MONASH University

MONASH BUSINESS SCHOOL

Tidy Time Series & Forecasting in R

tsibble feasts Fable

4. Seasonality and trends

robjhyndman.com/workshop2020

Outline

1. Time series decompositions
2. Lab Session 7

Outline

1. Time series decompositions
2. Lab Session 7

Time series decomposition

Trend-Cycle aperiodic changes in level over time.

Seasonal (almost) periodic changes in level due to seasonal factors (e.g., the quarter of the year, the month, or day of the week).

Additive decomposition

$$y_t = S_t + T_t + R_t$$

where $y_t =$ data at period t

$T_t =$ trend-cycle component at period t

$S_t =$ seasonal component at period t

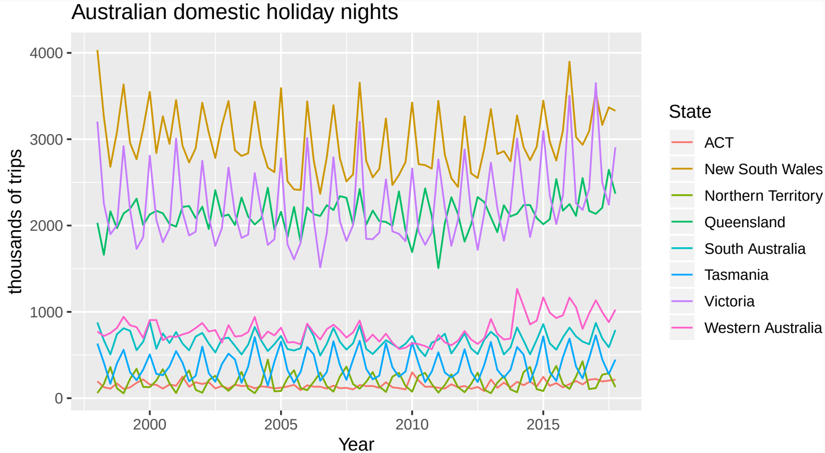
$R_t =$ remainder component at period t

STL decomposition

- STL: “Seasonal and Trend decomposition using Loess”
- Very versatile and robust.
- Seasonal component allowed to change over time, and rate of change controlled by user.
- Smoothness of trend-cycle also controlled by user.
- Optionally robust to outliers
- Not trading day or calendar adjustments.
- Only additive.
- Take logs to get multiplicative decomposition.
- Use Box-Cox transformations to get other decompositions.

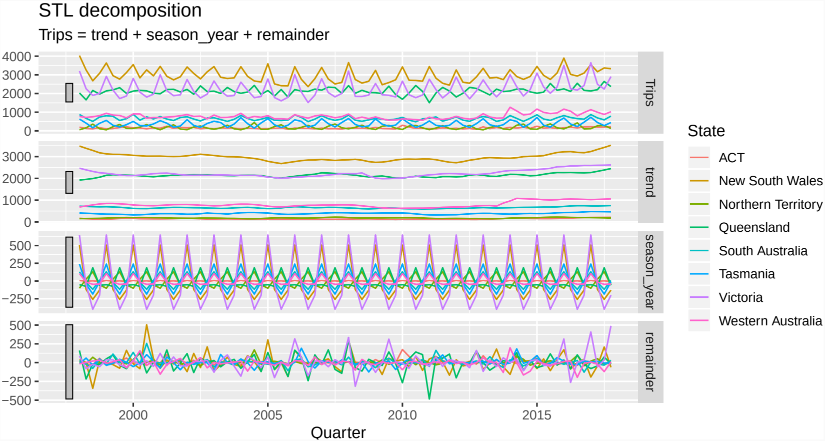
Australian holidays

```r
holidays %>% autoplot(Trips) +
  ylab("thousands of trips") + xlab("Year") +
  ggtitle("Australian domestic holiday nights")
```



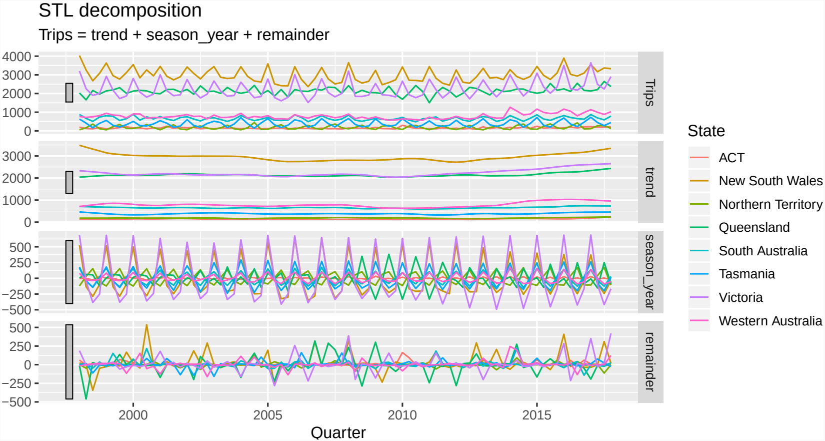
Holidays decomposition

```
holidays %>%
  STL(Trips ~ season(window="periodic"), robust=TRUE) %>%
  autoplot()
```



Holidays decomposition

```
holidays %>%
  STL(Trips ~ season(window = 5), robust = TRUE) %>%
  autoplot()
```



STL decomposition

```
holidays %>%
  STL(Trips ~ trend(window=15) + season(window=13),
      robust = TRUE)
```

- `trend(window = ?)` controls wiggliness of trend component.
- `season(window = ?)` controls variation on seasonal component.
- `STL()` chooses `season(window=13)` by default
- A large seasonal window is equivalent to setting `window="periodic"`.
- Odd numbers should be used for symmetry.

