



# Forecasting: principles and practice

Rob J Hyndman

3.3 Hierarchical forecasting

# Outline

**1 Hierarchical and grouped time series**

**2 hts package for R**

**3 Application: Australian tourism**

**4 Optimal forecast reconciliation**

**5 Lab Session 21**

**6 Temporal hierarchies**

**7 Lab session 22**

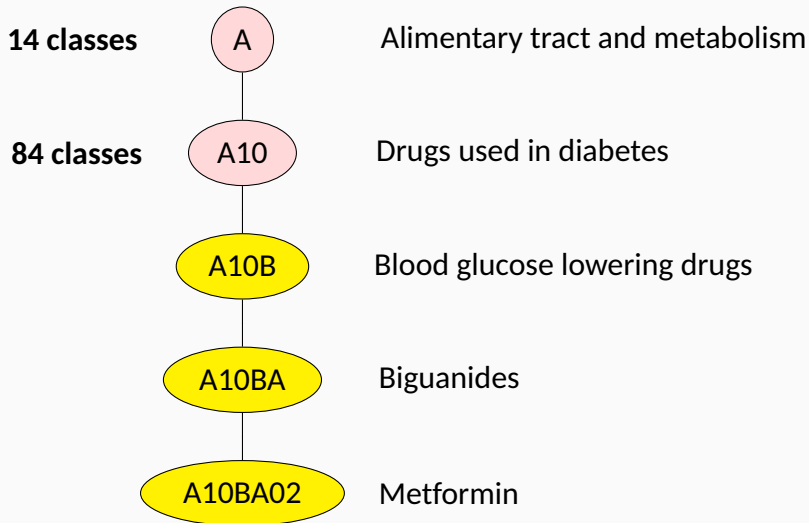
# Forecasting the PBS



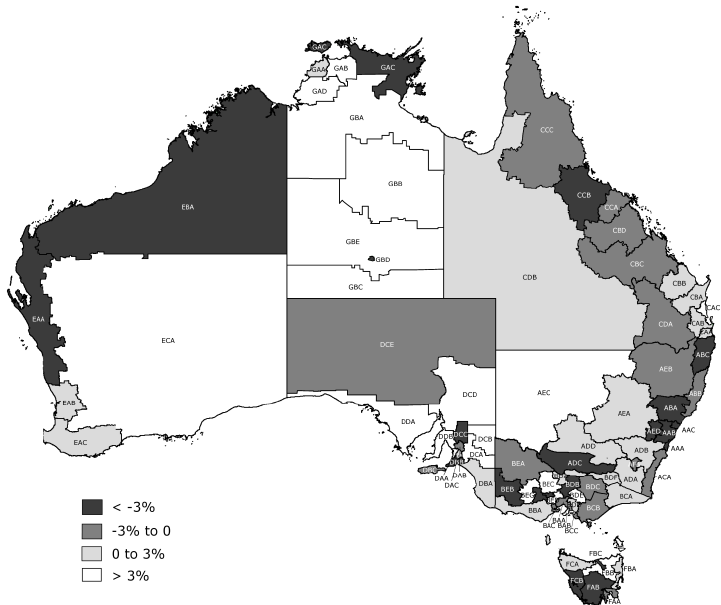
# ATC drug classification

- 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
- V Various

# ATC drug classification



# Australian tourism



# Australian tourism

- 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

# Spectacle sales

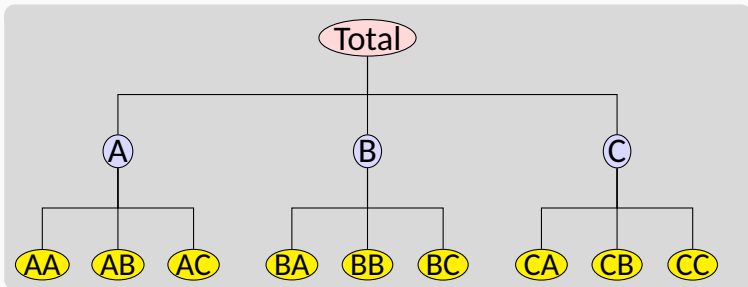


- Monthly UK sales data from 2000 – 2014
- Provided by a large spectacle manufacturer
- Split by brand (26), gender (3), price range (6), materials (4), and stores (600)
- About 1 million bottom-level series



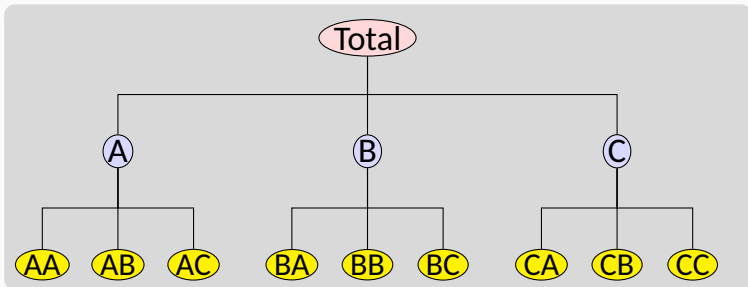
# Hierarchical time series

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



# Hierarchical time series

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

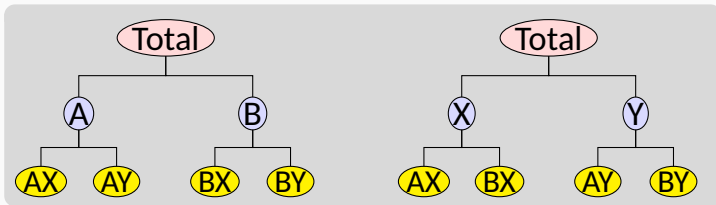


## Examples

- Pharmaceutical sales
- Tourism demand by region and purpose

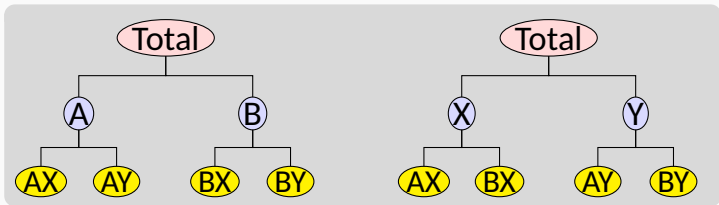
# Grouped time series

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

- Spectacle sales by brand, gender, stores, etc.
- Tourism by state and purpose of travel

# 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 available in the **hts** package in R.

# Outline

- 1 Hierarchical and grouped time series
- 2 hts package for R
- 3 Application: Australian tourism
- 4 Optimal forecast reconciliation
- 5 Lab Session 21
- 6 Temporal hierarchies
- 7 Lab session 22

# hts package for R



## hts: Hierarchical and Grouped Time Series

Methods for analysing and forecasting hierarchical and grouped time series

Version: 5.1.5

Depends: R ( $\geq 3.2.0$ ), forecast ( $\geq 8.1$ )

Imports: SparseM, Matrix, matrixcalc, parallel, utils, methods, graphics, grDevices

LinkingTo: Rcpp ( $\geq 0.11.0$ ), RcppEigen

Suggests: testthat, knitr, rmarkdown

Published: 2018-03-26

Author: Rob J Hyndman, Alan Lee, Earo Wang, Shanika Wickramasuriya

Maintainer: Rob J Hyndman <Earowang@gmail.com>

BugReports: <https://github.com/earowang/hts/issues>

License: GPL ( $\geq 2$ )

URL: <http://pkg.earo.me/hts>



# 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)))
```

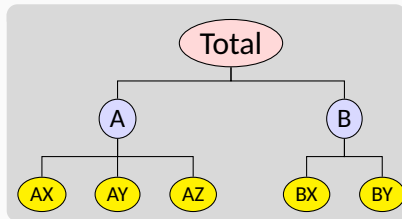
# Example using R

```
library(hts)
```

```
# bts is a matrix containing the bottom level time series
```

```
# nodes describes the hierarchical structure
```

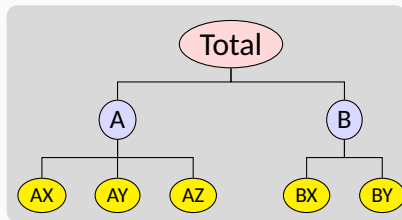
```
y <- hts(bts, nodes=list(2, c(3,2)))
```



# 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)
```

# forecast.gts() function

## Usage

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

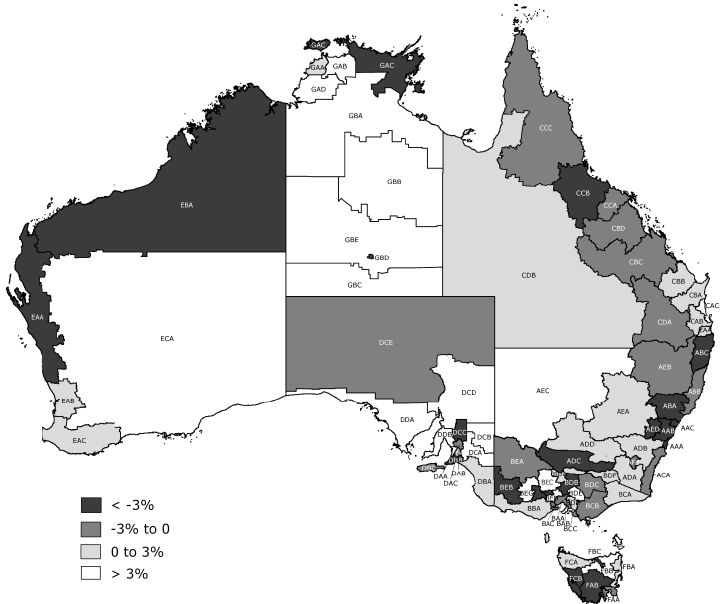
## Arguments

object	Hierarchical time series object of class gts.
h	Forecast horizon
method	Method for distributing forecasts within the hierarchy.
fmethod	Forecasting method to use
weights	Weights used for "optimal combination" method.
covariance	Shrinkage estimator or sample estimator for GLS covariance.
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 to be used

# Outline

- 1 Hierarchical and grouped time series
- 2 hts package for R
- 3 **Application: Australian tourism**
- 4 Optimal forecast reconciliation
- 5 Lab Session 21
- 6 Temporal hierarchies
- 7 Lab session 22

# Australian tourism

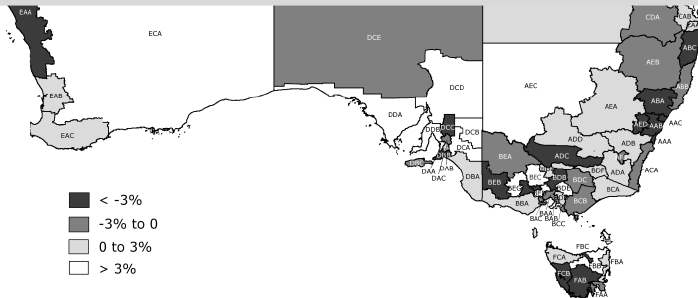


# Australian tourism

## Domestic visitor nights

Quarterly data: 1998 – 2006.

