fpp3_ch 2_Time_Series_Graphics

Ji Hyun Kim

2020 9 24

fpp3 : Ch 2 Time Series Graphics

ch2. TimeSeries Graphis

2.1 tsibble objects

- tsibble : 시간처리를 위해 tidyverse(의 tibble)를 확장한 객체

index variable

- 인덱스 변수: 시점을 표시하는 변수
- 예 : Year(연도)를 index로 하는 시계열 Observation의 생성

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------
------ tidyverse 1.3.0 --
```

```
## √ ggplot2 3.3.2     √ purrr   0.3.4
## √ tibble  3.0.3     √ dplyr   1.0.2
## √ tidyr   1.1.2     √ stringr 1.4.0
## √ readr   1.3.1     √ forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------------------------------------
tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(tsibble)
library(feasts)
```

```
## Loading required package: fabletools
```

```
library(fabletools)
```

```
y <- tsibble(Year=2015:2019, Observation=c(123,39,78,52,110), index=Year)
y
```

```
## # A tsibble: 5 x 2 [1Y]
##    Year Observation
##   <int>       <dbl>
## 1  2015         123
## 2  2016          39
## 3  2017          78
## 4  2018          52
## 5  2019         110
```

```
index(y)
```

```
## Year
```

- tibble 객체를 tsibble객체로 변환
    - 출력하면 [1M]: 월별 자료임을 표시해줌

```
z <- tibble(Month=c('2019 Jan', '2019 Feb', '2019 Mar', '2019 Apr', '2019 May'), Observation =
  c(50,23,34,30,25))
z %>%
  mutate(Month=yearmonth(Month)) %>%
  as_tsibble(index = Month)
```

```
## # A tsibble: 5 x 2 [1M]
##    Month Observation
##    <mth>       <dbl>
## 1 2019 1          50
## 2 2019 2          23
## 3 2019 3          34
## 4 2019 4          30
## 5 2019 5          25
```

```
class(z)
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

```
yearquarter(z$Month)
```

```
## <yearquarter[5]>
## [1] "2019 Q1" "2019 Q1" "2019 Q1" "2019 Q2" "2019 Q2"
## # Year starts on: January
```

```
yearmonth(z$Month)
```

```
## <yearmonth[5]>
## [1] "2019 1" "2019 2" "2019 3" "2019 4" "2019 5"
```

```
yearweek(z$Month)
```

```
## <yearweek[5]>
## [1] "2019 W01" "2019 W05" "2019 W09" "2019 W14" "2019 W18"
## # Week starts on: Monday
```

```
#as_date(z$Month), ymd(z$Month) 는 월별이므로 NA로 처리됨
```

- 인덱스 관련 함수

The key variable

- 시계열이 여러개 있으면 key 변수를 이용하여 필요한 시계열만 추출함
- 예: tsibbledate::olympic_running 올림픽 육상기록 312x4

```
library(tsibbledata)
class(olympic_running)
```

```
## [1] "tbl_ts"     "tbl_df"     "tbl"        "data.frame"
```

```
olympic_running
```

```
## # A tsibble: 312 x 4 [4Y]
## # Key:       Length, Sex [14]
##     Year Length Sex    Time
##    <int>  <int> <chr> <dbl>
##  1  1896    100 men    12
##  2  1900    100 men    11
##  3  1904    100 men    11
##  4  1908    100 men    10.8
##  5  1912    100 men    10.8
##  6  1916    100 men    NA
##  7  1920    100 men    10.8
##  8  1924    100 men    10.6
##  9  1928    100 men    10.8
## 10  1932    100 men    10.3
## # ... with 302 more rows
```

```
index(olympic_running)
```

```
## Year
```

Working with tsibble objects

- tsibbledata::PBS : 호주 월별 의료버험 약처방 65219x9
  - a10 : ATC2 == 'A10' : Antidiabetic drup(당뇨병 약) 매출
  - h02 : ATC2 == 'H02' : Corticosteroid drup(부신피질 호르몬제: 피부질환, 류마티스 등에 쓰임) 매출

```
PBS
```

```
## # A tsibble: 65,219 x 9 [1M]
## # Key:       Concession, Type, ATC1, ATC2 [336]
##      Month Concession  Type   ATC1  ATC1_desc    ATC2  ATC2_desc   Scripts  Cost
##      <mth> <chr>       <chr>  <chr> <chr>        <chr> <chr>         <dbl> <dbl>
##  1  1991 7 Concession~ Co-pa~ A     Alimentary ~ A01   STOMATOLOG~   18228 67877
##  2  1991 8 Concession~ Co-pa~ A     Alimentary ~ A01   STOMATOLOG~   15327 57011
##  3  1991 9 Concession~ Co-pa~ A     Alimentary ~ A01   STOMATOLOG~   14775 55020
##  4 1991 10 Concession~ Co-pa~ A     Alimentary ~ A01   STOMATOLOG~   15380 57222
##  5 1991 11 Concession~ Co-pa~ A     Alimentary ~ A01   STOMATOLOG~   14371 52120
##  6 1991 12 Concession~ Co-pa~ A     Alimentary ~ A01   STOMATOLOG~   15028 54299
##  7  1992 1 Concession~ Co-pa~ A     Alimentary ~ A01   STOMATOLOG~   11040 39753
##  8  1992 2 Concession~ Co-pa~ A     Alimentary ~ A01   STOMATOLOG~   15165 54405
##  9  1992 3 Concession~ Co-pa~ A     Alimentary ~ A01   STOMATOLOG~   16898 61108
## 10  1992 4 Concession~ Co-pa~ A     Alimentary ~ A01   STOMATOLOG~   18141 65356
## # ... with 65,209 more rows
```

```
# 키변수 ATC2가 'A10'(당뇨병약)만 추출
PBS %>%
  filter(ATC2 == "A10")
```

```
## # A tsibble: 816 x 9 [1M]
## # Key:       Concession, Type, ATC1, ATC2 [4]
##      Month Concession  Type   ATC1  ATC1_desc    ATC2  ATC2_desc  Scripts   Cost
##      <mth> <chr>       <chr>  <chr> <chr>        <chr> <chr>        <dbl>  <dbl>
##  1  1991 7 Concession~ Co-pa~ A     Alimentary ~ A10   ANTIDIABE~   89733 2.09e6
##  2  1991 8 Concession~ Co-pa~ A     Alimentary ~ A10   ANTIDIABE~   77101 1.80e6
##  3  1991 9 Concession~ Co-pa~ A     Alimentary ~ A10   ANTIDIABE~   76255 1.78e6
##  4 1991 10 Concession~ Co-pa~ A     Alimentary ~ A10   ANTIDIABE~   78681 1.85e6
##  5 1991 11 Concession~ Co-pa~ A     Alimentary ~ A10   ANTIDIABE~   70554 1.69e6
##  6 1991 12 Concession~ Co-pa~ A     Alimentary ~ A10   ANTIDIABE~   75814 1.84e6
##  7  1992 1 Concession~ Co-pa~ A     Alimentary ~ A10   ANTIDIABE~   64186 1.56e6
##  8  1992 2 Concession~ Co-pa~ A     Alimentary ~ A10   ANTIDIABE~   75899 1.73e6
##  9  1992 3 Concession~ Co-pa~ A     Alimentary ~ A10   ANTIDIABE~   89445 2.05e6
## 10  1992 4 Concession~ Co-pa~ A     Alimentary ~ A10   ANTIDIABE~   97315 2.23e6
## # ... with 806 more rows
```

```
# 키변수 ATC2가 'A10'(당뇨병약)에 대해 월, 양도여부, 지급형태, 비용만 추출
PBS %>%
  filter(ATC2 == "A10") %>%
  select(Month, Concession, Type, Cost)
```

```
## # A tsibble: 816 x 4 [1M]
## # Key:       Concession, Type [4]
##      Month Concession   Type           Cost
##      <mth> <chr>        <chr>         <dbl>
##  1  1991 7 Concessional Co-payments 2092878
##  2  1991 8 Concessional Co-payments 1795733
##  3  1991 9 Concessional Co-payments 1777231
##  4 1991 10 Concessional Co-payments 1848507
##  5 1991 11 Concessional Co-payments 1686458
##  6 1991 12 Concessional Co-payments 1843079
##  7  1992 1 Concessional Co-payments 1564702
##  8  1992 2 Concessional Co-payments 1732508
##  9  1992 3 Concessional Co-payments 2046102
## 10  1992 4 Concessional Co-payments 2225977
## # ... with 806 more rows
```

- 기술 통계량 : 월별 비용 합계(summarize는 그룹지정이 없으면 인덱스 변수 기준으로 요약함)

```
PBS %>%
  filter(ATC2=='A10') %>%
  select(Month, Concession, Type, Cost) %>%
  summarise(TotalC = sum(Cost))
```

```
## # A tsibble: 204 x 2 [1M]
##      Month  TotalC
##      <mth>   <dbl>
##  1  1991 7 3526591
##  2  1991 8 3180891
##  3  1991 9 3252221
##  4 1991 10 3611003
##  5 1991 11 3565869
##  6 1991 12 4306371
##  7  1992 1 5088335
##  8  1992 2 2814520
##  9  1992 3 2985811
## 10  1992 4 3204780
## # ... with 194 more rows
```

```
PBS %>%
  filter(ATC2=='A10') %>%
  select(Month, Cost) %>%
  summarise(TotalC = sum(Cost))
```

```
## # A tsibble: 204 x 2 [1M]
##      Month  TotalC
##      <mth>   <dbl>
##  1  1991 7 3526591
##  2  1991 8 3180891
##  3  1991 9 3252221
##  4 1991 10 3611003
##  5 1991 11 3565869
##  6 1991 12 4306371
##  7  1992 1 5088335
##  8  1992 2 2814520
##  9  1992 3 2985811
## 10  1992 4 3204780
## # ... with 194 more rows
```

```
PBS %>%
  filter(ATC2=='A10') %>%
  select(Cost) %>%
  summarise(TotalC = sum(Cost))
```

```
## # A tsibble: 204 x 2 [1M]
##      Month  TotalC
##      <mth>   <dbl>
##  1  1991 7 3526591
##  2  1991 8 3180891
##  3  1991 9 3252221
##  4 1991 10 3611003
##  5 1991 11 3565869
##  6 1991 12 4306371
##  7  1992 1 5088335
##  8  1992 2 2814520
##  9  1992 3 2985811
## 10  1992 4 3204780
## # ... with 194 more rows
```

```
a10 <- PBS %>%
  filter(ATC2=='A10') %>%
  select(Cost) %>%
  summarise(TotalC = sum(Cost)) %>%
  mutate(Cost = TotalC/1e6)
a10
```

```
## # A tsibble: 204 x 3 [1M]
##      Month  TotalC  Cost
##      <mth>   <dbl> <dbl>
##  1  1991 7 3526591  3.53
##  2  1991 8 3180891  3.18
##  3  1991 9 3252221  3.25
##  4 1991 10 3611003  3.61
##  5 1991 11 3565869  3.57
##  6 1991 12 4306371  4.31
##  7  1992 1 5088335  5.09
##  8  1992 2 2814520  2.81
##  9  1992 3 2985811  2.99
## 10  1992 4 3204780  3.20
## # ... with 194 more rows
```

read_csv/convert to a tsibble

- prison_population.csv

```
# tibble로 읽기. 보통 날짜는 자동으로 처리됨
prison <- readr::read_csv("https://OTexts.com/fpp3/extrafiles/prison_population.csv")
```

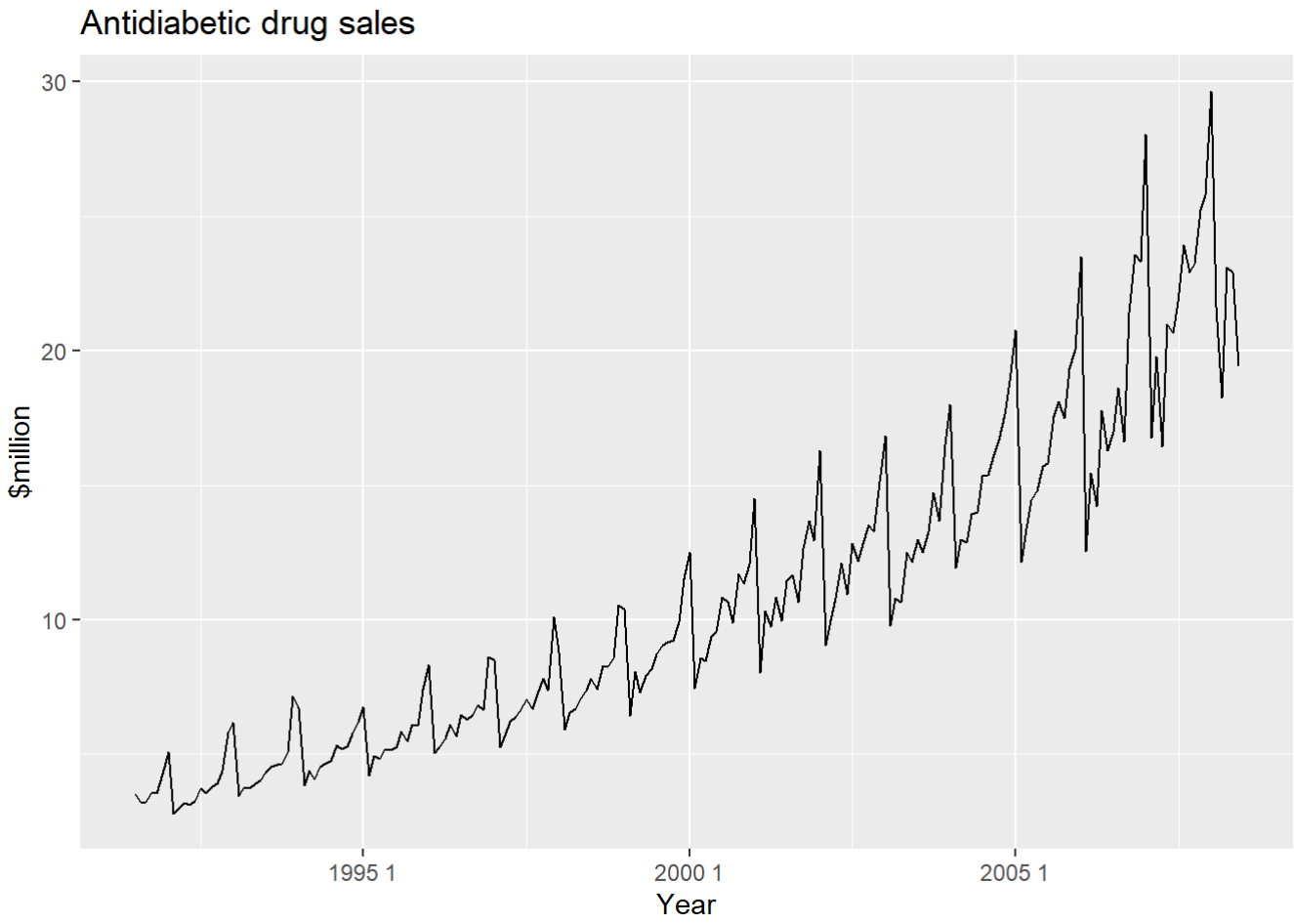
```
## Parsed with column specification:
## cols(
##   Date = col_date(format = ""),
##   State = col_character(),
##   Gender = col_character(),
##   Legal = col_character(),
##   Indigenous = col_character(),
##   Count = col_double()
## )
```

```
prison <- prison %>%
  mutate(Quarter = yearquarter(Date)) %>%
  select(-Date) %>%
  as_tsibble(key= c(State, Gender, Legal, Indigenous), index=Quarter)
```

2.2 Time plots

```
melsyd_economy <- ansett %>%
  filter(Airports == "MEL-SYD", Class=="Economy")
melsyd_economy %>%
  autoplot(Passengers) +
    labs(title = "Ansett economy class passengers", subtitle = "Melbourne-Sydney") +
    xlab("Year")
```


```
a10 %>% autoplot(Cost) +
  ggtitle("Antidiabetic drug sales") +
  ylab("$million") + xlab("Year")
```



