MONASH University

MONASH
BUSINESS
SCHOOL

Tidy time series analysis in R

tsibble feasts Fable

Rob J Hyndman

27 September 2019

Outline

Tidyverts packages

tidyverts.org

tsibble

tsibbledata

feasts

fable

Outline

Time series data

- Four-yearly Olympic winning times
- Annual Google profits
- Quarterly Australian beer production
- Monthly rainfall
- Weekly retail sales
- Daily IBM stock prices
- Hourly electricity demand
- 5-minute freeway traffic counts
- Time-stamped stock transaction data

tsibble objects

```
global_economy
```

```
## # A tsibble: 15,150 x 6 [1Y]
## # Key:       Country [263]
##     Year Country             GDP Imports Exports Population
##    <dbl> <fct>             <dbl>   <dbl>   <dbl>      <dbl>
##  1  1960 Afghanistan  537777811.    7.02    4.13    8996351
##  2  1961 Afghanistan  548888896.    8.10    4.45    9166764
##  3  1962 Afghanistan  546666678.    9.35    4.88    9345868
##  4  1963 Afghanistan  751111191.   16.9     9.17    9533954
##  5  1964 Afghanistan  800000044.   18.1     8.89    9731361
##  6  1965 Afghanistan 1006666638.   21.4    11.3     9938414
##  7  1966 Afghanistan 1399999967.   18.6     8.57   10152331
##  8  1967 Afghanistan 1673333418.   14.2     6.77   10372630
##  9  1968 Afghanistan 1373333367.   15.2     8.90   10604346
## 10  1969 Afghanistan 1408888922.   15.0    10.1    10854428
## # ... with 15,140 more rows
```

tsibble objects

```
global_economy
```

```
## # A tsibble: 15,150 x 6 [1Y]
## # Key:       Country [263]
##      Year Country             GDP Imports Exports Population
##     Index <fct>             <dbl>   <dbl>   <dbl>      <dbl>
##  1  1960 Afghanistan  537777811.    7.02    4.13    8996351
##  2  1961 Afghanistan  548888896.    8.10    4.45    9166764
##  3  1962 Afghanistan  546666678.    9.35    4.88    9345868
##  4  1963 Afghanistan  751111191.   16.9     9.17    9533954
##  5  1964 Afghanistan  800000044.   18.1     8.89    9731361
##  6  1965 Afghanistan 1006666638.   21.4    11.3     9938414
##  7  1966 Afghanistan 1399999967.   18.6     8.57   10152331
##  8  1967 Afghanistan 1673333418.   14.2     6.77   10372630
##  9  1968 Afghanistan 1373333367.   15.2     8.90   10604346
## 10  1969 Afghanistan 1408888922.   15.0    10.1    10854428
## # ... with 15,140 more rows
```

tsibble objects

```
global_economy
```

```
## # A tsibble: 15,150 x 6 [1Y]
## # Key:       Country [263]
##      Year Country               GDP Imports Exports Population
##      Index Key                 <dbl>   <dbl>   <dbl>      <dbl>
##  1   1960 Afghanistan  537777811.    7.02    4.13    8996351
##  2   1961 Afghanistan  548888896.    8.10    4.45    9166764
##  3   1962 Afghanistan  546666678.    9.35    4.88    9345868
##  4   1963 Afghanistan  751111191.   16.9     9.17    9533954
##  5   1964 Afghanistan  800000044.   18.1     8.89    9731361
##  6   1965 Afghanistan 1006666638.   21.4    11.3     9938414
##  7   1966 Afghanistan 1399999967.   18.6     8.57   10152331
##  8   1967 Afghanistan 1673333418.   14.2     6.77   10372630
##  9   1968 Afghanistan 1373333367.   15.2     8.90   10604346
## 10   1969 Afghanistan 1408888922.   15.0    10.1    10854428
## # ... with 15,140 more rows
```

tsibble objects

```
global_economy
```

```
## # A tsibble: 15,150 x 6 [1Y]
## # Key:       Country [263]
##     Year Country             GDP Imports Exports Population
##     Index Key          Measured variables
##  1  1960 Afghanistan  537777811.    7.02    4.13    8996351
##  2  1961 Afghanistan  548888896.    8.10    4.45    9166764
##  3  1962 Afghanistan  546666678.    9.35    4.88    9345868
##  4  1963 Afghanistan  751111191.   16.9     9.17    9533954
##  5  1964 Afghanistan  800000044.   18.1     8.89    9731361
##  6  1965 Afghanistan 1006666638.   21.4    11.3     9938414
##  7  1966 Afghanistan 1399999967.   18.6     8.57   10152331
##  8  1967 Afghanistan 1673333418.   14.2     6.77   10372630
##  9  1968 Afghanistan 1373333367.   15.2     8.90   10604346
## 10  1969 Afghanistan 1408888922.   15.0    10.1    10854428
## # ... with 15,140 more rows
```

tsibble objects

```r
tourism
```

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:       Region, State, Purpose [304]
##    Quarter Region   State Purpose  Trips
##      <qtr> <chr>    <chr> <chr>    <dbl>
##  1 1998 Q1 Adelaide 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.
## # ... with 24,310 more rows
```

tsibble objects

