Assignment_1_Data_624

Mubashira Qari

2026-02-01

Contents

2.10 Exercises

1. Explore the following four time series: Bricks from aus_production, Lynx from pelt, Close from gafa_stock, Demand from vic_elec.

Use ? (or help()) to find out about the data in each series.

Bricks from aus_production

```
library(fpp3)
```

```
## Registered S3 method overwritten by 'tsibble':
##   method              from
##   as_tibble.grouped_df dplyr
```

```
## -- Attaching packages ---------------------------------------- fpp3 1.0.2 --
```

```
## v tibble      3.2.1     v tsibble      1.1.6
## v dplyr       1.1.4     v tsibbledata 0.4.1
## v tidyr       1.3.1     v feasts      0.4.2
## v lubridate   1.9.4     v fable       0.5.0
## v ggplot2     4.0.0
```

```
## Warning: package 'dplyr' was built under R version 4.3.3
```

```
## Warning: package 'lubridate' was built under R version 4.3.3
```

```
## Warning: package 'tsibble' was built under R version 4.3.3
```

```
## Warning: package 'tsibbledata' was built under R version 4.3.3

## -- 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()
```

```
# Access help documentation
# ?aus_production
```

What is the time interval of each series?

Quarterly production of selected commodities in Australia.

Description

Quarterly estimates of selected indicators of manufacturing production in Australia.

Format

Time series of class tsibble.

Details

aus_production is a half-hourly tsibble with six values:

- **Beer:** Beer production in megalitres.

- **Tobacco:** Tobacco and cigarette production in tonnes.

- **Bricks:** Clay brick production in millions of bricks.

- **Cement:** Portland cement production in thousands of tonnes.

- **Electricity:** Electricity production in gigawatt hours.

- **Gas:** Gas production in petajoules.

Source
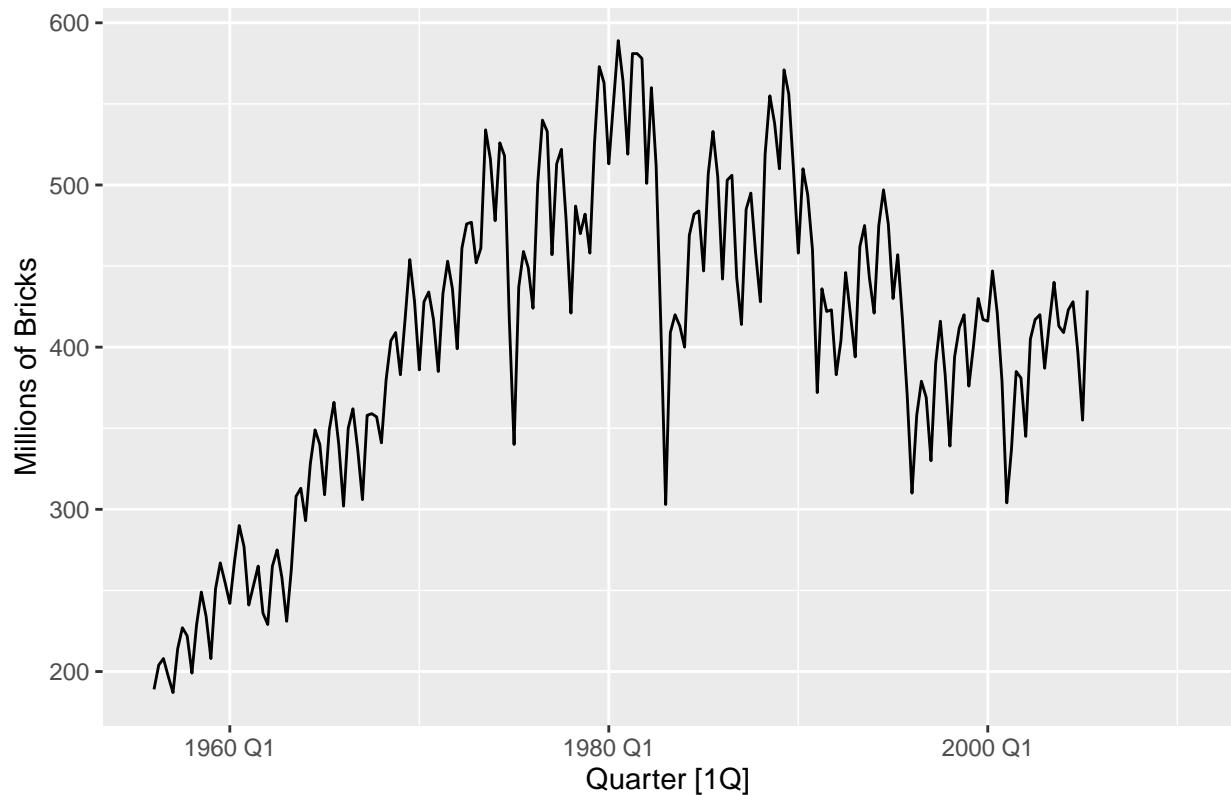
Australian Bureau of Statistics, catalogue number 8301.0.55.001 table 1.

Use autoplot() to produce a time plot of each series.

```
# Plot of the Bricks time series
aus_production |>
  autoplot(Bricks) +
  labs(title = "Australian Quarterly Clay Brick Production", y = "Millions of Bricks")
```

```
## Warning: Removed 20 rows containing missing values or values outside the scale range
## ('geom_line()').
```

Australian Quarterly Clay Brick Production

Lynx from pelt

```
# Access help documentation
# ?pelt
```

What is the time interval of each series?

This is Annual Pelt trading records

Description

Hudson Bay Company trading records for Snowshoe Hare and Canadian Lynx furs from 1845 to 1935. This data contains trade records for all areas of the company.

Format

Time series of class tsibble

Details

pelt is an annual tsibble with two values:
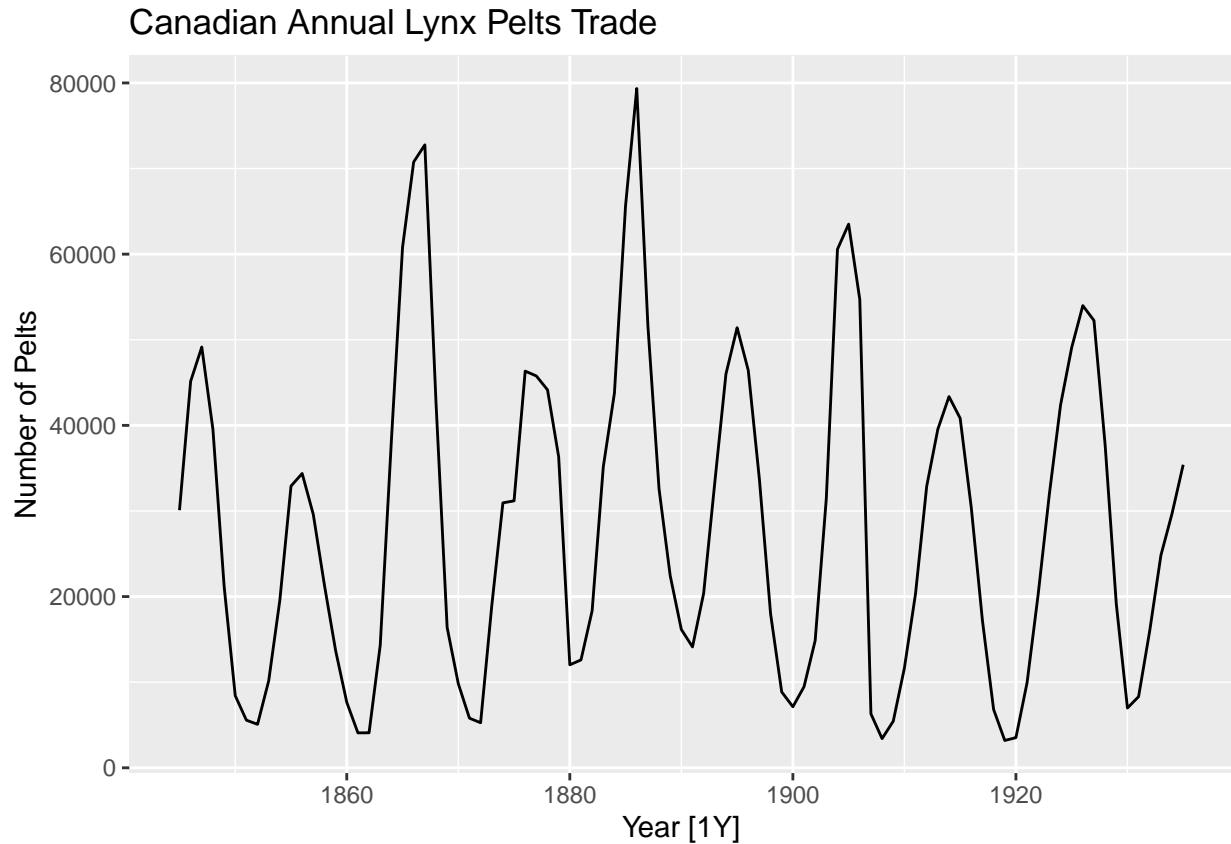
- **Hare:** The number of Snowshoe Hare pelts traded.

- **Lynx:** The number of Canadian Lynx pelts traded.

Source

Hudson Bay Company

Use autoplot() to produce a time plot of each series.

```
# Plot of the Lynx time series
pelt |>
  autoplot(Lynx) +
  labs(title = "Canadian Annual Lynx Pelts Trade", y = "Number of Pelts")
```

Close from gafa_stock

```
# Access help documentation
# ?gafa_stock
```

What is the time interval of each series?

