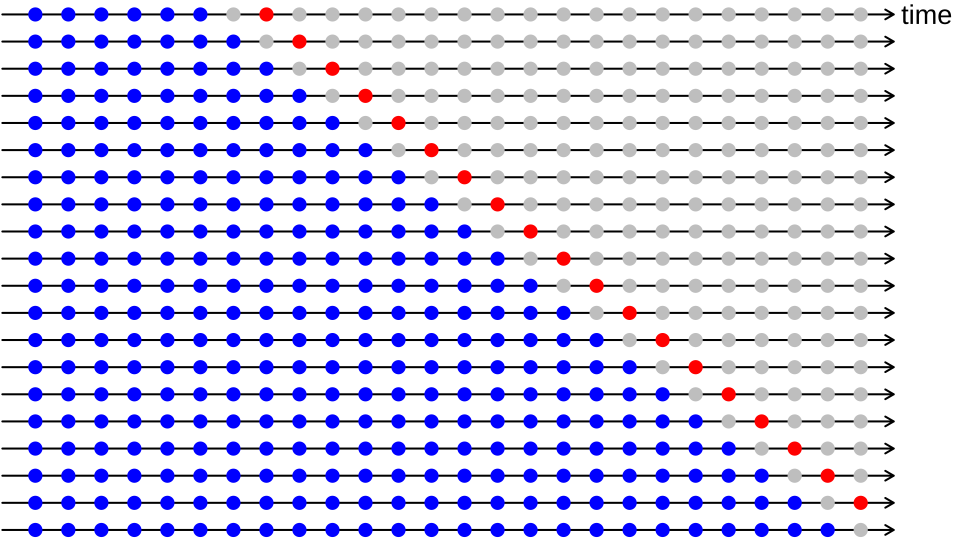
Forecast evaluation

Training sets **Test sets $h = 1$**



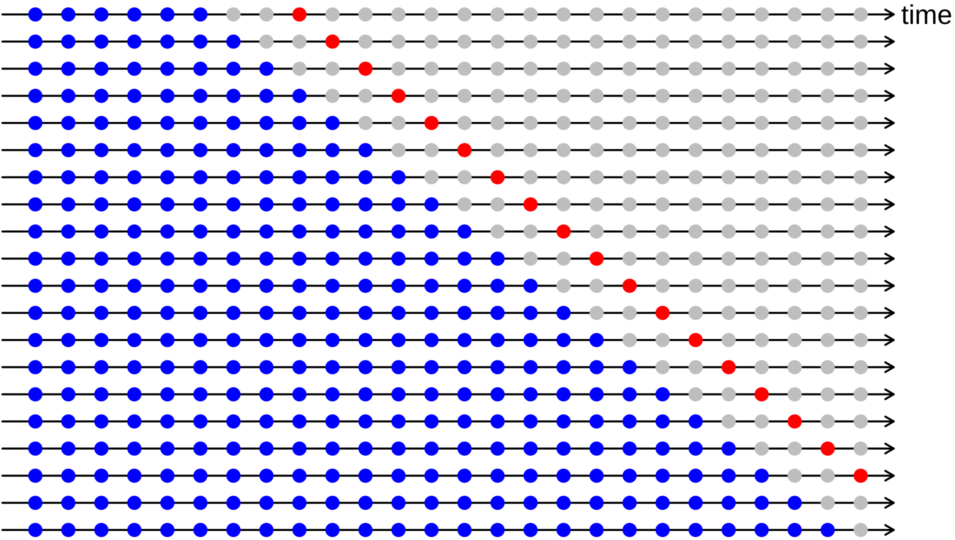
Forecast evaluation

Training sets **Test sets $h = 2$**



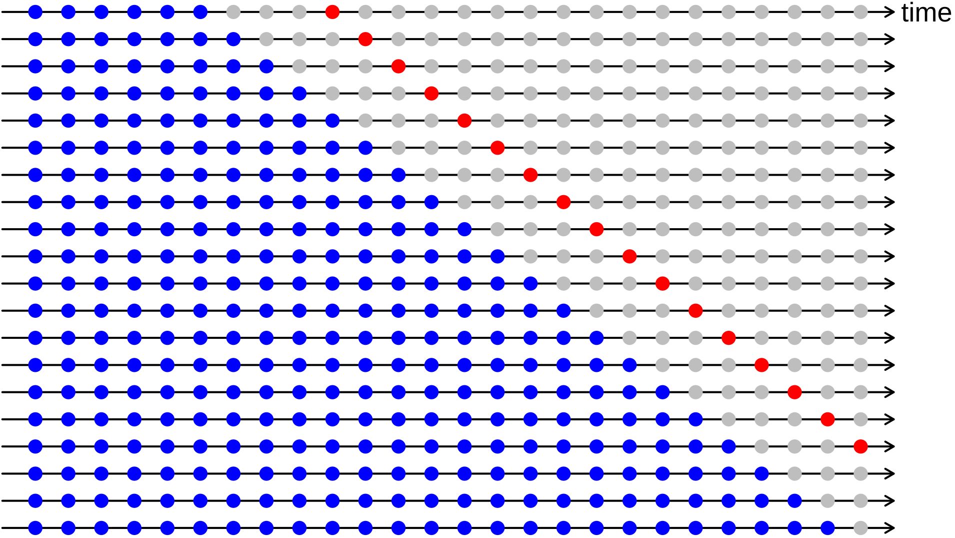
Forecast evaluation

Training sets **Test sets $h = 3$**

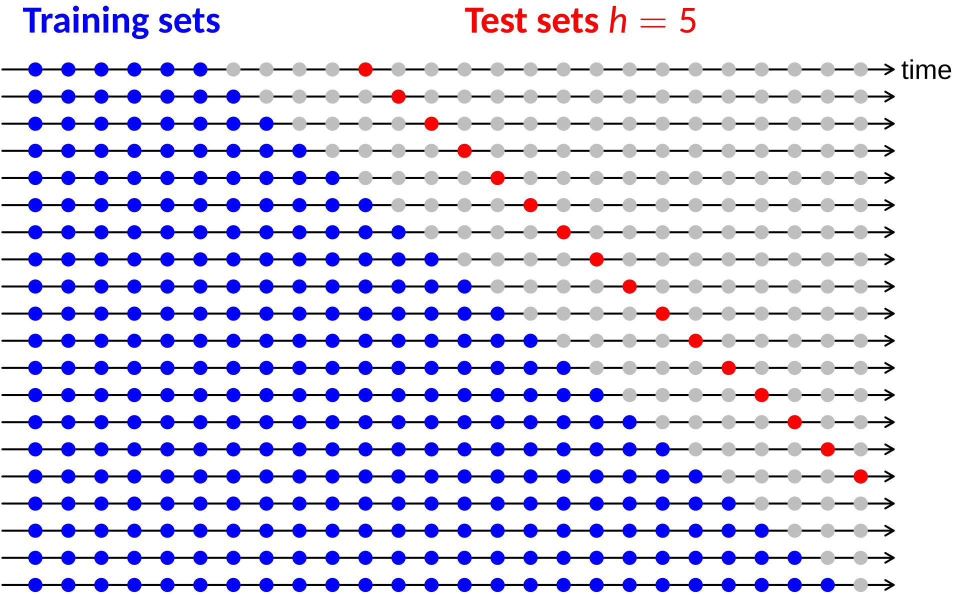


Forecast evaluation

Training sets **Test sets $h = 4$**

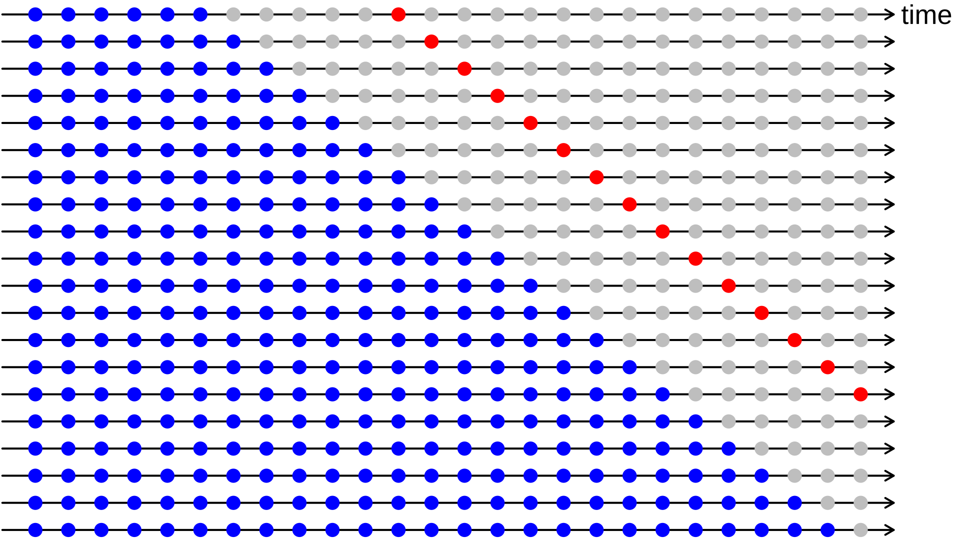


Forecast evaluation



Forecast evaluation

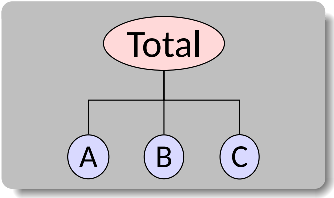
Training sets **Test sets $h = 6$**



Hierarchy: states, zones, regions

RMSE	$h = 1$	$h = 2$	$h = 3$	$h = 4$	$h = 5$	$h = 6$	**Ave**
				Forecast horizon			
				Australia			
Base	1762.04	1770.29	1766.02	1818.82	1705.35	1721.17	**1757.28**
Bottom	1736.92	1742.69	1722.79	1752.74	1666.73	1687.43	**1718.22**
OLS	1747.60	1757.68	1751.77	1800.67	1686.00	1706.45	**1741.69**
WLS	1705.21	1715.87	1703.75	1729.56	1627.79	1661.24	**1690.57**
GLS	1704.64	1715.60	1705.31	1729.04	1626.36	1661.64	**1690.43**
