

Feasts and fables: Time series analysis using R



Rob J Hyndman

robjhyndman.com/rba2023

Outline

- 1 tsibble: Time series data
- 2 feasts: Data visualization
- 3 feasts: Time series features
- 4 fable: Forecasting
- 5 fable: Forecast reconciliation

Outline

- 1 tsibble: Time series data
- 2 feasts: Data visualization
- 3 feasts: Time series features
- 4 fable: Forecasting
- 5 fable: Forecast reconciliation

Tidyverts packages

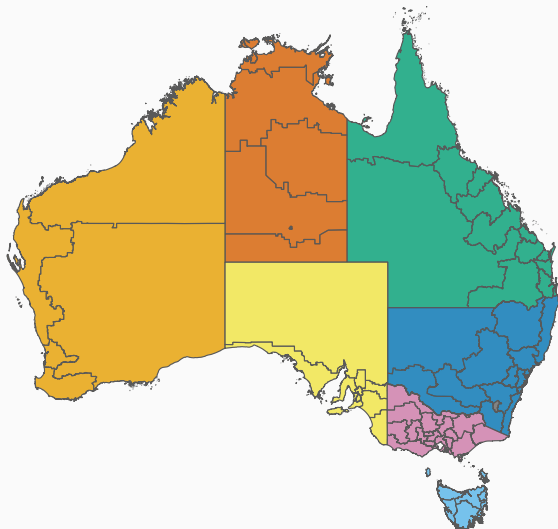
tidyverts.org

```
"r library(fpp3) "
```

```
-- Attaching packages ----- fpp3 0.5.0 --  
✓ tibble      3.1.8      ✓ tsibble      1.1.2  
✓ dplyr       1.1.0      ✓ tsibbledata  0.4.1  
✓ tidyr       1.3.0      ✓ feasts       0.3.0  
✓ lubridate   1.9.2      ✓ fable        0.3.2  
✓ ggplot2     3.4.1      ✓ fabletools   0.3.2  
  
-- Conflicts ----- fpp3_conflicts --  
X lubridate::date()      masks base::date()  
X dplyr::filter()        masks stats::filter()  
X tsibble::intersect()   masks base::intersect()  
X tsibble::interval()    masks lubridate::interval()  
X dplyr::lag()            masks stats::lag()  
X tsibble::setdiff()     masks base::setdiff()  
X tsibble::union()       masks base::union()
```



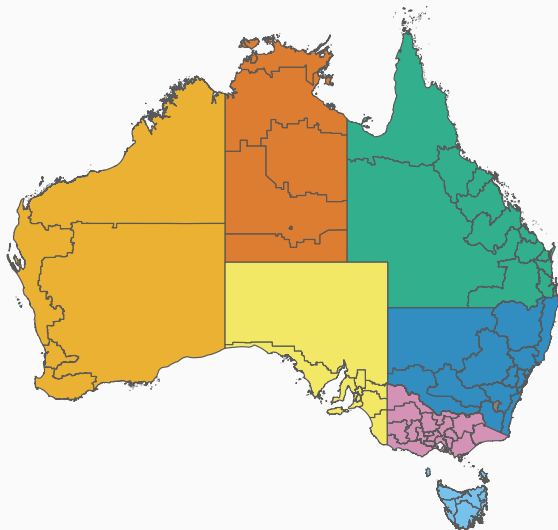
Australian tourism regions



State

	Queensland
	Northern Territory
	Western Australia
	South Australia
	New South Wales
	Australian Capital Territory
	Victoria
	Tasmania

Australian tourism regions



- Quarterly data on visitor nights: 1998 – 2017
- From *National Visitor Survey*, interviews of 120,000 Australians aged 15+.
- Geographical hierarchy split by
 - ▶ 8 states and territories
 - ▶ 76 regions
- Purpose:
 - ▶ Holidays
 - ▶ Business
 - ▶ Visiting friends & relatives
 - ▶ Other

tsibble objects

tourism

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:      Region, State, Purpose [304]
##   Quarter Region State Purpose Trips
##   <qtr> <chr>    <fct> <chr>    <dbl>
## 1 1998 Q1 Adelaide SA      Business 135.
## 2 1998 Q2 Adelaide SA      Business 110.
## 3 1998 Q3 Adelaide SA      Business 166.
## 4 1998 Q4 Adelaide SA      Business 127.
## 5 1999 Q1 Adelaide SA      Business 137.
## 6 1999 Q2 Adelaide SA      Business 200.
## 7 1999 Q3 Adelaide SA      Business 169.
## 8 1999 Q4 Adelaide SA      Business 134.
## 9 2000 Q1 Adelaide SA      Business 154.
## 10 2000 Q2 Adelaide SA      Business 169.
```

Domestic visitor
nights in thousands
by state/region and
purpose.

tsibble objects

tourism

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:      Region, State, Purpose [304]
##   Quarter Region  State Purpose  Trips
##   Index      <chr>   <fct> <chr>    <dbl>
## 1 1998 Q1 Adelaide SA      Business 135.
## 2 1998 Q2 Adelaide SA      Business 110.
## 3 1998 Q3 Adelaide SA      Business 166.
## 4 1998 Q4 Adelaide SA      Business 127.
## 5 1999 Q1 Adelaide SA      Business 137.
## 6 1999 Q2 Adelaide SA      Business 200.
## 7 1999 Q3 Adelaide SA      Business 169.
## 8 1999 Q4 Adelaide SA      Business 134.
## 9 2000 Q1 Adelaide SA      Business 154.
## 10 2000 Q2 Adelaide SA      Business 169.
```

Domestic visitor
nights in thousands
by state/region and
purpose.

tsibble objects

tourism

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:      Region, State, Purpose [304]
##   Quarter Region State Purpose Trips
##   Index    Keys      <dbl>
## 1 1998 Q1 Adelaide SA      Business 135.
## 2 1998 Q2 Adelaide SA      Business 110.
## 3 1998 Q3 Adelaide SA      Business 166.
## 4 1998 Q4 Adelaide SA      Business 127.
## 5 1999 Q1 Adelaide SA      Business 137.
## 6 1999 Q2 Adelaide SA      Business 200.
## 7 1999 Q3 Adelaide SA      Business 169.
## 8 1999 Q4 Adelaide SA      Business 134.
## 9 2000 Q1 Adelaide SA      Business 154.
## 10 2000 Q2 Adelaide SA      Business 169.
```

Domestic visitor
nights in thousands
by state/region and
purpose.

tsibble objects

tourism