This is Daily GAFA stock prices

Description

Historical stock prices from 2014-2018 for Google, Amazon, Facebook and Apple. All prices are in $USD.

Format

Time series of class tsibble

Details

gafa_stock is a tsibble containing data on irregular trading days:

- **Open:** The opening price for the stock.

- **High:** The stock's highest trading price.

- **Low:** The stock's lowest trading price.

- **Close:** The closing price for the stock.

- **Adj_Close:** The adjusted closing price for the stock.

- **Volume:** The amount of stock traded.

Each stock is uniquely identified by one key:

- **Symbol:** The ticker symbol for the stock.
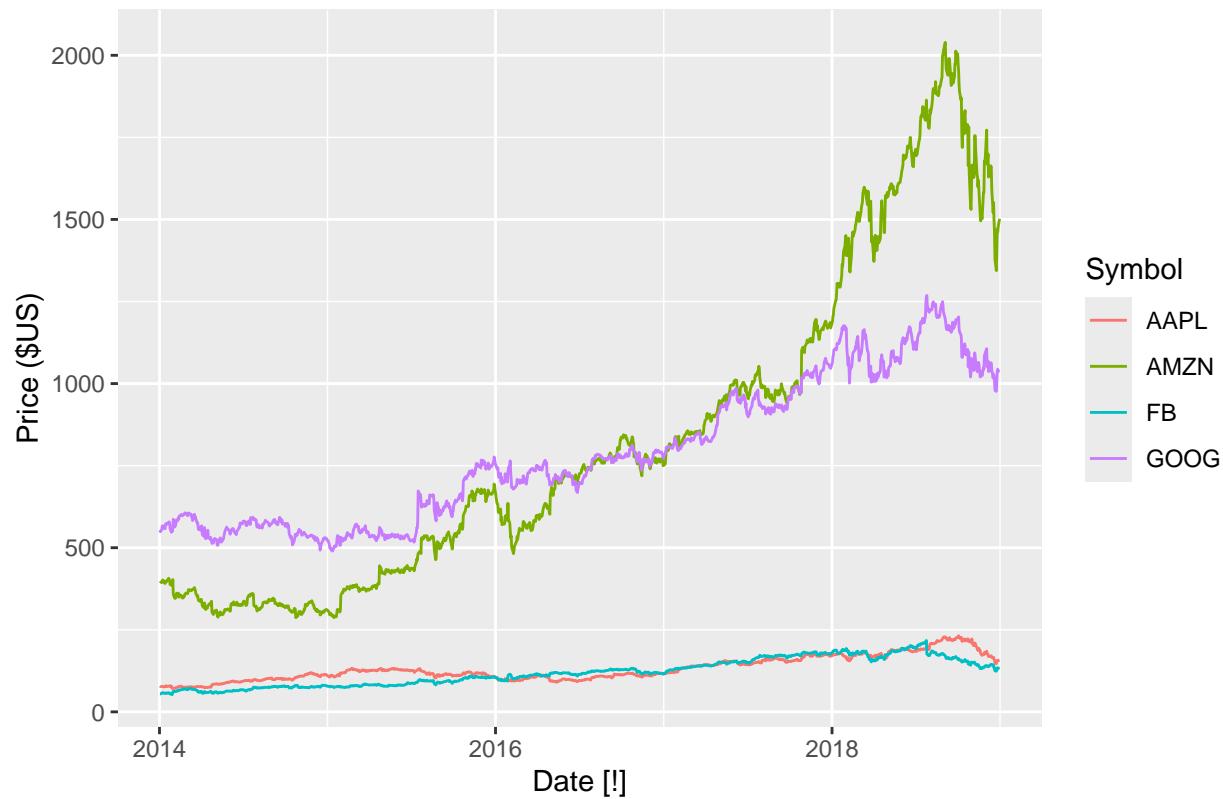
Source

Yahoo Finance historical data

Use autoplot() to produce a time plot of each series.

```
# Plot of the Closing prices for all stocks
gafa_stock |>
  autoplot(Close) +
  labs(title = "GAFA Daily Closing Stock Prices", y = "Price ($US)")
```

GAFA Daily Closing Stock Prices

Demand from vic_elec

```
# Access help documentation
# ?vic_elec
```

What is the time interval of each series?

Half-hourly electricity demand for Victoria, Australia

Description

vic_elec is a half-hourly tsibble with three values:

- **Demand:** Total electricity demand in MWh.

- **Temperature:** Temperature of Melbourne (BOM site 086071).

- **Holiday:** Indicator for if that day is a public holiday.

Format

Time series of class tsibble.

Details
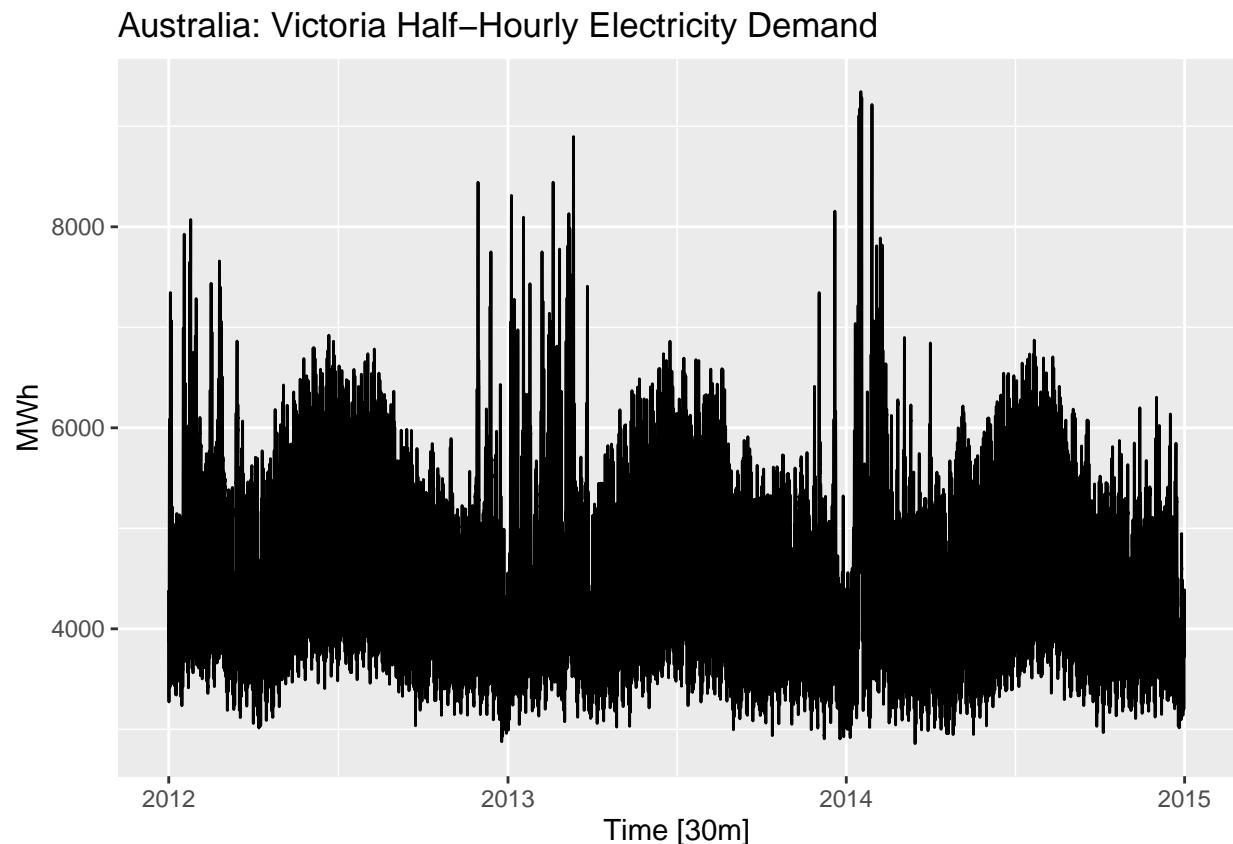
This data is for operational demand, which is the demand met by local scheduled generating units, semi-scheduled generating units, and non-scheduled intermittent generating units of aggregate capacity larger than 30 MWh, and by generation imports to the region. The operational demand excludes the demand met by non-scheduled non-intermittent generating units, non-scheduled intermittent generating units of aggregate capacity smaller than 30 MWh, exempt generation (e.g. rooftop solar, gas tri-generation, very small wind farms, etc), and demand of local scheduled loads. It also excludes some very large industrial users (such as mines or smelters).