				States			
Base	399.77	404.16	401.92	407.26	395.38	401.17	**401.61**
Bottom	404.29	406.95	404.96	409.02	399.80	401.55	**404.43**
OLS	404.47	407.62	405.43	413.79	401.10	404.90	**406.22**
WLS	398.84	402.12	400.71	405.03	394.76	398.23	**399.95**
GLS	398.84	402.16	400.86	405.03	394.59	398.22	**399.95**
				Regions			
Base	93.15	93.38	93.45	93.79	93.50	93.56	**93.47**
Bottom	93.15	93.38	93.45	93.79	93.50	93.56	**93.47**
OLS	93.28	93.53	93.64	94.17	93.78	93.88	**93.71**
WLS	93.02	93.32	93.38	93.72	93.39	93.53	**93.39**
GLS	92.98	93.27	93.34	93.66	93.34	93.46	**93.34**

Hierarchical time series

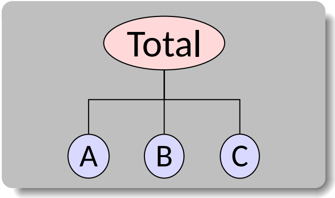


y_t : observed aggregate of all series at time t.

$y_{X,t}$: observation on series X at time t.

$\boldsymbol{b}_t$: vector of all series at bottom level in time t.

Hierarchical time series

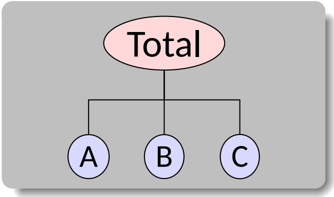


y_t : observed aggregate of all series at time t.

$y_{X,t}$: observation on series X at time t.