```r
tourism
```

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:       Region, State, Purpose [304]
##    Quarter Region   State Purpose  Trips
##    Index   <chr>    <chr> <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.
## # ... with 24,310 more rows
```

tsibble objects

```r
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.
## # ... with 24,310 more rows
```

tsibble objects

```r
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.
## # ... with 24,310 more rows
```

tsibble objects

```
tourism
```

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:       Region, State, Purpose [304]
##    Quarter Region   State Purpose  Trips
##      <qtr> <chr>    <chr> <chr>    <dbl>
##  1 1998 Q1 Adelaide 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.
## # ... with 24,310 more rows
```

Index | Keys | Measure

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

feasts package

tidyverts.org

tsibble

tsibbledata

feasts

fable

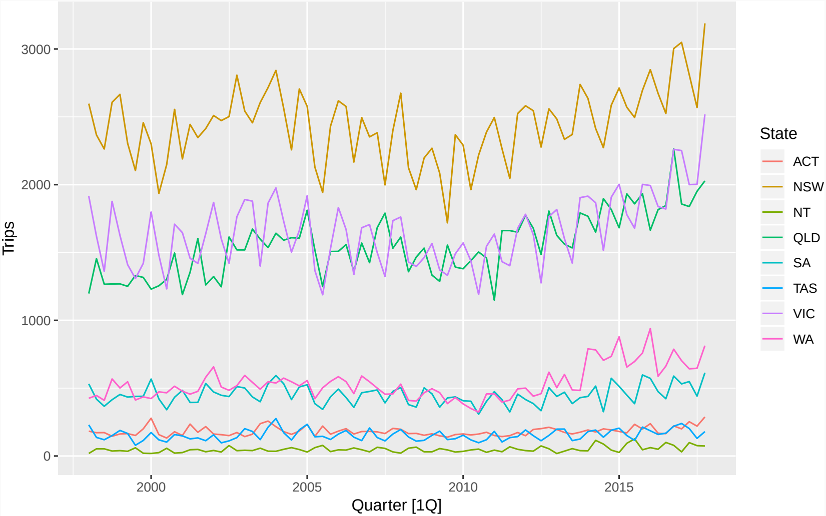
Holidays by state

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

```
## # A tsibble: 640 x 3 [1Q]
## # Key:       State [8]
##    Quarter State Trips
##      <qtr> <chr> <dbl>
##  1 1998 Q1 ACT    183.
##  2 1998 Q2 ACT    172.
##  3 1998 Q3 ACT    173.
##  4 1998 Q4 ACT    146.
##  5 1999 Q1 ACT    162.
##  6 1999 Q2 ACT    165.
##  7 1999 Q3 ACT    151.
##  8 1999 Q4 ACT    200.
##  9 2000 Q1 ACT    279.
## 10 2000 Q2 ACT    157.
```

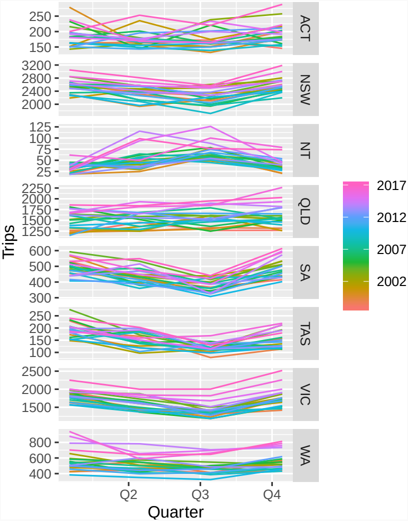
Time plots

```
holidays %>% autoplot(Trips)
```



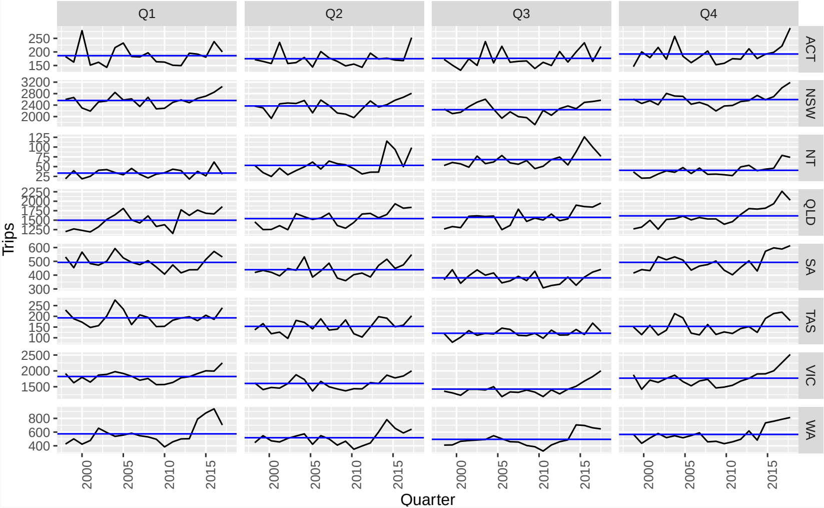
Season plots