Source

Australian Energy Market Operator.
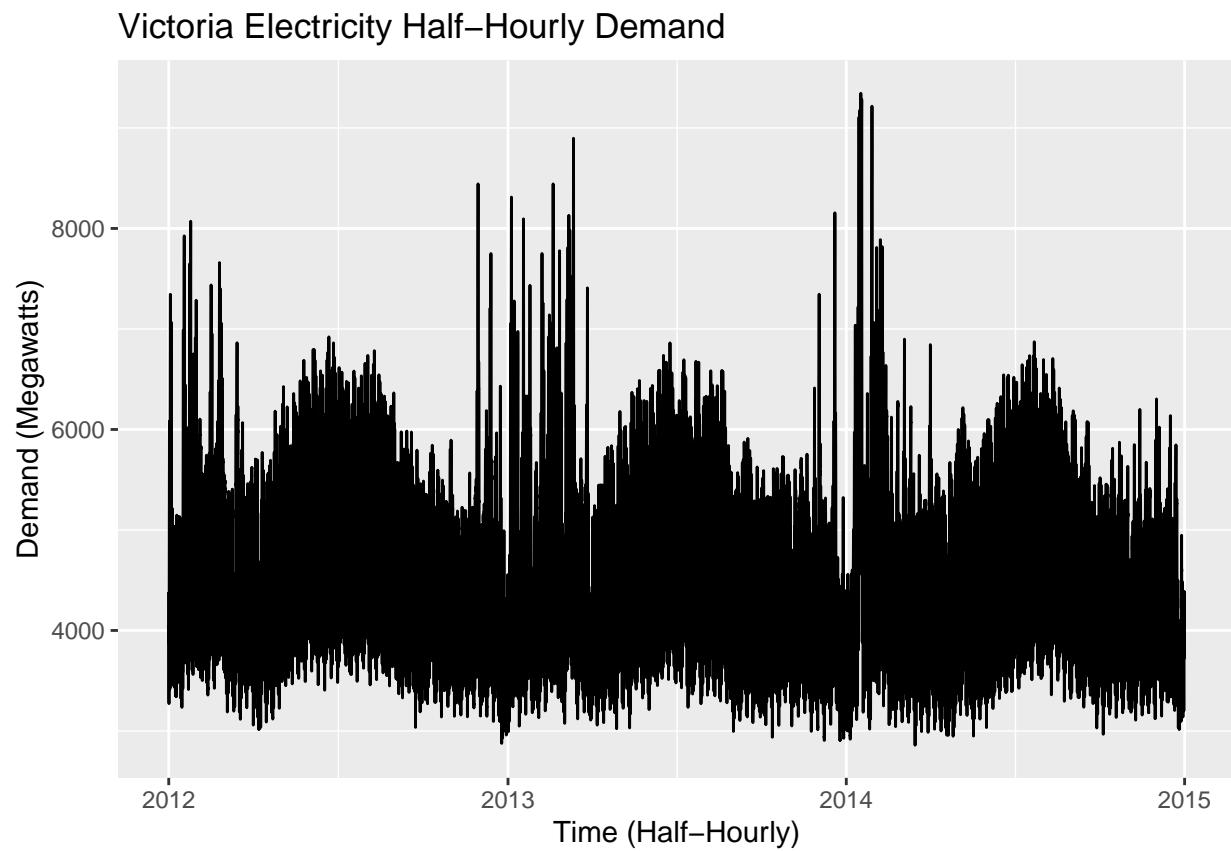
Use autoplot() to produce a time plot of each series.

```
# Plot of the Electricity Demand
vic_elec |>
  autoplot(Demand) +
  labs(title = "Australia: Victoria Half-Hourly Electricity Demand", y = "MWh")
```

For the last plot, modify the axis labels and title.

```r
# Plot of the Electricity Demand
vic_elec |>
  autoplot(Demand) +
labs(
    title = "Victoria Electricity Half-Hourly Demand",
    y = "Demand (Megawatts)",
    x = "Time (Half-Hourly)"
  )
```


2. Use filter() to find what days corresponded to the peak closing price for each of the four stocks in gafa_stock.

```r
# view(gafa_stock)
```

```r
gafa_stock |>
  group_by(Symbol) |>
  filter(Close == max(Close)) |>
  arrange(desc(Close)) -> gafa_peak_closing_price
```

```r
# view(gafa_peak_closing_price)
```

3. Download the file tute1.csv from the book website....

**a. You can read the data into R with the following script:**

```
tute1 <- readr::read_csv("https://raw.githubusercontent.com/uzmabb182/Data_624_Predictive_Analytics/ref
# View(tute1)
```
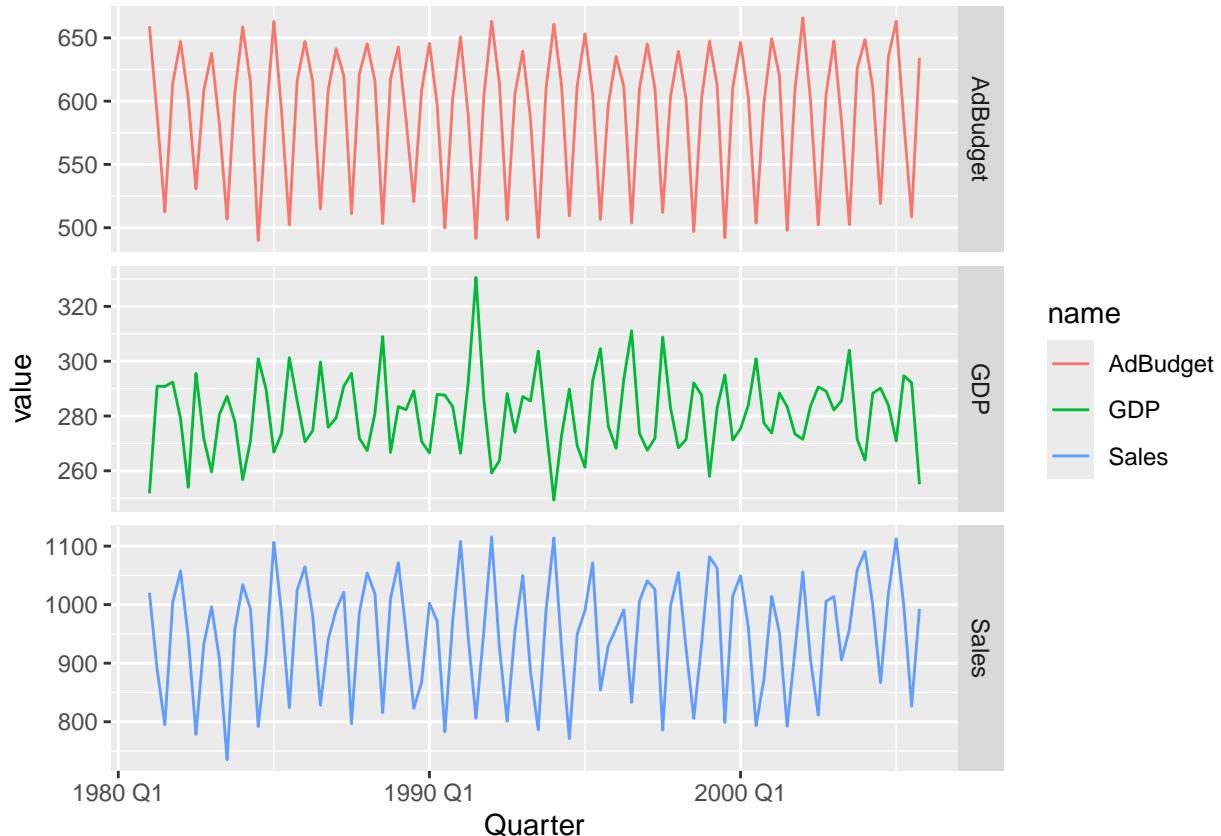
b. Convert the data to time series

```
mytimeseries <- tute1 |>
  mutate(Quarter = yearquarter(Quarter)) |>
  as_tsibble(index = Quarter)

# view(mytimeseries)
```
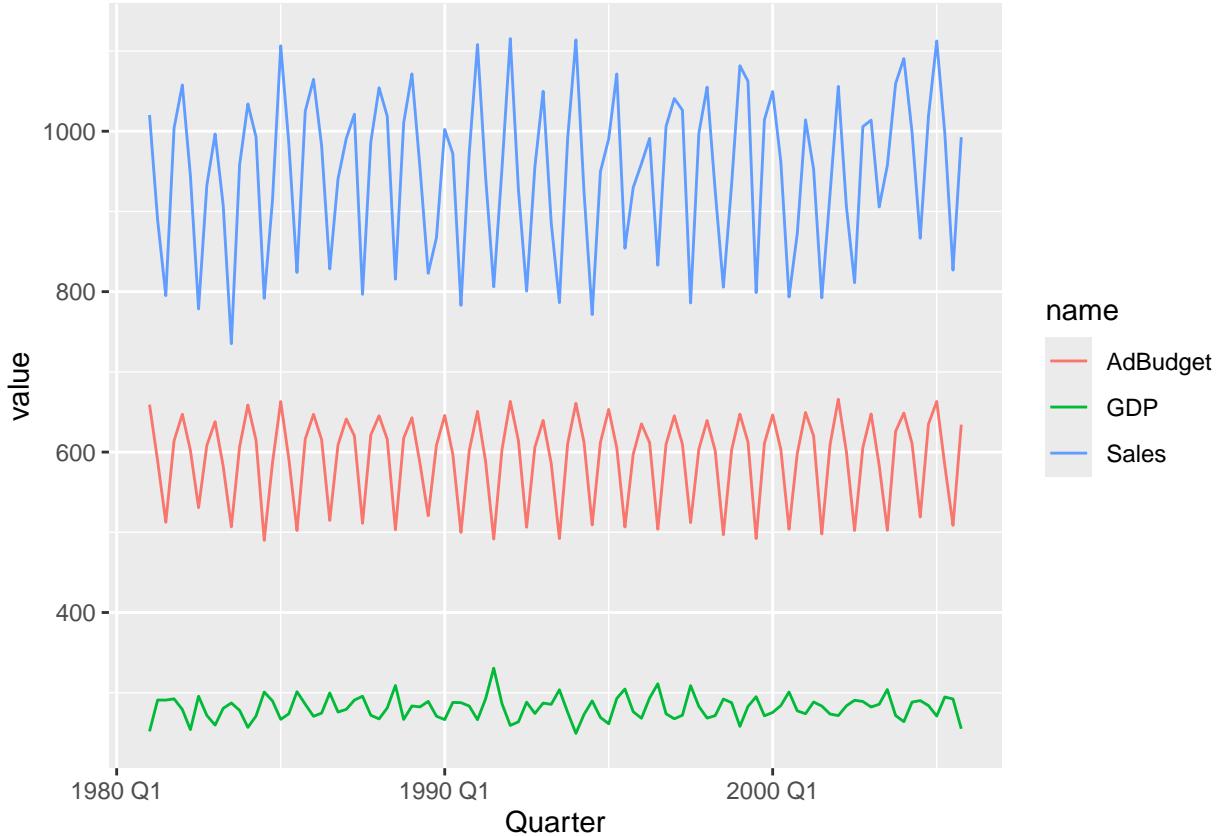
c. Construct time series plots of each of the three series

```
mytimeseries |>
  pivot_longer(-Quarter) |>
  ggplot(aes(x = Quarter, y = value, colour = name)) +
  geom_line() +
  facet_grid(name ~ ., scales = "free_y")
```