$\boldsymbol{b}_t$: vector of all series at bottom level in time t.

Hierarchical time series



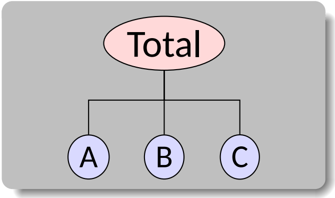
y_t : observed aggregate of all series at time t.

$y_{X,t}$: observation on series X at time t.

$\boldsymbol{b}_t$: vector of all series at bottom level in time t.

$$\boldsymbol{y}_t = [y_t, y_{A,t}, y_{B,t}, y_{C,t}]' = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_{A,t} \\ y_{B,t} \\ y_{C,t} \end{pmatrix}$$

Hierarchical time series



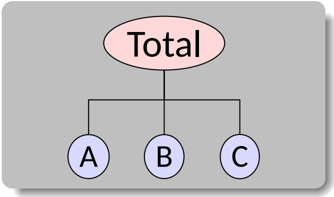
y_t : observed aggregate of all series at time t.

$y_{X,t}$: observation on series X at time t.

$\boldsymbol{b}_t$: vector of all series at bottom level in time t.

$$\boldsymbol{y}_t = [y_t, y_{A,t}, y_{B,t}, y_{C,t}]' = \underbrace{\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\boldsymbol{S}} \begin{pmatrix} y_{A,t} \\ y_{B,t} \\ y_{C,t} \end{pmatrix}$$

Hierarchical time series



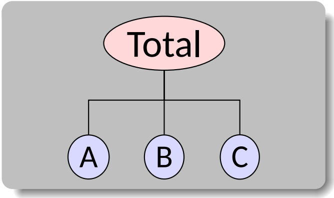
y_t : observed aggregate of all series at time t.