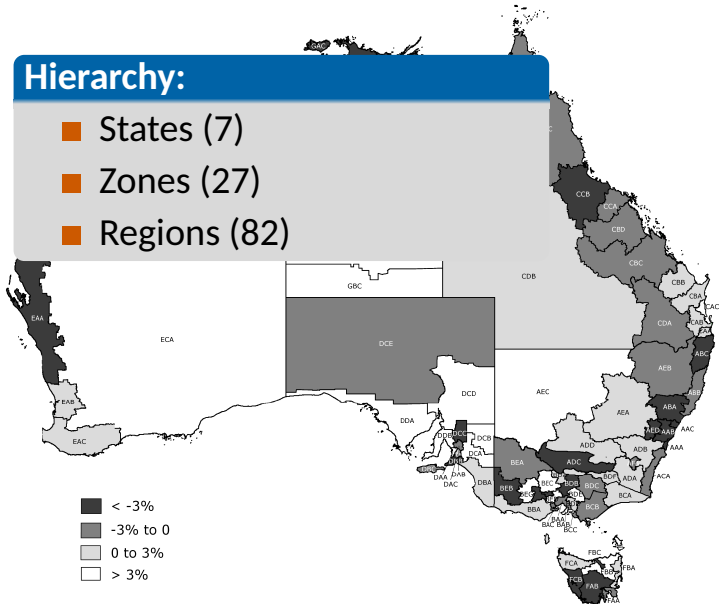
From: *National Visitor Survey*, based on annual interviews of 120,000 Australians aged 15+, collected by Tourism Research Australia.



# Australian tourism

## Hierarchy:

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





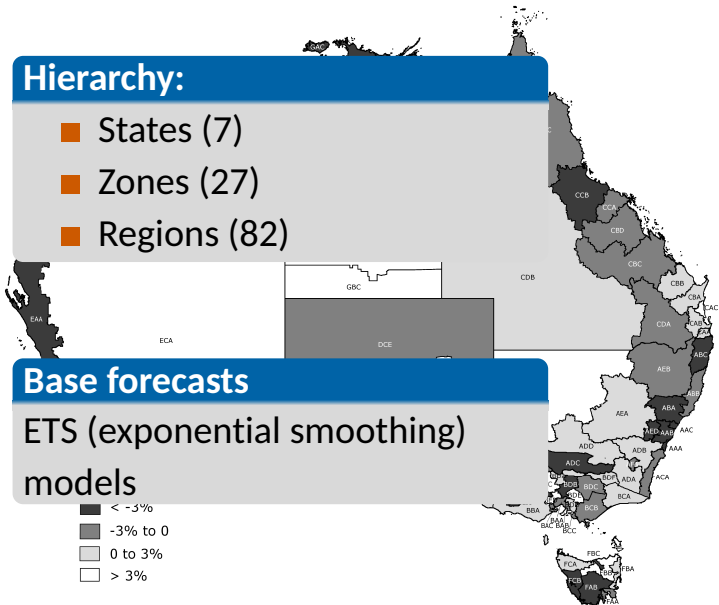
# Australian tourism

## Hierarchy:

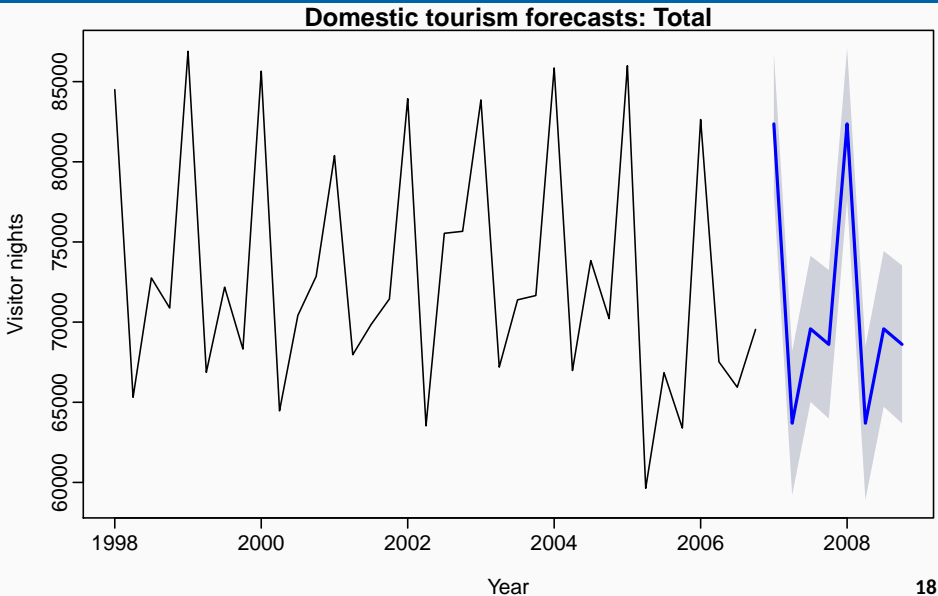
- States (7)
- Zones (27)
- Regions (82)