Check what happens when you don't include facet_grid().

```
mytimeseries |>
  pivot_longer(-Quarter) |>
  ggplot(aes(x = Quarter, y = value, colour = name)) +
  geom_line()
```


facet_grid with scales = "free_y" allow each variable to have its own y-axis range. Without it all variables will fit on the same y-axis. The small variable will look like a flat straight line at the bottom of the chart and the pattern will not be displayed clearly.

4. The USgas package contains data on the demand for natural gas in the US.

a. The USgas package contains data on the demand for natural gas in the US

```
library(USgas)
```

```
## Warning: package 'USgas' was built under R version 4.3.3
```

```
# view(us_total)
```

b. Create a tsibble from us_total with year as the index and state as the key.
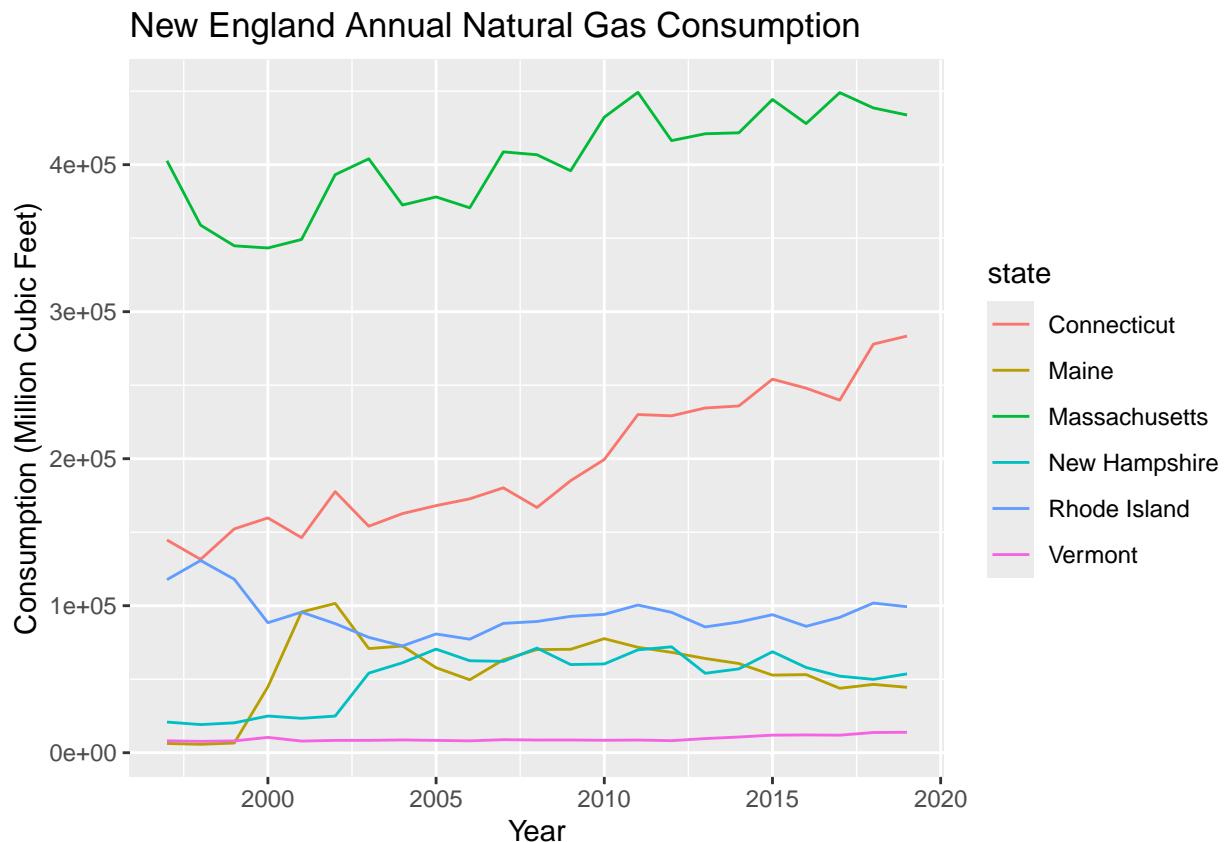
```
# Convert to tsibble
us_total_tsibble <- us_total |>
  as_tsibble(index = year, key = state)

# view(us_total_tsibble)
```

c. Plot the annual natural gas consumption by state for the New England area (comprising the states of Maine, Vermont, New Hampshire, Massachusetts, Connecticut and Rhode Island).

```
# Define the New England states
new_england_states <- c("Maine", "Vermont", "New Hampshire",
              "Massachusetts", "Connecticut", "Rhode Island")

us_total_tsibble |>
  filter(state %in% new_england_states) |>
  autoplot(y) +
  labs(
    title = "New England Annual Natural Gas Consumption",
    y = "Consumption (Million Cubic Feet)",
    x = "Year"
  )
```

Massachusetts consumes significantly more natural gas than the other states, which makes the lines for states like Vermont or Rhode Island appear quite low on the chart. This is a common scaling issue when comparing states of vastly different population sizes.

5a. Download tourism.xlsx from the book website and read it into R using readxl::read_excel().

```
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.3.3
```

```
tourism_data <- readxl::read_excel("C:/Users/Uzma/CUNY-SPS-Assignments/Data_624/Data_624_Predictive_Anal
```

```
# head(tourism_data)
```

b. Create a tsibble which is identical to the tourism tsibble from the tsibble package.

```
library(tsibble)
# the index is 'Quarter' and keys are Region, State, Purpose
tourism_tsibble <- as_tsibble(tourism)

tourism_tsibble
```

```
## # 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.
## # i 24,310 more rows
```

c. Find what combination of Region and Purpose had the maximum number of overnight trips on average.

```
tourism_tsibble %>%
  as_tibble() %>%
  group_by(Region, Purpose) %>%
  summarise(Avg_Trips = mean(Trips), .groups = "drop") %>%
  # This is the new part:
  filter(Avg_Trips == max(Avg_Trips))
```

```
## # A tibble: 1 x 3
##   Region Purpose  Avg_Trips
##   <chr>  <chr>        <dbl>
## 1 Sydney Visiting      747.
```

When we use group_by(Region, Purpose), the data becomes "grouped."
If you don't ungroup it, any future calculations like filter or mutate will happen inside those groups, which can lead to weird errors or unexpected results.
When .groups = "drop" is applied, it tells R that calculation of average step is finished and now remove all the groups and turn it back into a normal, flat dataframe.

d. Create a new tsibble which combines the Purposes and Regions, and just has total trips by State.

```
state_tourism <- tourism_tsibble %>%
  # 1. Group by the new key State
  # This automatically drops Region and Purpose from the grouping
  group_by(State) %>%

  # 2. Sum the trips
  # tsibble automatically groups by Quarter
  summarise(Trips = sum(Trips))

# View the result
state_tourism
```

```
## # 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.
## # i 630 more rows
```

8. Use the following graphics functions: autoplot(), gg_season(), gg_subseries(), gg_lag(), ACF() and explore features from the following time series: "Total Private" Employed from us_employment, Bricks from aus_production, Hare from pelt, "H02" Cost from PBS, and Barrels from us_gasoline.