$y_{X,t}$: observation on series X at time t.

$\boldsymbol{b}_t$: vector of all series at bottom level in time t.

$$\boldsymbol{y}_t = [y_t, y_{A,t}, y_{B,t}, y_{C,t}]' = \underbrace{\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\boldsymbol{S}} \underbrace{\begin{pmatrix} y_{A,t} \\ y_{B,t} \\ y_{C,t} \end{pmatrix}}_{\boldsymbol{b}_t}$$

Hierarchical time series



y_t : observed aggregate of all series at time t.

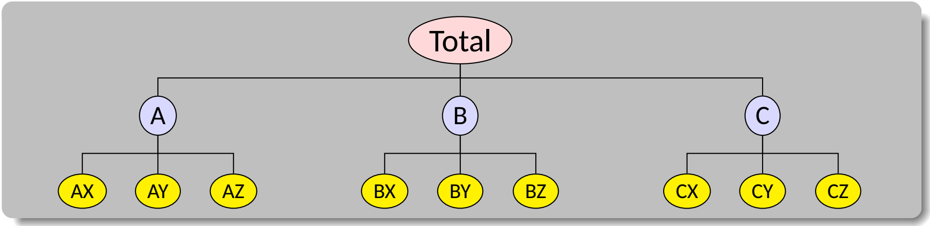
$y_{X,t}$: observation on series X at time t.

$\boldsymbol{b}_t$: vector of all series at bottom level in time t.

$$\boldsymbol{y}_t = [y_t, y_{A,t}, y_{B,t}, y_{C,t}]' = \underbrace{\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\boldsymbol{S}} \underbrace{\begin{pmatrix} y_{A,t} \\ y_{B,t} \\ y_{C,t} \end{pmatrix}}_{\boldsymbol{b}_t}$$

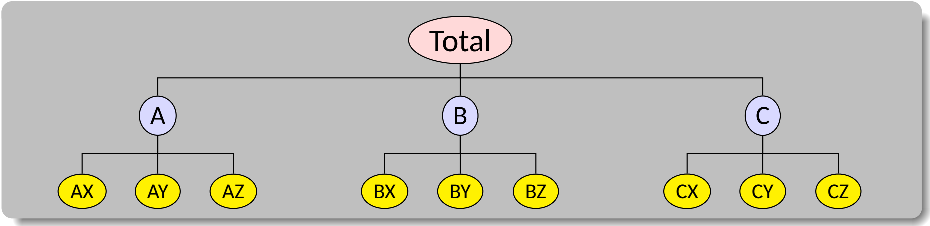
$$\boxed{\boldsymbol{y}_t = \boldsymbol{S}\boldsymbol{b}_t}$$

Hierarchical time series



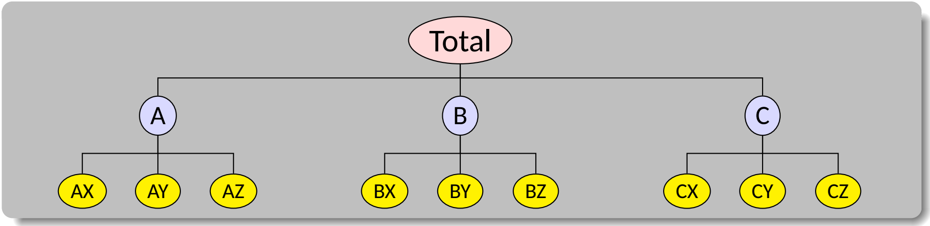
$$\boldsymbol{y}_t = \begin{pmatrix} y_t \\ y_{A,t} \\ y_{B,t} \\ y_{C,t} \\ y_{AX,t} \\ y_{AY,t} \\ y_{AZ,t} \\ y_{BX,t} \\ y_{BY,t} \\ y_{BZ,t} \\ y_{CX,t} \\ y_{CY,t} \\ y_{CZ,t} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0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

Hierarchical time series



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

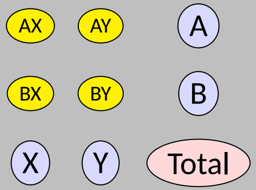
Hierarchical time series



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

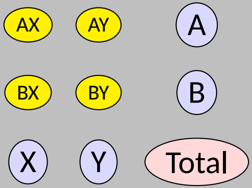
$$\boldsymbol{y}_t = \boldsymbol{S}\boldsymbol{b}_t$$