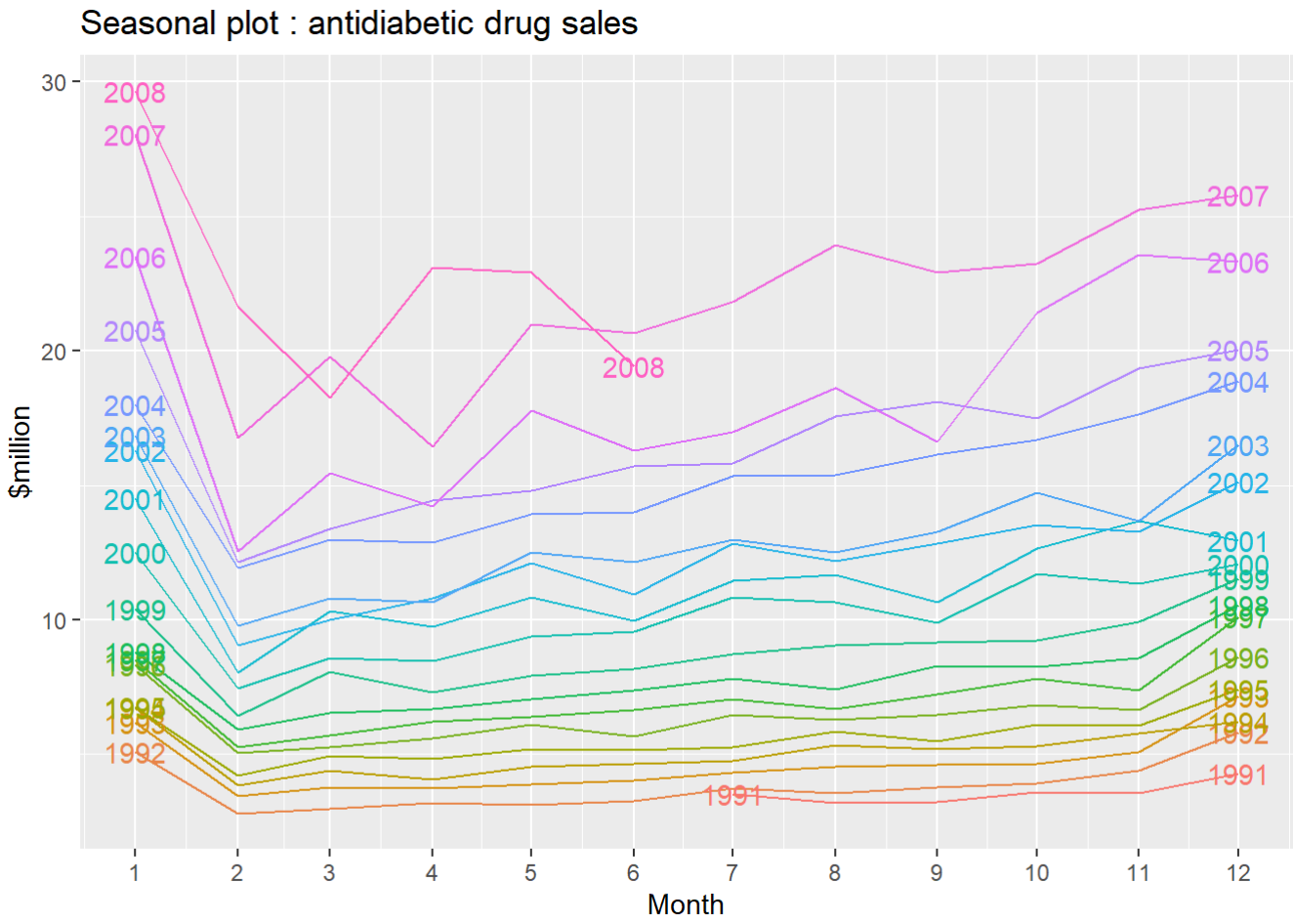
2.3 Time series patterns

시계열의 4가지 성분

- 추세성분(Trend)
- 계절성분(Seasonal)
- 순환성분(Cyclic)
- 랜덤성분, 잔여성분(Random,Residual)

2.4 Seasonal plots

```
library(feasts)
a10 %>% gg_season(Cost, labels = "both") +
  ylab("$million")+
  ggtitle("Seasonal plot : antidiabetic drug sales")
```

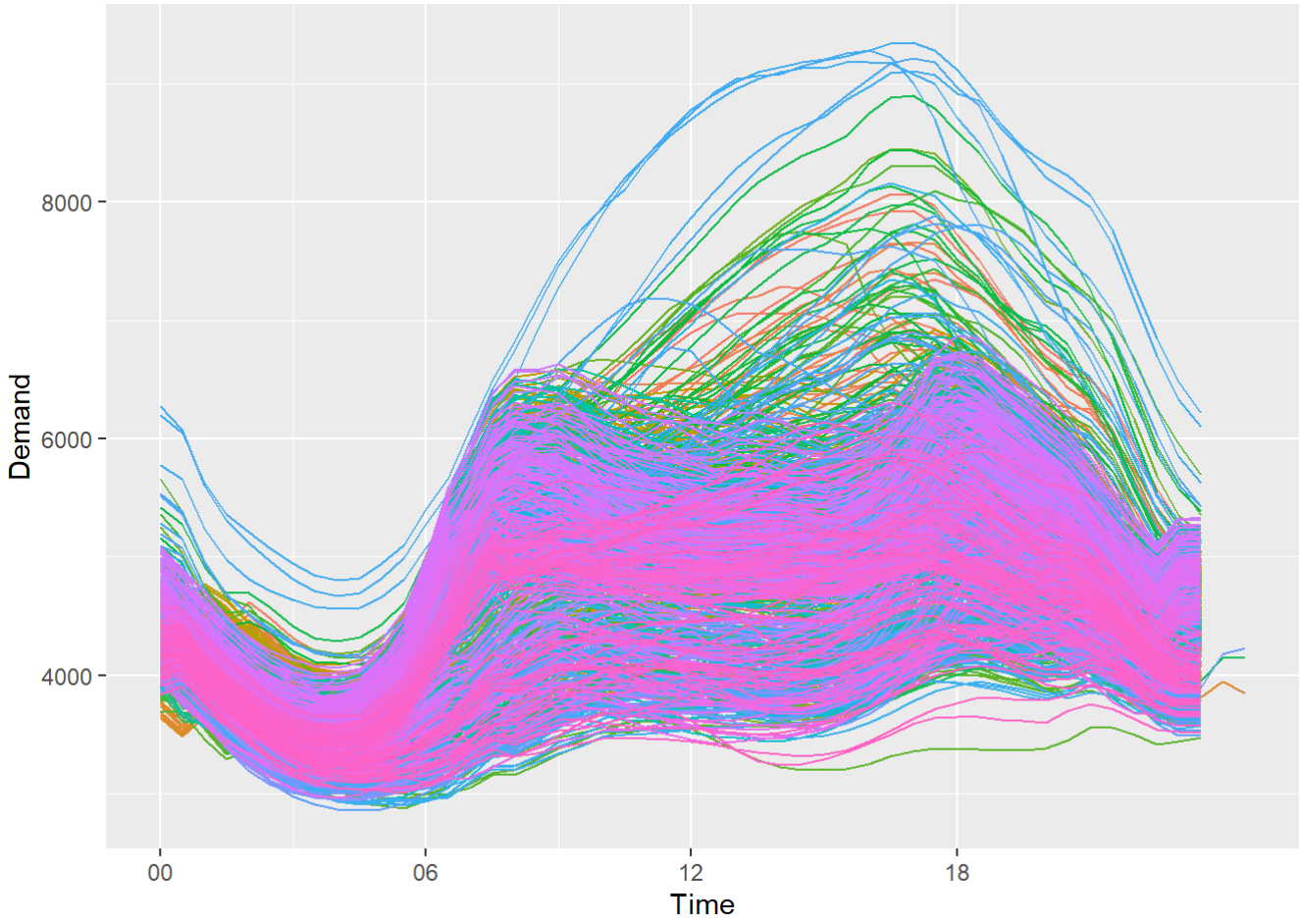


Multiple seasonal periods

```
vic_elec
```

```
## # A tsibble: 52,608 x 5 [30m] <Australia/Melbourne>
##    Time                Demand Temperature Date       Holiday
##    <dttm>               <dbl>       <dbl> <date>     <lgl>
##  1 2012-01-01 00:00:00  4383.        21.4 2012-01-01 TRUE
##  2 2012-01-01 00:30:00  4263.        21.0 2012-01-01 TRUE
##  3 2012-01-01 01:00:00  4049.        20.7 2012-01-01 TRUE
##  4 2012-01-01 01:30:00  3878.        20.6 2012-01-01 TRUE
##  5 2012-01-01 02:00:00  4036.        20.4 2012-01-01 TRUE
##  6 2012-01-01 02:30:00  3866.        20.2 2012-01-01 TRUE
##  7 2012-01-01 03:00:00  3694.        20.1 2012-01-01 TRUE
##  8 2012-01-01 03:30:00  3562.        19.6 2012-01-01 TRUE
##  9 2012-01-01 04:00:00  3433.        19.1 2012-01-01 TRUE
## 10 2012-01-01 04:30:00  3359.        19.0 2012-01-01 TRUE
## # ... with 52,598 more rows
```

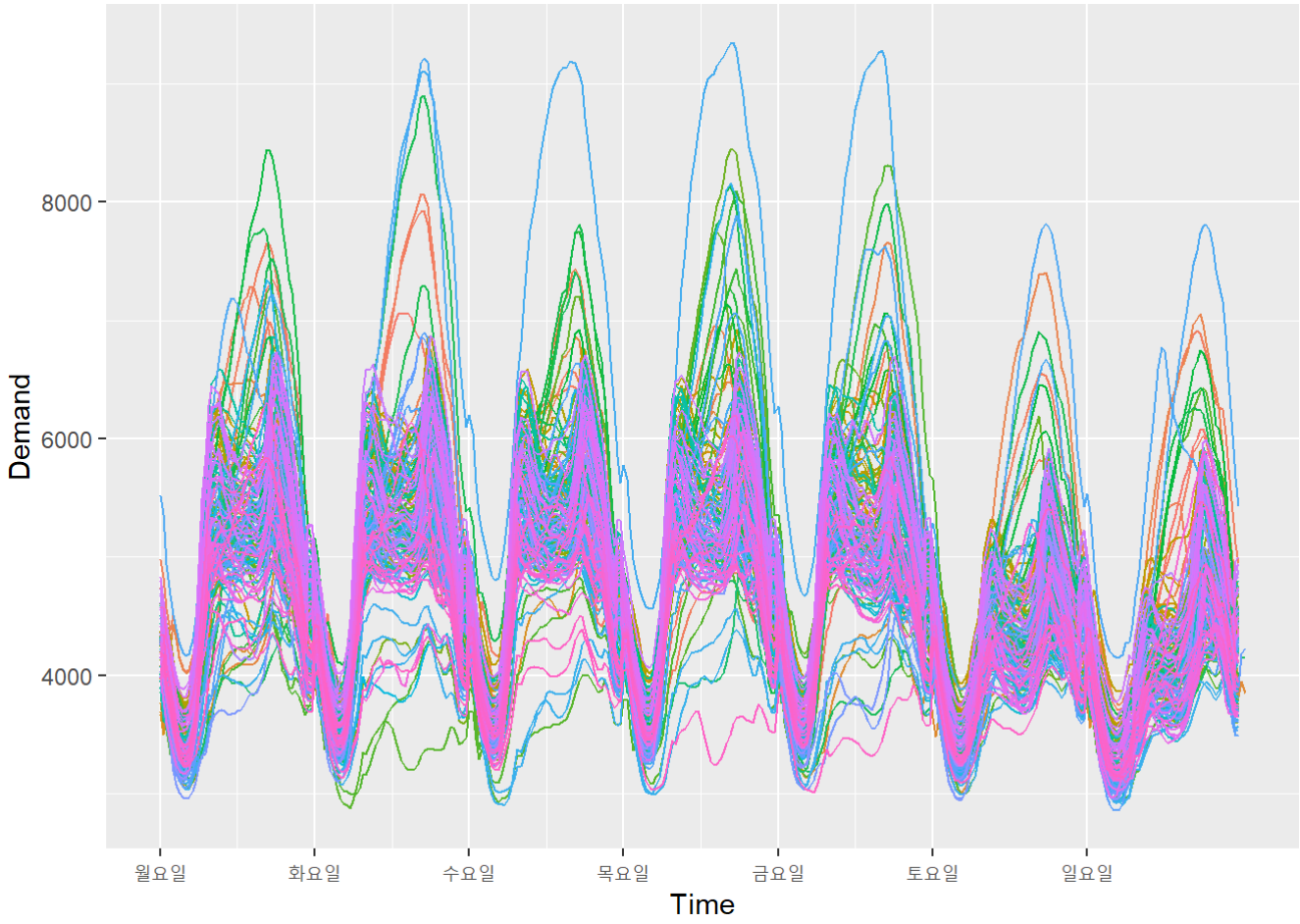
```
vic_elec %>%
  gg_season(Demand,period = "day") +
  theme(legend.position = "none")
```




```
vic_elec
```

```
## # A tsibble: 52,608 x 5 [30m] <Australia/Melbourne>
##    Time                Demand Temperature Date       Holiday
##    <dttm>               <dbl>       <dbl> <date>     <lgl>
##  1 2012-01-01 00:00:00  4383.        21.4 2012-01-01 TRUE
##  2 2012-01-01 00:30:00  4263.        21.0 2012-01-01 TRUE
##  3 2012-01-01 01:00:00  4049.        20.7 2012-01-01 TRUE
##  4 2012-01-01 01:30:00  3878.        20.6 2012-01-01 TRUE
##  5 2012-01-01 02:00:00  4036.        20.4 2012-01-01 TRUE
##  6 2012-01-01 02:30:00  3866.        20.2 2012-01-01 TRUE
##  7 2012-01-01 03:00:00  3694.        20.1 2012-01-01 TRUE
##  8 2012-01-01 03:30:00  3562.        19.6 2012-01-01 TRUE
##  9 2012-01-01 04:00:00  3433.        19.1 2012-01-01 TRUE
## 10 2012-01-01 04:30:00  3359.        19.0 2012-01-01 TRUE
## # ... with 52,598 more rows
```

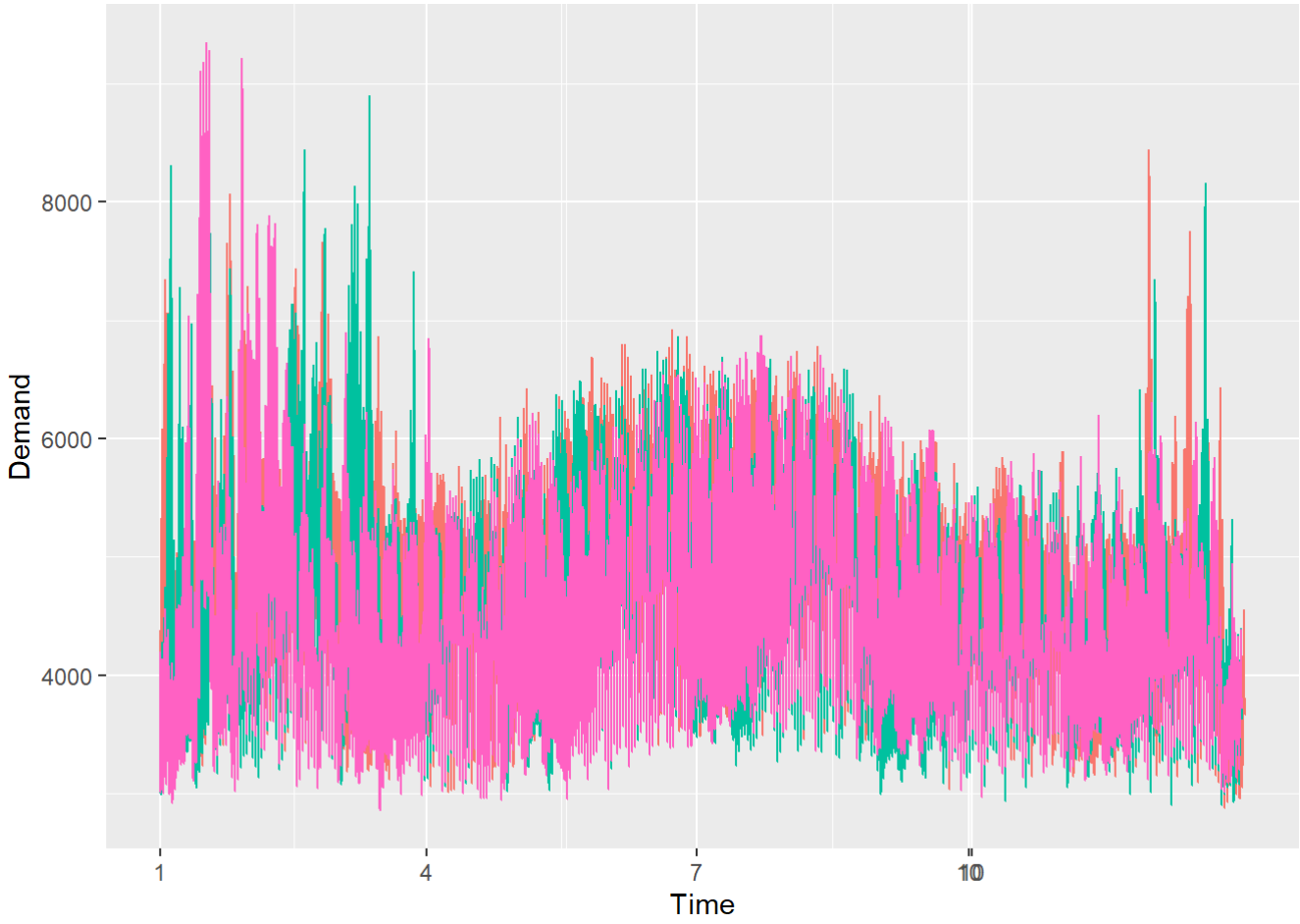
```
vic_elec %>%
  gg_season(Demand,period = "week") +
  theme(legend.position = "none")
```




```
vic_elec
```