1. Total Private Employed from us_employment
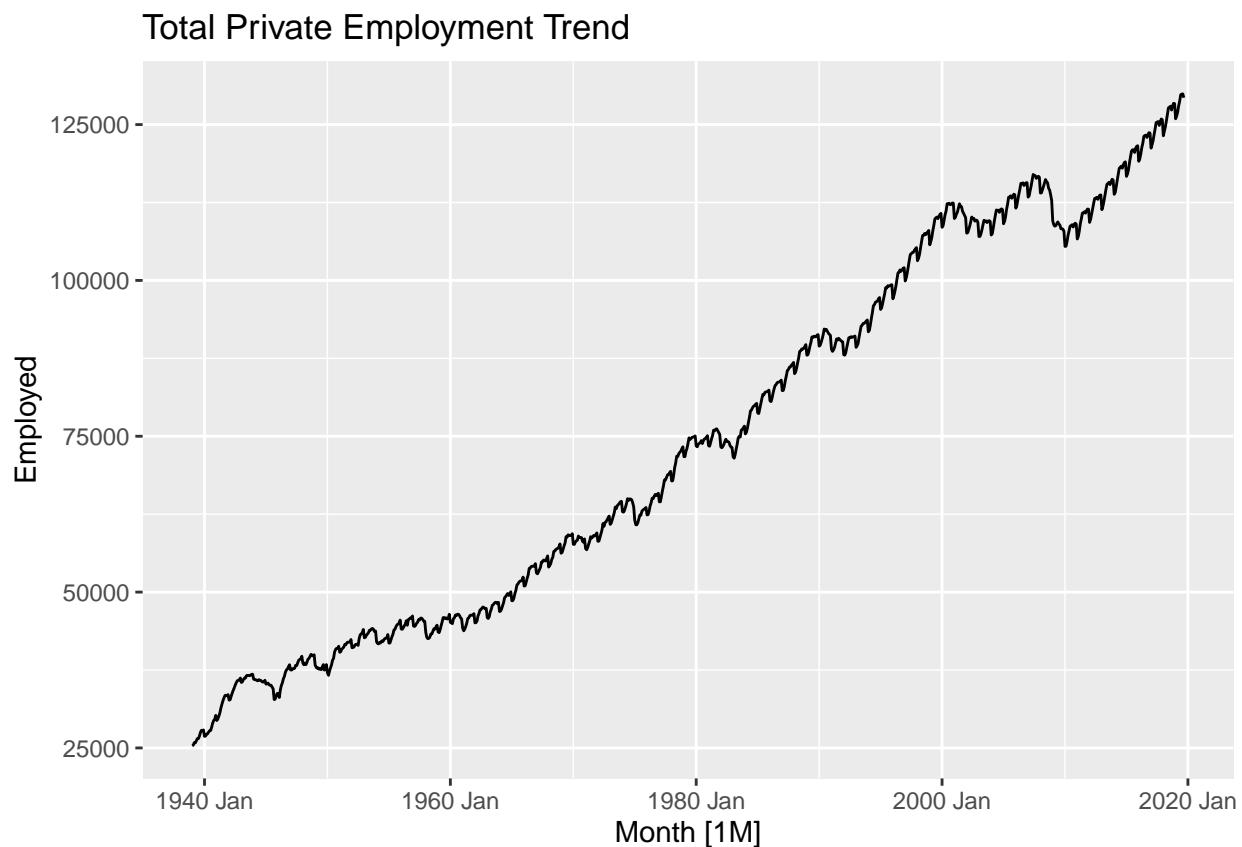
```r
library(fpp3)

#view(us_employment)

# Filter the data
total_private <- us_employment |>
  filter(Title == "Total Private")

# 1. Time Plot (Trend)
total_private |> autoplot(Employed) +
  labs(title = "Total Private Employment Trend")
```

Total Private Employment Trend
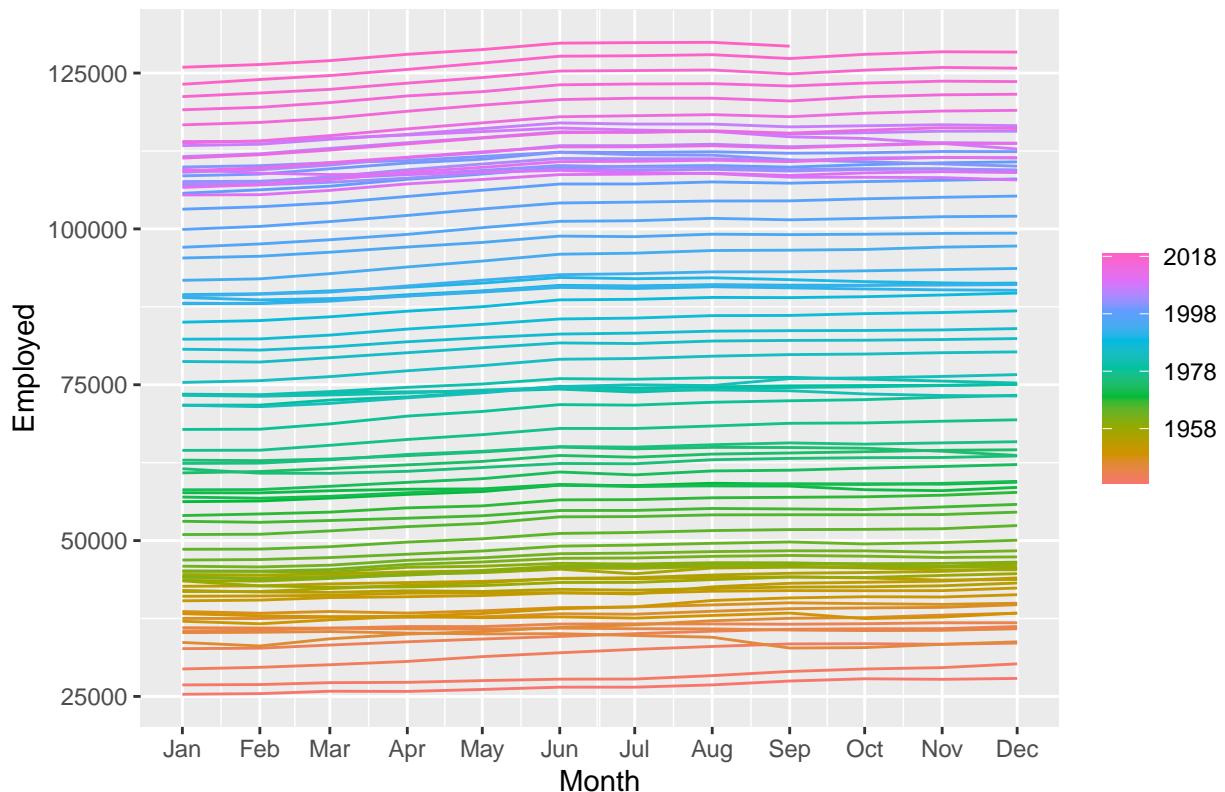

```r
# 2. Seasonal Plot (Seasonality within years)
total_private |> gg_season(Employed) +
  labs(title = "Seasonal Plot")
```

```
## Warning: 'gg_season()' was deprecated in feasts 0.4.2.
## i Please use 'ggtime::gg_season()' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```
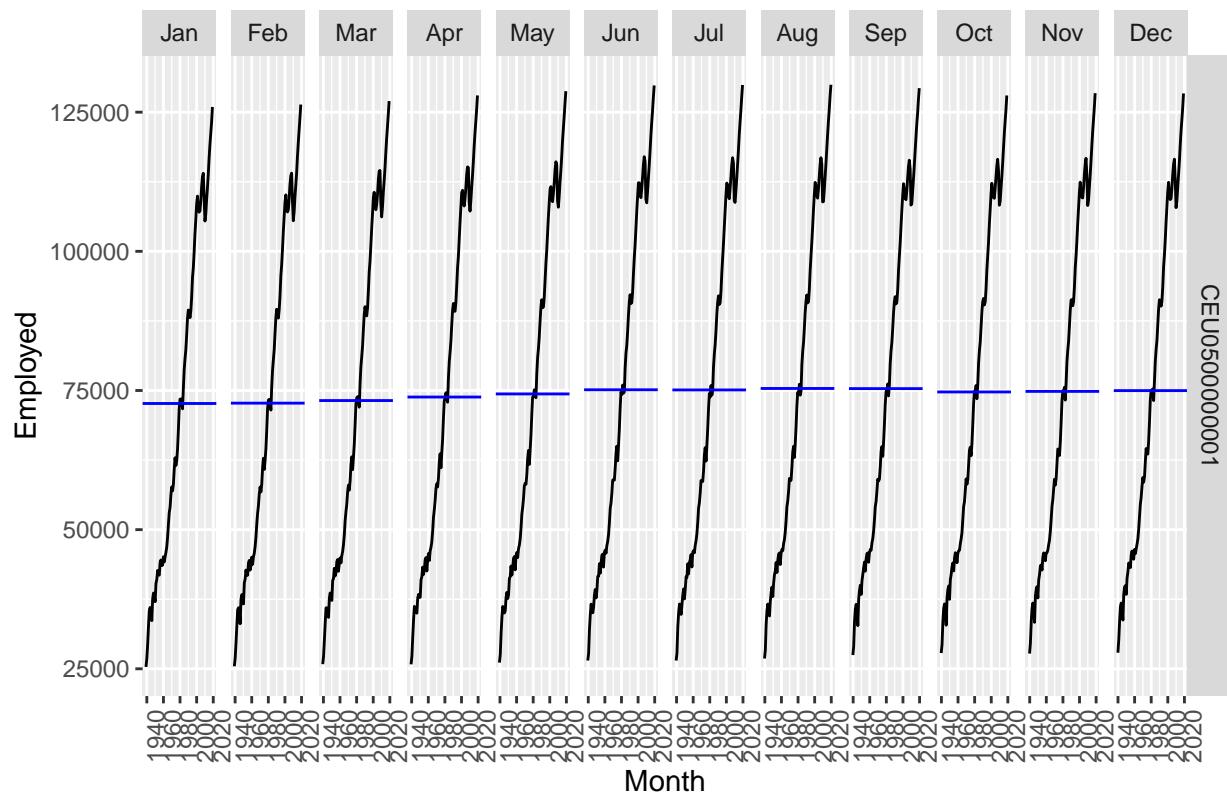
Seasonal Plot