Grouped data



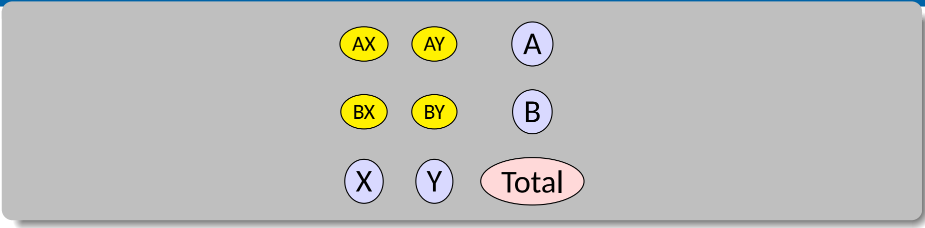
$$\boldsymbol{y}_t = \begin{pmatrix} y_t \\ y_{A,t} \\ y_{B,t} \\ y_{X,t} \\ y_{Y,t} \\ y_{AX,t} \\ y_{AY,t} \\ y_{BX,t} \\ y_{BY,t} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0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

Grouped data



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

Grouped data



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

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

Hierarchical and grouped time series

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

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

where

- $\boldsymbol{y}_t$ is a vector of all series at time t
- $\boldsymbol{b}_t$ is a vector of the most disaggregated series at time t
- $\boldsymbol{S}$ is a "summing matrix" containing the aggregation constraints.

Forecasting notation

Let $\hat{\boldsymbol{y}}_n(h)$ be vector of initial h-step forecasts, made at time n, stacked in same order as $\boldsymbol{y}_t$.
(In general, they will not "add up".)

Reconciled forecasts must be of the form:

$$\tilde{\boldsymbol{y}}_n(h) = \boldsymbol{S}\boldsymbol{P}\hat{\boldsymbol{y}}_n(h)$$

for some matrix $\boldsymbol{P}$.

- $\boldsymbol{P}$ extracts and combines base forecasts $\hat{\boldsymbol{y}}_n(h)$ to get bottom-level forecasts.

Forecasting notation

Let $\hat{\boldsymbol{y}}_n(h)$ be vector of initial h-step forecasts, made at time n, stacked in same order as $\boldsymbol{y}_t$.
(In general, they will not "add up".)

Reconciled forecasts must be of the form:

$$\tilde{\boldsymbol{y}}_n(h) = \boldsymbol{S}\boldsymbol{P}\hat{\boldsymbol{y}}_n(h)$$

for some matrix $\boldsymbol{P}$.

- $\boldsymbol{P}$ extracts and combines base forecasts $\hat{\boldsymbol{y}}_n(h)$ to get bottom-level forecasts.
- $\boldsymbol{S}$ adds them up

Forecasting notation

Let $\hat{\boldsymbol{y}}_n(h)$ be vector of initial h-step forecasts, made at time n, stacked in same order as $\boldsymbol{y}_t$.
(In general, they will not "add up".)

Reconciled forecasts must be of the form:

$$\tilde{\boldsymbol{y}}_n(h) = \boldsymbol{S}\boldsymbol{P}\hat{\boldsymbol{y}}_n(h)$$

for some matrix $\boldsymbol{P}$.

- $\boldsymbol{P}$ extracts and combines base forecasts $\hat{\boldsymbol{y}}_n(h)$ to get bottom-level forecasts.
- $\boldsymbol{S}$ adds them up

Forecasting notation

Let $\hat{\boldsymbol{y}}_n(h)$ be vector of initial h-step forecasts, made at time n, stacked in same order as $\boldsymbol{y}_t$.
(In general, they will not "add up".)

Reconciled forecasts must be of the form:

$$\tilde{\boldsymbol{y}}_n(h) = \boldsymbol{S}\boldsymbol{P}\hat{\boldsymbol{y}}_n(h)$$

for some matrix $\boldsymbol{P}$.

- $\boldsymbol{P}$ extracts and combines base forecasts $\hat{\boldsymbol{y}}_n(h)$ to get bottom-level forecasts.
- $\boldsymbol{S}$ adds them up

Forecasting notation

Let $\hat{\boldsymbol{y}}_n(h)$ be vector of initial h-step forecasts, made at time n, stacked in same order as $\boldsymbol{y}_t$.
(In general, they will not "add up".)