## Base forecasts

## ETS (exponential smoothing) models

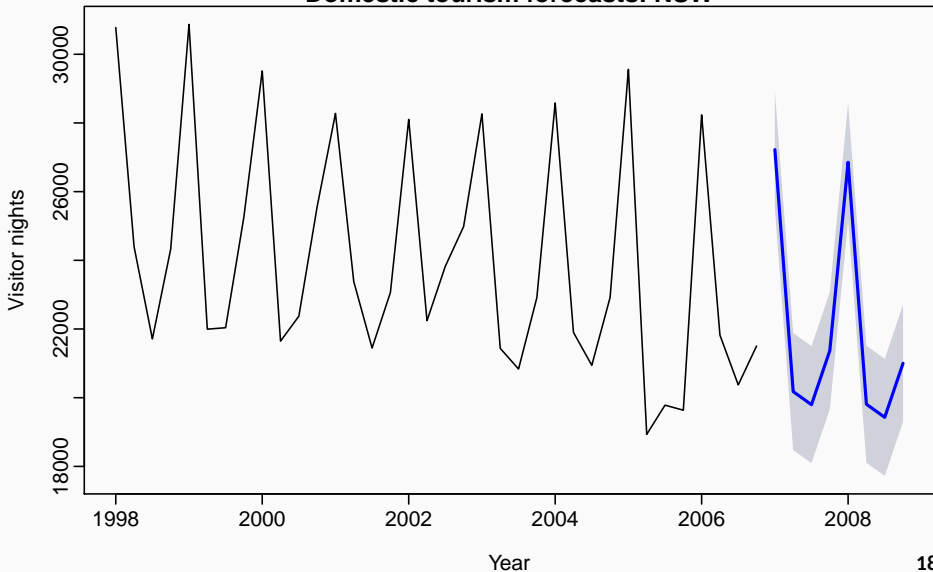


# Base forecasts

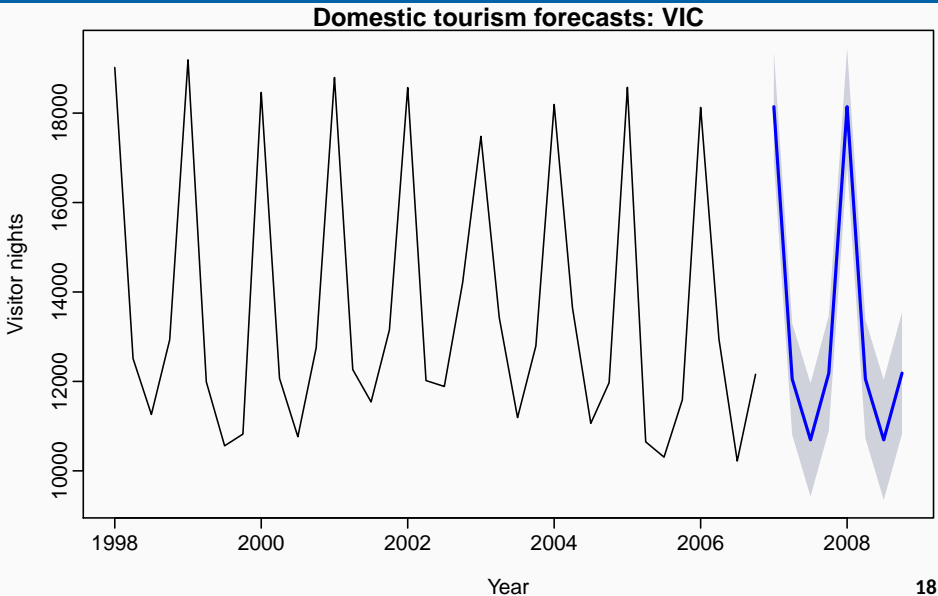


# Base forecasts

**Domestic tourism forecasts: NSW**

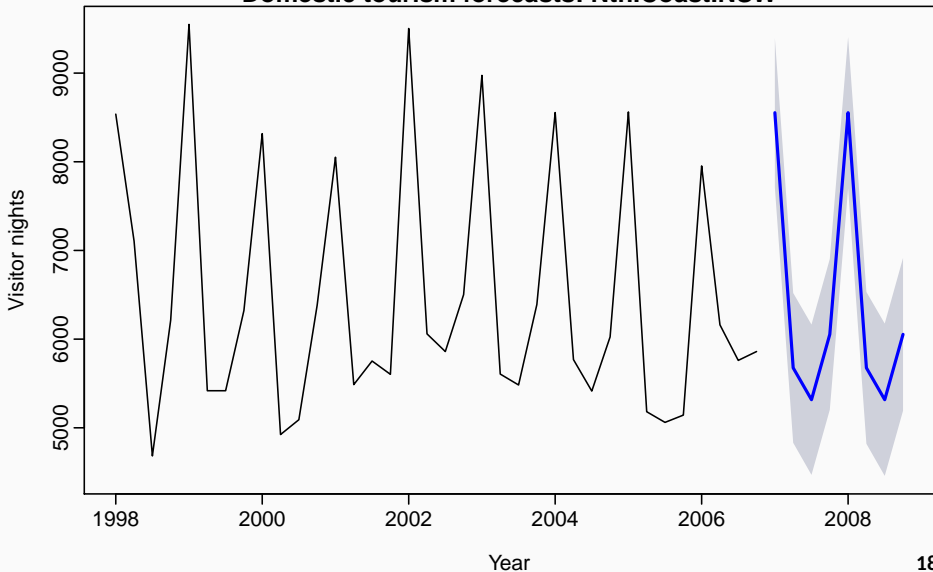


# Base forecasts



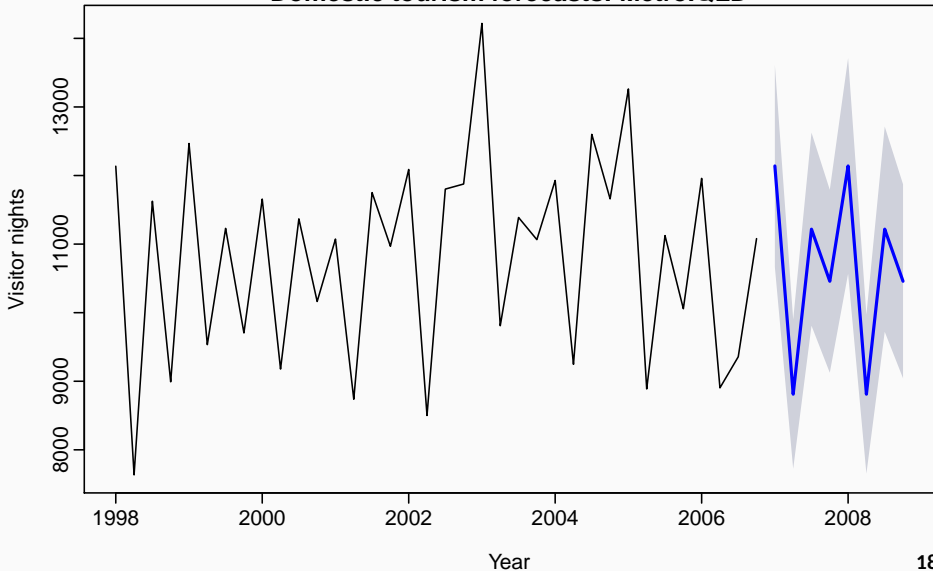
# Base forecasts

**Domestic tourism forecasts: Nth.Coast.NSW**

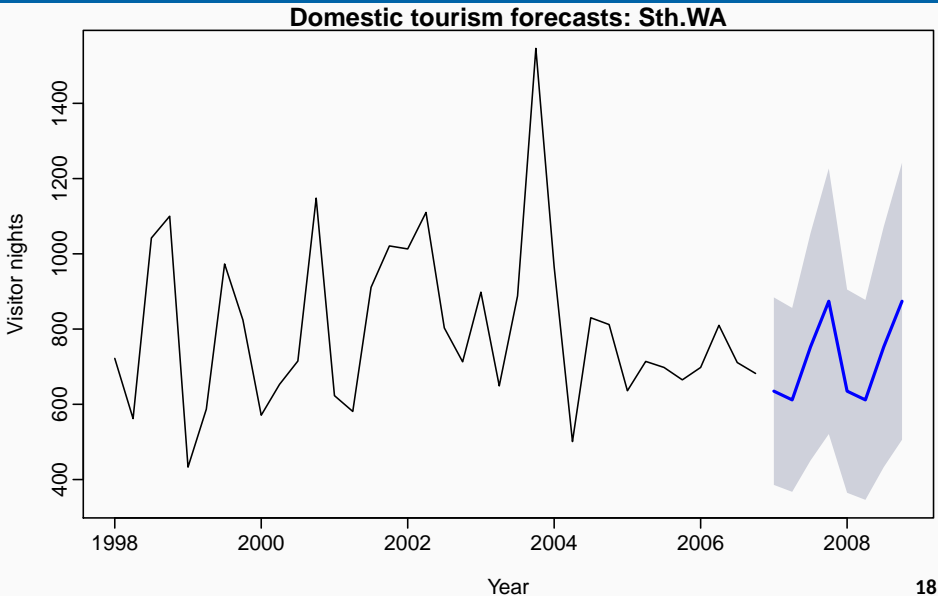


# Base forecasts

**Domestic tourism forecasts: Metro.QLD**

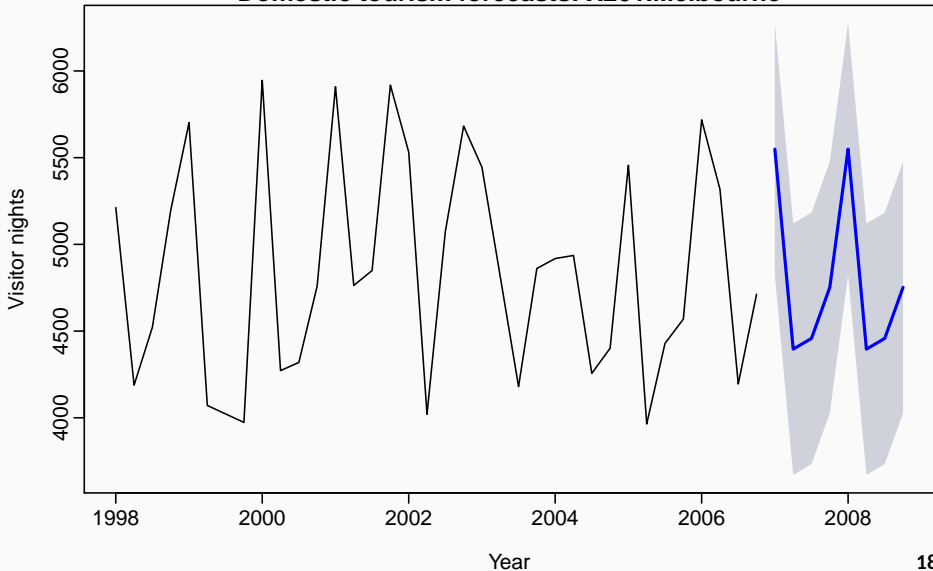


# Base forecasts



# Base forecasts

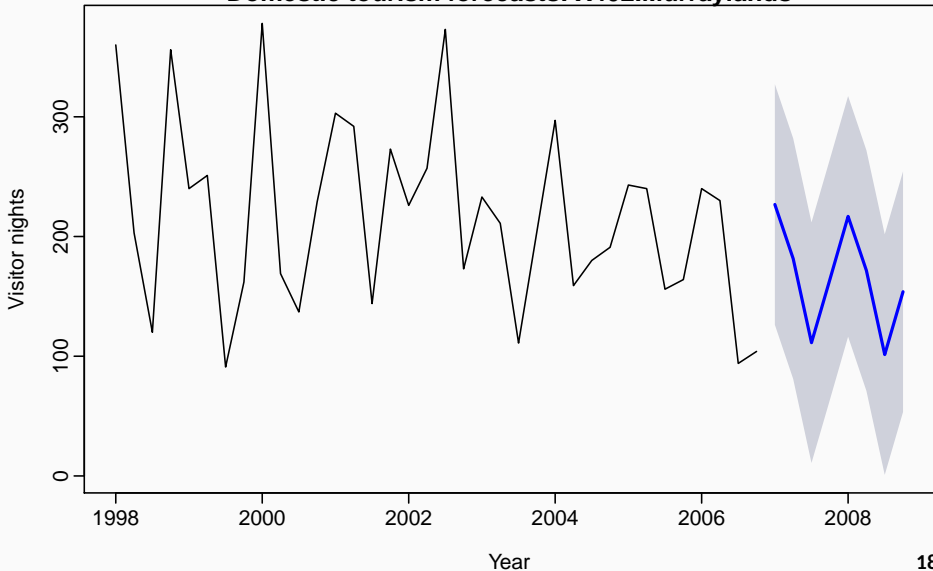
Domestic tourism forecasts: X201.Melbourne