```
## # A tsibble: 24,320 x 5 [1Q]
```

```
## # Key:      Region, State, Purpose [304]
```

```
##   Quarter Region State Purpose Trips
```

```
##   Index      Keys      Measure
```

```
## 1 1998 Q1 Adelaide SA      Business 135.
```

```
## 2 1998 Q2 Adelaide SA      Business 110.
```

```
## 3 1998 Q3 Adelaide SA      Business 166.
```

```
## 4 1998 Q4 Adelaide SA      Business 127.
```

```
## 5 1999 Q1 Adelaide SA      Business 137.
```

```
## 6 1999 Q2 Adelaide SA      Business 200.
```

```
## 7 1999 Q3 Adelaide SA      Business 169.
```

```
## 8 1999 Q4 Adelaide SA      Business 134.
```

```
## 9 2000 Q1 Adelaide SA      Business 154.
```

```
## 10 2000 Q2 Adelaide SA      Business 169.
```

Domestic visitor
nights in thousands
by state/region and
purpose.

tsibble objects

- A `tsibble` allows storage and manipulation of multiple time series in R.
- It contains:
 - ▶ An index: time information about the observation
 - ▶ Measured variable(s): numbers of interest
 - ▶ Key variable(s): optional unique identifiers for each series
- It works with tidyverse functions.

Outline

- 1 tsibble: Time series data
- 2 feasts: Data visualization
- 3 feasts: Time series features
- 4 fable: Forecasting
- 5 fable: Forecast reconciliation

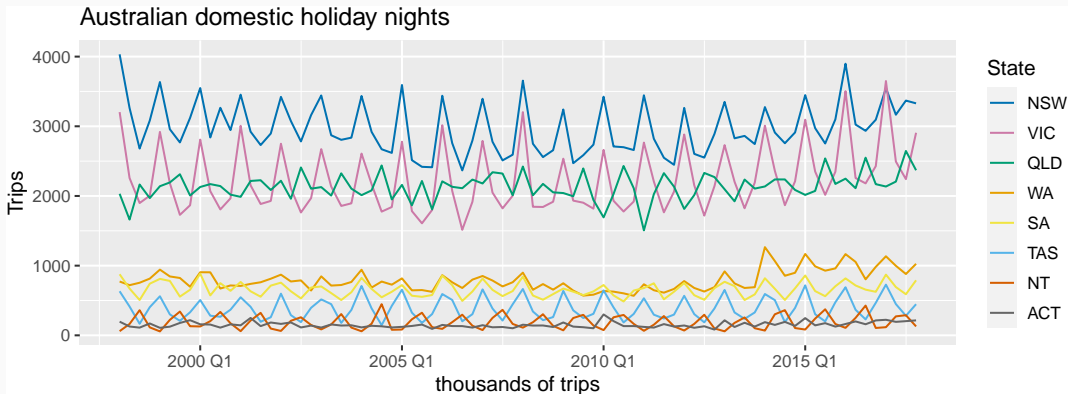
Australian holidays

```
holidays <- tourism |>
  filter(Purpose == "Holiday") |>
  group_by(State) |>
  summarise(Trips = sum(Trips))
```

```
## # A tsibble: 640 x 3 [1Q]
## # Key:      State [8]
##   State Quarter Trips
##   <fct>    <qtr> <dbl>
## 1 NSW      1998 Q1 4033.
## 2 NSW      1998 Q2 3262.
## 3 NSW      1998 Q3 2681.
## 4 NSW      1998 Q4 3083.
## 5 NSW      1999 Q1 3635.
## 6 NSW      1999 Q2 2958.
## 7 NSW      1999 Q3 2768.
## 8 NSW      1999 Q4 3121.
## 9 NSW      2000 Q1 3548.
```

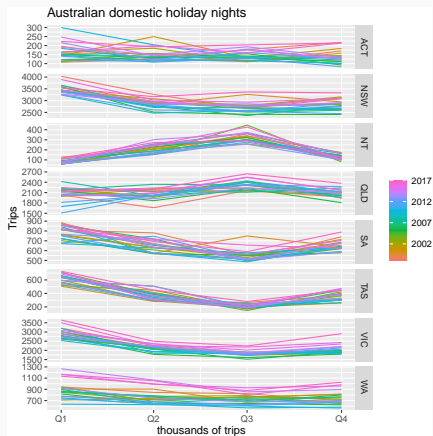
Australian holidays

```
holidays |> autoplot(Trips) +  
  labs(x = "thousands of trips", title = "Australian domestic holiday nights")
```



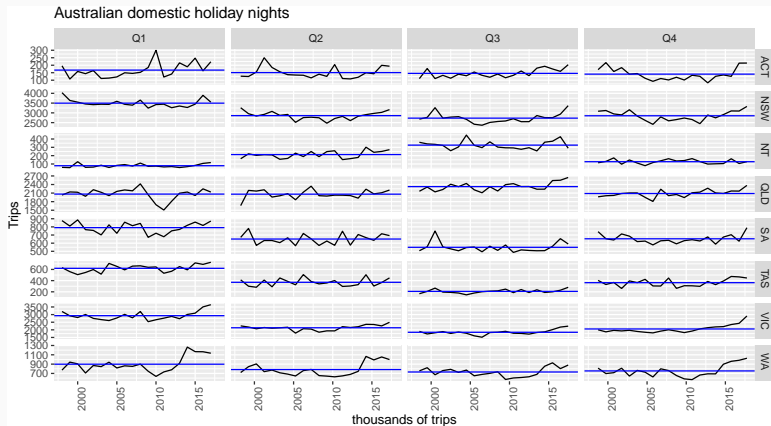
Seasonal plots

```
holidays |> gg_season(Trips) +  
  labs(x = "thousands of trips", title = "Australian domestic holiday nights")
```



Seasonal subseries plots

```
holidays |> gg_subseries(Trips) +  
  labs(x = "thousands of trips", title = "Australian domestic holiday nights")
```



Outline

- 1 tsibble: Time series data
- 2 feasts: Data visualization
- 3 feasts: Time series features
- 4 fable: Forecasting
- 5 fable: Forecast reconciliation

Strength of seasonality and trend

STL decomposition

$$y_t = T_t + S_t + R_t$$

Seasonal strength

$$\max \left(0, 1 - \frac{\text{Var}(R_t)}{\text{Var}(S_t + R_t)} \right)$$

Trend strength

$$\max \left(0, 1 - \frac{\text{Var}(R_t)}{\text{Var}(T_t + R_t)} \right)$$

Feature extraction and statistics