Reconciled forecasts must be of the form:

$$\tilde{\boldsymbol{y}}_n(h) = \boldsymbol{S}\boldsymbol{P}\hat{\boldsymbol{y}}_n(h)$$

for some matrix $\boldsymbol{P}$.

- $\boldsymbol{P}$ extracts and combines base forecasts $\hat{\boldsymbol{y}}_n(h)$ to get bottom-level forecasts.
- $\boldsymbol{S}$ adds them up

Optimal combination forecasts

Main result

The best (minimum sum of variances) unbiased forecasts are obtained when $\boldsymbol{P} = (\boldsymbol{S}'\Sigma_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\Sigma_h^{-1}$, where Σ_h is the h-step base forecast error covariance matrix.

$$\tilde{\boldsymbol{y}}_n(h) = \boldsymbol{S}(\boldsymbol{S}'\Sigma_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\Sigma_h^{-1}\hat{\boldsymbol{y}}_n(h)$$

Reconciled forecasts — Base forecasts

Problem: Σ_h hard to estimate, especially for $h > 1$.

Solutions:

- Ignore Σ_h (OLS)
- Assume Σ_h diagonal (WLS) [Default in `hts`]
- Try to estimate Σ_h (GLS)

Optimal combination forecasts

Main result

The best (minimum sum of variances) unbiased forecasts are obtained when $\boldsymbol{P} = (\boldsymbol{S}'\Sigma_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\Sigma_h^{-1}$, where Σ_h is the *h*-step base forecast error covariance matrix.

$$\tilde{\boldsymbol{y}}_n(h) = \boldsymbol{S}(\boldsymbol{S}'\Sigma_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\Sigma_h^{-1}\hat{\boldsymbol{y}}_n(h)$$

Reconciled forecasts

Base forecasts

Problem: Σ_h hard to estimate, especially for $h > 1$.

Solutions:

- Ignore Σ_h (OLS)
- Assume Σ_h diagonal (WLS) [Default in `hts`]
- Try to estimate Σ_h (GLS)

Optimal combination forecasts

Main result

The best (minimum sum of variances) unbiased forecasts are obtained when $\boldsymbol{P} = (\boldsymbol{S}'\Sigma_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\Sigma_h^{-1}$, where Σ_h is the *h*-step base forecast error covariance matrix.

$$\tilde{\boldsymbol{y}}_n(h) = \boldsymbol{S}(\boldsymbol{S}'\Sigma_h^{-1}\boldsymbol{S})^{-1}\boldsymbol{S}'\Sigma_h^{-1}\hat{\boldsymbol{y}}_n(h)$$

Reconciled forecasts

Base forecasts

Problem: Σ_h hard to estimate, especially for $h > 1$.

Solutions:

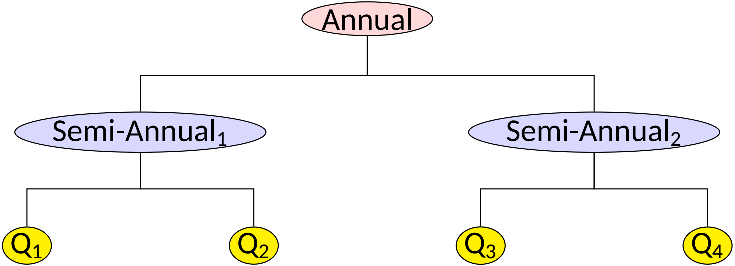
- Ignore Σ_h (OLS)
- Assume Σ_h diagonal (WLS) [Default in `hts`]
- Try to estimate Σ_h (GLS)

Outline

Lab Session 15

Outline

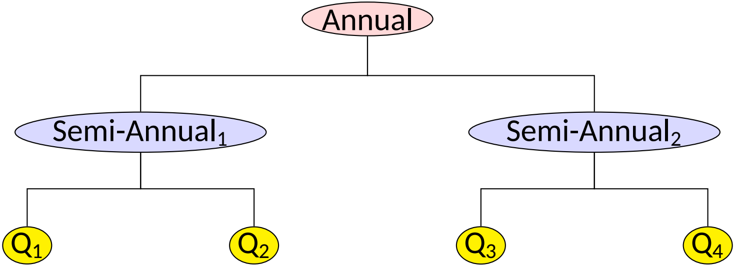
Temporal hierarchies



Basic idea:

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

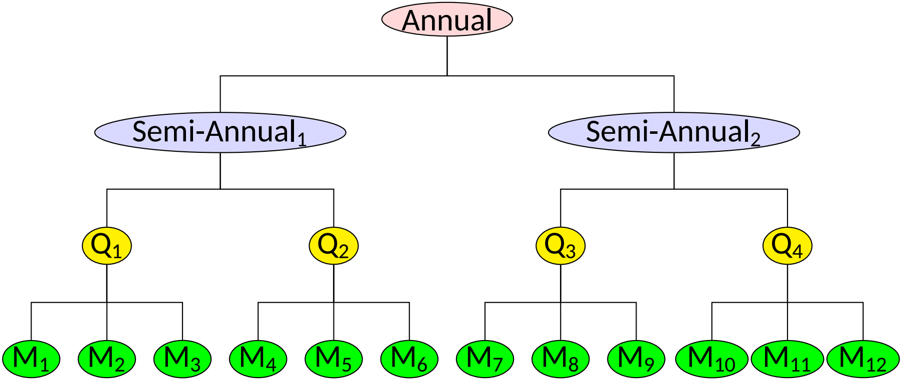
Temporal hierarchies



Basic idea:

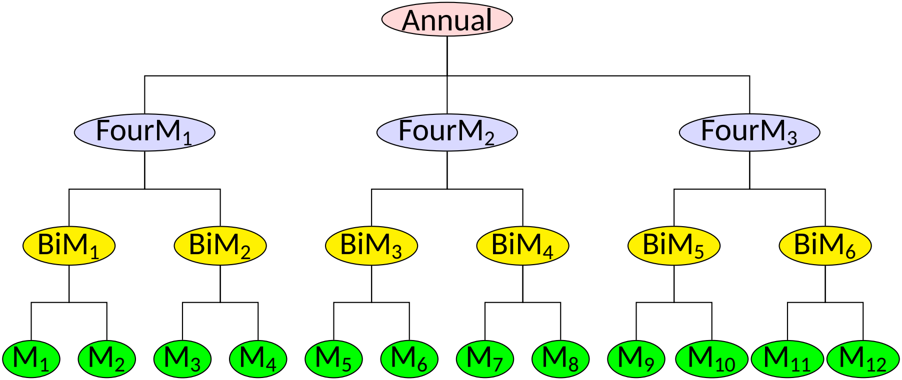
- Forecast series at each available frequency.
- Optimally reconcile forecasts within the same year.

Monthly series



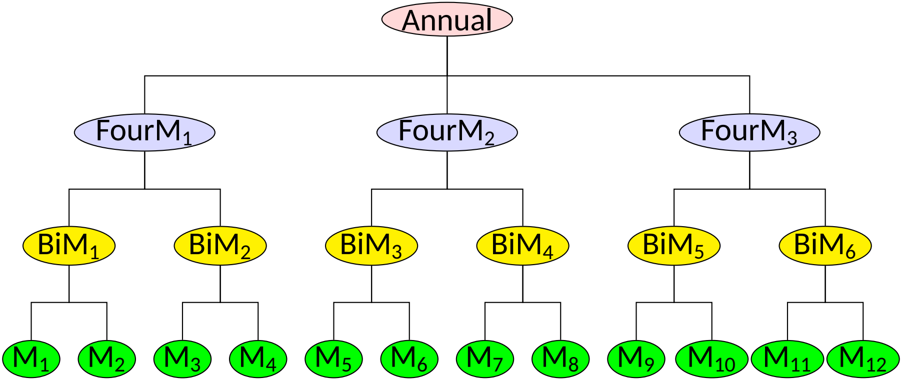
- $k = 2, 4, 12$ nodes
- $k = 3, 6, 12$ nodes
- Why not $k = 2, 3, 4, 6, 12$ nodes?

Monthly series



- $k = 2, 4, 12$ nodes
- $k = 3, 6, 12$ nodes
- Why not $k = 2, 3, 4, 6, 12$ nodes?

Monthly series



- $k = 2, 4, 12$ nodes
- $k = 3, 6, 12$ nodes
- Why not $k = 2, 3, 4, 6, 12$ nodes?

Monthly data

$$\underbrace{\begin{pmatrix} A \\ SemiA_1 \\ SemiA_2 \\ FourM_1 \\ FourM_2 \\ FourM_3 \\ Q_1 \\ \vdots \\ Q_4 \\ BiM_1 \\ \vdots \\ BiM_6 \\ M_1 \\ \vdots \\ M_{12} \end{pmatrix}}_{(28\times 1)} = \underbrace{\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & \vdots & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & \vdots & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ & & & & & & \boldsymbol{I}_{12} & & & & & \end{pmatrix}}_{\boldsymbol{S}} \underbrace{\begin{pmatrix} M_1 \\ M_2 \\ M_3 \\ M_4 \\ M_5 \\ M_6 \\ M_7 \\ M_8 \\ M_9 \\ M_{10} \\ M_{11} \\ M_{12} \end{pmatrix}}_{\boldsymbol{B}_t}$$

In general

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.