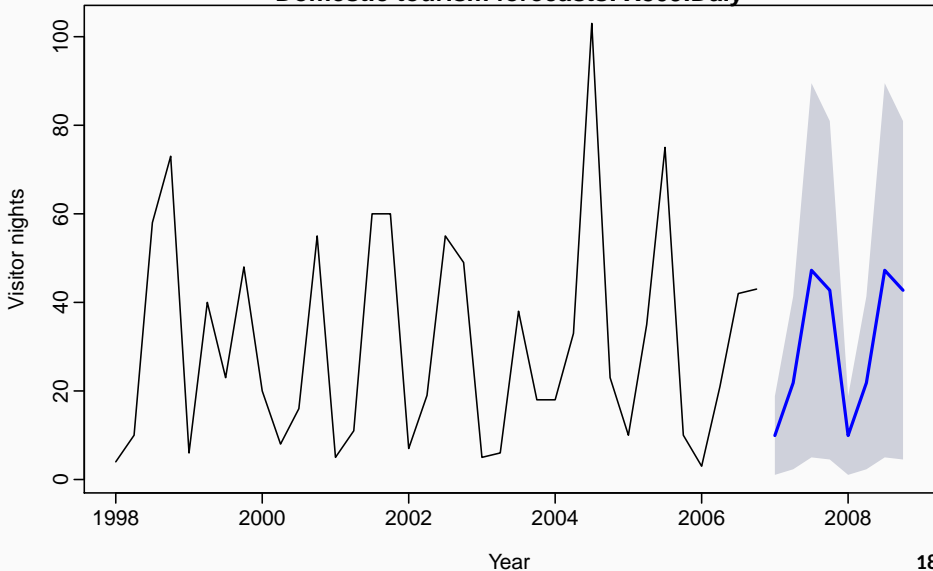
# Base forecasts

Domestic tourism forecasts: X402.Murraylands

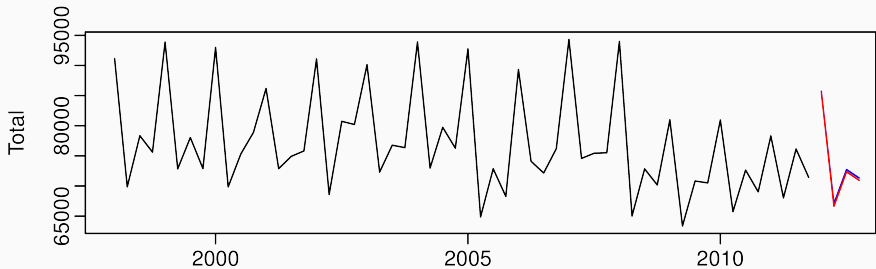


# Base forecasts

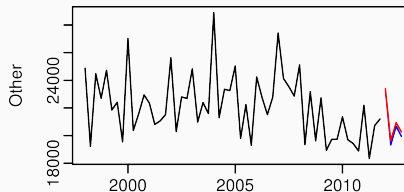
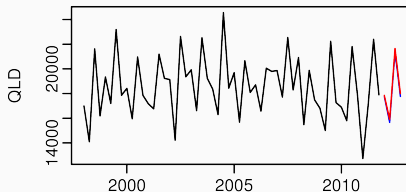
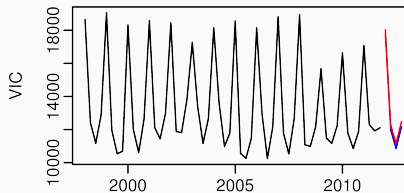
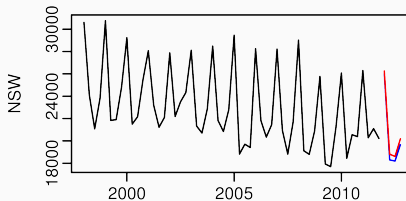
Domestic tourism forecasts: X809.Daly



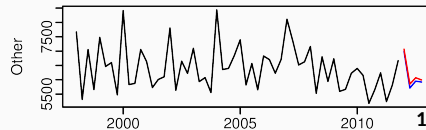
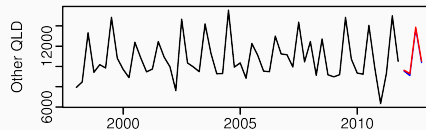
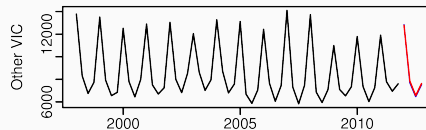
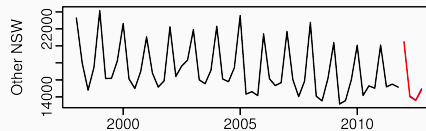
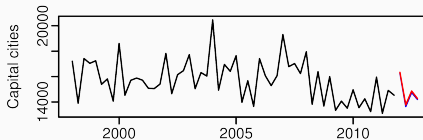
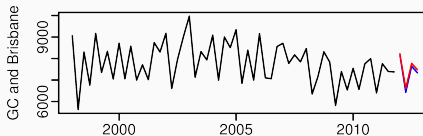
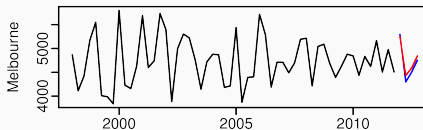
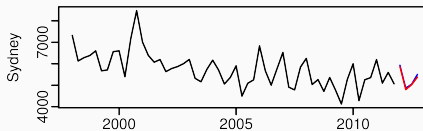
# Reconciled forecasts



# Reconciled forecasts



# Reconciled forecasts



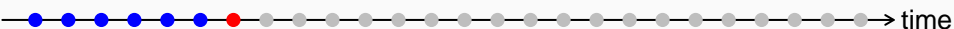
# Forecast evaluation

- Select models using all observations;
- Re-estimate models using first 12 observations and generate 1- to 8-step-ahead forecasts;
- Increase sample size one observation at a time, re-estimate models, generate forecasts until the end of the sample;
- In total 24 1-step-ahead, 23 2-steps-ahead, up to 17 8-steps-ahead for forecast evaluation.