```r
holidays %>% gg_season(Trips)
```



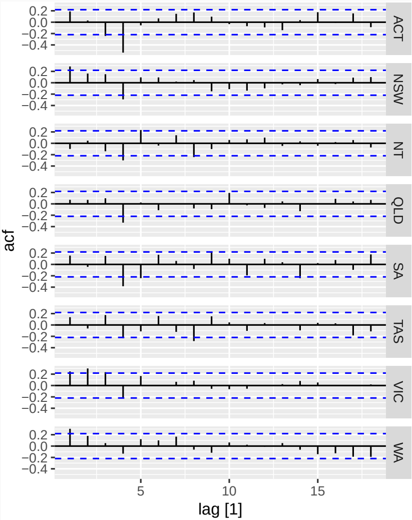
Graphics

```
holidays %>% gg_subseries(Trips)
```



Graphics

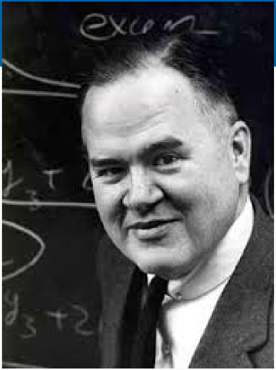
```
holidays %>% ACF(difference(Trips, 4)) %>% autoplot()
```



Features

Cognostics

Computer-produced diagnostics (Tukey and Tukey, 1985).



John W Tukey

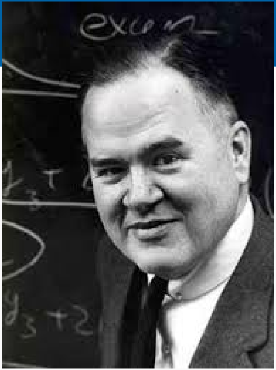
Features

Cognostics

Computer-produced diagnostics (Tukey and Tukey, 1985).

Examples for time series

- lag correlation
- size and direction of trend
- strength of seasonality
- timing of peak seasonality
- spectral entropy



John W Tukey

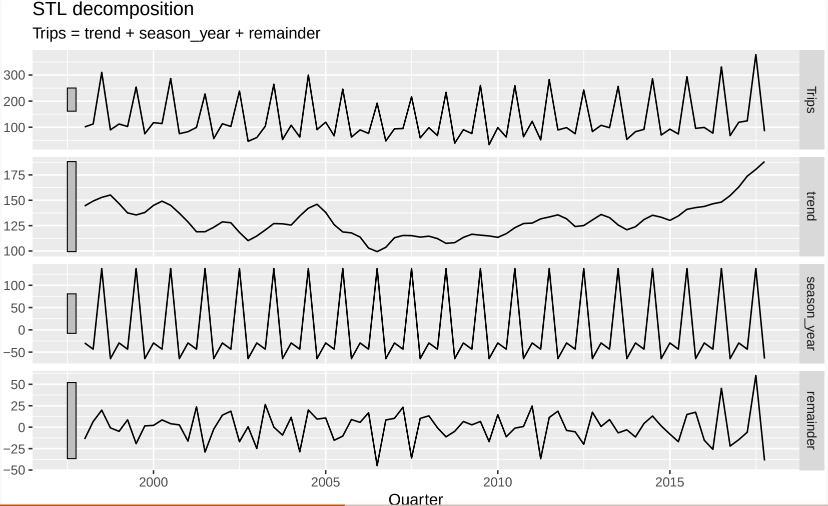
Extracting single time series

```r
snowy <- tourism %>%
  filter(
    Region=="Snowy Mountains",
    Purpose=="Holiday"
  )
snowy
```

```
## # A tsibble: 80 x 5 [1Q]
## # Key:       Region, State, Purpose [1]
##    Quarter Region          State Purpose Trips
##      <qtr> <chr>           <chr> <chr>   <dbl>
##  1 1998 Q1 Snowy Mountains NSW   Holiday 101.
##  2 1998 Q2 Snowy Mountains NSW   Holiday 112.
##  3 1998 Q3 Snowy Mountains NSW   Holiday 310.
##  4 1998 Q4 Snowy Mountains NSW   Holiday  89.8
##  5 1999 Q1 Snowy Mountains NSW   Holiday 112.
```

Decompositions