```
tourism |> features(Trips, feat_stl)
```

```
## # A tibble: 304 x 12
##   Region      State Purpose trend~1 seaso~2 seaso~3 seaso~4 spiki~5 linea~6
##   <chr>      <fct> <chr>    <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1 Adelaide    SA    Busine~  0.464  0.407      3      1 1.58e+2 -5.31
## 2 Adelaide    SA    Holiday  0.554  0.619      1      2 9.17e+0 49.0
## 3 Adelaide    SA    Other    0.746  0.202      2      1 2.10e+0 95.1
## 4 Adelaide    SA    Visiti~  0.435  0.452      1      3 5.61e+1 34.6
## 5 Adelaide Hil~ SA    Busine~  0.464  0.179      3      0 1.03e-1 0.968
## 6 Adelaide Hil~ SA    Holiday  0.528  0.296      2      1 1.77e-1 10.5
## 7 Adelaide Hil~ SA    Other    0.593  0.404      2      2 4.44e-4 4.28
## 8 Adelaide Hil~ SA    Visiti~  0.488  0.254      0      3 6.50e+0 34.2
## 9 Alice Springs NT    Busine~  0.534  0.251      0      1 1.69e-1 23.8
## 10 Alice Springs NT    Holiday  0.381  0.832      3      1 7.39e-1 -19.6
## # ... with 294 more rows, 3 more variables: curvature <dbl>,
## #   stl_e_acf1 <dbl>, stl_e_acf10 <dbl>, and abbreviated variable names
## #   1: trend_strength, 2: seasonal_strength_year, 3: seasonal_peak_year,
## #   4: seasonal_trough_year, 5: spikiness, 6: linearity
```

Feature extraction and statistics

```
tourism |>  
  features(Trips, feat_stl) |>  
  ggplot(aes(x = trend_strength, y = seasonal_strength_year, col = Purpose)) +  
  geom_point() +  
  facet_wrap(vars(State))
```



Feature extraction and statistics

```
tourism |>
  features(Trips, feat_stl) |>
  ggplot(aes(x = trend_strength, y = seasonal_strength_year, col = Purpose)) +
  geom_point() +
  facet_wrap(vars(State))
```



- Holidays more seasonal than other travel.
- WA has strongest trends.

Feature extraction and statistics

Find the most seasonal time series:

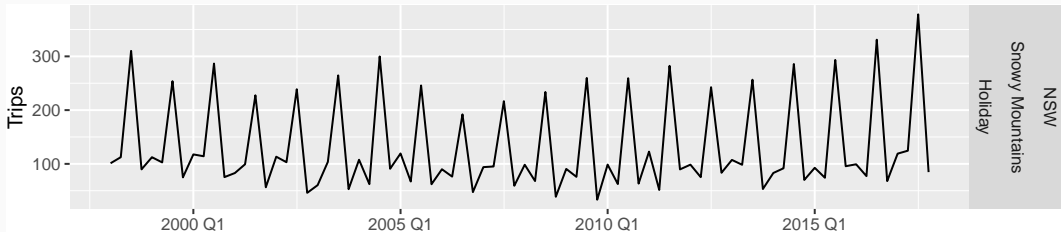
```
most_seasonal <- tourism |>  
  features(Trips, feat_stl) |>  
  filter(seasonal_strength_year == max(seasonal_strength_year))
```

Feature extraction and statistics

Find the most seasonal time series:

```
most_seasonal <- tourism |>  
  features(Trips, feat_stl) |>  
  filter(seasonal_strength_year == max(seasonal_strength_year))
```

```
tourism |>  
  right_join(most_seasonal, by = c("State", "Region", "Purpose")) |>  
  ggplot(aes(x = Quarter, y = Trips)) +  
  geom_line() +  
  facet_grid(vars(State, Region, Purpose))
```



Feature extraction and statistics

```
tourism_features <- tourism |>  
  features(Trips, feature_set(pkgs = "feasts"))
```

All features from the feasts package

```
## # A tibble: 304 x 51  
##   Region      State Purpose trend~1 seaso~2 seaso~3 seaso~4 spiki~5 linea~6  
##   <chr>      <fct> <chr>    <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  
## 1 Adelaide  SA      Busine~  0.464  0.407      3      1 1.58e+2 -5.31  
## 2 Adelaide  SA      Holiday 0.554  0.619      1      2 9.17e+0 49.0  
## 3 Adelaide  SA      Other    0.746  0.202      2      1 2.10e+0 95.1  
## 4 Adelaide  SA      Visiti~  0.435  0.452      1      3 5.61e+1 34.6  
## 5 Adelaide Hil~ SA      Busine~  0.464  0.179      3      0 1.03e-1 0.968  
## 6 Adelaide Hil~ SA      Holiday 0.528  0.296      2      1 1.77e-1 10.5  
## 7 Adelaide Hil~ SA      Other    0.593  0.404      2      2 4.44e-4 4.28  
## 8 Adelaide Hil~ SA      Visiti~  0.488  0.254      0      3 6.50e+0 34.2  
## 9 Alice Springs NT      Busine~  0.534  0.251      0      1 1.69e-1 23.8  
## 10 Alice Springs NT      Holiday 0.381  0.832      3      1 7.39e-1 -19.6  
## # ... with 294 more rows, 42 more variables: curvature <dbl>,  
## #   stl_e_acf1 <dbl>, stl_e_acf10 <dbl>, acf1 <dbl>, acf10 <dbl>,  
## #   diff1_acf1 <dbl>, diff1_acf10 <dbl>, diff2_acf1 <dbl>, diff2_acf10 <dbl>,  
## #   season_acf1 <dbl>, pacf5 <dbl>, diff1_pacf5 <dbl>, diff2_pacf5 <dbl>,  
## #   season_pacf <dbl>, zero_run_mean <dbl>, nonzero_squared_cv <dbl>,  
## #   zero_start_prop <dbl>, zero_end_prop <dbl>, lambda_guerrero <dbl>,
```


Feature extraction and statistics