# Forecast evaluation

Training sets

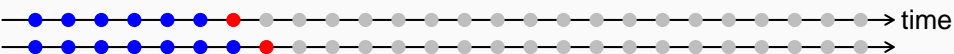
Test sets  $h = 1$



# Forecast evaluation

Training sets

Test sets  $h = 1$

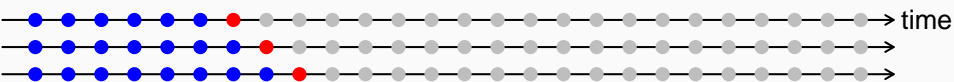




# Forecast evaluation

Training sets

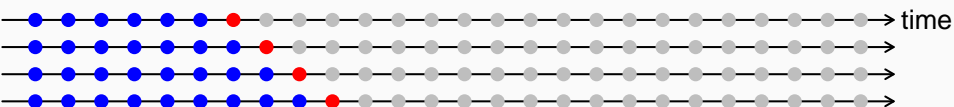
Test sets  $h = 1$



# Forecast evaluation

Training sets

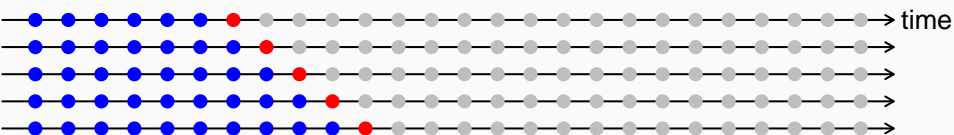
Test sets  $h = 1$



# Forecast evaluation

Training sets

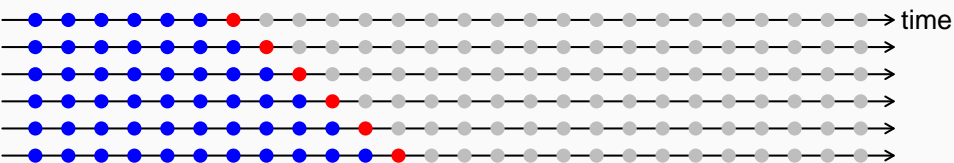
Test sets  $h = 1$



# Forecast evaluation

Training sets

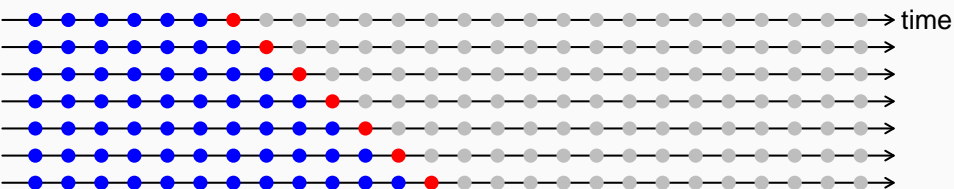
Test sets  $h = 1$



# Forecast evaluation

Training sets

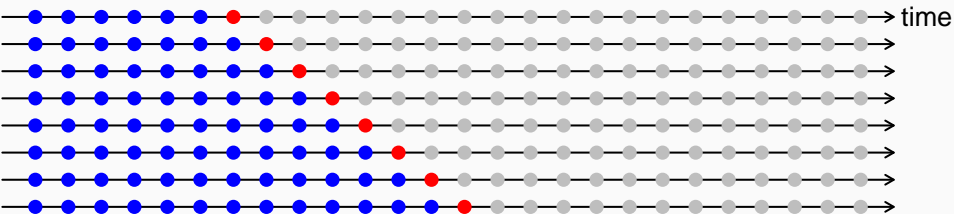
Test sets  $h = 1$



# Forecast evaluation

Training sets

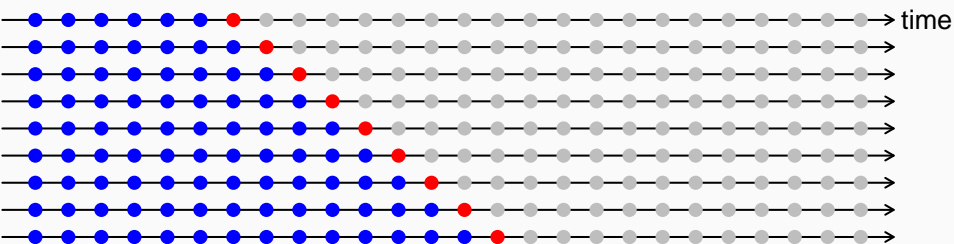
Test sets  $h = 1$



# Forecast evaluation

Training sets

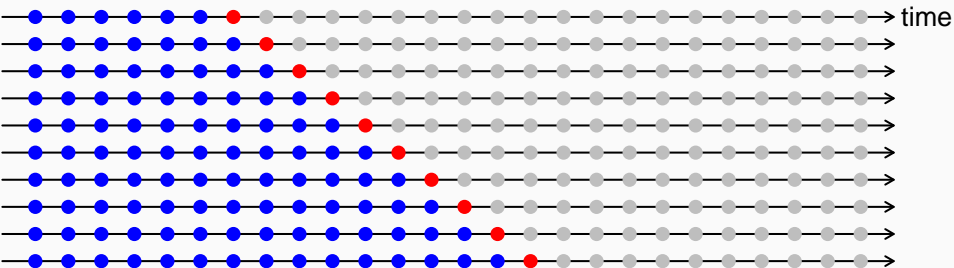
Test sets  $h = 1$



# Forecast evaluation

Training sets

Test sets  $h = 1$

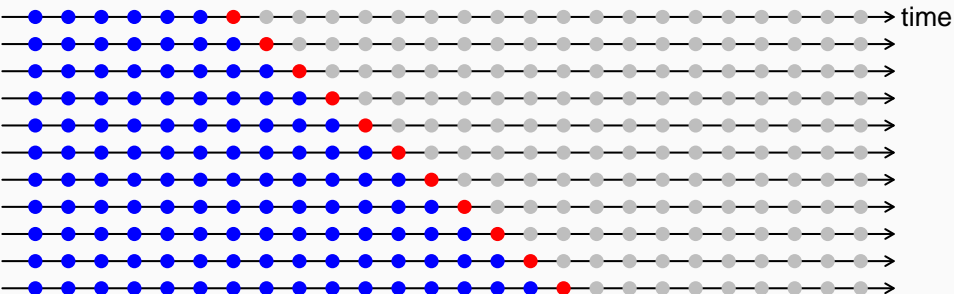




# Forecast evaluation

Training sets

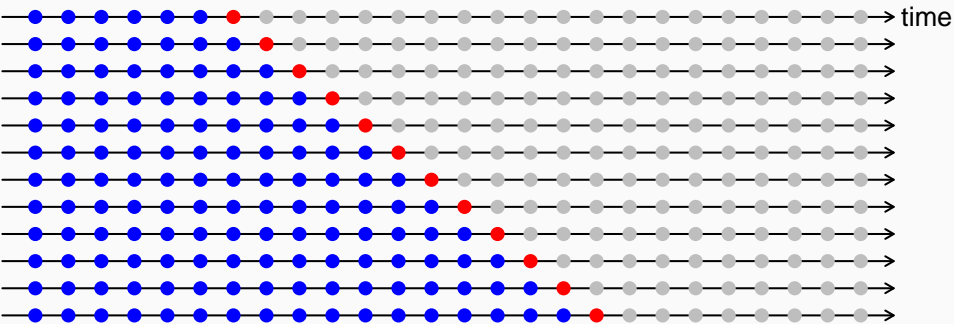
Test sets  $h = 1$



# Forecast evaluation

Training sets

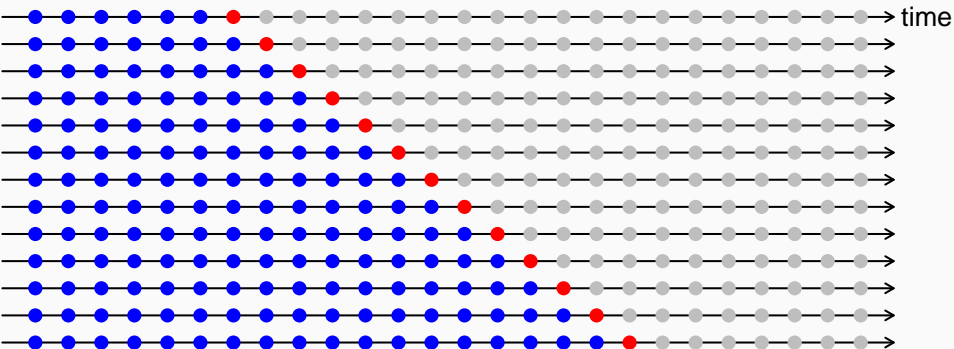
Test sets  $h = 1$



# Forecast evaluation

Training sets

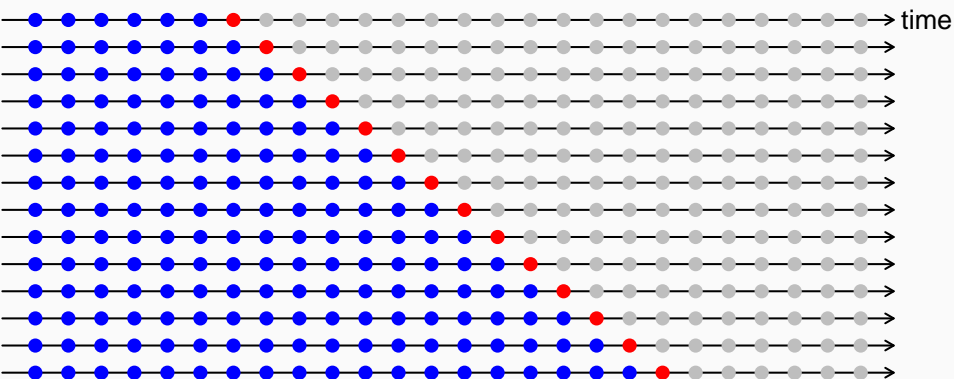
Test sets  $h = 1$



# Forecast evaluation

Training sets

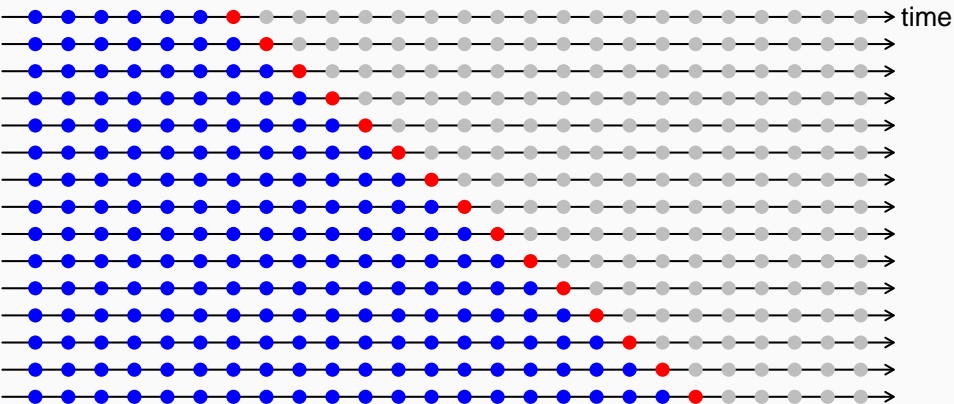
Test sets  $h = 1$



# Forecast evaluation

Training sets

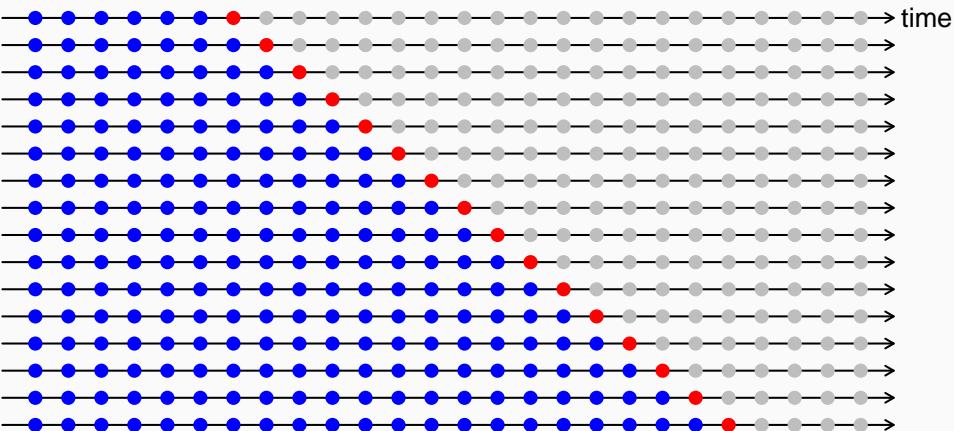
Test sets  $h = 1$



# Forecast evaluation

Training sets

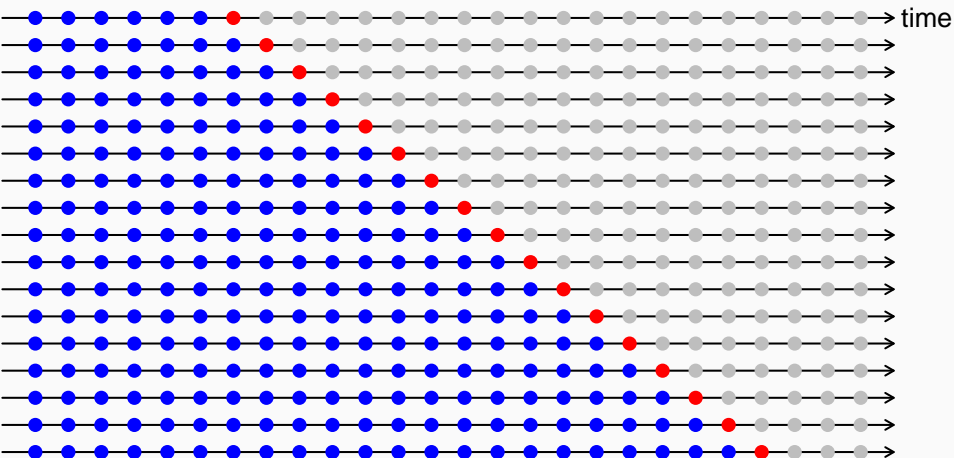
Test sets  $h = 1$



# Forecast evaluation

Training sets

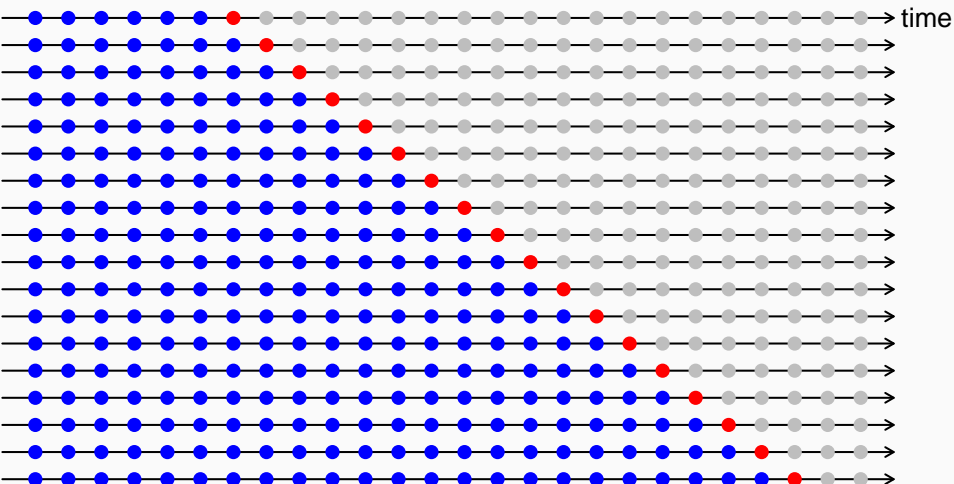
Test sets  $h = 1$



# Forecast evaluation

Training sets

Test sets  $h = 1$

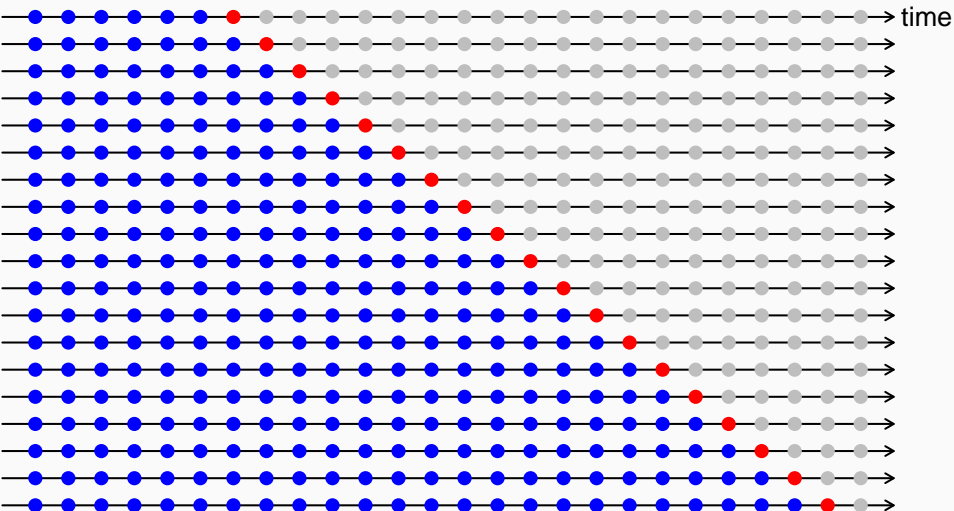




# Forecast evaluation

Training sets

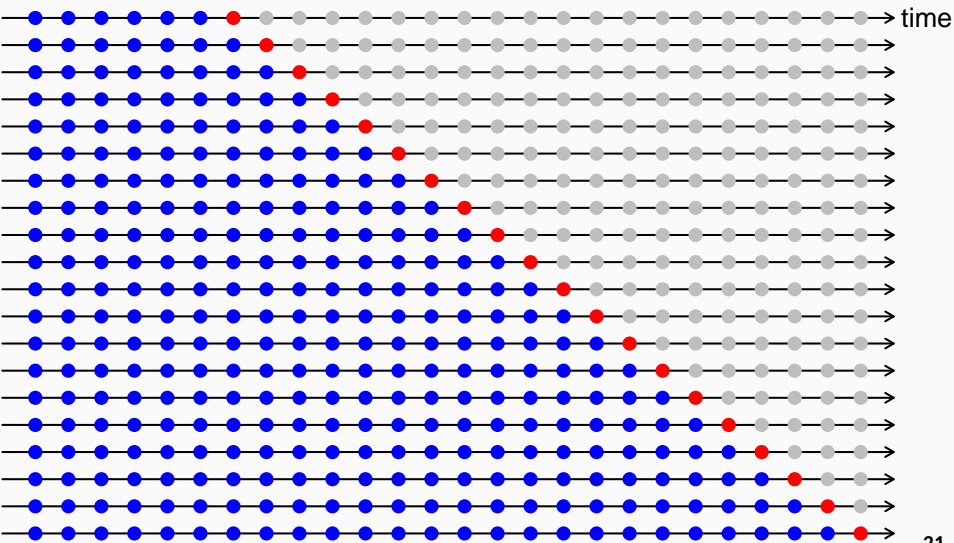
Test sets  $h = 1$



# Forecast evaluation

Training sets

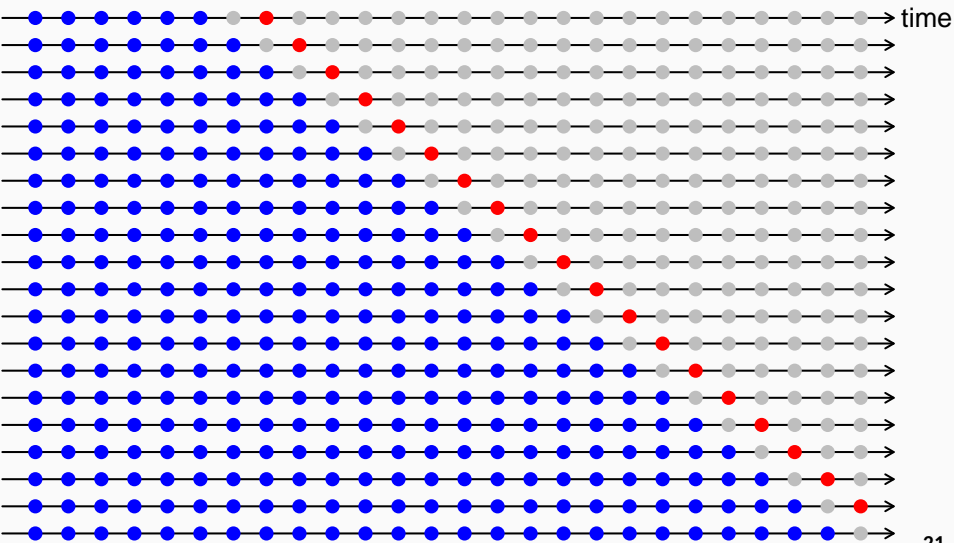