```
snowy %>% STL(Trips ~ season(window = "periodic")) %>%
  autoplot()
```



Candidate features

STL decomposition

$$Y_t = S_t + T_t + R_t$$

Candidate features

STL decomposition

$$Y_t = S_t + T_t + R_t$$

- Strength of seasonality: $\max\left(0, 1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - T_t)}\right)$
- Strength of trend: $\max\left(0, 1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - S_t)}\right)$
- Timing of seasonal peaks/troughs
- Linearity/curvature of trend
- Spikiness of remainder
- Autocorrelations of remainder $(R_1, \dots, R_T)$

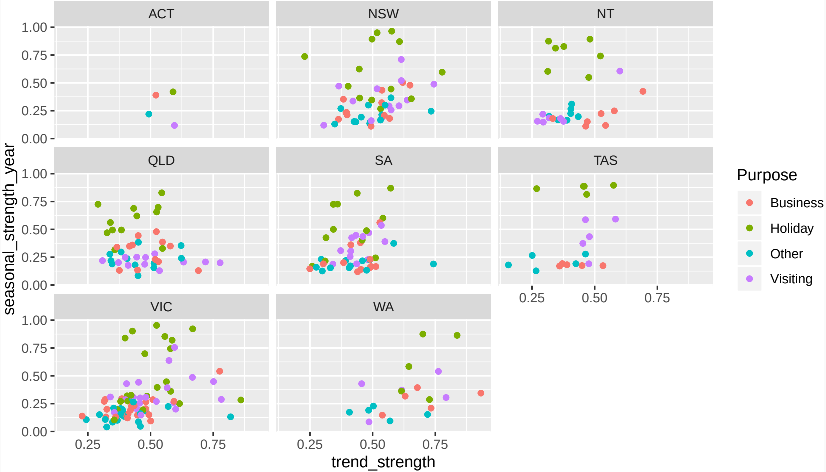
Feature extraction and statistics

```r
tourism %>% features(Trips, feature_set(tags="stl"))
```

```
## # A tibble: 304 x 12
##    Region State Purpose trend_strength seasonal_streng~
##    <chr>  <chr> <chr>            <dbl>            <dbl>
##  1 Adela~ SA    Busine~          0.451            0.380
##  2 Adela~ SA    Holiday          0.541            0.601
##  3 Adela~ SA    Other            0.743            0.189
##  4 Adela~ SA    Visiti~          0.433            0.446
##  5 Adela~ SA    Busine~          0.453            0.140
##  6 Adela~ SA    Holiday          0.512            0.244
##  7 Adela~ SA    Other            0.584            0.374
##  8 Adela~ SA    Visiti~          0.481            0.228
##  9 Alice~ NT    Busine~          0.526            0.224
## 10 Alice~ NT    Holiday          0.377            0.827
## # ... with 294 more rows, and 7 more variables:
## #   seasonal_peak_year <dbl>, seasonal_trough_year <dbl>,
## #   spikiness <dbl>, linearity <dbl>, curvature <dbl>,
## #   stl_e_acf1 <dbl>, stl_e_acf10 <dbl>
```

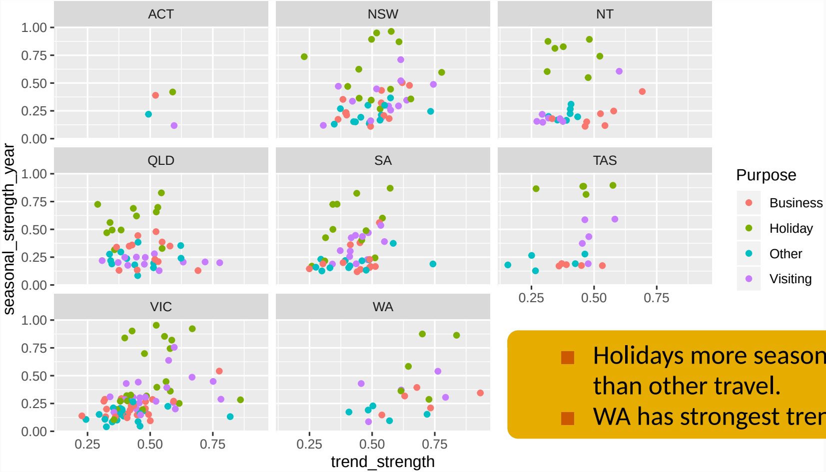
Feature extraction and statistics

```r
tourism %>% features(Trips, feature_set(tags="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, feature_set(tags="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:

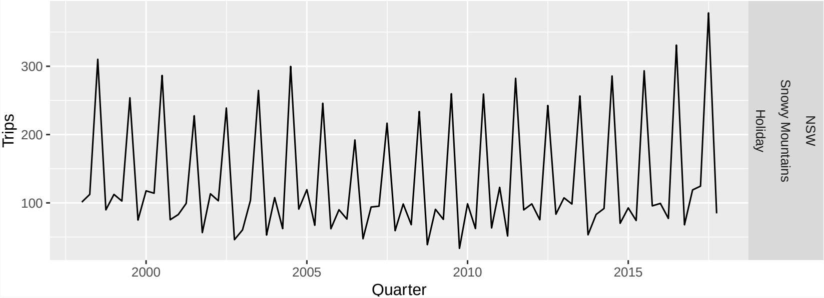
```
most_seasonal <- tourism %>%
  features(Trips, feature_set(tags="stl")) %>%
  filter(seasonal_strength_year == max(seasonal_strength_year))
```