In general

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.

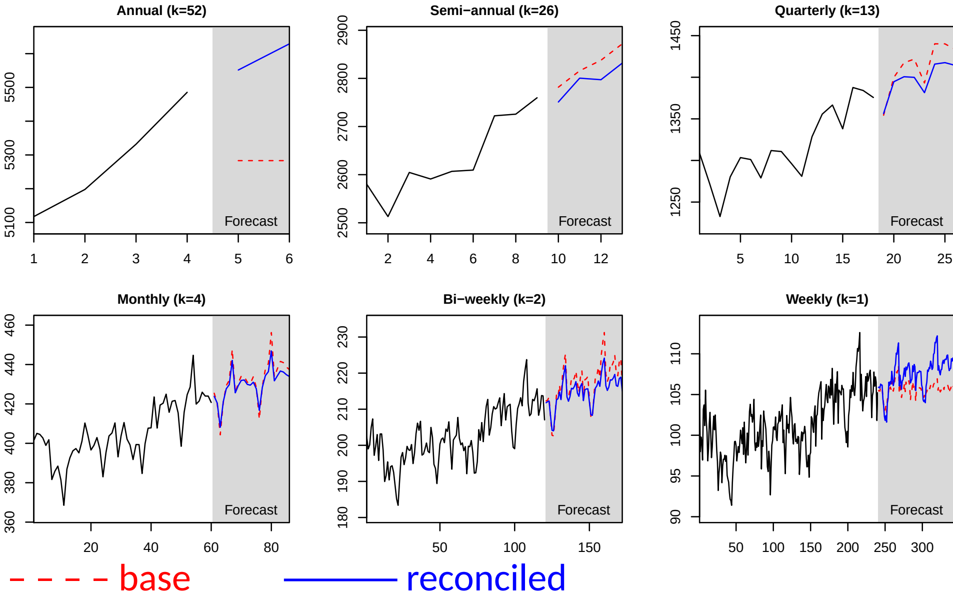
In general

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.

UK Accidents and Emergency Demand



UK Accidents and Emergency Demand

1. Type 1 Departments — Major A&E
2. Type 2 Departments — Single Specialty
3. Type 3 Departments — Other A&E/Minor Injury
4. Total Attendances
5. Type 1 Departments — Major A&E $>$ 4 hrs
6. Type 2 Departments — Single Specialty $>$ 4 hrs
7. Type 3 Departments — Other A&E/Minor Injury $>$ 4 hrs
8. Total Attendances $>$ 4 hrs
9. Emergency Admissions via Type 1 A&E
10. Total Emergency Admissions via A&E
11. Other Emergency Admissions (i.e., not via A&E)
12. Total Emergency Admissions
13. Number of patients spending $>$ 4 hrs from decision to admission

UK Accidents and Emergency Demand

- **Minimum training set**: all data except the last year
- Base forecasts using `auto.arima()`.
- Mean Absolute Scaled Errors for 1, 4 and 13 weeks ahead using a rolling origin.

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

UK Accidents and Emergency Demand

- **Minimum training set**: all data except the last year
- Base forecasts using `auto.arima()`.
- Mean Absolute Scaled Errors for 1, 4 and 13 weeks ahead using a rolling origin.

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

UK Accidents and Emergency Demand

- **Minimum training set**: all data except the last year
- Base forecasts using `auto.arima()`.
- Mean Absolute Scaled Errors for 1, 4 and 13 weeks ahead using a rolling origin.

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

UK Accidents and Emergency Demand

- **Minimum training set**: all data except the last year
- Base forecasts using `auto.arima()`.
- Mean Absolute Scaled Errors for 1, 4 and 13 weeks ahead using a rolling origin.

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

UK Accidents and Emergency Demand

- **Minimum training set**: all data except the last year
- Base forecasts using `auto.arima()`.
- Mean Absolute Scaled Errors for 1, 4 and 13 weeks ahead using a rolling origin.

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

thief package for R

thief: Temporal HIErarchical Forecasting

Install from CRAN

```
install.packages("thief")
```

Usage

```
library(thief)
thief(y)
```

thief package for R

thief: Temporal HIErarchical Forecasting

Install from CRAN

```
install.packages("thief")
```

Usage

```
library(thief)
thief(y)
```

Outline

Lab Session 16