```
# 3. Subseries Plot (Seasonality across years)
total_private |> gg_subseries(Employed) +
  labs(title = "Subseries Plot")
```
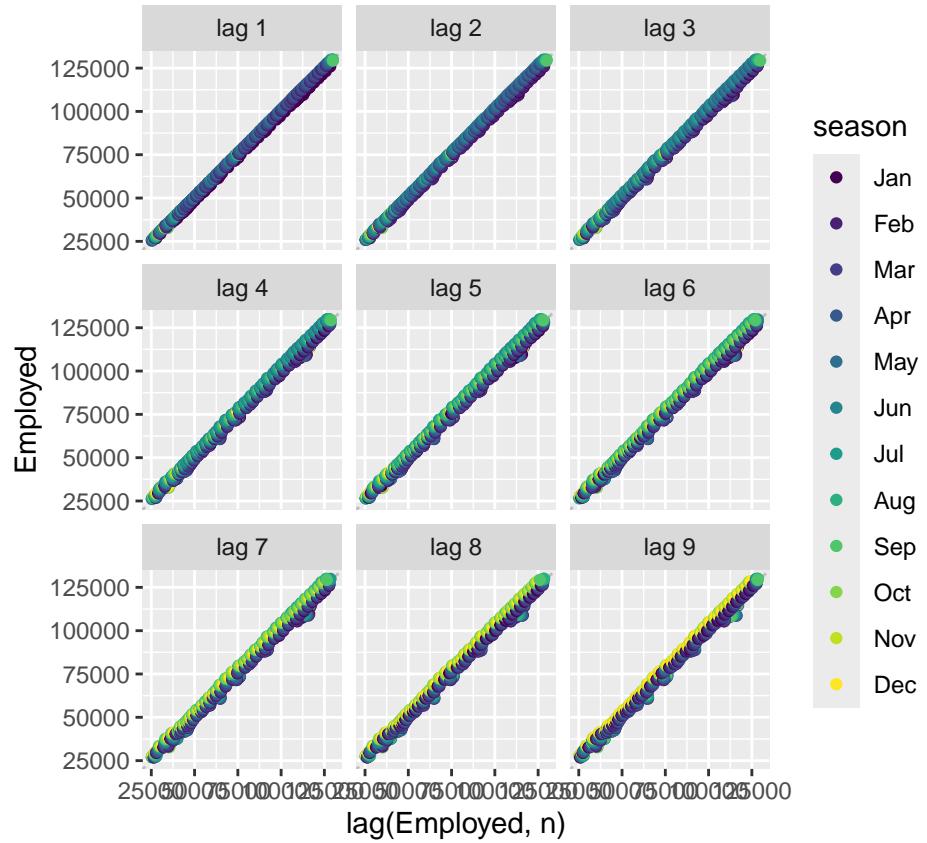
```
## Warning: `gg_subseries()` was deprecated in feasts 0.4.2.
## i Please use `ggtime::gg_subseries()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

Subseries Plot
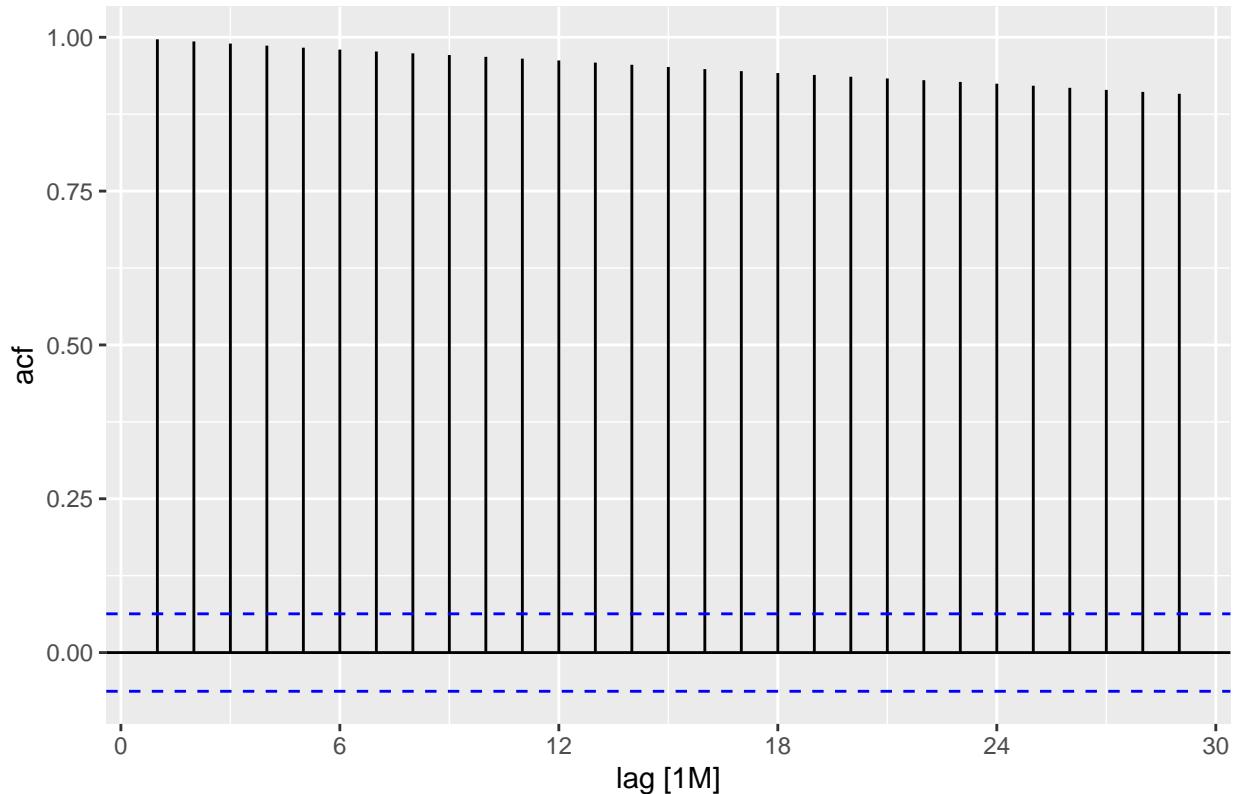

```r
# 4. Lag Plot (Autocorrelation visual)
total_private |> gg_lag(Employed, geom="point")
```

```
## Warning: 'gg_lag()' was deprecated in feasts 0.4.2.
## i Please use 'ggtime::gg_lag()' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```