Feature extraction and statistics

Find the most seasonal time series:

```
most_seasonal <- tourism %>%
  features(Trips, feature_set(tags="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

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

All features from the feasts package

```
## # A tibble: 304 x 47
##    Region State Purpose trend_strength seasonal_streng~
##    <chr>  <chr> <chr>            <dbl>            <dbl>
##  1 Adela~ SA    Busine~          0.451            0.380
##  2 Adela~ SA    Holiday          0.541            0.601
##  3 Adela~ SA    Other            0.743            0.189
##  4 Adela~ SA    Visiti~          0.433            0.446
##  5 Adela~ SA    Busine~          0.453            0.140
##  6 Adela~ SA    Holiday          0.512            0.244
##  7 Adela~ SA    Other            0.584            0.374
##  8 Adela~ SA    Visiti~          0.481            0.228
##  9 Alice~ NT    Busine~          0.526            0.224
## 10 Alice~ NT    Holiday          0.377            0.827
## # ... with 294 more rows, and 42 more variables:
## #   seasonal_peak_year <dbl>, seasonal_trough_year <dbl>,
## #   spikiness <dbl>, linearity <dbl>, curvature <dbl>,
## #   stl_e_acf1 <dbl>, stl_e_acf10 <dbl>, acf1 <dbl>,
## #   acf10 <dbl>, diff1_acf1 <dbl>, diff1_acf10 <dbl>,
```

Feature extraction and statistics