```
pcs <- tourism_features |>
  select(-State, -Region, -Purpose) |>
  prcomp(scale = TRUE) |>
  broom::augment(tourism_features)
```

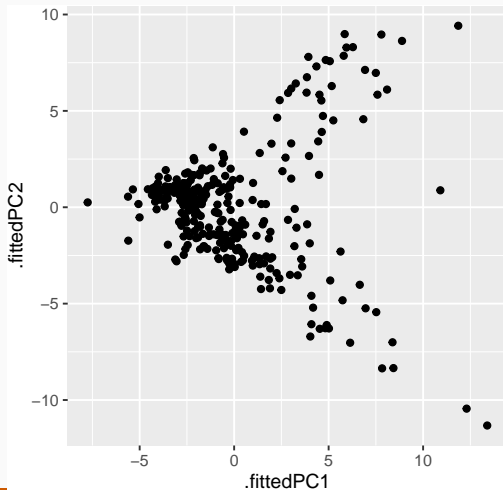
Principal components based on all features from the feasts package

```
## # A tibble: 304 x 100
##   .rownames Region      State Purpose trend~1 seaso~2 seaso~3 seaso~4 spiki~5
##   <chr>      <chr>      <fct> <chr>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1 1          Adelaide SA      Busine~ 0.464    0.407      3      1 1.58e+2
## 2 2          Adelaide SA      Holiday 0.554    0.619      1      2 9.17e+0
## 3 3          Adelaide SA      Other   0.746    0.202      2      1 2.10e+0
## 4 4          Adelaide SA      Visiti~ 0.435    0.452      1      3 5.61e+1
## 5 5          Adelaide H~ SA      Busine~ 0.464    0.179      3      0 1.03e-1
## 6 6          Adelaide H~ SA      Holiday 0.528    0.296      2      1 1.77e-1
## 7 7          Adelaide H~ SA      Other   0.593    0.404      2      2 4.44e-4
## 8 8          Adelaide H~ SA      Visiti~ 0.488    0.254      0      3 6.50e+0
## 9 9          Alice Spri~ NT      Busine~ 0.534    0.251      0      1 1.69e-1
## 10 10         Alice Spri~ NT      Holiday 0.381    0.832      3      1 7.39e-1
## # ... with 294 more rows, 91 more variables: linearity <dbl>,
## #   curvature <dbl>, stl_e_acf1 <dbl>, stl_e_acf10 <dbl>, acf1 <dbl>,
## #   acf10 <dbl>, diff1_acf1 <dbl>, diff1_acf10 <dbl>, diff2_acf1 <dbl>,
## #   diff2_acf10 <dbl>, season_acf1 <dbl>, pacf5 <dbl>, diff1_pacf5 <dbl>,
```

Feature extraction and statistics

```
pcs |> ggplot(aes(x=.fittedPC1, y=.fittedPC2)) +  
  geom_point() + theme(aspect.ratio=1)
```

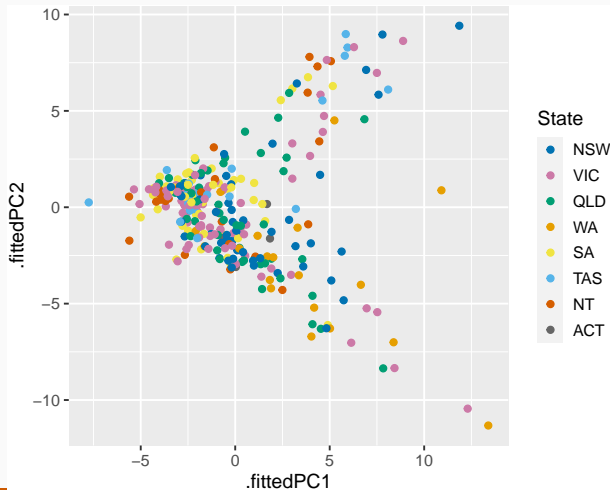
Principal components
based on all features
from the feasts
package



Feature extraction and statistics

```
pcs |> ggplot(aes(x=.fittedPC1, y=.fittedPC2, col=State)) +  
  geom_point() + theme(aspect.ratio=1)
```

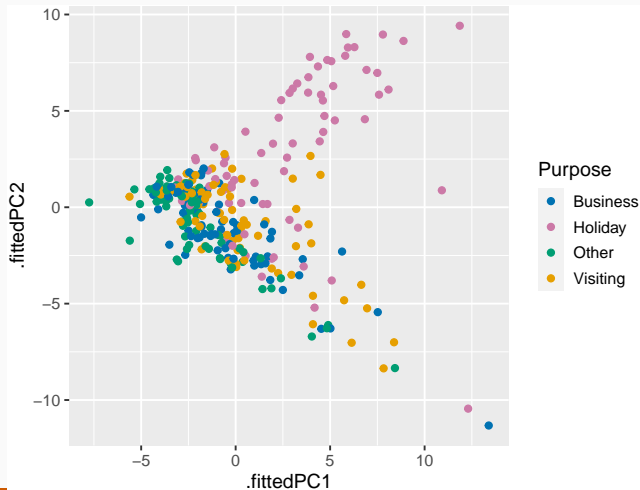
Principal components
based on all features
from the feasts
package



Feature extraction and statistics

```
pcs |> ggplot(aes(x=.fittedPC1, y=.fittedPC2, col=Purpose)) +  
  geom_point() + theme(aspect.ratio=1)
```

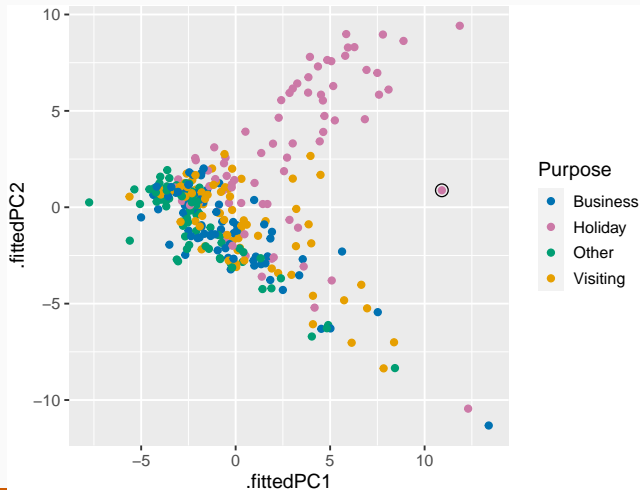
Principal components
based on all features
from the feasts
package



Feature extraction and statistics