Test sets  $h = 1$



# Forecast evaluation

Training sets

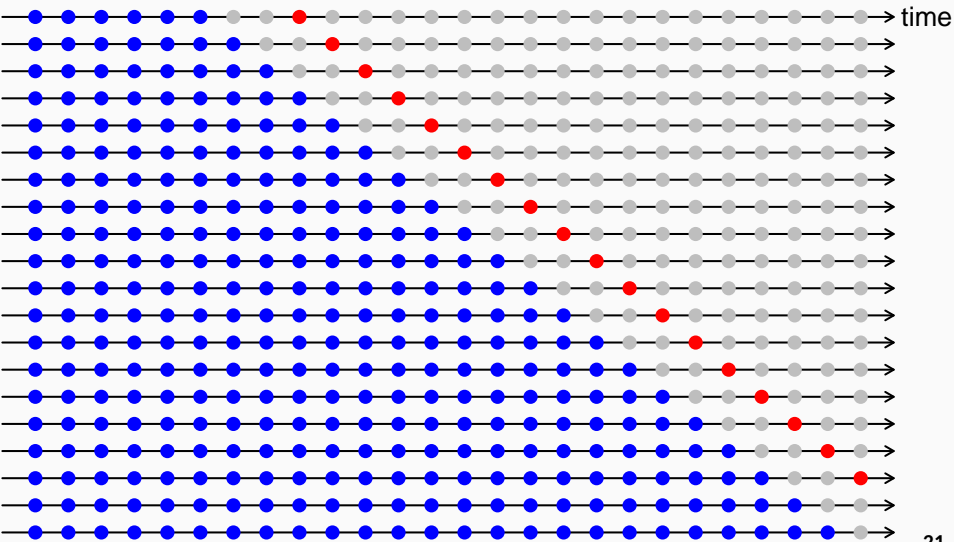
Test sets  $h = 2$



# Forecast evaluation

Training sets

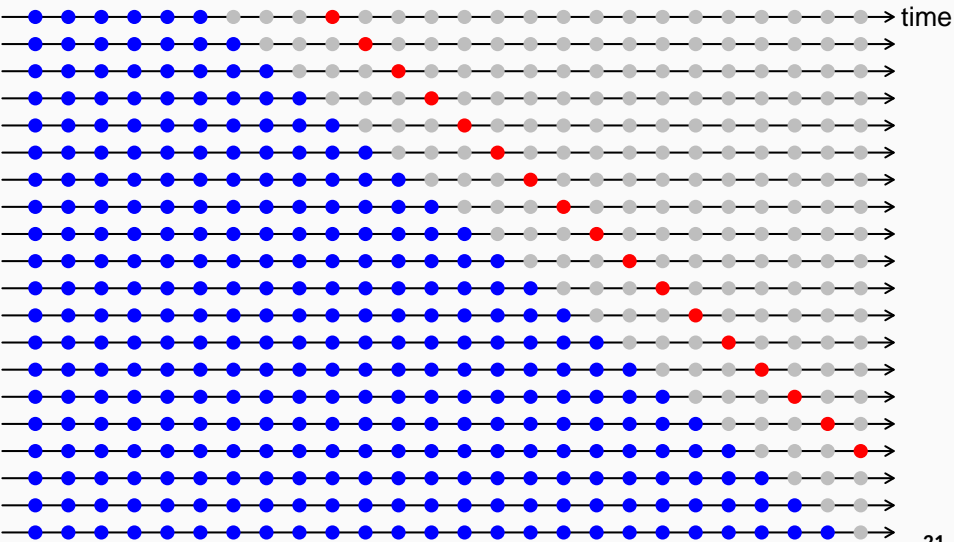
Test sets  $h = 3$



# Forecast evaluation

Training sets

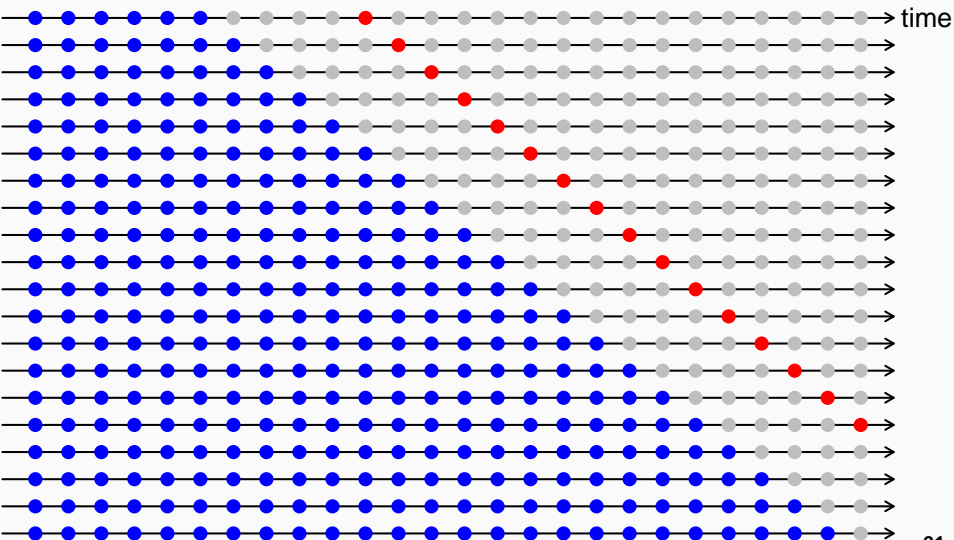
Test sets  $h = 4$



# Forecast evaluation

Training sets

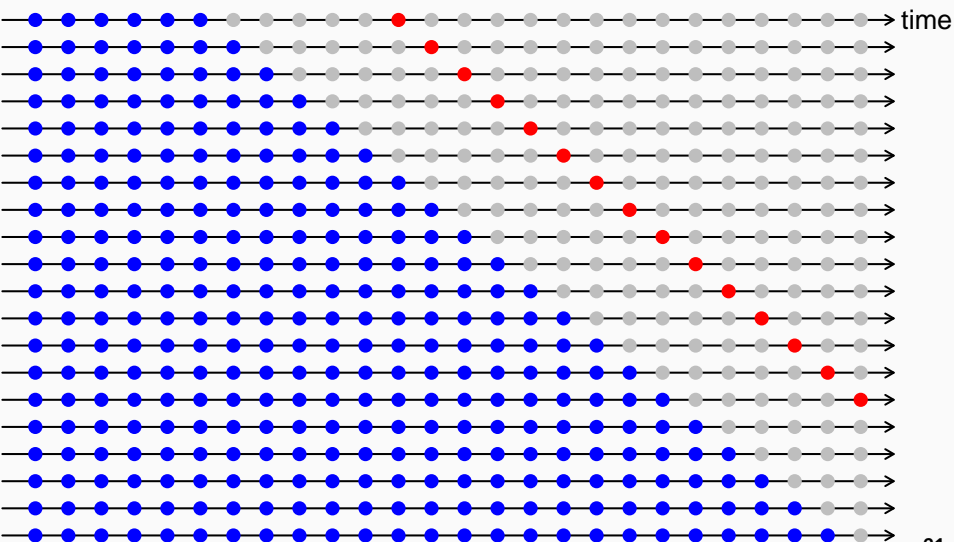
Test sets  $h = 5$



# Forecast evaluation

Training sets

Test sets  $h = 6$



# Hierarchy: states, zones, regions

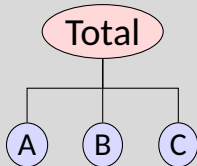
RMSE	Forecast horizon						Ave
	$h = 1$	$h = 2$	$h = 3$	$h = 4$	$h = 5$	$h = 6$	
Australia							
Base	1762.04	1770.29	1766.02	1818.82	1705.35	1721.17	1757.28
Bottom	1736.92	1742.69	1722.79	1752.74	1666.73	1687.43	1718.22
WLS	1705.21	1715.87	1703.75	1729.56	1627.79	1661.24	1690.57
GLS	1704.64	1715.60	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
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
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



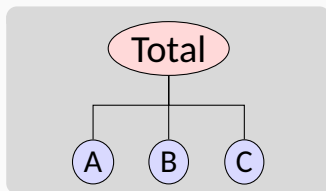
# Outline

- 1 Hierarchical and grouped time series
- 2 hts package for R
- 3 Application: Australian tourism
- 4 Optimal forecast reconciliation
- 5 Lab Session 21
- 6 Temporal hierarchies
- 7 Lab session 22

# Hierarchical time series



# Hierarchical time series

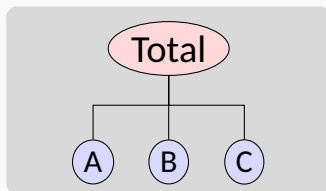


$y_t$  : observed aggregate of all series at time  $t$ .

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

$b_t$  : vector of all series at bottom level in time  $t$ .

# Hierarchical time series



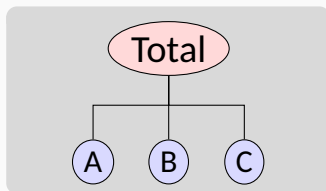
$y_t$  : observed aggregate of all series at time  $t$ .

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

$b_t$  : vector of all series at bottom level in time  $t$ .

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

# Hierarchical time series



$y_t$  : observed aggregate of all series at time  $t$ .

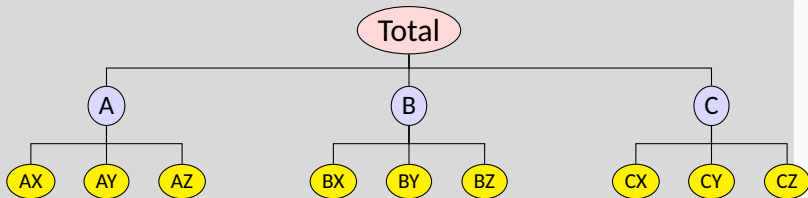