```
# 5. ACF Plot (Autocorrelation statistical)
total_private |> ACF(Employed) |> autoplot() +
  labs(title = "ACF Plot")
```

ACF Plot

Seasonality, Cyclicity, and Trend:

Trend Analysis:

There is a strong, consistent upward trend over time.

Seasonality Analysis:

Yes, there is a clear annual seasonal pattern.

Cyclicity Analysis:

There is evidence of a business cycle, most notably the dip around Global Financial Crisis of 2008.

What do you learn about the series?

The private sector of US economy has grown steadily over the decades.

What can you say about the seasonal patterns?

Employment consistently drops in January after the holiday season and tends to peak in the summer/late year.
The seasonal plot shows lines that are very paralle showing the stable seasonal pattern.
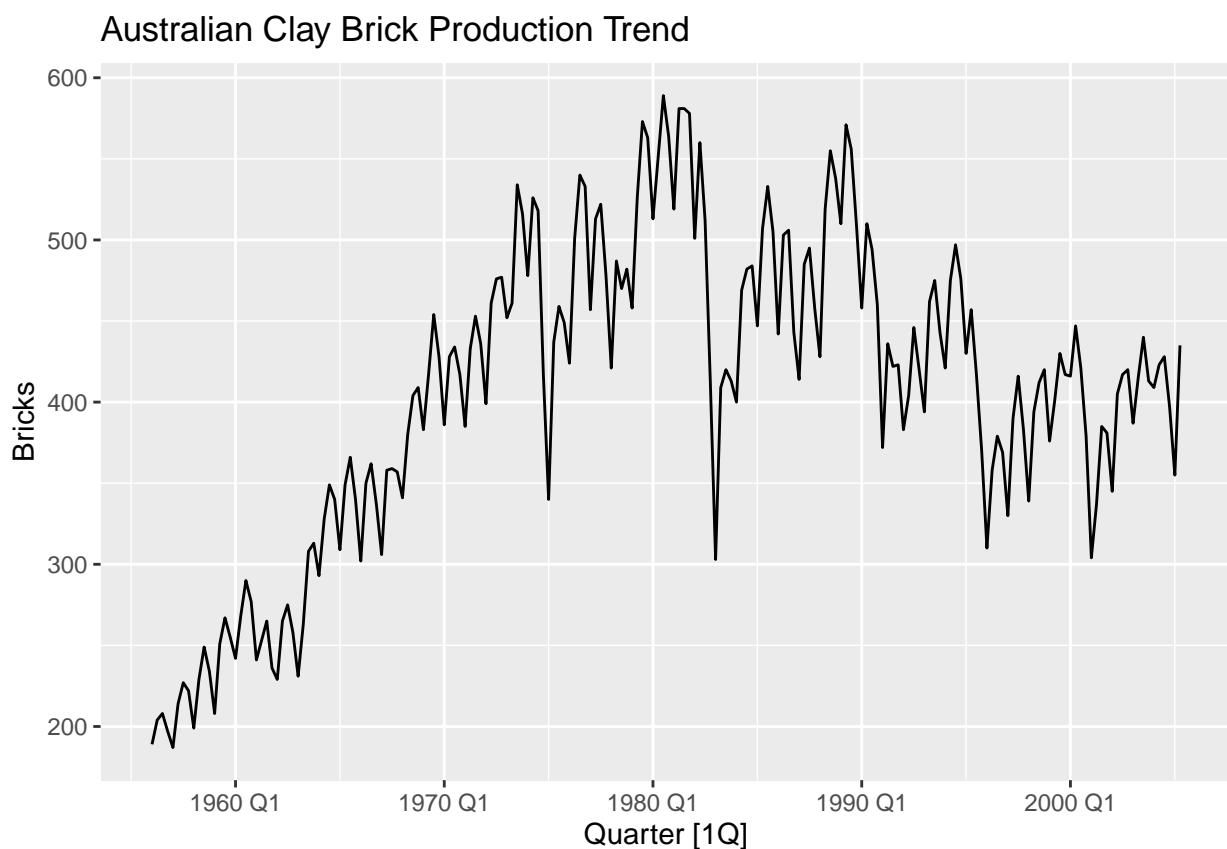
Can you identify any unusual years?

2008-2009: You can see a significant break in the trend where employment drops sharply due to the recession.

2. Bricks from aus_production
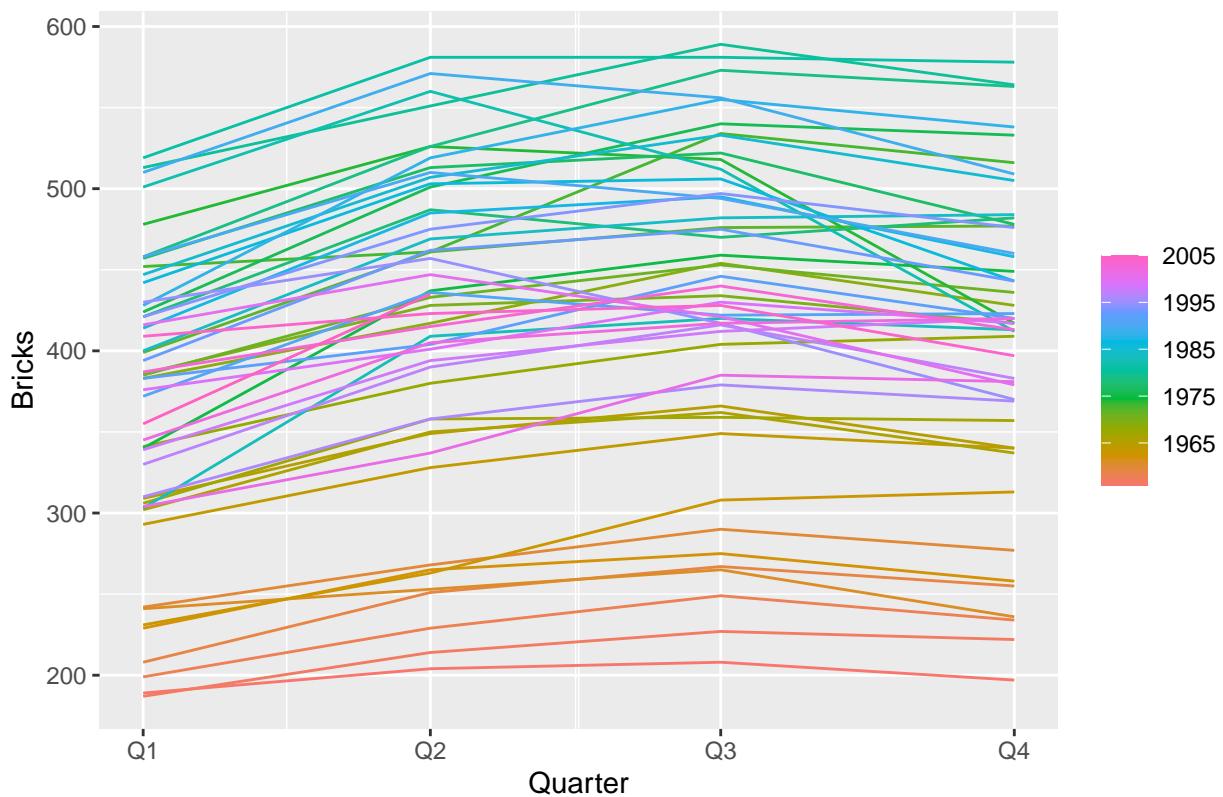
```
# Filter the data (removing NAs at the end)
bricks <- aus_production |>
  filter(!is.na(Bricks))

# 1. Time Plot
bricks |> autoplot(Bricks) +
  labs(title = "Australian Clay Brick Production Trend")
```
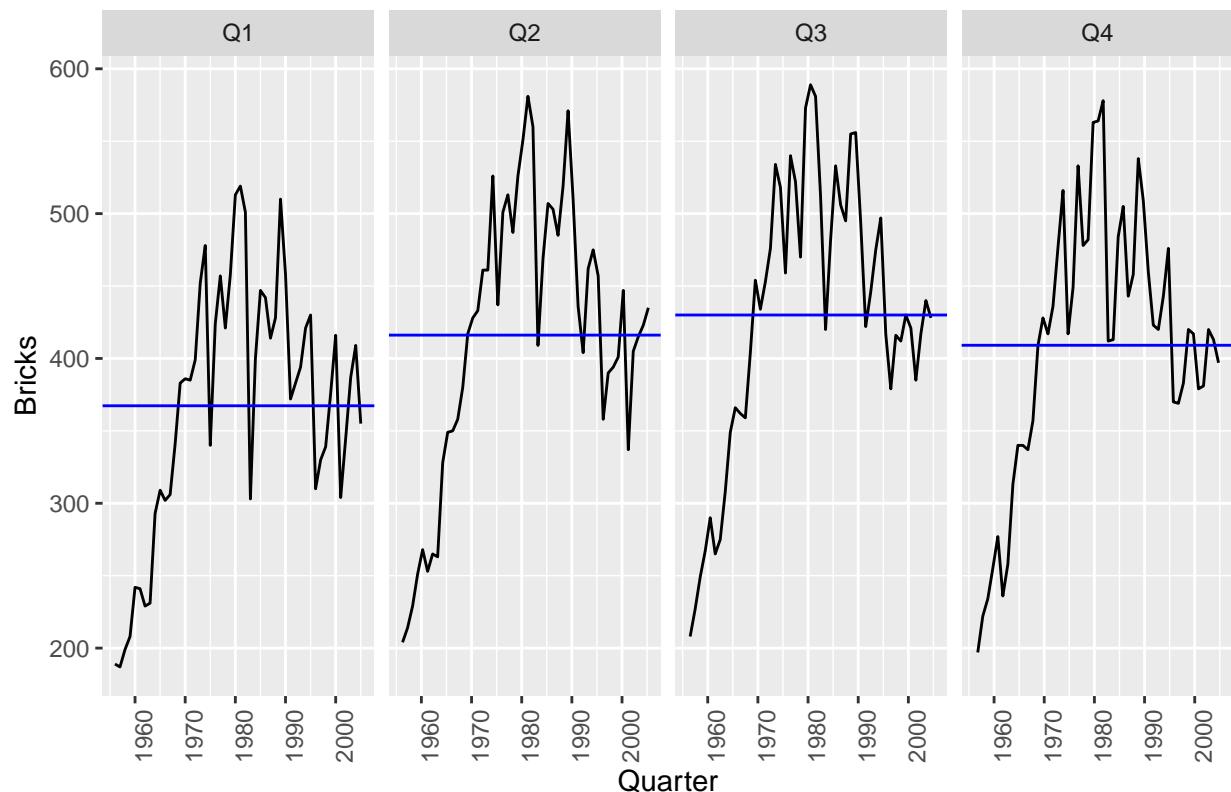