```
pcs |> ggplot(aes(x=.fittedPC1, y=.fittedPC2, col=Purpose)) +  
  geom_point() + theme(aspect.ratio=1)
```

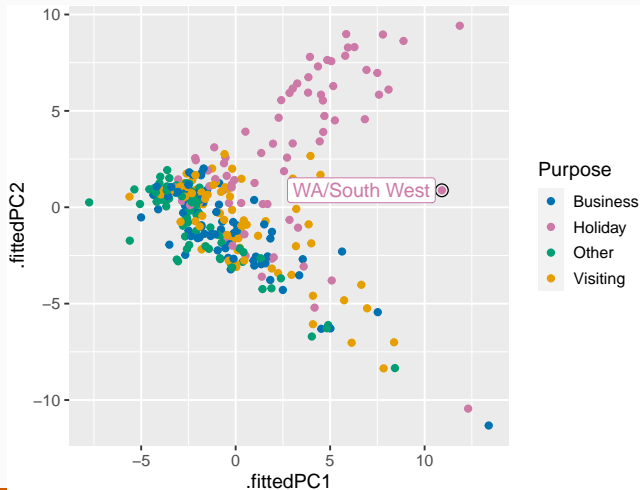
Principal components
based on all features
from the feasts
package



Feature extraction and statistics

```
pcs |> ggplot(aes(x=.fittedPC1, y=.fittedPC2, col=Purpose)) +  
  geom_point() + theme(aspect.ratio=1)
```

Principal components
based on all features
from the feasts
package



Outline

- 1 tsibble: Time series data
- 2 feasts: Data visualization
- 3 feasts: Time series features
- 4 fable: Forecasting
- 5 fable: Forecast reconciliation

Model fitting

```
training <- tourism |>
  filter(year(Quarter) <= 2015)
fit <- training |>
  model(
    snaive = SNAIVE(Trips),
    naive = NAIVE(Trips),
    ets = ETS(Trips),
    arima = ARIMA(Trips)
  )
```

```
## # A mable: 304 x 7
```

```
## # Key:      Region, State, Purpose [304]
```

##	Region	State	Purpose	snaive	naive	ets	arima
##	<chr>	<fct>	<chr>	<model>	<model>	<model>	<model>
##	1 Adelaide	SA	Busine~	<SNAIVE>	<NAIVE>	<ETS(M,N,A)>	<ARIMA(0,0,0)(1,0,1)[4] w/ mean>
##	2 Adelaide	SA	Holiday	<SNAIVE>	<NAIVE>	<ETS(M,N,A)>	<ARIMA(0,0,0)(2,0,0)[4] w/ mean>
##	3 Adelaide	SA	Other	<SNAIVE>	<NAIVE>	<ETS(M,A,N)>	<ARIMA(0,1,1) w/ drift>
##	4 Adelaide	SA	Visiti~	<SNAIVE>	<NAIVE>	<ETS(A,N,A)>	<ARIMA(0,0,0)(1,0,1)[4] w/ mean>
##	5 Adelaide	~ SA	Busine~	<SNAIVE>	<NAIVE>	<ETS(A,N,N)>	<ARIMA(0,0,0) w/ mean>
##	6 Adelaide	~ SA	Holiday	<SNAIVE>	<NAIVE>	<ETS(A,A,N)>	<ARIMA(0,0,0) w/ mean>
##	7 Adelaide	~ SA	Other	<SNAIVE>	<NAIVE>	<ETS(A,N,N)>	<ARIMA(2,1,1)(2,0,0)[4]>

Model fitting

```
fit |>
  filter(Purpose == "Holiday", Region == "Snowy Mountains") |>
  select(arima) |>
  report()
```

```
## Series: Trips
## Model: ARIMA(1,0,0)(0,1,2)[4]
##
## Coefficients:
##          ar1      sma1      sma2
##          0.223  -0.639  -0.288
## s.e.  0.121    0.253    0.163
##
## sigma^2 estimated as 461.3:  log likelihood=-307
## AIC=622   AICc=622   BIC=631
```

Model fitting

```
augment(fit)
```

```
## # A tsibble: 87,552 x 9 [1Q]
## # Key:           Region, State, Purpose, .model [1,216]
##   Region   State Purpose .model Quarter Trips .fitted .resid .innov
##   <chr>    <fct> <chr>   <chr>   <qtr> <dbl>   <dbl>   <dbl>   <dbl>
## 1 Adelaide SA     Business snaive 1998 Q1  135.    NA     NA     NA
## 2 Adelaide SA     Business snaive 1998 Q2  110.    NA     NA     NA
## 3 Adelaide SA     Business snaive 1998 Q3  166.    NA     NA     NA
## 4 Adelaide SA     Business snaive 1998 Q4  127.    NA     NA     NA
## 5 Adelaide SA     Business snaive 1999 Q1  137.   135.    2.37    2.37
## 6 Adelaide SA     Business snaive 1999 Q2  200.   110.   89.9   89.9
## 7 Adelaide SA     Business snaive 1999 Q3  169.   166.    3.32    3.32
## 8 Adelaide SA     Business snaive 1999 Q4  134.   127.    7.20    7.20
## 9 Adelaide SA     Business snaive 2000 Q1  154.   137.   16.6   16.6
## 10 Adelaide SA     Business snaive 2000 Q2  169.   200.  -31.1  -31.1
## # ... with 87,542 more rows
```

Ljung-Box test

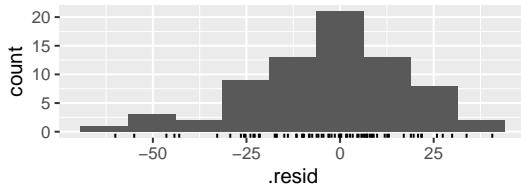
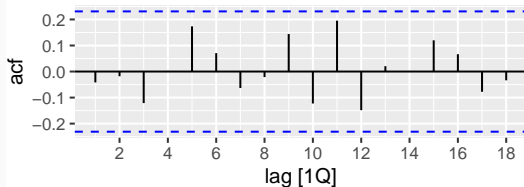
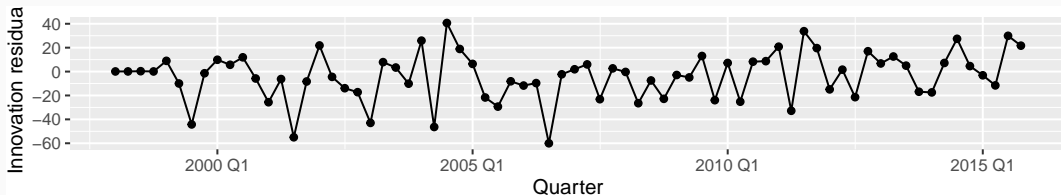
```
augment(fit) |>  
  filter(Purpose == "Holiday", Region == "Snowy Mountains", .model == "arima") |>  
  features(.resid, ljung_box, dof = 3, lag = 8)
```