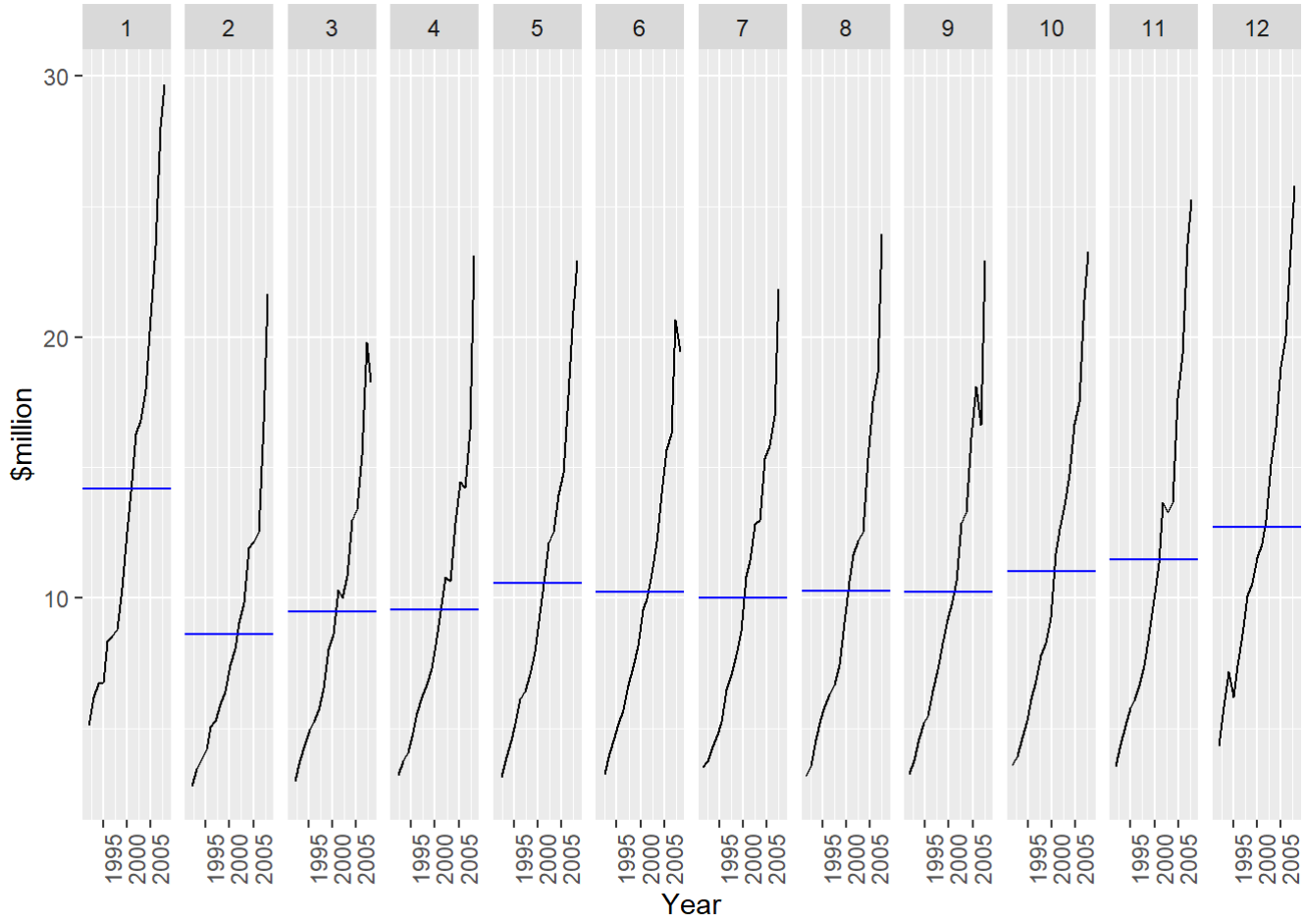
```
## # A tsibble: 52,608 x 5 [30m] <Australia/Melbourne>
##    Time                Demand Temperature Date       Holiday
##    <dttm>               <dbl>       <dbl> <date>     <lgl>
##  1 2012-01-01 00:00:00  4383.        21.4 2012-01-01 TRUE
##  2 2012-01-01 00:30:00  4263.        21.0 2012-01-01 TRUE
##  3 2012-01-01 01:00:00  4049.        20.7 2012-01-01 TRUE
##  4 2012-01-01 01:30:00  3878.        20.6 2012-01-01 TRUE
##  5 2012-01-01 02:00:00  4036.        20.4 2012-01-01 TRUE
##  6 2012-01-01 02:30:00  3866.        20.2 2012-01-01 TRUE
##  7 2012-01-01 03:00:00  3694.        20.1 2012-01-01 TRUE
##  8 2012-01-01 03:30:00  3562.        19.6 2012-01-01 TRUE
##  9 2012-01-01 04:00:00  3433.        19.1 2012-01-01 TRUE
## 10 2012-01-01 04:30:00  3359.        19.0 2012-01-01 TRUE
## # ... with 52,598 more rows
```

```
vic_elec %>%
  gg_season(Demand,period = "year") +
  theme(legend.position = "none")
```



2.5 Seasonal subseries plots

```
a10 %>%
  gg_subseries(Cost)+
  ylab("$million") +
  xlab("Year")
```




```
  ggtitle("Seasonal subseries plot : antidiabetic drug sales")
```

```
## $title
## [1] "Seasonal subseries plot : antidiabetic drug sales"
##
## attr(,"class")
## [1] "labels"
```

Example: Australian holidya tourism

tsibble::tourism

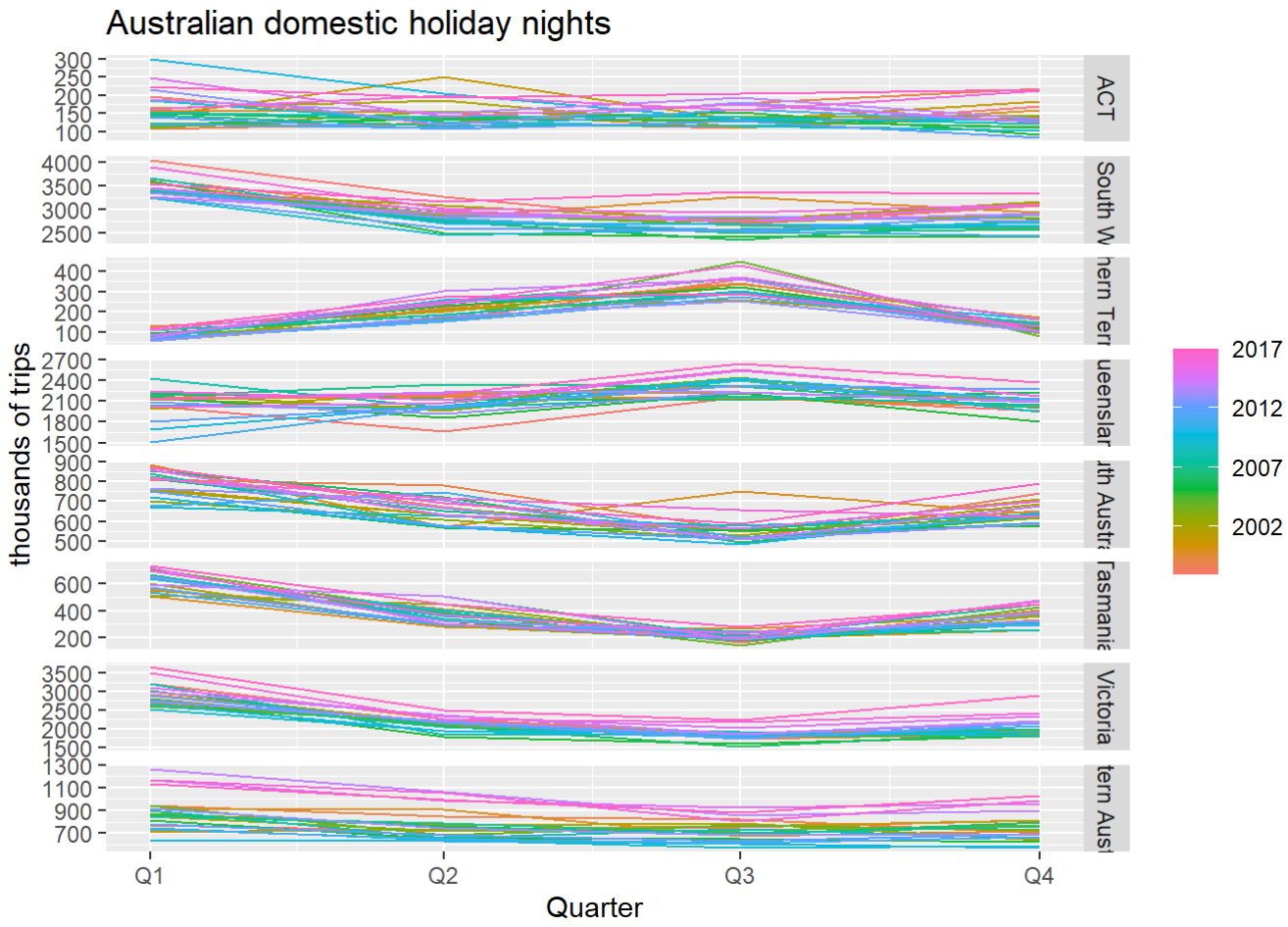
```
tourism
```

```
## # 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 South Australia Business  135.
##  2 1998 Q2 Adelaide South Australia Business  110.
##  3 1998 Q3 Adelaide South Australia Business  166.
##  4 1998 Q4 Adelaide South Australia Business  127.
##  5 1999 Q1 Adelaide South Australia Business  137.
##  6 1999 Q2 Adelaide South Australia Business  200.
##  7 1999 Q3 Adelaide South Australia Business  169.
##  8 1999 Q4 Adelaide South Australia Business  134.
##  9 2000 Q1 Adelaide South Australia Business  154.
## 10 2000 Q2 Adelaide South Australia Business  169.
## # ... with 24,310 more rows
```

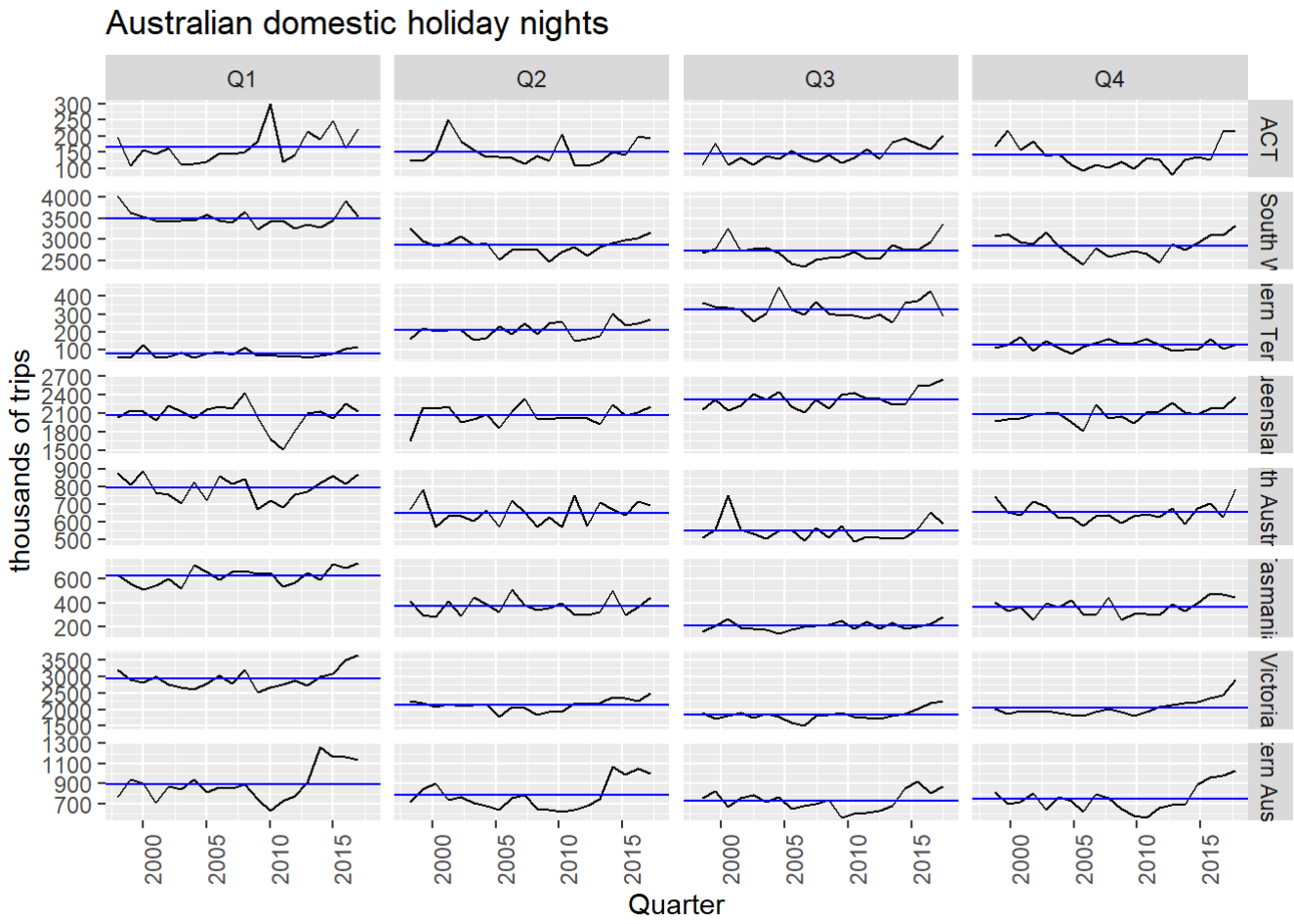
```
holidays <- tourism %>%
  filter(Purpose == "Holiday") %>%
  group_by(State) %>%
  summarise(Trips = sum(Trips))
holidays
```

```
## # A tsibble: 640 x 3 [1Q]
## # Key:       State [8]
##    State Quarter Trips
##    <chr>   <qtr> <dbl>
##  1 ACT   1998 Q1  196.
##  2 ACT   1998 Q2  127.
##  3 ACT   1998 Q3  111.
##  4 ACT   1998 Q4  170.
##  5 ACT   1999 Q1  108.
##  6 ACT   1999 Q2  125.
##  7 ACT   1999 Q3  178.
##  8 ACT   1999 Q4  218.
##  9 ACT   2000 Q1  158.
## 10 ACT   2000 Q2  155.
## # ... with 630 more rows
```

```
holidays %>% gg_season(Trips) +
  ylab("thousands of trips") +
  ggtitle("Australian domestic holiday nights")
```




```
holidays %>% gg_subseries(Trips) +
  ylab("thousands of trips") +
  ggtitle("Australian domestic holiday nights")
```