```r
pcs <- tourism_features %>% select(-State, -Region, -Purpose) %>%
  prcomp(scale=TRUE) %>% augment(tourism_features)
```

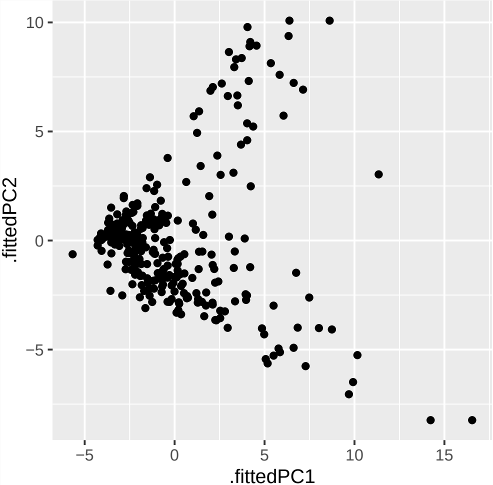
```
## # A tibble: 304 x 92
##    .rownames Region State Purpose trend_strength
##    <fct>     <chr>  <chr> <chr>            <dbl>
##  1 1         Adela~ SA    Busine~          0.451
##  2 2         Adela~ SA    Holiday          0.541
##  3 3         Adela~ SA    Other            0.743
##  4 4         Adela~ SA    Visiti~          0.433
##  5 5         Adela~ SA    Busine~          0.453
##  6 6         Adela~ SA    Holiday          0.512
##  7 7         Adela~ SA    Other            0.584
##  8 8         Adela~ SA    Visiti~          0.481
##  9 9         Alice~ NT    Busine~          0.526
## 10 10        Alice~ NT    Holiday          0.377
## # ... with 294 more rows, and 87 more variables:
## #   seasonal_strength_year <dbl>, seasonal_peak_year <dbl>,
## #   seasonal_trough_year <dbl>, spikiness <dbl>,
## #   linearity <dbl>, curvature <dbl>, stl_e_acf1 <dbl>,
## #   stl_e_acf10 <dbl>, acf1 <dbl>, acf10 <dbl>,
```

Principal components based on all features from the feasts package

Feature extraction and statistics

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

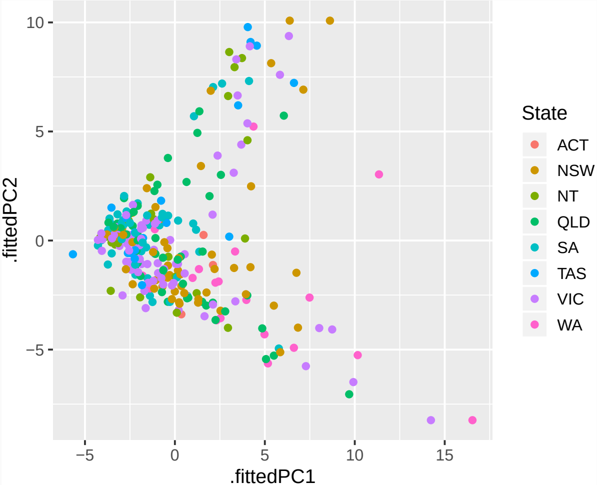
Principal components based on all features from the feasts package



Feature extraction and statistics

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

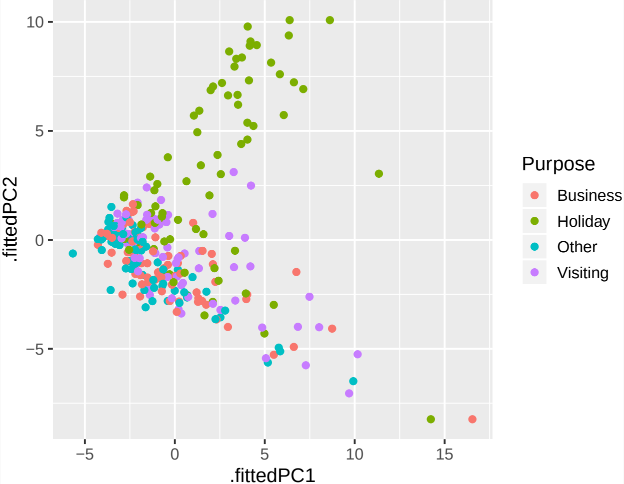
Principal components based on all features from the feasts package



Feature extraction and statistics

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

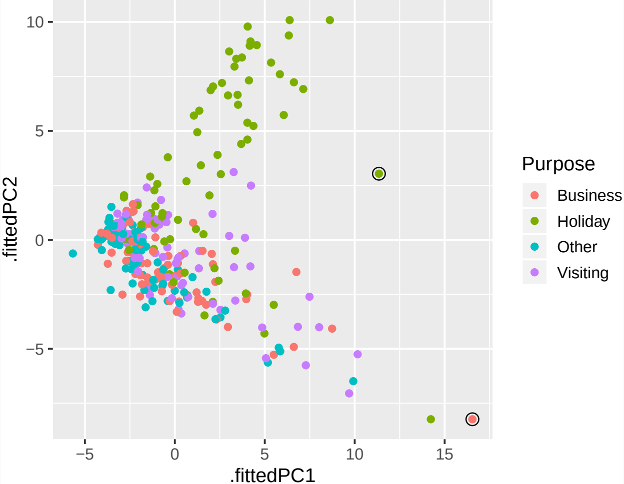
Principal components based on all features from the feasts package



Feature extraction and statistics

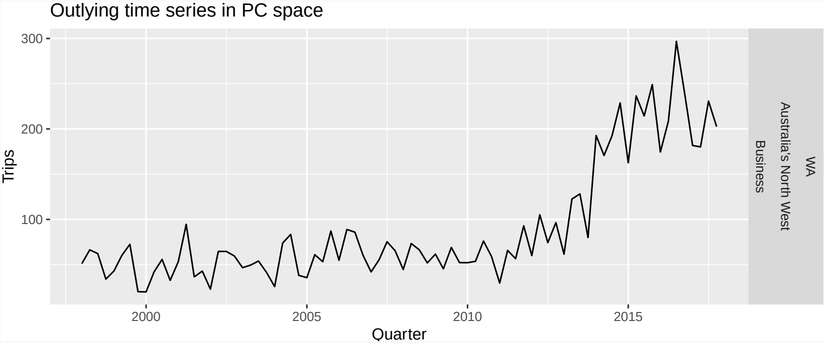
```
pcs %>% ggplot(aes(x=.fittedPC1, y=.fittedPC2, col=Purpose)) +
  geom_point() + theme(aspect.ratio=1)
```

Principal components based on all features from the feasts package



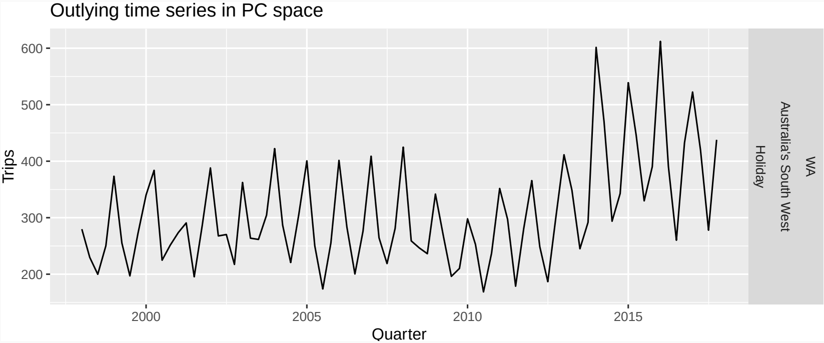
Feature extraction and statistics

```
pcs %>%
  filter(.fittedPC1 == max(.fittedPC1)) %>%
  left_join(tourism, by = c("State", "Region", "Purpose")) %>%
  ggplot(aes(x = Quarter, y = Trips)) +
    geom_line() +
    facet_grid(vars(State,Region,Purpose)) +
    ggtitle("Outlying time series in PC space") +
    theme(legend.position = "none")
```



Feature extraction and statistics

```
pcs %>%
  filter(.fittedPC1 > 10 & .fittedPC2 > 2.5) %>%
  left_join(tourism, by = c("State", "Region", "Purpose")) %>%
  ggplot(aes(x = Quarter, y = Trips)) +
    geom_line() +
    facet_grid(vars(State,Region,Purpose)) +
    ggtitle("Outlying time series in PC space") +
    theme(legend.position = "none")
```



Outline

fable package

tidyverts.org

tsibble

tsibbledata

feasts

fable

Model estimation

The `model()` function estimates models for tsibbles.