```
## # A tibble: 1 x 6
```

```
##   Region      State Purpose .model lb_stat lb_pvalue  
##   <chr>      <fct> <chr>  <chr>   <dbl>   <dbl>  
## 1 Snowy Mountains NSW   Holiday arima     4.45     0.486
```

gg_tsresiduals() function

```
fit |>  
  filter(Purpose == "Holiday", Region == "Snowy Mountains") |>  
  select(arima) |>  
  gg_tsresiduals()
```



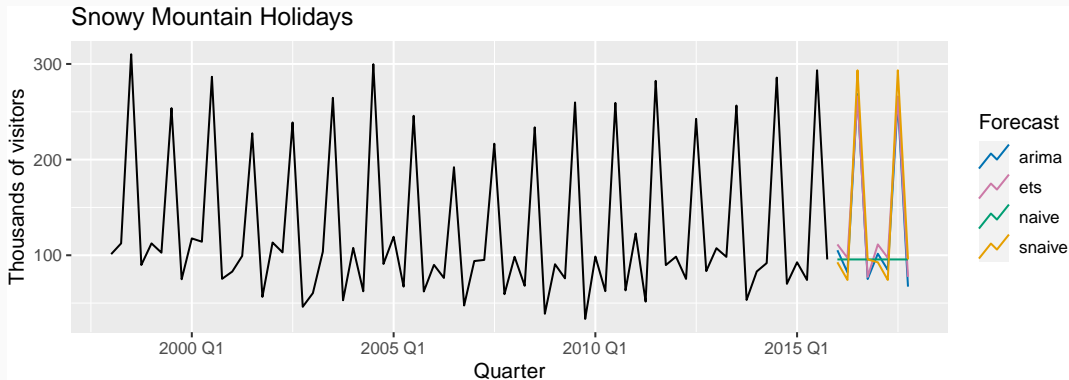
Producing forecasts

```
fc <- fit |>  
  forecast(h = "2 years")
```

```
## # A tibble: 9,728 x 7 [1Q]  
## # Key:   Region, State, Purpose, .model [1,216]  
##   Region State Purpose .model Quarter      Trips .mean  
##   <chr>   <fct> <chr>   <chr>   <qtr>     <dist> <dbl>  
## 1 Adelaide SA      Business snaive 2016 Q1 N(143, 2128) 143.  
## 2 Adelaide SA      Business snaive 2016 Q2 N(168, 2128) 168.  
## 3 Adelaide SA      Business snaive 2016 Q3 N(176, 2128) 176.  
## 4 Adelaide SA      Business snaive 2016 Q4 N(187, 2128) 187.  
## 5 Adelaide SA      Business snaive 2017 Q1 N(143, 4257) 143.  
## 6 Adelaide SA      Business snaive 2017 Q2 N(168, 4257) 168.  
## 7 Adelaide SA      Business snaive 2017 Q3 N(176, 4257) 176.  
## 8 Adelaide SA      Business snaive 2017 Q4 N(187, 4257) 187.  
## 9 Adelaide SA      Business naive   2016 Q1 N(187, 2635) 187.  
## 10 Adelaide SA      Business naive   2016 Q2 N(187, 5270) 187.  
## # ... with 9,718 more rows
```

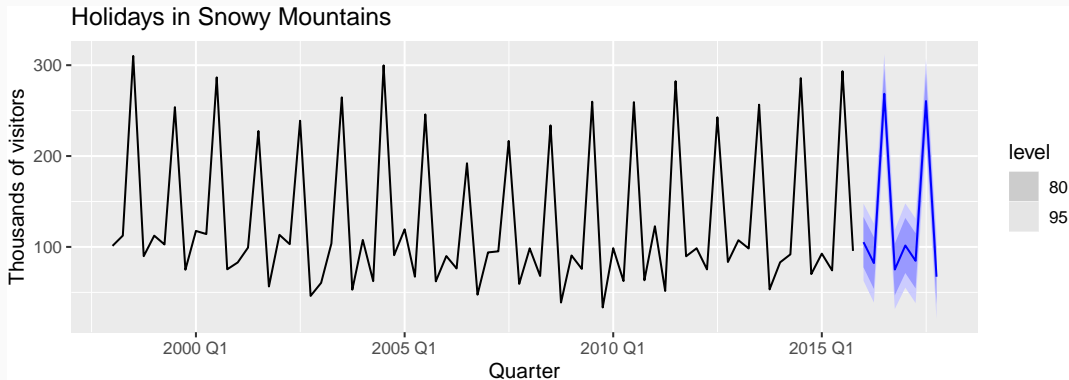
Visualising forecasts

```
fc |>  
  filter(Purpose == "Holiday", Region == "Snowy Mountains") |>  
  autoplot(training, level = NULL) +  
  labs(title = "Snowy Mountain Holidays", y = "Thousands of visitors") +  
  guides(color = guide_legend(title = "Forecast"))
```



Visualising forecasts

```
fc |>  
  filter(Purpose == "Holiday", Region == "Snowy Mountains", .model == "arma") |>  
  autoplot(training) +  
  labs(title = "Holidays in Snowy Mountains", y = "Thousands of visitors") +  
  guides(color = guide_legend(title = "Forecast"))
```

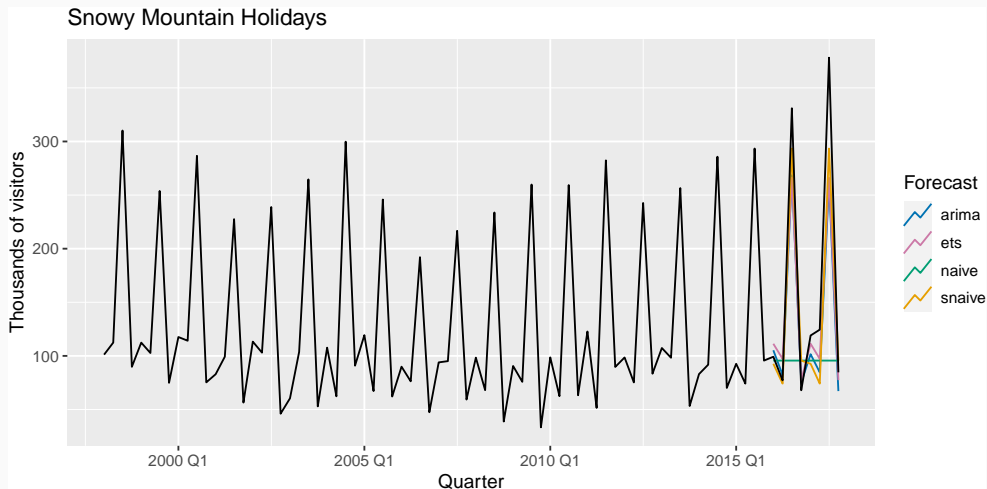


Prediction intervals