2.6 Scatterplots

- 다수의 시계열은 산점도로 비교 가능
- 2014년 빅토리아 30분 간격 전기수요

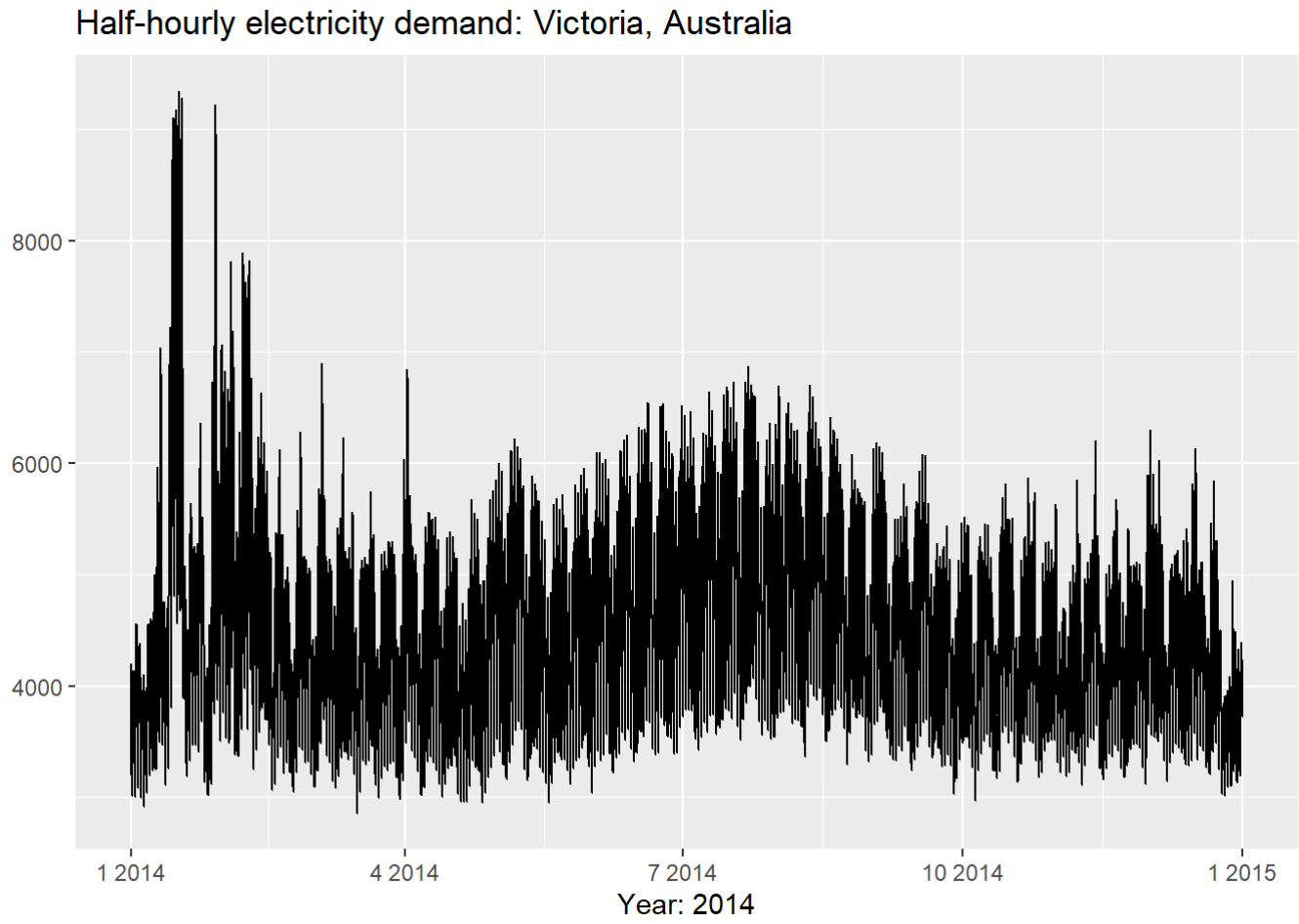
```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:tsibble':
##
##     interval
```

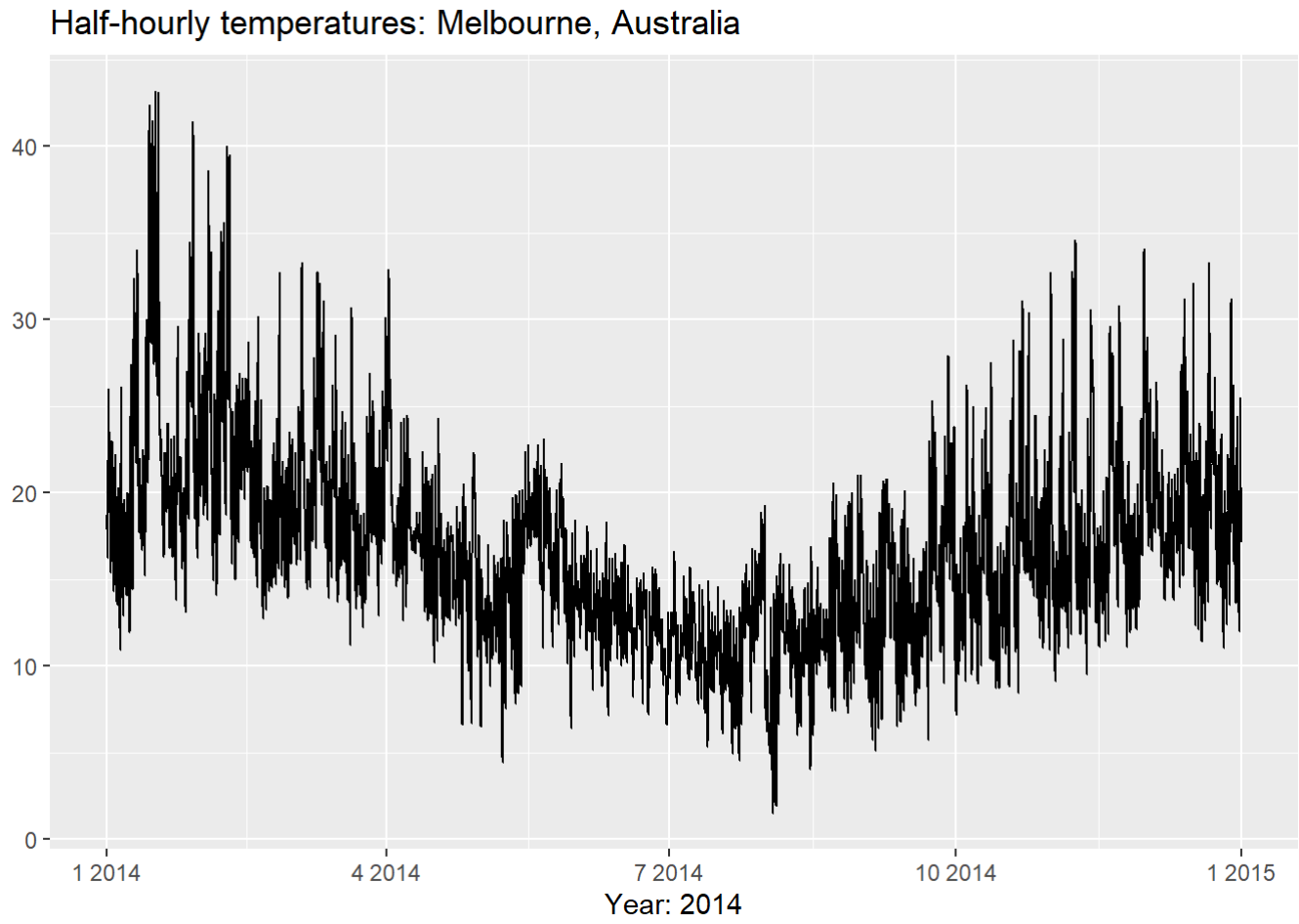
```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
vic_elec %>%
  filter(year(Time) == 2014) %>%
  autoplot(Demand) +
  xlab("Year: 2014") + ylab(NULL) +
  ggtitle("Half-hourly electricity demand: Victoria, Australia")
```



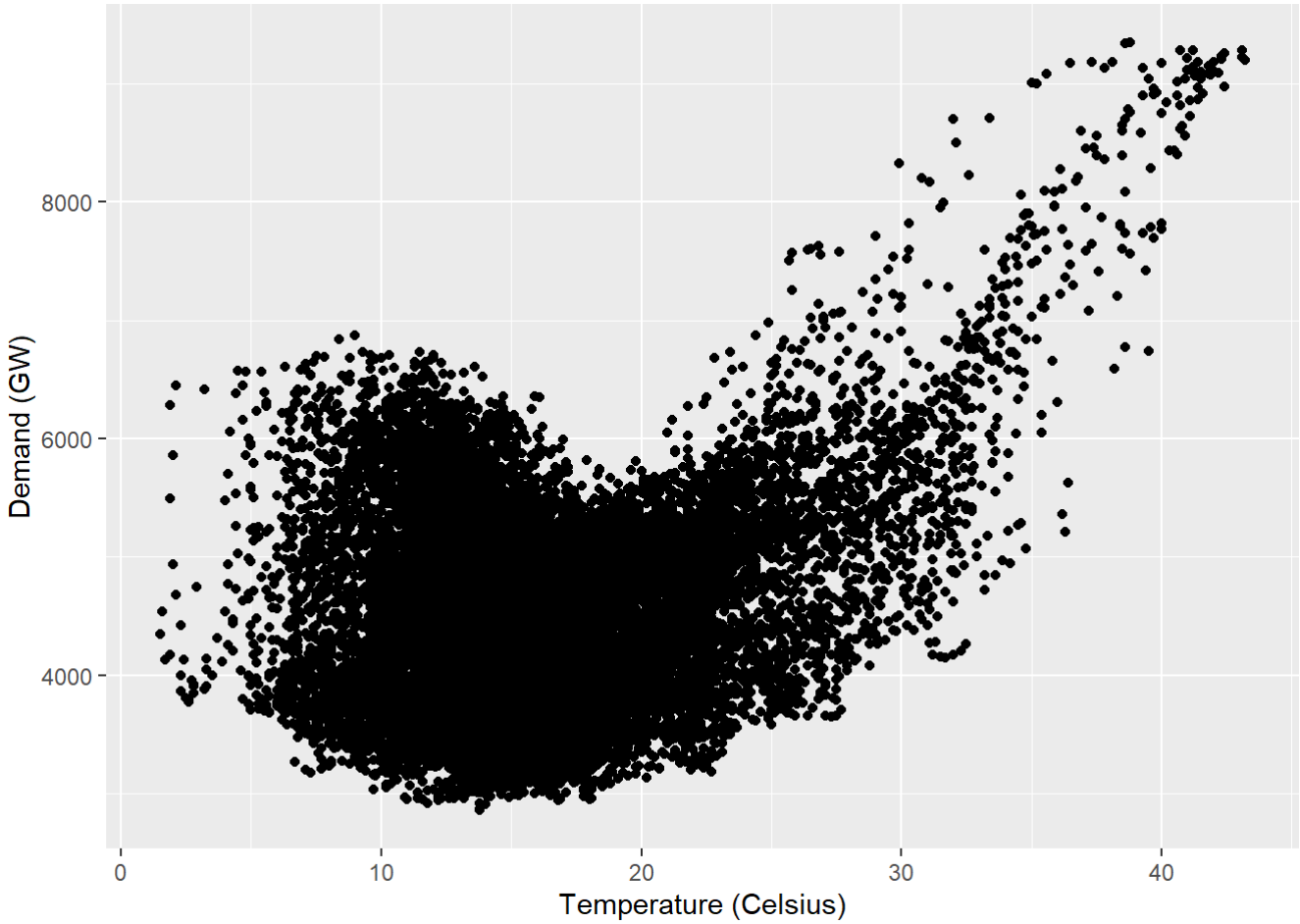
- 2014년 빅토리아 기온

```
vic_elec %>%
  filter(year(Time) == 2014) %>%
  autoplot(Temperature) +
  xlab("Year: 2014") + ylab(NULL) +
  ggtitle("Half-hourly temperatures: Melbourne, Australia")
```



- 기온과 전기수요:
  - 기온이 높으면 전기수요도 높은 경향
  - 기온이 낮으면 전기수요가 높은 경향

```
vic_elec %>%
  filter(year(Time) == 2014) %>%
  ggplot(aes(x = Temperature, y = Demand)) +
    geom_point() +
    ylab("Demand (GW)") + xlab("Temperature (Celsius)")
```



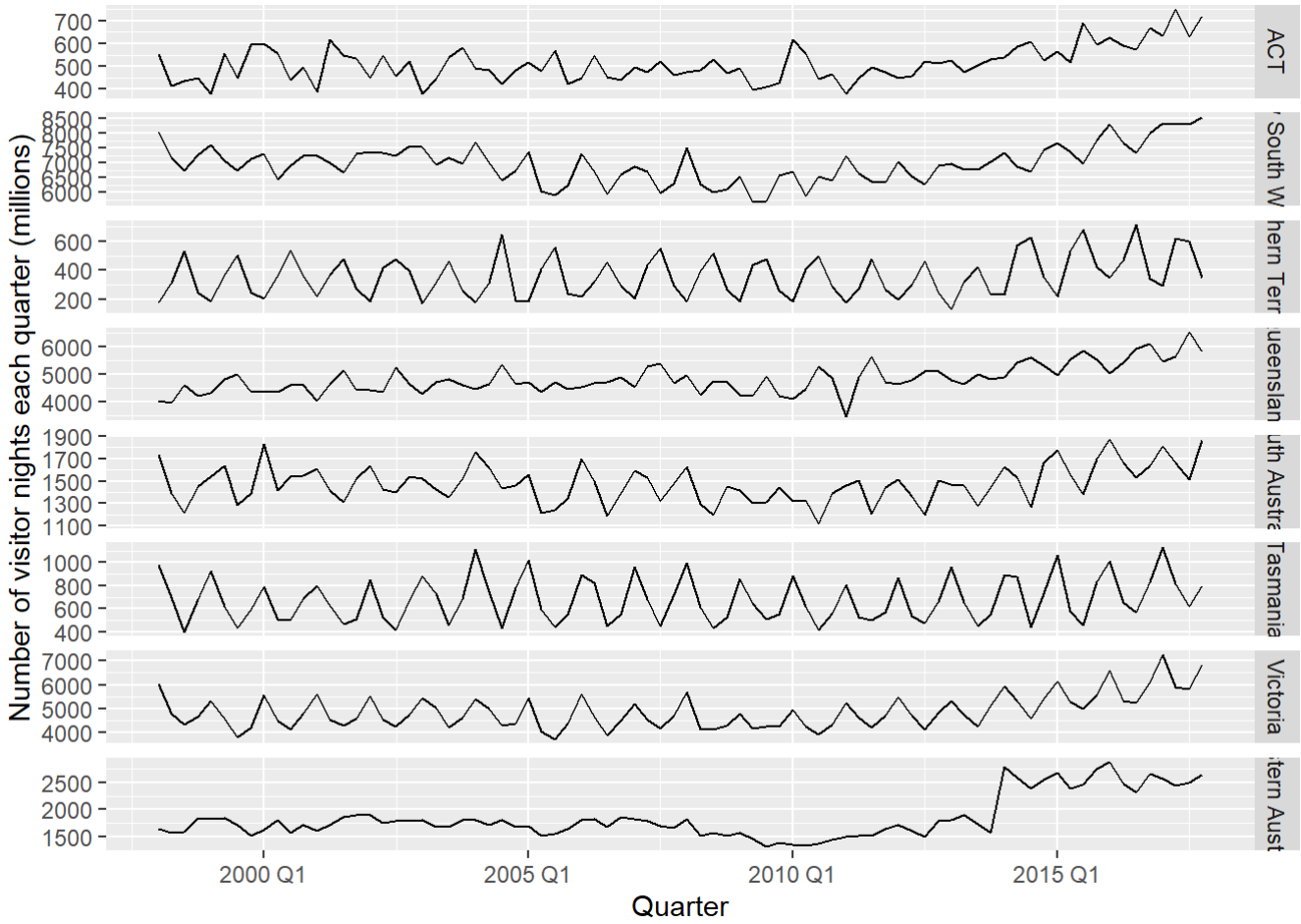
Correlation

Scatterplot matrices

```
visitors <- tourism %>%
  group_by(State) %>%
  summarise(Trips = sum(Trips))
visitors
```

```
## # A tsibble: 640 x 3 [1Q]
## # Key:       State [8]
##    State Quarter Trips
##    <chr>   <qtr> <dbl>
##  1 ACT   1998 Q1  551.
##  2 ACT   1998 Q2  416.
##  3 ACT   1998 Q3  436.
##  4 ACT   1998 Q4  450.
##  5 ACT   1999 Q1  379.
##  6 ACT   1999 Q2  558.
##  7 ACT   1999 Q3  449.
##  8 ACT   1999 Q4  595.
##  9 ACT   2000 Q1  600.
## 10 ACT   2000 Q2  557.
## # ... with 630 more rows
```

```
visitors %>%
  ggplot(aes(x = Quarter, y = Trips)) +
    geom_line() +
    facet_grid(vars(State), scales = "free_y") +
    ylab("Number of visitor nights each quarter (millions)")
```