$y_{X,t}$  : observation on series  $X$  at time  $t$ .

$b_t$  : vector of all series at bottom level in time  $t$ .

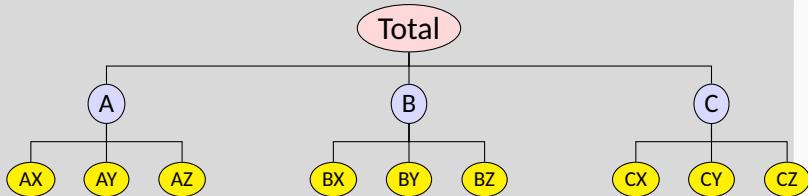
$$\mathbf{y}_t = \begin{pmatrix} y_t \\ y_{A,t} \\ y_{B,t} \\ y_{C,t} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathbf{S}} \underbrace{\begin{pmatrix} y_{A,t} \\ y_{B,t} \\ y_{C,t} \end{pmatrix}}_{\mathbf{b}_t}$$

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

# Hierarchical time series

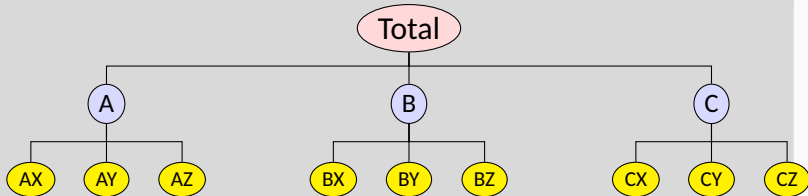


# Hierarchical time series



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

# Hierarchical time series

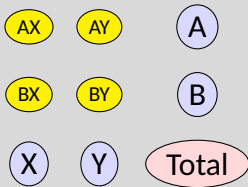


$$\mathbf{y}_t = \begin{pmatrix} y_t \\ y_{A,t} \\ y_{B,t} \\ y_{C,t} \\ y_{AX,t} \\ y_{AY,t} \\ y_{AZ,t} \\ y_{BX,t} \\ y_{BY,t} \\ y_{BZ,t} \\ y_{CX,t} \\ y_{CY,t} \\ y_{CZ,t} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0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

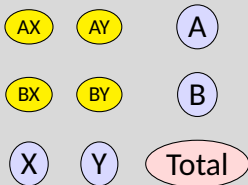
$$\mathbf{y}_t = \mathbf{S} \mathbf{b}_t$$



# Grouped time series

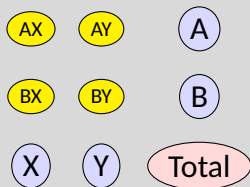


# Grouped time series



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

# Grouped time series



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

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

# Hierarchical and grouped time series

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

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

where

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

## Forecasting notation

Let  $\hat{\mathbf{y}}_n(h)$  be vector of initial  $h$ -step forecasts, made at time  $n$ , stacked in same order as  $\mathbf{y}_t$ .

## Forecasting notation

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

# Forecasting notation

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

Reconciled forecasts must be of the form:

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

for some matrix  $\mathbf{P}$ .