```
fc |> hilo(level = 95)
```

```
## # A tsibble: 9,728 x 8 [1Q]
## # Key:      Region, State, Purpose, .model [1,216]
##   Region    State Purpose .model Quarter      Trips .mean      `95%`
##   <chr>     <fct> <chr>   <chr>    <qtr>      <dist> <dbl>      <hilo>
## 1 Adelaide SA      Business snaive 2016 Q1 N(143, 2128) 143. [52.4, 233]95
## 2 Adelaide SA      Business snaive 2016 Q2 N(168, 2128) 168. [77.5, 258]95
## 3 Adelaide SA      Business snaive 2016 Q3 N(176, 2128) 176. [86.0, 267]95
## 4 Adelaide SA      Business snaive 2016 Q4 N(187, 2128) 187. [96.3, 277]95
## 5 Adelaide SA      Business snaive 2017 Q1 N(143, 4257) 143. [14.9, 271]95
## 6 Adelaide SA      Business snaive 2017 Q2 N(168, 4257) 168. [40.0, 296]95
## 7 Adelaide SA      Business snaive 2017 Q3 N(176, 4257) 176. [48.6, 304]95
## 8 Adelaide SA      Business snaive 2017 Q4 N(187, 4257) 187. [58.9, 315]95
## 9 Adelaide SA      Business naive  2016 Q1 N(187, 2635) 187. [86.1, 287]95
## 10 Adelaide SA     Business naive  2016 Q2 N(187, 5270) 187. [44.4, 329]95
## # ... with 9,718 more rows
```


Measures of forecast accuracy



Measures of forecast accuracy

```
fc |>
```

```
accuracy(tourism)
```

```
## # A tibble: 1,216 x 13
```

```
##   .model Region State Purpose .type      ME    RMSE    MAE      MPE    MAPE    MASE    RMSSE
##   <chr>   <chr> <fct> <chr>   <chr>   <dbl> <dbl> <dbl>   <dbl> <dbl> <dbl> <dbl>
## 1 arima  Adela~ SA    Busine~ Test    20.8   29.0   27.0     10.5   15.0  0.850  0.628
## 2 arima  Adela~ SA    Holiday Test    21.7   31.1   25.5     10.6   13.1  1.17   1.15
## 3 arima  Adela~ SA    Other   Test     9.79   13.7   12.0     10.6   14.2  0.887  0.772
## 4 arima  Adela~ SA    Visiti~ Test    32.2   36.1   32.2     13.3   13.3  1.04   0.956
## 5 arima  Adela~ SA    Busine~ Test     0.634   4.65   3.13    -Inf    Inf    0.935  0.771
## 6 arima  Adela~ SA    Holiday Test     6.13    7.24   6.13     35.1   35.1  1.07   0.899
## 7 arima  Adela~ SA    Other   Test    -0.923   1.52   1.43   -192.   206.   1.12   0.783
## 8 arima  Adela~ SA    Visiti~ Test     5.67   12.0   10.3    -54.5  107.   1.36   0.903
## 9 arima  Alice~ NT    Busine~ Test     9.11   12.2   10.5     26.6   43.5  1.71   1.56
## 10 arima Alice~ NT    Holiday Test    -0.536   9.66   8.54    -32.8   56.2  0.883  0.803
## # ... with 1,206 more rows, and 1 more variable: ACF1 <dbl>
```

Measures of forecast accuracy

```
fc |>  
  accuracy(tourism) |>  
  summarise(RMSSE = sqrt(mean(RMSSE^2)), .by=".model")
```

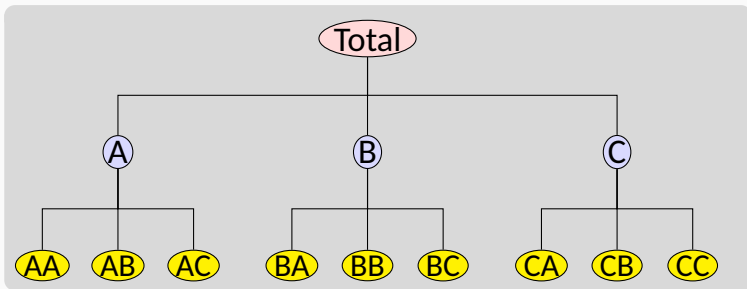
```
## # A tibble: 4 x 2  
##   .model RMSSE  
##   <chr>   <dbl>  
## 1 arima  1.04  
## 2 ets    0.996  
## 3 naive  1.36  
## 4 snaive 1.18
```

Outline

- 1 tsibble: Time series data
- 2 feasts: Data visualization
- 3 feasts: Time series features
- 4 fable: Forecasting
- 5 fable: Forecast reconciliation

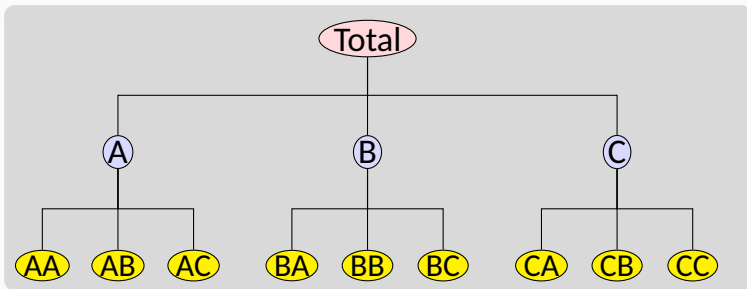
Hierarchical time series

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.

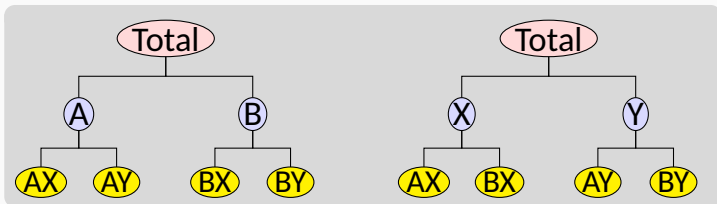


Examples

- Tourism demand by states, zones, regions

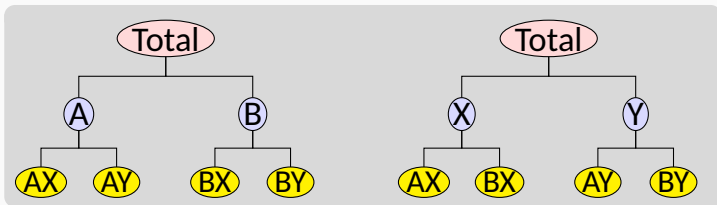
Grouped time series

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

Creating aggregates