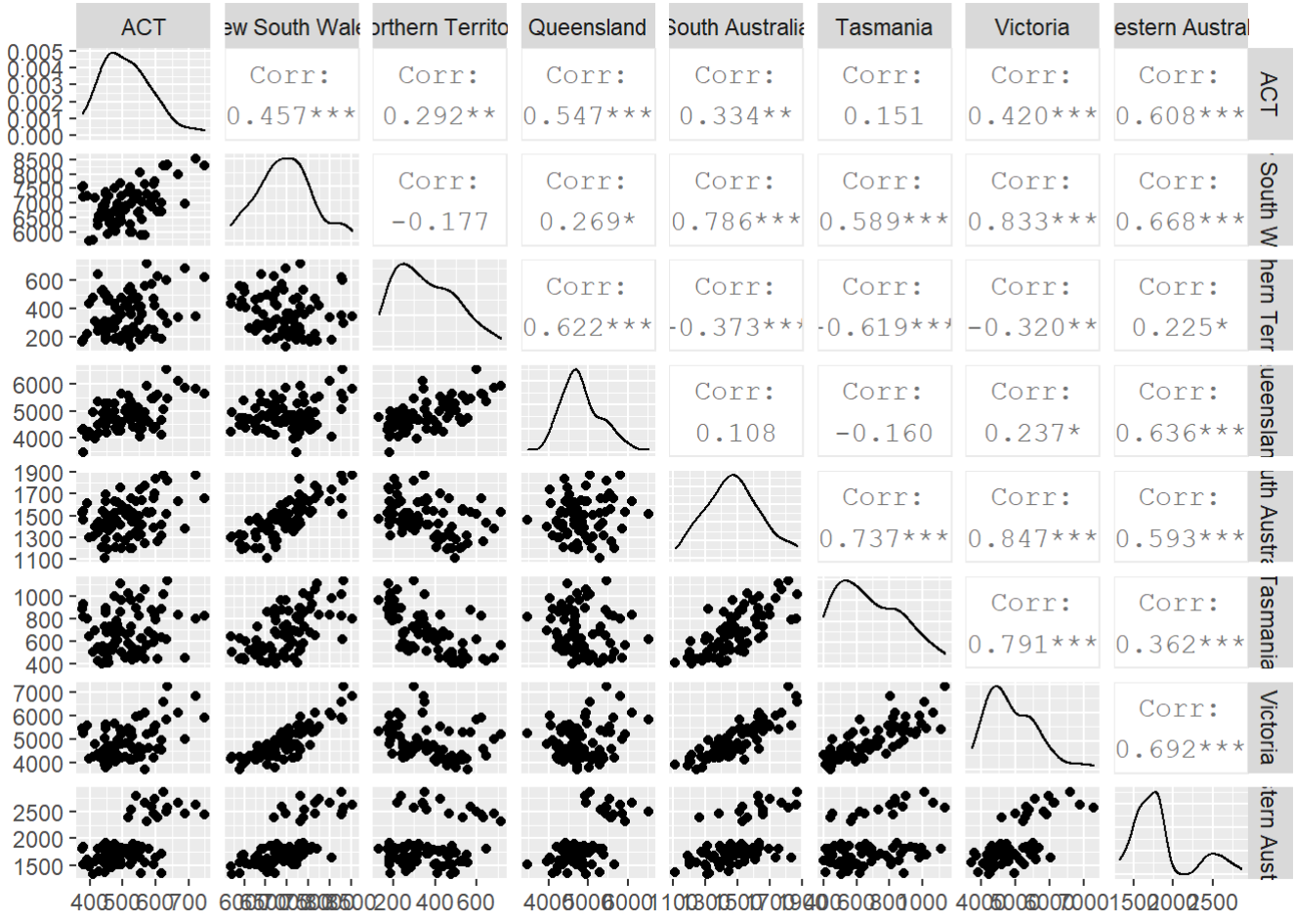

```
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
visitors %>%
  spread(State,Trips) %>% head()
```

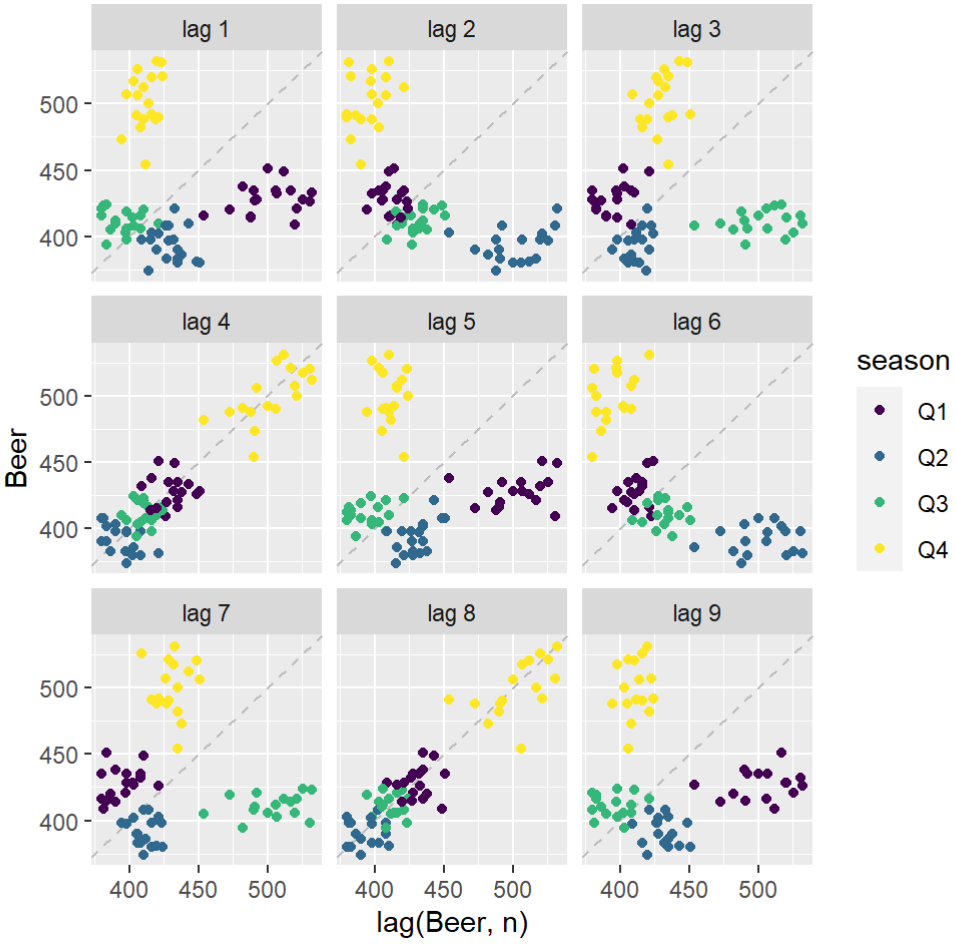
```
## # A tsibble: 6 x 9 [1Q]
##   Quarter   ACT `New South Wale~ `Northern Terri~ Queensland `South Australi~
##     <qtr> <dbl>            <dbl>            <dbl>      <dbl>            <dbl>
## 1 1998 Q1  551.             8040.              181.       4041.            1735.
## 2 1998 Q2  416.             7166.              314.       3968.            1395.
## 3 1998 Q3  436.             6748.              528.       4594.            1213.
## 4 1998 Q4  450.             7282.              248.       4203.            1453.
## 5 1999 Q1  379.             7585.              185.       4332.            1541.
## 6 1999 Q2  558.             7054.              366.       4824.            1636.
## # ... with 3 more variables: Tasmania <dbl>, Victoria <dbl>, `Western
## #   Australia` <dbl>
```

```
visitors %>%
  spread(State, Trips) %>%
  GGally::ggpairs(columns = 2:9)
```



2.7 Lag plots

```
recent_production <- aus_production %>%
  filter(year(Quarter) >= 1992)
recent_production %>% gg_lag(Beer, geom="point")
```



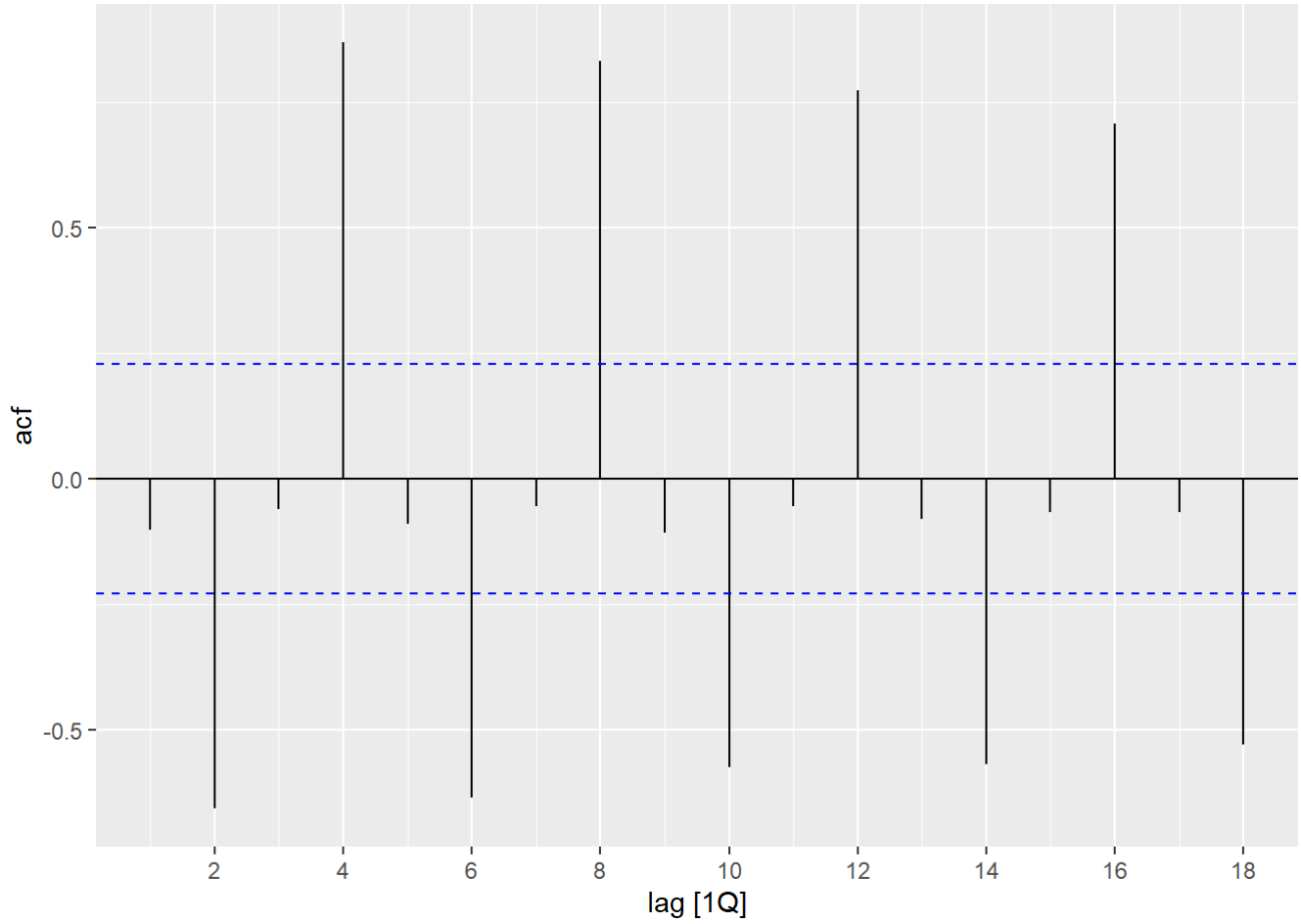
2.8 ACF(Autocorrelation)

- 자기 상관계수(Autocorelation, ACF, ACF함수): 원시계열과 시차변수간 상관계수 $ r_k = $
- 예 : 맥주 생산량 (분기별 자료)

```
recent_production %>% ACF(Beer, lag_max = 9)
```

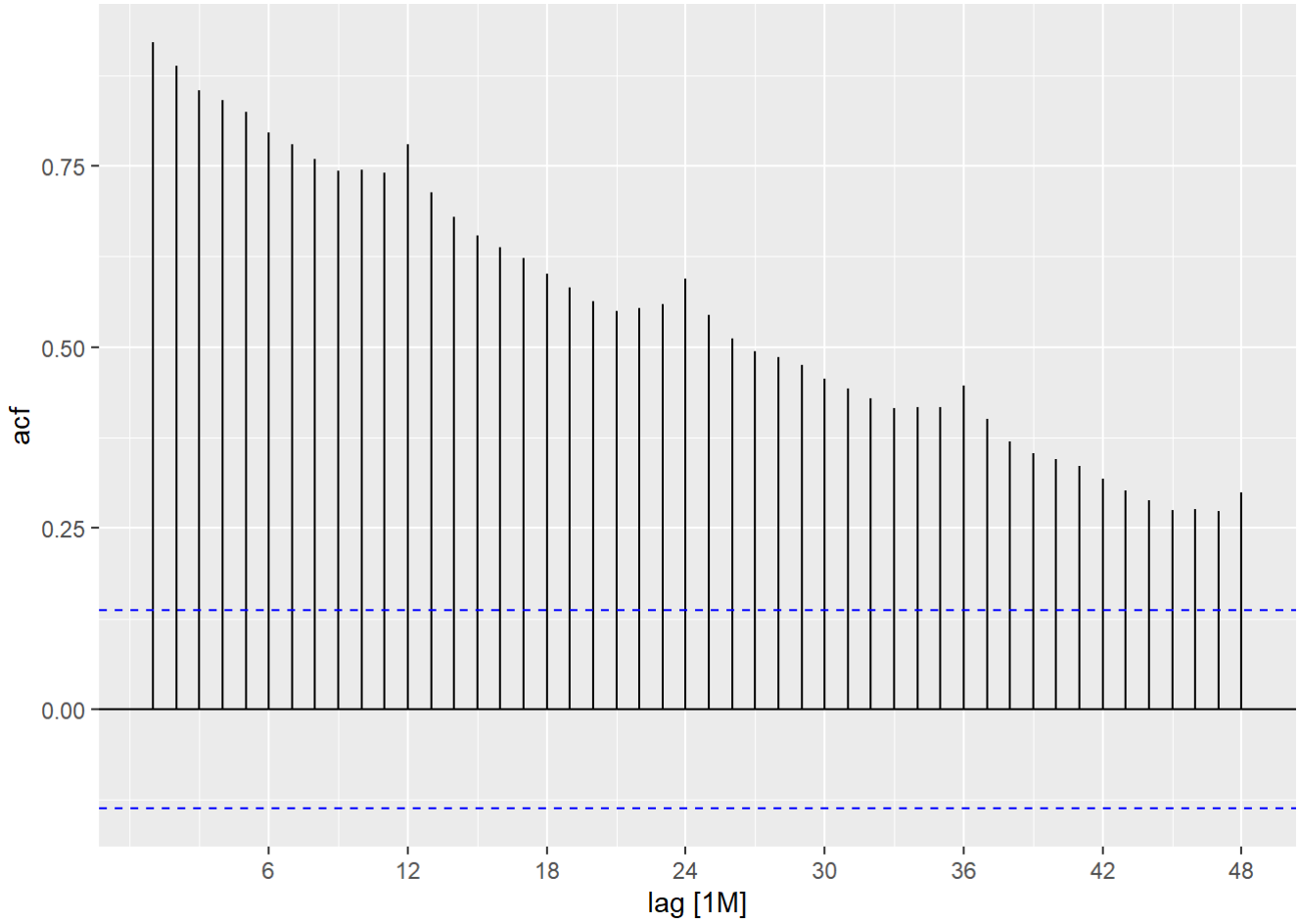
```
## # A tsibble: 9 x 2 [1Q]
##     lag     acf
##   <lag>   <dbl>
## 1    1Q -0.102
## 2    2Q -0.657
## 3    3Q -0.0603
## 4    4Q  0.869
## 5    5Q -0.0892
## 6    6Q -0.635
## 7    7Q -0.0542
## 8    8Q  0.832
## 9    9Q -0.108
```

```
recent_production %>% ACF(Beer) %>% autoplot()
```