```
tourism %>%
  model(
    snaive = SNAIVE(Trips),
    ets = ETS(Trips),
    arima = ARIMA(Trips)
  )
```

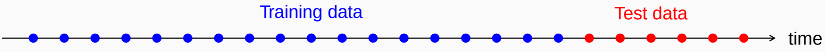
```
## # A mable: 304 x 6
## # Key:     Region, State, Purpose [304]
##    Region     State Purpose  snaive   ets       arima
##    <chr>      <chr> <chr>    <model>  <model>   <model>
##  1 Adelaide   SA    Business <SNAIV~ <ETS(M,~ <ARIMA(0,0,0)(1,~
##  2 Adelaide   SA    Holiday  <SNAIV~ <ETS(A,~ <ARIMA(0,0,0)(1,~
##  3 Adelaide   SA    Other    <SNAIV~ <ETS(M,~ <ARIMA(0,1,1) w/~
##  4 Adelaide   SA    Visiting <SNAIV~ <ETS(A,~ <ARIMA(0,0,0)(1,~
##  5 Adelaide ~ SA    Business <SNAIV~ <ETS(A,~ <ARIMA(0,0,0) w/~
##  6 Adelaide ~ SA    Holiday  <SNAIV~ <ETS(A,~ <ARIMA(0,1,1)>
##  7 Adelaide ~ SA    Other    <SNAIV~ <ETS(A,~ <ARIMA(0,1,2)(0,~
##  8 Adelaide ~ SA    Visiting <SNAIV~ <ETS(M,~ <ARIMA(0,1,1)>
```

Producing forecasts

```r
tourism %>%
  model(
    snaive = SNAIVE(Trips),
    ets = ETS(Trips),
    arima = ARIMA(Trips)
  ) %>%
  forecast(h= "3 years")
```

```
## # A fable: 10,944 x 7 [1Q]
## # Key:     Region, State, Purpose, .model [912]
##    Region   State Purpose  .model Quarter Trips .distribution
##    <chr>    <chr> <chr>    <chr>    <qtr> <dbl> <dist>
##  1 Adelaide SA    Business snaive 2018 Q1  129. N(129, 2018)
##  2 Adelaide SA    Business snaive 2018 Q2  174. N(174, 2018)
##  3 Adelaide SA    Business snaive 2018 Q3  185. N(185, 2018)
##  4 Adelaide SA    Business snaive 2018 Q4  197. N(197, 2018)
```

Training and test sets



- A model which fits the training data well will not necessarily forecast well.
- Forecast accuracy is based only on the test set.

Forecast errors

Forecast "error": the difference between an observed value and its forecast.

$$e_{T+h} = y_{T+h} - \hat{y}_{T+h|T},$$

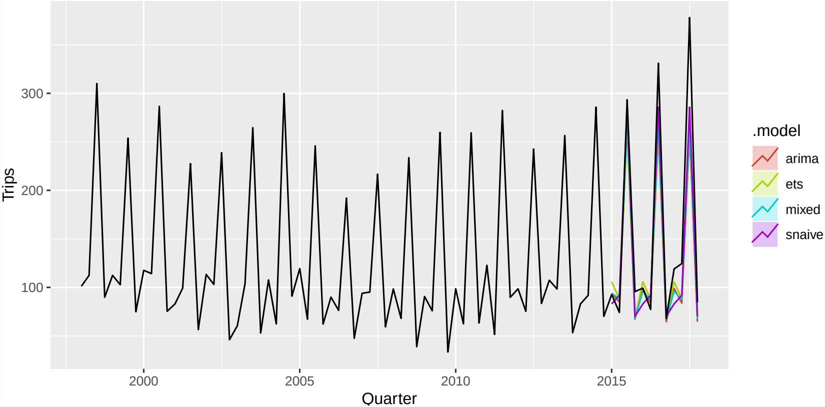
where the training data is given by $\{y_1, \dots, y_T\}$

Forecast errors

```r
train <- tourism %>%
  filter(year(Quarter) <= 2014)
fit <- train %>%
  model(
    ets = ETS(Trips),
    arima = ARIMA(Trips),
    snaive = SNAIVE(Trips)
  ) %>%
  mutate(mixed = (ets+arima+snaive)/3)
fc <- fit %>% forecast(h="3 years")
```

Forecast errors