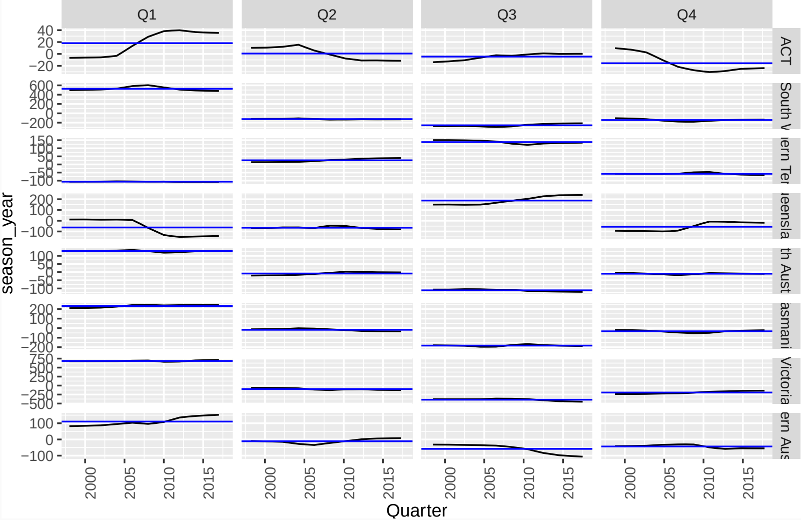
Holidays decomposition

```
dcmp <- holidays %>% STL(Trips)
dcmp
```

```
## # A dable:           640 x 7 [1Q]
## # Key:               State [8]
## # STL Decomposition: Trips = trend + season_year +
## #   remainder
##    State    Quarter Trips trend season_year remainder
##    <chr>      <qtr> <dbl> <dbl>       <dbl>     <dbl>
##   1 ACT      1998 Q1  196.  171.       -6.60      32.3
##   2 ACT      1998 Q2  127.  156.       10.3      -39.7
##   3 ACT      1998 Q3  111.  142.      -13.9      -17.2
##   4 ACT      1998 Q4  170.  130.        9.76      30.3
##   5 ACT      1999 Q1  108.  135.       -6.35     -20.7
##   6 ACT      1999 Q2  125.  148.       10.5      -33.9
##   7 ACT      1999 Q3  178.  166.      -13.2       25.5
##   8 ACT      1999 Q4  218.  177.        8.56      32.0
##   9 ACT      2000 Q1  158.  169.       -6.09      -4.74
##  10 ACT      2000 Q2  155.  151.       10.7       -7.00
```

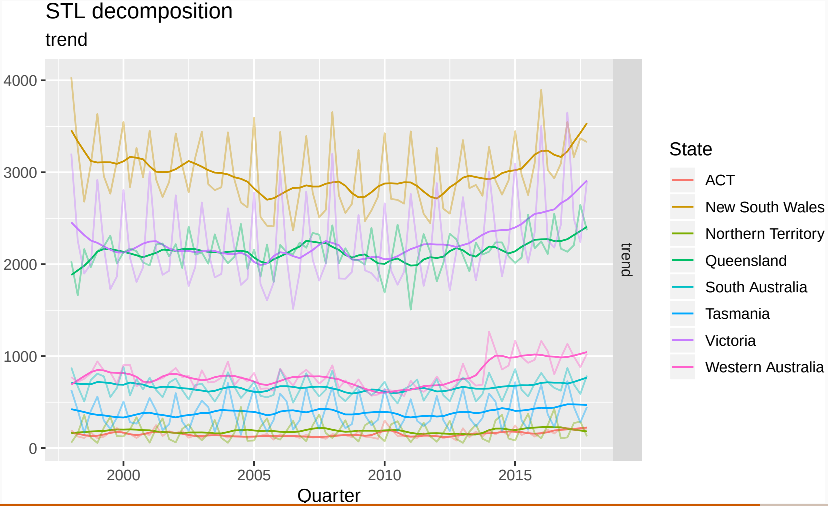
Holidays decomposition

```
dcmp %>% gg_subseries(season_year)
```



Holidays decomposition

```r
autoplot(dcmp, trend, scale_bars=FALSE) +
  autolayer(holidays, alpha=0.4)
```



Outline

1 Time series decompositions

2 Lab Session 7

Lab Session 7

Repeat the decomposition using

```
holidays %>%
  STL(Trips ~ season(window=7) + trend(window=11)) %>%
  autoplot()
```

What happens as you change `season(window = ???)` and `trend(window = ???)`?