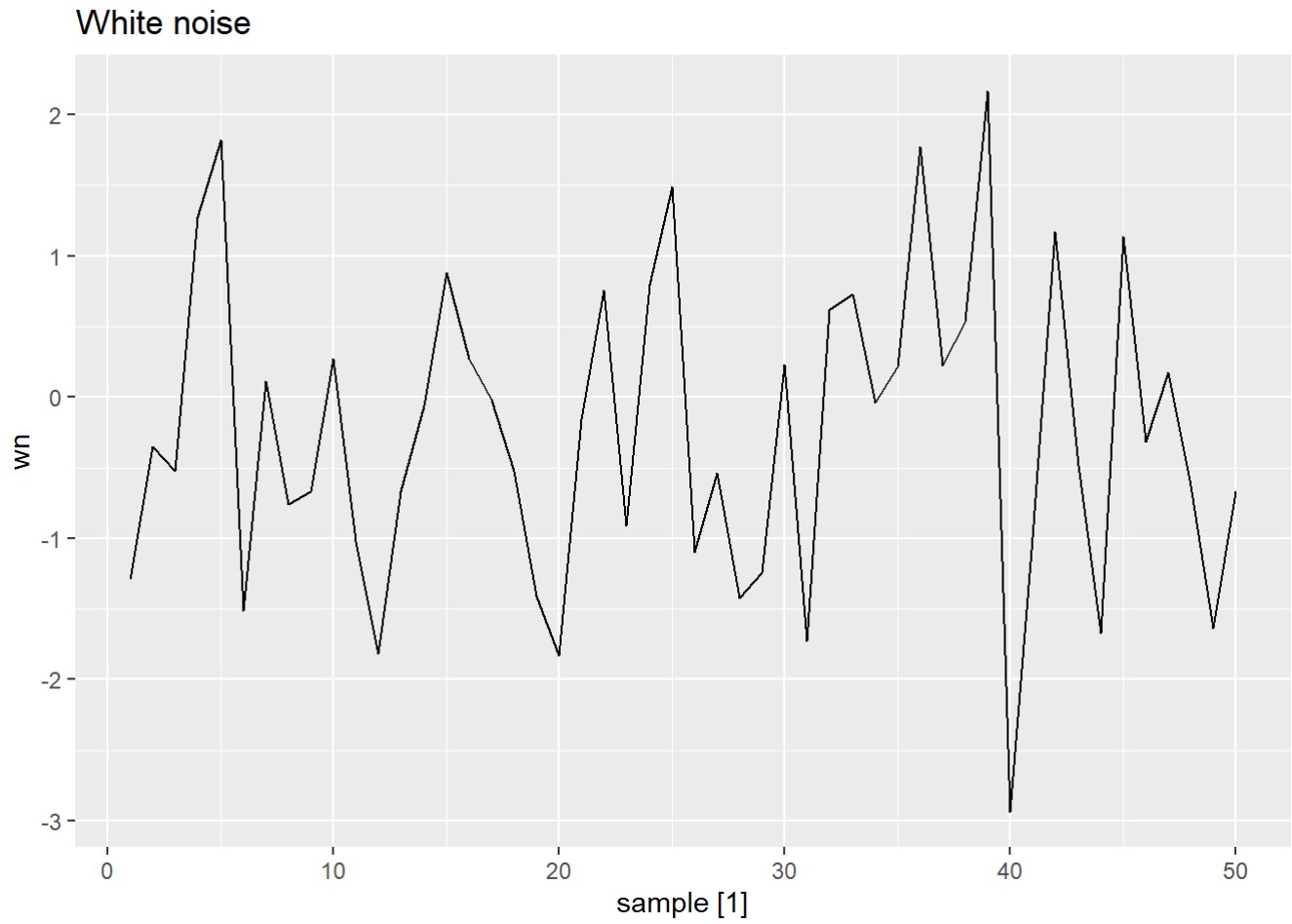
Trend and seasonality in ACF

```
a10 %>% ACF(Cost, lag_max = 48) %>% autoplot()
```

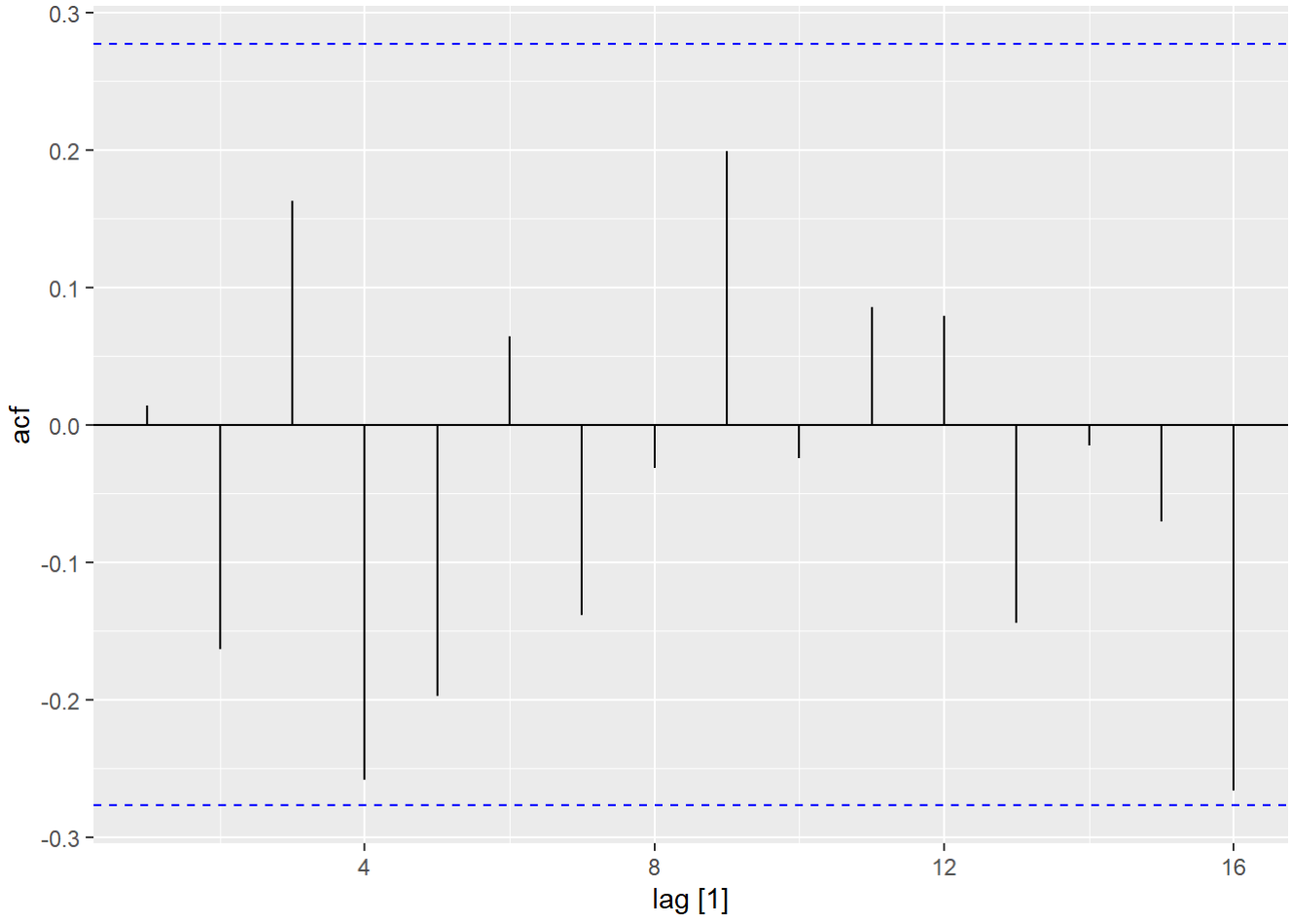


2.9 White noise

```
set.seed(30)
y <- tsibble(sample = 1:50, wn = rnorm(50), index = sample)
y %>% autoplot(wn) + ggtitle("White noise")
```




```
y %>% ACF(wn) %>% autoplot()
```

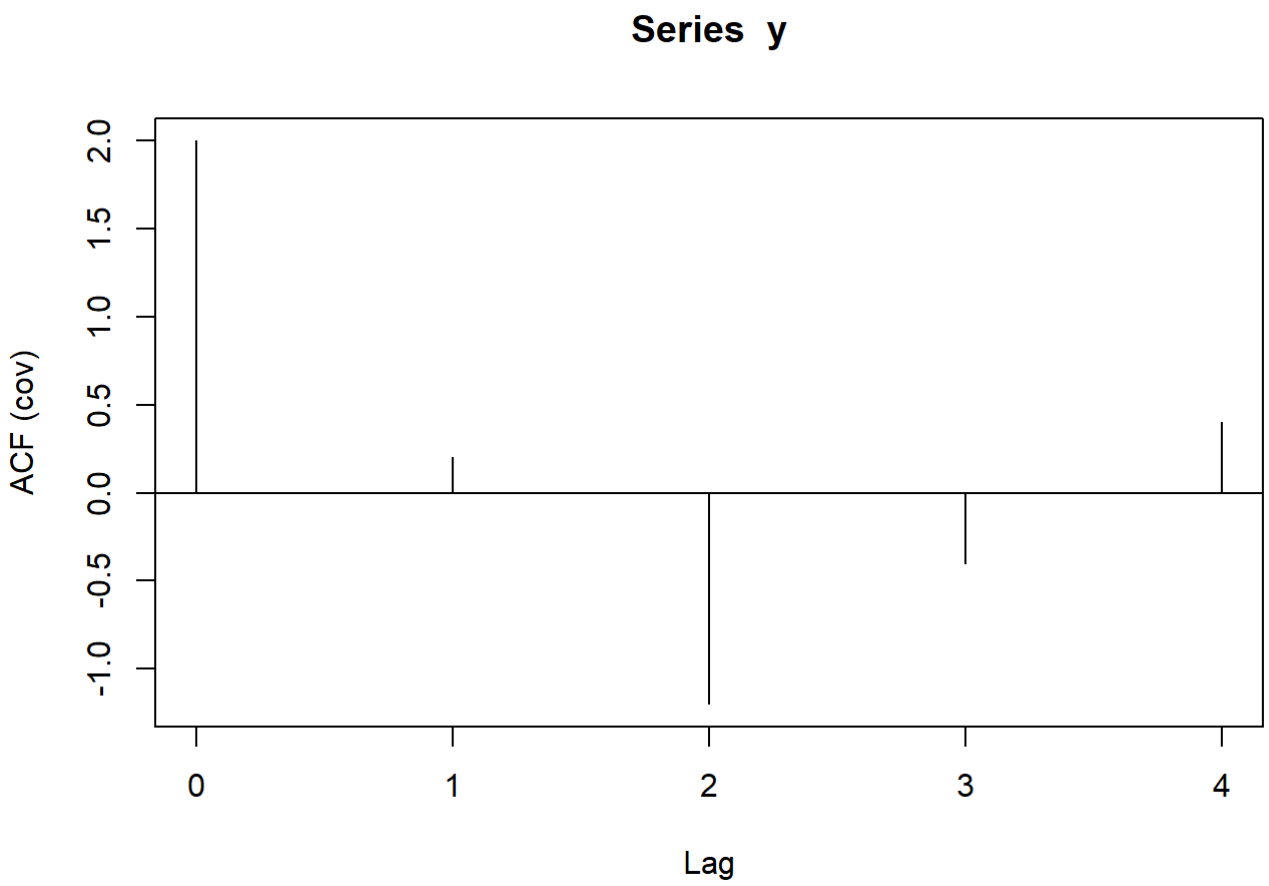


2.10 Exercise

2.11 Further reading

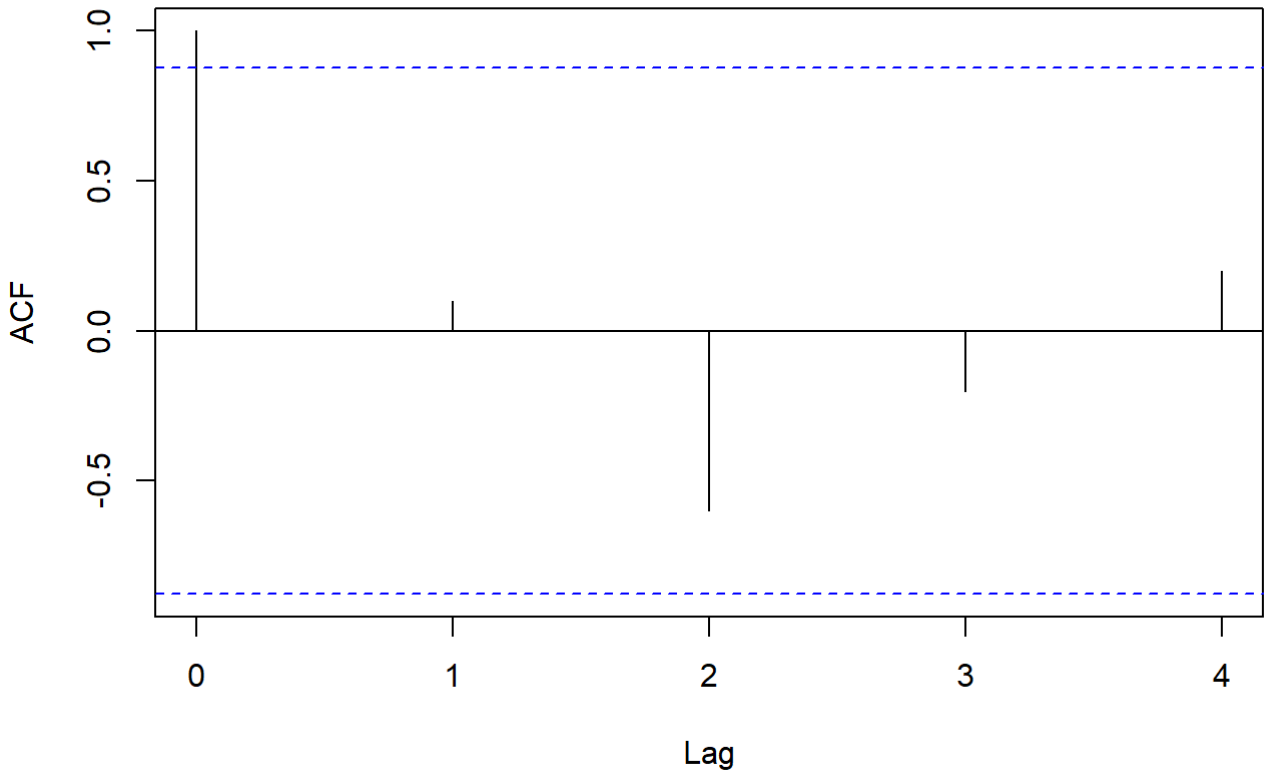
2.12 Appendix

```
y <- c(4,2,0,1,3)
print(acf(y, type = 'cov'))
```