```
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*>      <fct*>      <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> NSW <aggregated> 8040.
## 7 1998 Q1 <aggregated> VIC <aggregated> 6010.
## 8 1998 Q1 <aggregated> QLD <aggregated> 4041.
## 9 1998 Q1 <aggregated> WA <aggregated> 1641.
## 10 1998 Q1 <aggregated> SA <aggregated> 1735.
## 11 1998 Q1 <aggregated> TAS <aggregated> 982.
## 12 1998 Q1 <aggregated> NT <aggregated> 181.
## 13 1998 Q1 <aggregated> ACT <aggregated> 551.
```

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.

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, 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 using `reconcile()`.

Forecast reconciliation

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

```
## # A tibble: 1,700 x 7 [1Q]
## # Key:   Purpose, State, Region, .model [850]
##   Purpose State Region      .model Quarter      Trips .mean
##   <chr*>   <fct*> <chr*>      <chr>      <qtr>      <dist> <dbl>
## 1 Business NSW   Blue Mountains ets        2018 Q1 N(20, 140) 19.7
## 2 Business NSW   Blue Mountains ets        2018 Q2 N(20, 140) 19.7
## 3 Business NSW   Blue Mountains ets_adjusted 2018 Q1 N(20, 133) 20.2
## 4 Business NSW   Blue Mountains ets_adjusted 2018 Q2 N(21, 143) 20.5
## 5 Business NSW   Capital Country ets        2018 Q1 N(36, 202) 36.1
## 6 Business NSW   Capital Country ets        2018 Q2 N(36, 202) 36.1
## 7 Business NSW   Capital Country ets_adjusted 2018 Q1 N(37, 190) 37.5
## 8 Business NSW   Capital Country ets_adjusted 2018 Q2 N(38, 194) 38.2
```

Hierarchical and grouped time series

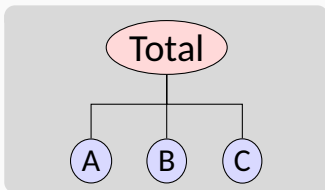
Every collection of time series with aggregation constraints can be written as

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

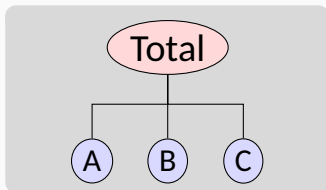
where

- \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.

Hierarchical time series



Hierarchical time series

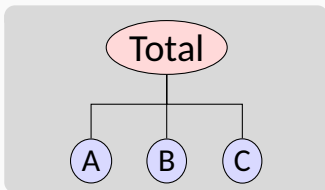


y_t : observed aggregate of all series at time t .

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

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

Hierarchical time series



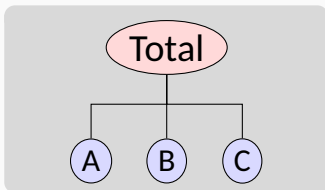
y_t : observed aggregate of all series at time t .

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

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

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

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 .

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

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

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:

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

for some matrix \mathbf{G} .

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:

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

for some matrix \mathbf{G} .

- \mathbf{G} 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 $\mathbf{G} = (\mathbf{S}'\mathbf{W}_h^{-1}\mathbf{S})^{-1}\mathbf{S}'\mathbf{W}_h^{-1}$, where \mathbf{W}_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 $\mathbf{G} = (\mathbf{S}'\mathbf{W}_h^{-1}\mathbf{S})^{-1}\mathbf{S}'\mathbf{W}_h^{-1}$, where \mathbf{W}_h is the h -step base forecast error covariance matrix.

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

Problem: \mathbf{W}_h hard to estimate, especially for $h > 1$.

Solutions:

- Ignore \mathbf{W}_h (OLS) [`min_trace(method='ols')`]
- Assume $\mathbf{W}_h = k_h\mathbf{W}_1$ is diagonal (WLS)
[`min_trace(method='wls')`]
- Assume $\mathbf{W}_h = k_h\mathbf{W}_1$ and estimate it (GLS)

Example: Australian tourism

```
tourism_agg <- tourism |>
  aggregate_key(Purpose * (State / Region),
    Trips = sum(Trips)
  )
fc <- tourism_agg |>
  filter(year(Quarter) <= 2015) |>
  model(
    ets = ETS(Trips),
    arima = ARIMA(Trips)
  ) |>
  mutate(
    comb = (ets + arima) / 2
  ) |>
  reconcile(
    ets_adj = min_trace(ets),
    arima_adj = min_trace(arima),
```

Forecast evaluation

```
accuracy(fc, tourism_agg)
```

```
## # A tibble: 2,550 x 13
```

[illegible]

Forecast evaluation

```
accuracy(fc, tourism_agg) |>  
  summarise(RMSSE = sqrt(mean(RMSSE^2)), by=".model") |>  
  arrange(RMSSE)
```

```
## # A tibble: 1 x 2  
##   RMSSE by  
##   <dbl> <chr>  
## 1  1.06 .model
```

More information

- Slides and papers: **robjhyndman.com**
- Packages: **tidyverts.org**
- Forecasting textbook using tsibble, feasts and fable packages: **OTexts.com/fpp3**

Find me at ...

 **[@robjhyndman](https://twitter.com/robjhyndman)**

 **[@robjhyndman](https://github.com/robjhyndman)**

 **robjhyndman.com**

 **rob.hyndman@monash.edu**