```
fc %>%
  filter(Region=="Snowy Mountains", Purpose=="Holiday") %>%
  autoplot(level=NULL) +
  autolayer(snowy, Trips)
```



Measures of forecast accuracy

$$
\begin{aligned}
y_{T+h} &= (T+h)\text{th observation, } h = 1, \dots, H \\
\hat{y}_{T+h|T} &= \text{its forecast based on data up to time } T. \\
e_{T+h} &= y_{T+h} - \hat{y}_{T+h|T}
\end{aligned}
$$

$$
\begin{aligned}
\text{MAE} &= \text{mean}(|e_{T+h}|) \\
\text{MSE} &= \text{mean}(e_{T+h}^2) \qquad\qquad \text{RMSE} = \sqrt{\text{mean}(e_{T+h}^2)} \\
\text{MAPE} &= 100\text{mean}(|e_{T+h}|/|y_{T+h}|)
\end{aligned}
$$

Measures of forecast accuracy

$$\begin{aligned} y_{T+h} &= (T+h)\text{th observation}, h = 1,\dots,H \\ \hat{y}_{T+h|T} &= \text{its forecast based on data up to time } T. \\ e_{T+h} &= y_{T+h} - \hat{y}_{T+h|T} \end{aligned}$$

$$\begin{aligned} \text{MAE} &= \text{mean}(|e_{T+h}|) \\ \text{MSE} &= \text{mean}(e^2_{T+h}) \qquad \text{RMSE} = \sqrt{\text{mean}(e^2_{T+h})} \\ \text{MAPE} &= 100\text{mean}(|e_{T+h}|/|y_{T+h}|) \end{aligned}$$

- MAE, MSE, RMSE are all scale dependent.
- MAPE is scale independent but is only sensible if $y_t \gg 0$ for all t, and y has a natural zero.

Measures of forecast accuracy

Mean Absolute Scaled Error

$$\text{MASE} = \text{mean}(|e_{T+h}|/Q)$$

where Q is a stable measure of the scale of the time series $\{y_t\}$.

Proposed by Hyndman and Koehler (IJF, 2006).

For non-seasonal time series,

$$Q = (T-1)^{-1}\sum_{t=2}^{T}|y_t - y_{t-1}|$$

works well. Then MASE is equivalent to MAE relative to a naïve method.

Measures of forecast accuracy

Mean Absolute Scaled Error

$$\text{MASE} = \text{mean}(|e_{T+h}|/Q)$$

where Q is a stable measure of the scale of the time series $\{y_t\}$.

Proposed by Hyndman and Koehler (IJF, 2006).

For seasonal time series,

$$Q = (T-m)^{-1} \sum_{t=m+1}^{T} |y_t - y_{t-m}|$$

works well. Then MASE is equivalent to MAE relative to a seasonal naïve method.

Measures of forecast accuracy

```r
accuracy(fc, tourism)
```

```
## # A tibble: 1,216 x 12
##    .model Region State Purpose .type    ME  RMSE   MAE      MPE
##    <chr>  <chr>  <chr> <chr>   <chr> <dbl> <dbl> <dbl>    <dbl>
##  1 arima  Adela~ SA    Busine~ Test  22.5  28.5  25.3     11.9
##  2 arima  Adela~ SA    Holiday Test  21.9  34.8  28.0      9.93
##  3 arima  Adela~ SA    Other   Test   4.71 17.5  14.6      0.529
##  4 arima  Adela~ SA    Visiti~ Test  32.8  37.1  32.8     13.7
##  5 arima  Adela~ SA    Busine~ Test   1.31  5.58  3.57 -Inf
##  6 arima  Adela~ SA    Holiday Test   6.46  7.43  6.46   37.4
##  7 arima  Adela~ SA    Other   Test   1.35  2.79  1.93  -31.0
##  8 arima  Adela~ SA    Visiti~ Test   8.37 12.6  10.4     -3.98
##  9 arima  Alice~ NT    Busine~ Test   9.85 12.2  10.7     34.4
## 10 arima  Alice~ NT    Holiday Test   4.80 11.3   9.30     4.46
## # ... with 1,206 more rows, and 3 more variables: MAPE <dbl>,
## #   MASE <dbl>, ACF1 <dbl>
```

Measures of forecast accuracy

```r
accuracy(fc, tourism) %>%
  group_by(.model) %>%
  summarise(
    RMSE = mean(RMSE),
    MAE = mean(MAE),
    MASE = mean(MASE)
  ) %>%
  arrange(RMSE)
```

```
## # A tibble: 4 x 4
##   .model  RMSE   MAE  MASE
##   <chr>  <dbl> <dbl> <dbl>
## 1 mixed   19.8  16.0 0.997
## 2 ets     20.2  16.4 1.00
## 3 snaive  21.5  17.3 1.17
## 4 arima   21.9  17.8 1.07
```

Acknowledgements

Mitchell O'Hara-Wild

Earo Wang

tidyverts.org

robjhyndman.com