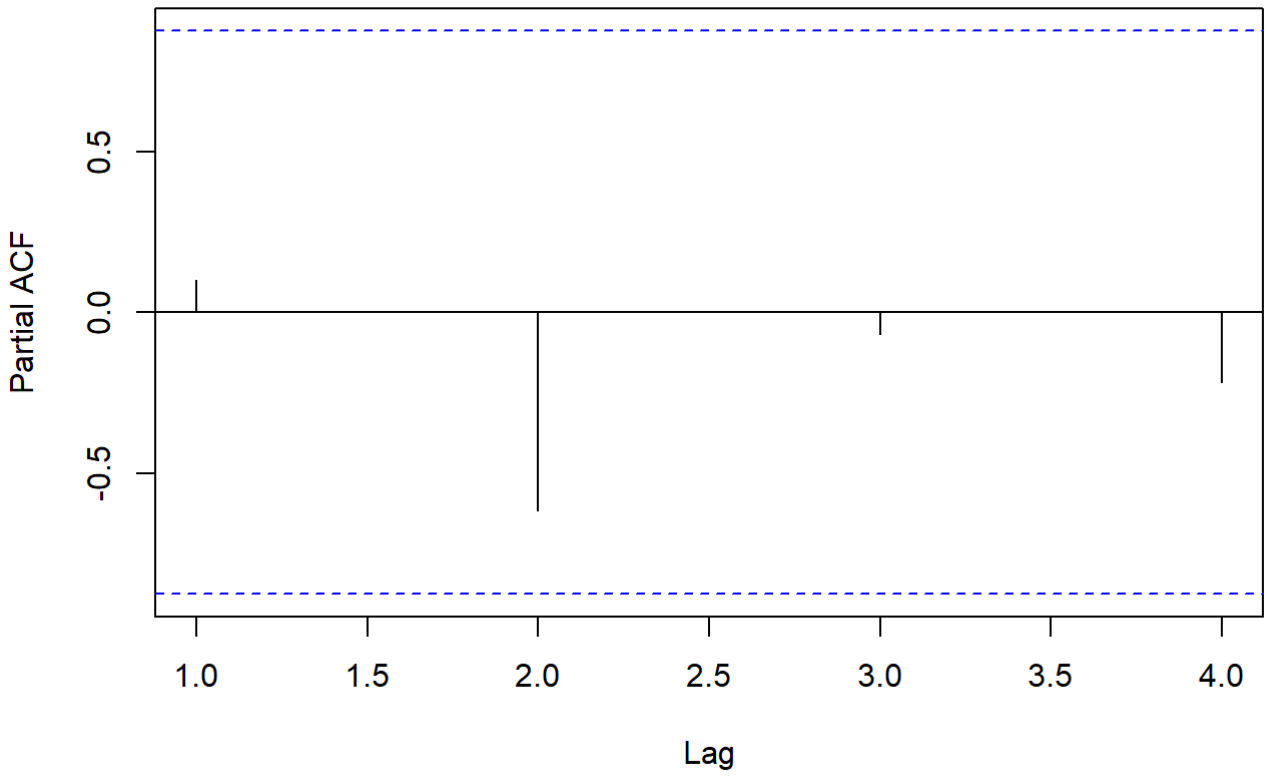

```
##
## Autocovariances of series 'y', by lag
##
##    0    1    2    3    4
##  2.0  0.2 -1.2 -0.4  0.4
```

```
print(acf(y,type='cor'))
```




```
##
## Autocorrelations of series 'y', by lag
##
##    0    1    2    3    4
##  1.0  0.1 -0.6 -0.2  0.2
```

```
print(acf(y,type='partial'))
```




```
##
## Partial autocorrelations of series 'y', by lag
##
##      1      2      3      4
##  0.100 -0.616 -0.067 -0.217
```

```
cbind(y, stats::lag(y),stats::lag(y,2), stats::lag(y,3), stats::lag(y,4))
```

```
##      y
## [1,] 4 4 4 4 4
## [2,] 2 2 2 2 2
## [3,] 0 0 0 0 0
## [4,] 1 1 1 1 1
## [5,] 3 3 3 3 3
```

```
library(tidyverse)
cbind(y, lag(y), lag(y,n=2), lag(y,n=3))
```

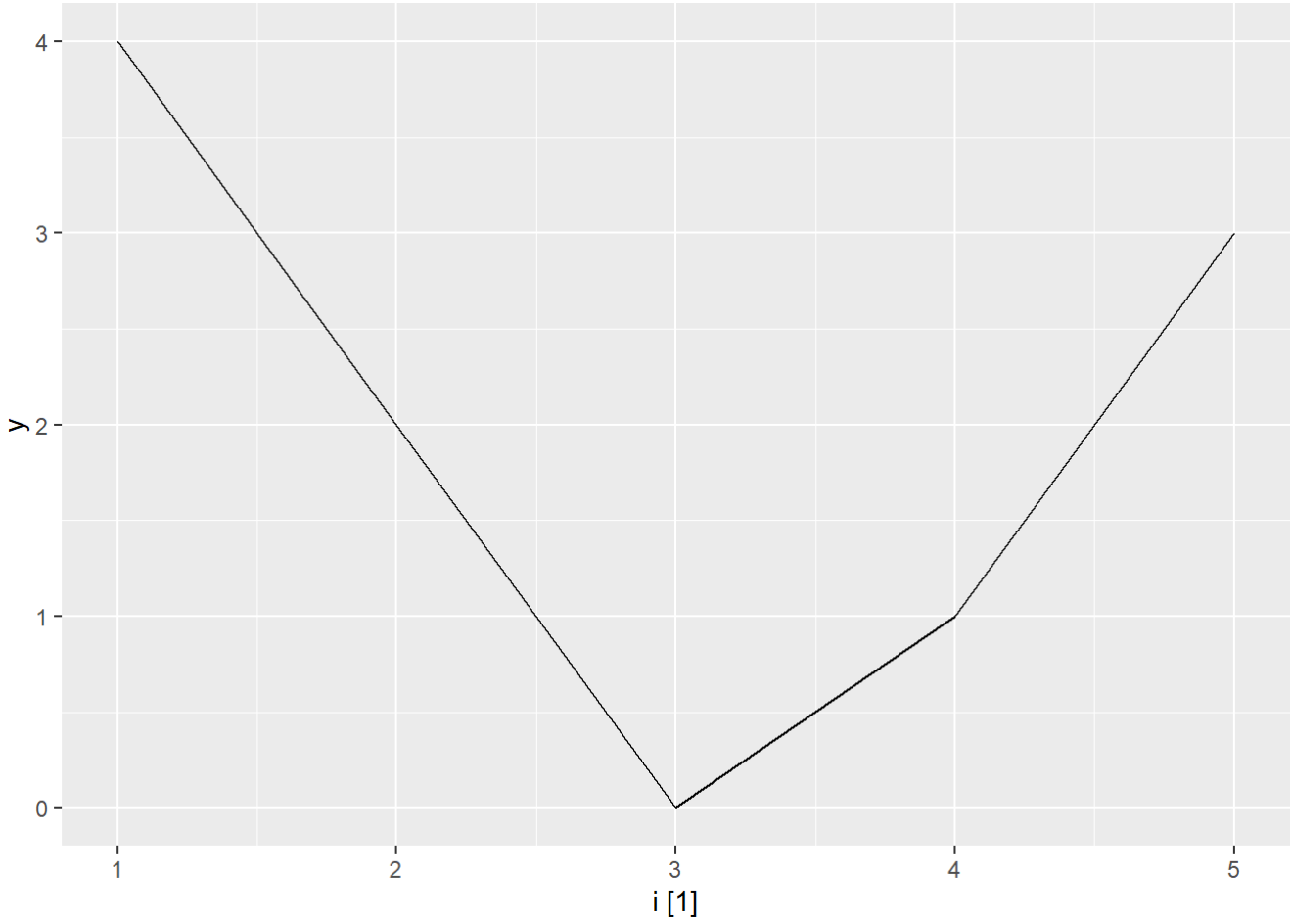
```
##      y
## [1,] 4 NA NA NA
## [2,] 2  4 NA NA
## [3,] 0  2  4 NA
## [4,] 1  0  2  4
## [5,] 3  1  0  2
```

```
cbind(y, lead(y), lead(y,n=2), lead(y, n=3))
```

```
##      y
## [1,] 4  2  0  1
## [2,] 2  0  1  3
## [3,] 0  1  3 NA
## [4,] 1  3 NA NA
## [5,] 3 NA NA NA
```

- feasts::ACF : 모두 1차부터 계산

```
tsb <- tsibble(i=1:5, y=c(4,2,0,1,3), index=i)
autoplot(tsb,y)
```




```
ACF(tsb,y,type='cov')
```

```
## # A tsibble: 4 x 2 [1]
##     lag   acf
##   <lag> <dbl>
## 1     1   0.2
## 2     2  -1.2
## 3     3  -0.4
## 4     4   0.4
```

```
ACF(tsb,y,type='cor')
```

```
## # A tsibble: 4 x 2 [1]
##     lag    acf
##   <lag>  <dbl>
## 1     1  0.100
## 2     2 -0.600
## 3     3 -0.200
## 4     4  0.200
```

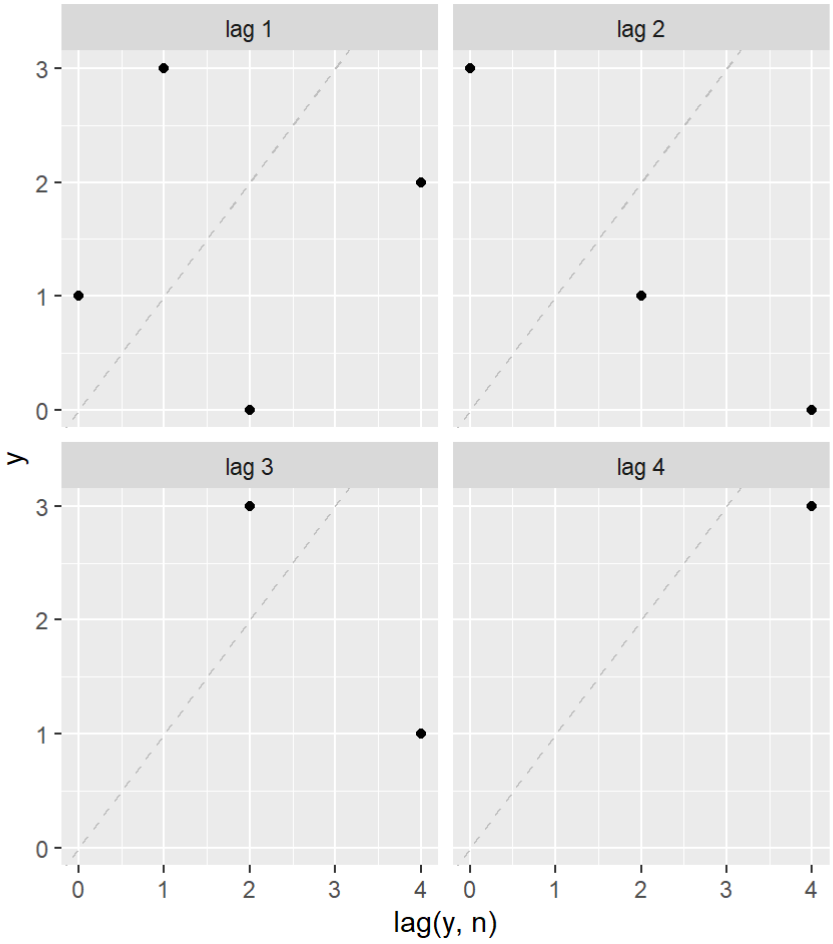
```
ACF(tsb,y,type='par')
```

```
## # A tsibble: 4 x 2 [1]
##     lag     acf
##   <lag>   <dbl>
## 1     1  0.100
## 2     2 -0.616
## 3     3 -0.0674
## 4     4 -0.217
```

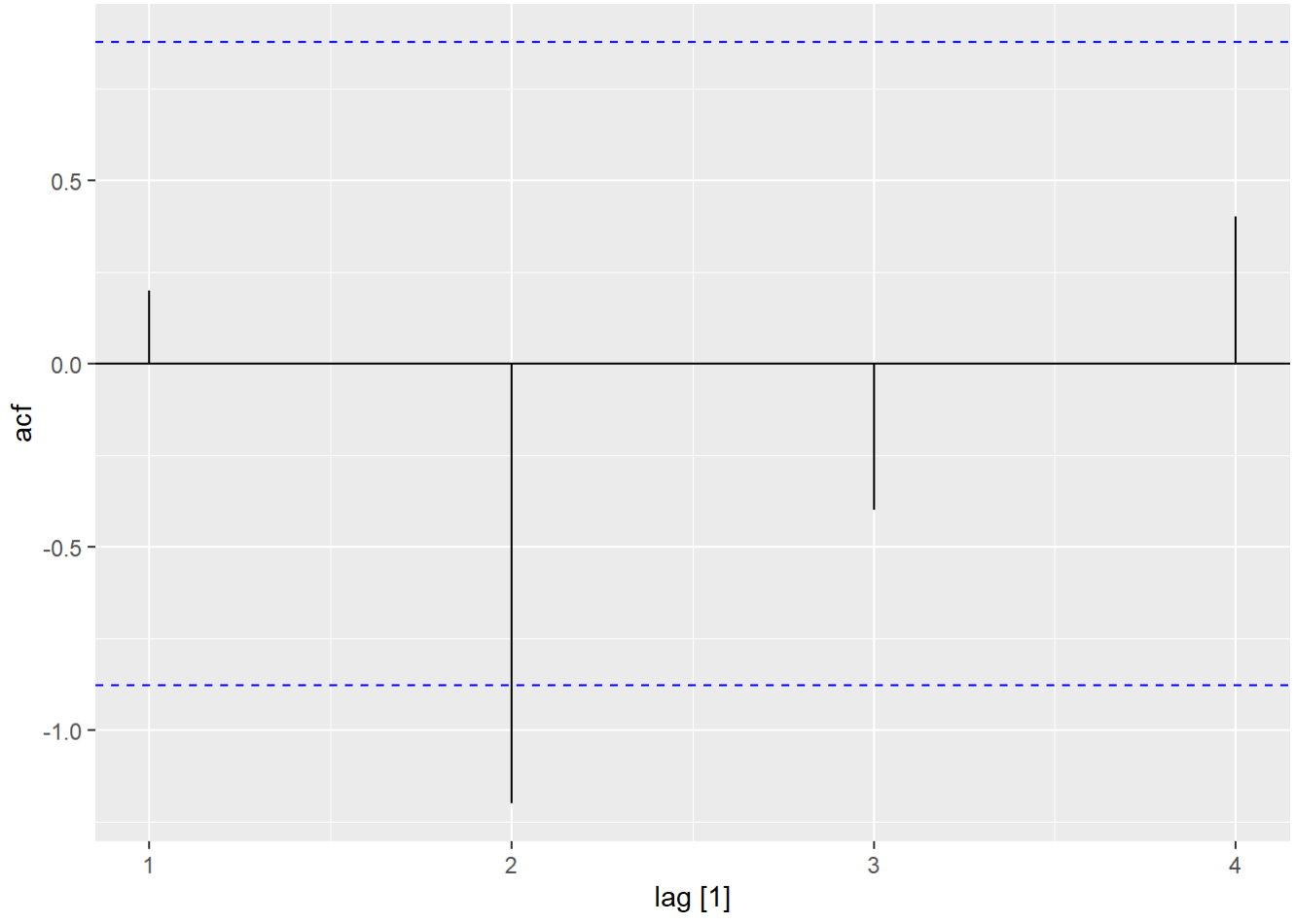
```
PACF(tsb,y)
```

```
## # A tsibble: 4 x 2 [1]
##     lag    pacf
##   <lag>   <dbl>
## 1     1  0.100
## 2     2 -0.616
## 3     3 -0.0674
## 4     4 -0.217
```