# Forecasting notation

Let  $\hat{\mathbf{y}}_n(h)$  be vector of initial  $h$ -step forecasts, made at time  $n$ , stacked in same order as  $\mathbf{y}_t$ .

(In general, they will not “add up”.)

Reconciled forecasts must be of the form:

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

for some matrix  $\mathbf{P}$ .

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



# Optimal combination forecasts

## Main result

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

# Optimal combination forecasts

## Main result

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

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

**Problem:**  $\Sigma_h$  hard to estimate, especially for  $h > 1$ .

## Solutions:

- Ignore  $\Sigma_h$  (OLS)
- Assume  $\Sigma_h$  diagonal (WLS) [Default in hts]
- Try to estimate  $\Sigma_h$  (GLS)

# Features

- Covariates can be included in initial forecasts.
- Adjustments can be made to initial forecasts at any level.
- Very simple and flexible method. Can work with *any* hierarchical or grouped time series.
- Conceptually easy to implement: regression of base forecasts on structure matrix.

# Outline

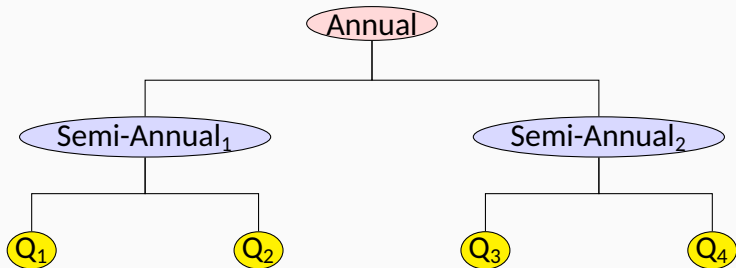
- 1 Hierarchical and grouped time series
- 2 hts package for R
- 3 Application: Australian tourism
- 4 Optimal forecast reconciliation
- 5 Lab Session 21
- 6 Temporal hierarchies
- 7 Lab session 22

# Lab Session 21

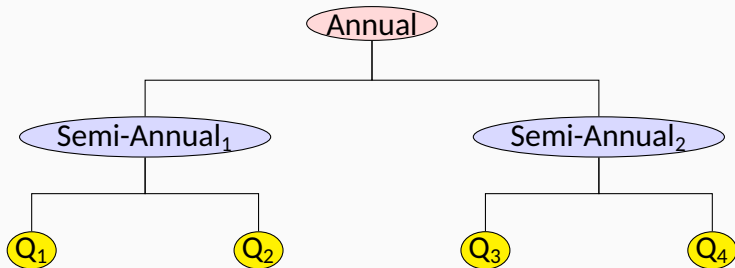
# Outline

- 1 Hierarchical and grouped time series
- 2 hts package for R
- 3 Application: Australian tourism
- 4 Optimal forecast reconciliation
- 5 Lab Session 21
- 6 Temporal hierarchies
- 7 Lab session 22

# Temporal hierarchies



# Temporal hierarchies

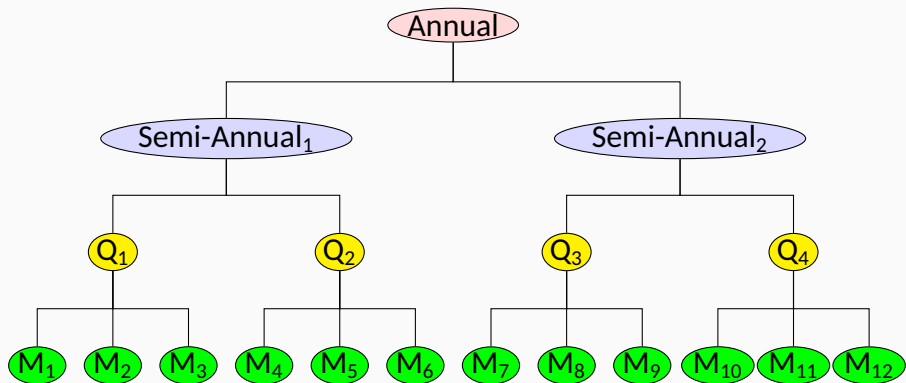


## Basic idea:

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

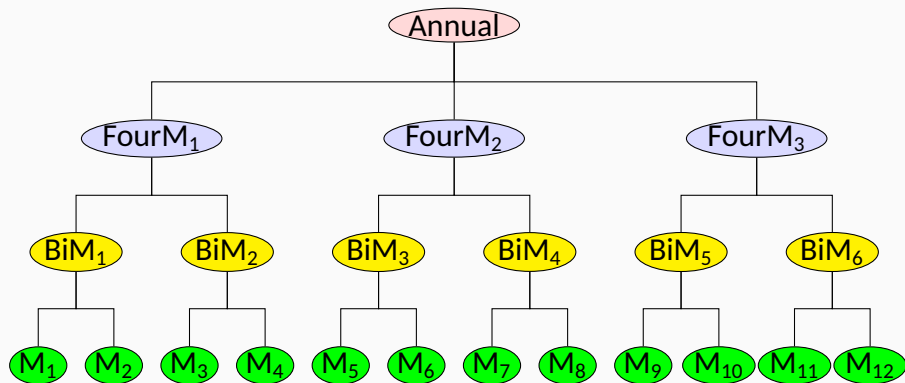


# Monthly series



■  $k = 2, 4, 12$  nodes

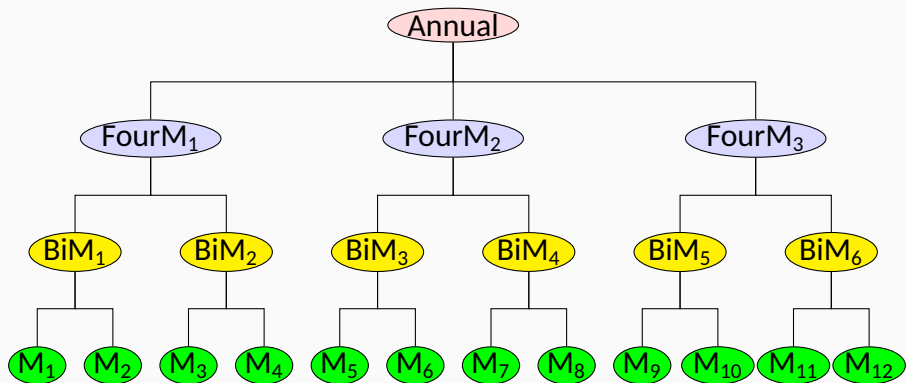
# Monthly series



■  $k = 2, 4, 12$  nodes

■  $k = 3, 6, 12$  nodes

# Monthly series



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

# Monthly data

$$\underbrace{\begin{pmatrix} A \\ \text{Semi}A_1 \\ \text{Semi}A_2 \\ \text{Four}M_1 \\ \text{Four}M_2 \\ \text{Four}M_3 \\ Q_1 \\ \vdots \\ Q_4 \\ \text{Bi}M_1 \\ \vdots \\ \text{Bi}M_6 \\ M_1 \\ \vdots \\ M_{12} \end{pmatrix}}_{(28 \times 1)} = \underbrace{\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & \vdots & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & \vdots & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}}_{\substack{I_{12} \\ S}} \underbrace{\begin{pmatrix} M_1 \\ M_2 \\ M_3 \\ M_4 \\ M_5 \\ M_6 \\ M_7 \\ M_8 \\ M_9 \\ M_{10} \\ M_{11} \\ M_{12} \end{pmatrix}}_{b_t}$$

## In general

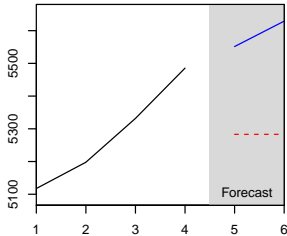
For a time series  $y_1, \dots, y_T$ , observed at frequency  $m$ , we generate aggregate series

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

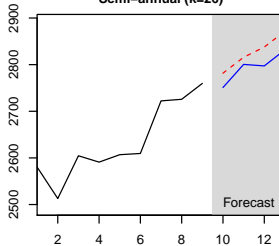
- $k \in F(m) = \{\text{factors of } m\}$ .
- A single unique hierarchy is only possible when there are no coprime pairs in  $F(m)$ .
- $M_k = m/k$  is seasonal period of aggregated series.

# UK Accidents and Emergency Demand}

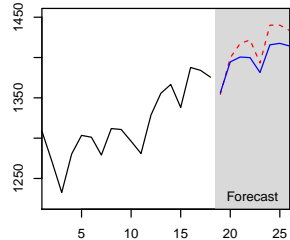
Annual ( $k=52$ )



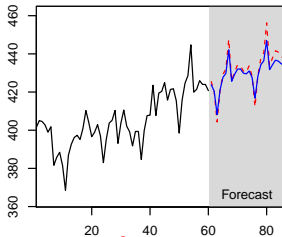
Semi-annual ( $k=26$ )



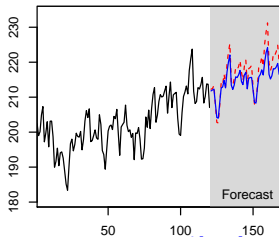
Quarterly ( $k=13$ )



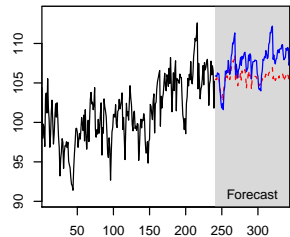
Monthly ( $k=4$ )



Bi-weekly ( $k=2$ )



Weekly ( $k=1$ )



--- base

— reconciled

# 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

# 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.



# 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	<i>h</i>	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: Temporal HIErarchical Forecasting

## thief: Temporal HIErarchical Forecasting

### Install from CRAN

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

### Usage

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

# Outline

- 1 Hierarchical and grouped time series
- 2 hts package for R
- 3 Application: Australian tourism
- 4 Optimal forecast reconciliation
- 5 Lab Session 21
- 6 Temporal hierarchies
- 7 Lab session 22

# Lab Session 22