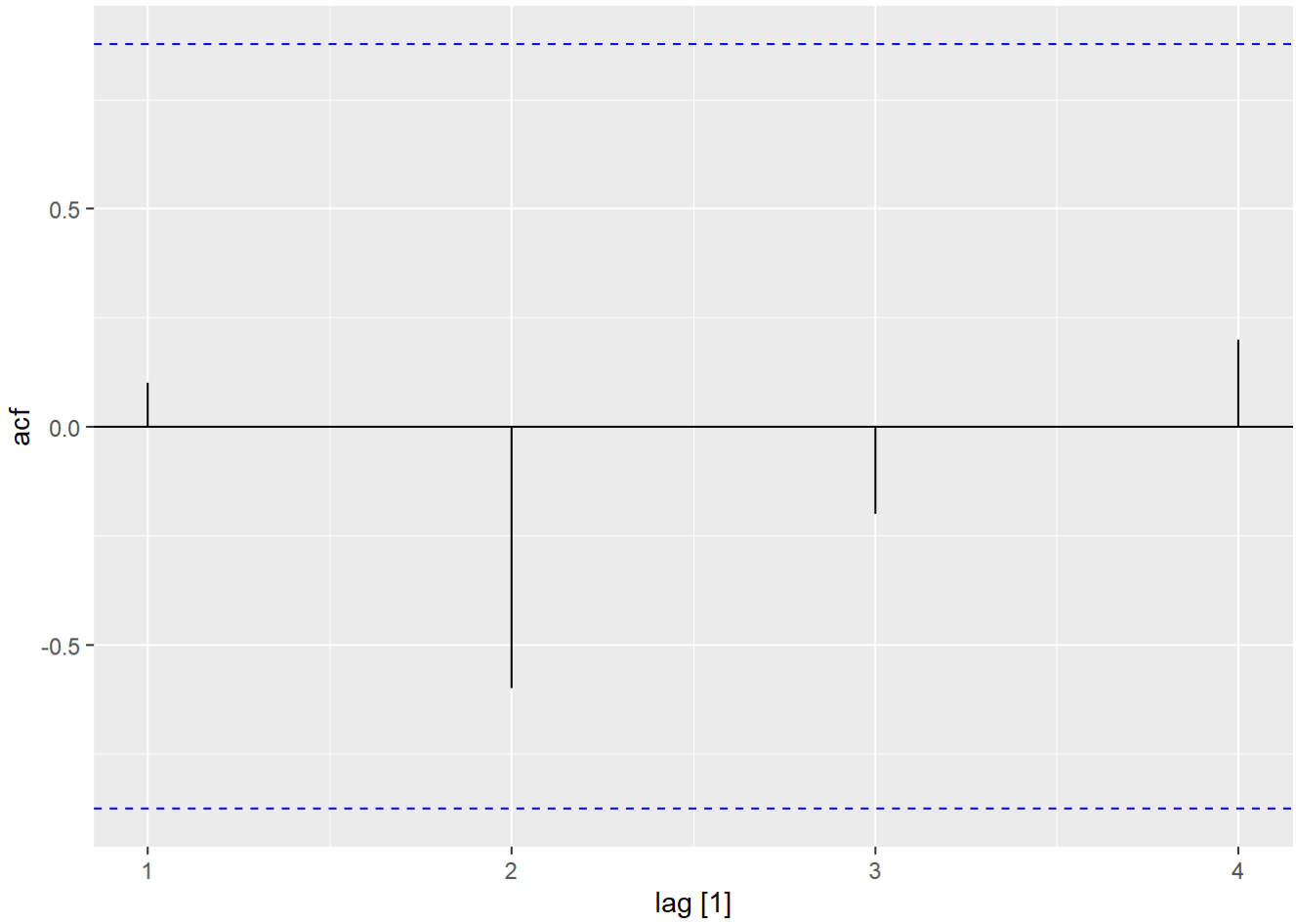
```
tsb %>% gg_lag(y, geom='point')
```



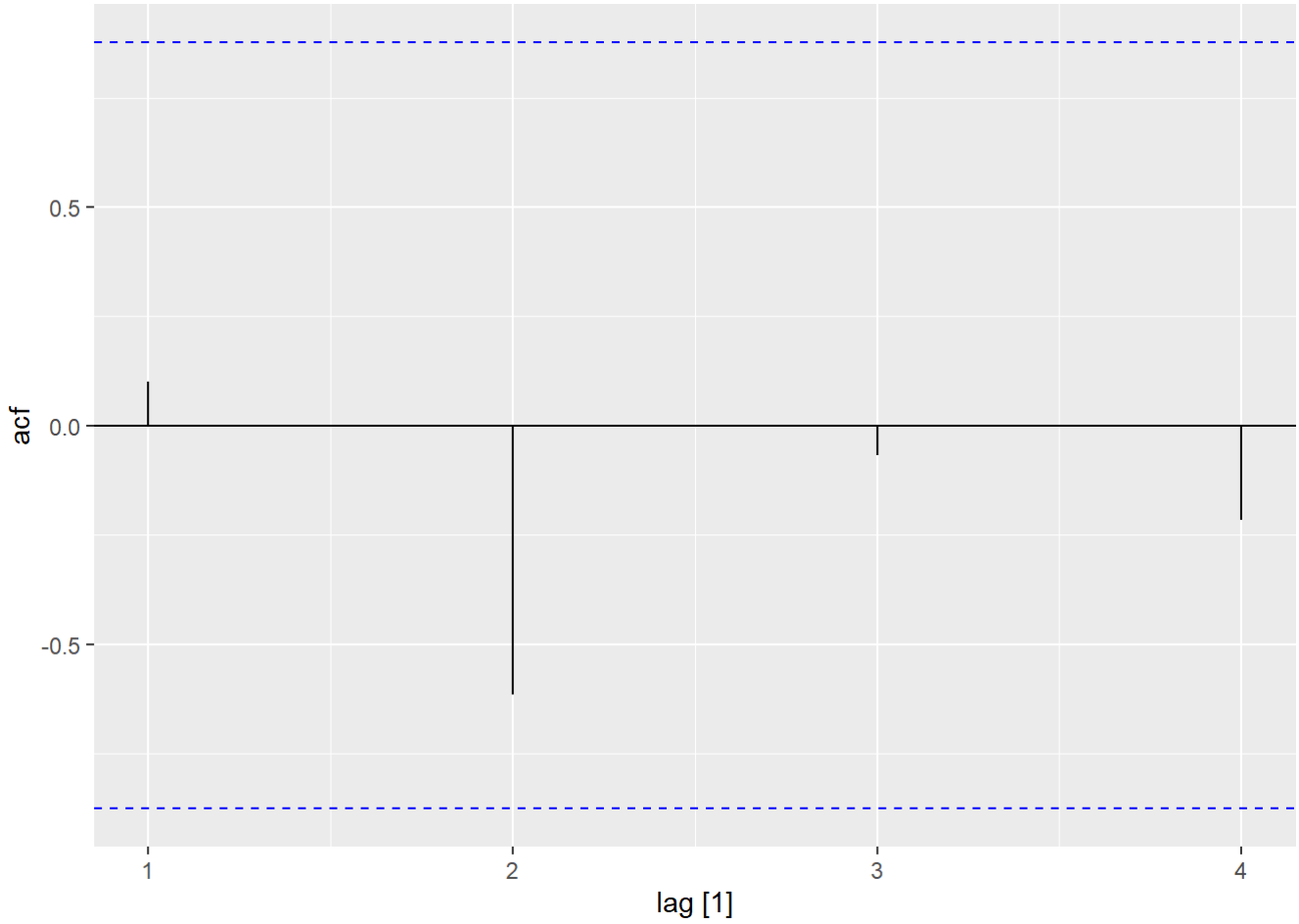

```
tsb %>% ACF(y, lag_max = 4, type='cov') %>% autoplot()
```



```
tsb %>% ACF(y, lag_max = 4) %>% autoplot()
```



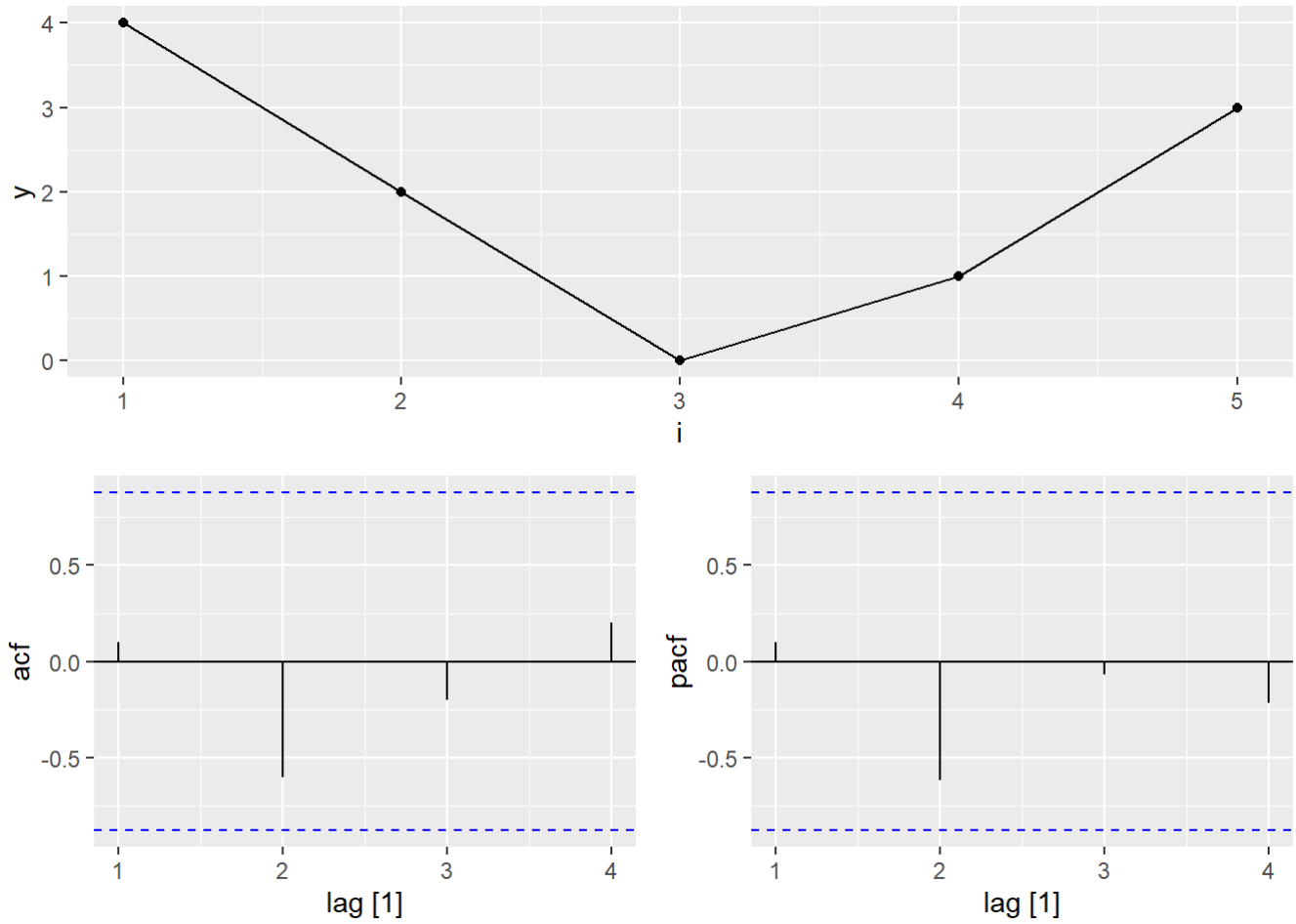

```
tsb %>% ACF(y, lag_max = 4, type='partial') %>% autoplot()
```




```
tsb %>% features(y, ljung_box, lag=4, dof=0)
```

```
## # A tibble: 1 x 2
##   lb_stat lb_pvalue
##     <dbl>     <dbl>
## 1    6.39     0.172
```

```
feasts::gg_tsdisplay(tsb,y,plot='partial')
```




```
LDF <- read.csv(textConnection('
day, a, y
1, ss, 50
1, gg, 100
2, ss, 60
2, gg, 110
3, ss, 80
3, gg, 150'))
LDF
```

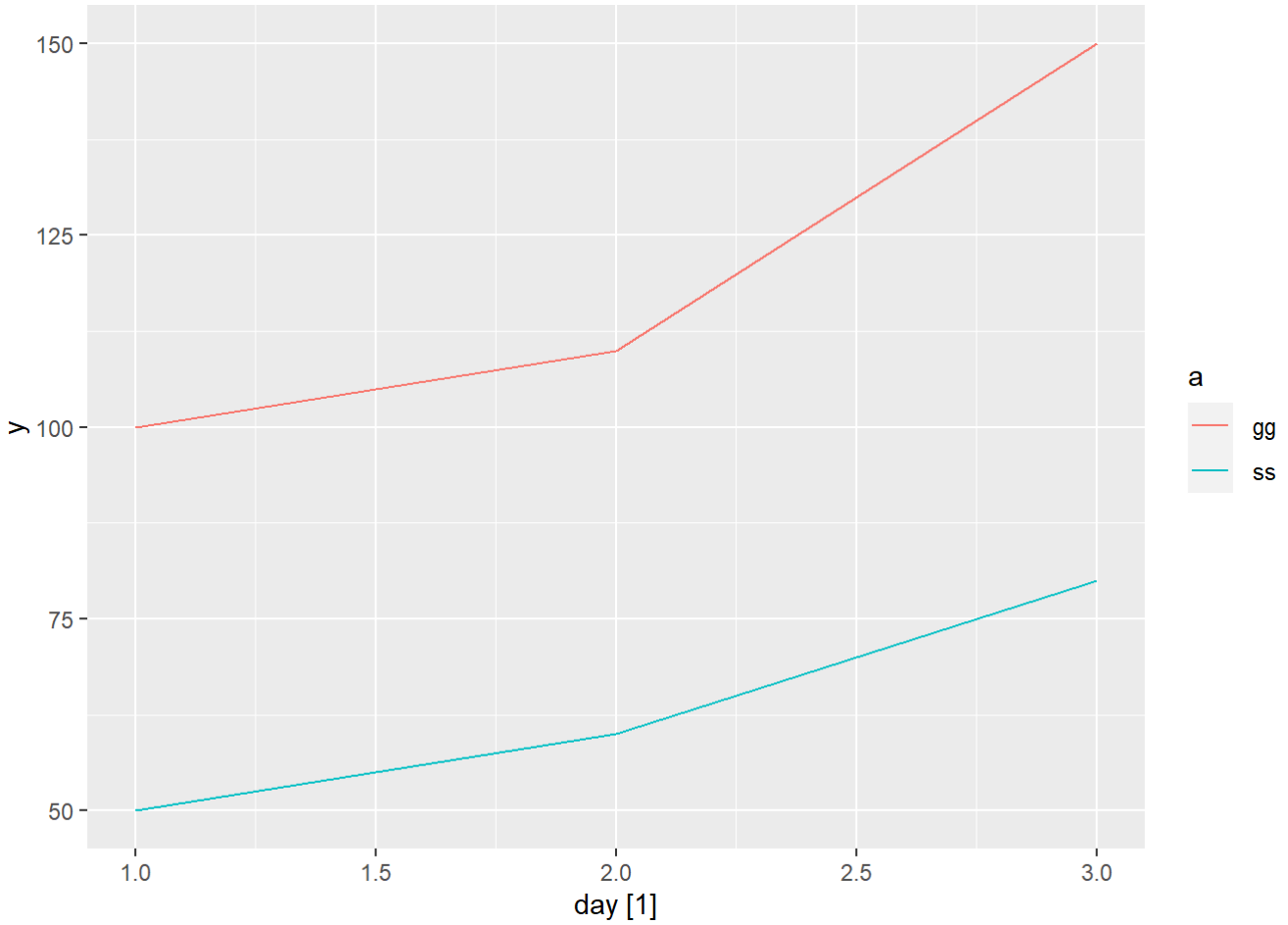
```
##   day  a   y
## 1   1 ss  50
## 2   1 gg 100
## 3   2 ss  60
## 4   2 gg 110
## 5   3 ss  80
## 6   3 gg 150
```

```
tsb <- as_tsibble(LDF, key=a, index=day)
tsb
```

```
## # A tsibble: 6 x 3 [1]
## # Key:       a [2]
##     day a          y
##   <int> <chr> <int>
## 1     1 " gg"   100
## 2     2 " gg"   110
## 3     3 " gg"   150
## 4     1 " ss"    50
## 5     2 " ss"    60
## 6     3 " ss"    80
```

```
autoplot(tsb)
```

```
## Plot variable not specified, automatically selected `.vars = y`
```




```
filter(tsb, a=='gg')
```

```
## # A tsibble: 0 x 3 [?]
## # Key:       a [0]
## # ... with 3 variables: day <int>, a <chr>, y <int>
```

```
filter(tsb, day<3)
```

```
## # A tsibble: 4 x 3 [1]
## # Key:       a [2]
##     day a          y
##   <int> <chr> <int>
## 1     1 " gg"   100
## 2     2 " gg"   110
## 3     1 " ss"    50
## 4     2 " ss"    60
```