```
# 2. Seasonal Plot
bricks |> gg_season(Bricks) +
  labs(title = "Bricks Seasonal Plot")
```

Bricks Seasonal Plot
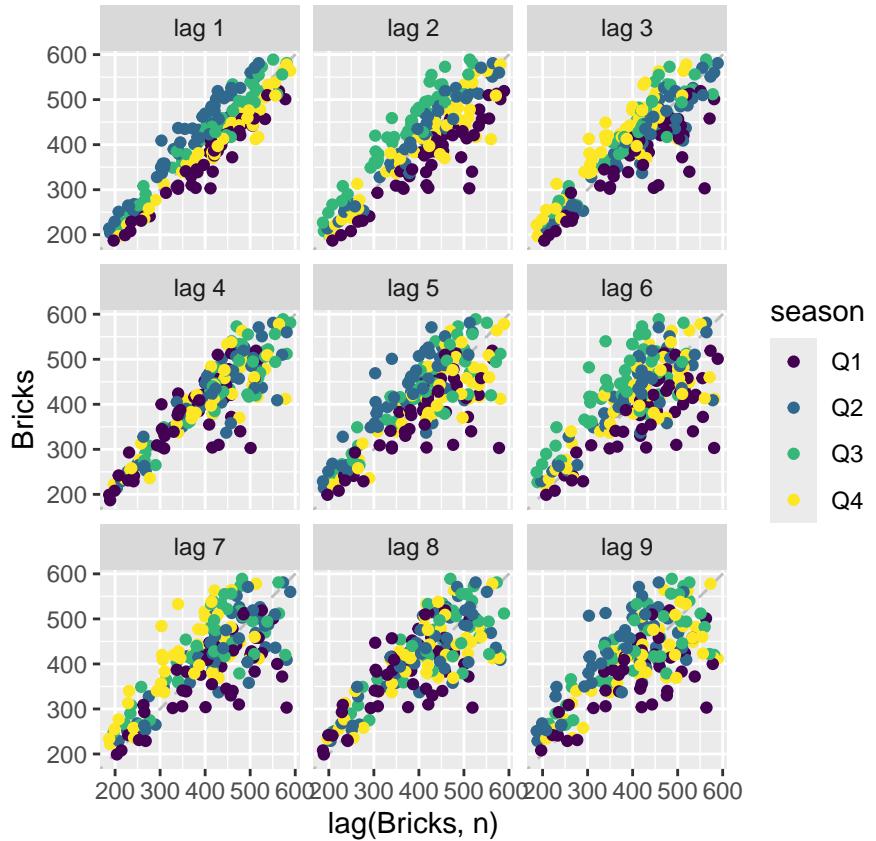
```
# 3. Subseries Plot
bricks |> gg_subseries(Bricks) +
  labs(title = "Bricks Subseries Plot")
```

20

Bricks Subseries Plot
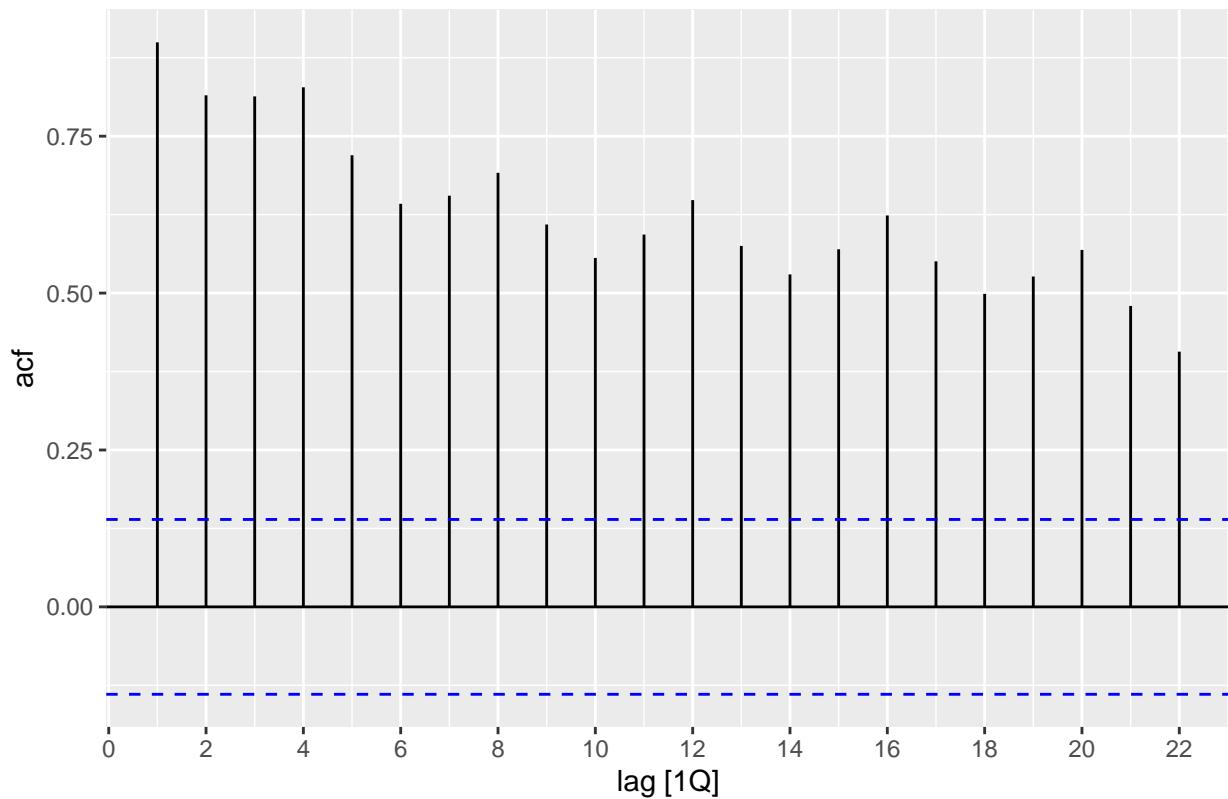

```r
# 4. Lag Plot
bricks |> gg_lag(Bricks, geom="point")
```

```
# 5. ACF Plot
bricks |> ACF(Bricks) |> autoplot() +
  labs(title = "Bricks ACF Plot")
```

Bricks ACF Plot

Seasonality, Cyclicity, and Trend:

Trend Analysis:

There is an upward trend until about 1980 after which the trend flattens and slightly declines.

Seasonality Analysis:

As you can see that the the jagged pattern repeats every 4 points showing strong quarterly seasonality.

Cyclicity Analysis:

Yes, there are clear boom and bust cycles in construction that span several years starting on top of the trend.

What do you learn about the series?

Brick production is highly sensitive to the housing market cycles. It stopped growing around 1980.

What can you say about the seasonal patterns?

Q1 is lowest: Production almost always drops in Quarter 1 due to may be Australian summer holidays.

Q3 is highest: Production tends to peak in Quarter 3.

Can you identify any unusual years?

The shift around 1980-1982 is notable and the long-term growth stops.
This suggest a structural change in the industry due to a switch to other building materials probably.

3. Hare from pelt

SKIPPED - Annual Data:

We cannot plot seasonality if you don't have seasons.
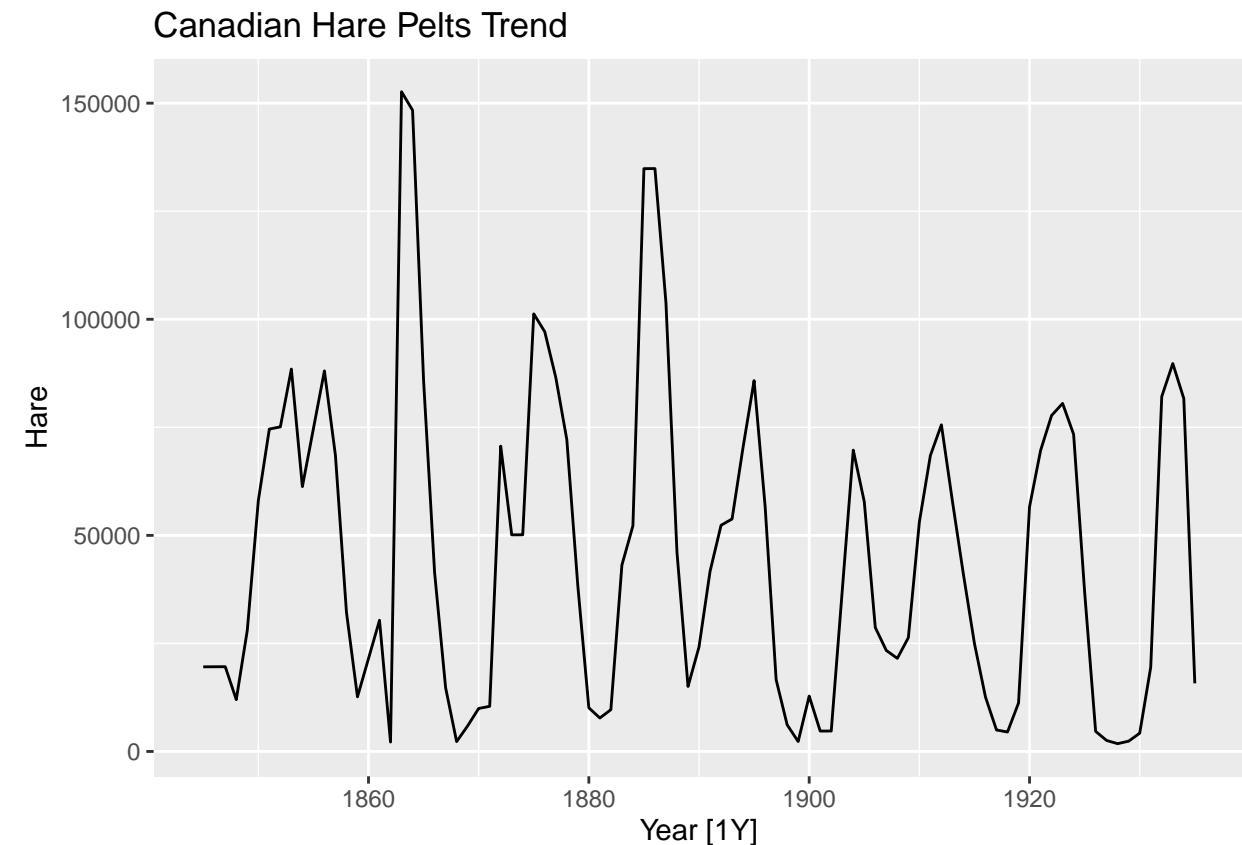
The pelt dataset only has one observation per year

gg_season() tries to put "Month" or "Quarter" on the x-axis to compare Jan 1845 vs. Jan 1846.

gg_subseries() tries to group all "Januarys" together.

Since there are no months or quarters in annual data, R has nothing to plot on the x-axis for a seasonal plot.

```
# 1. Time Plot
pelt |> autoplot(Hare) +
  labs(title = "Canadian Hare Pelts Trend")
```
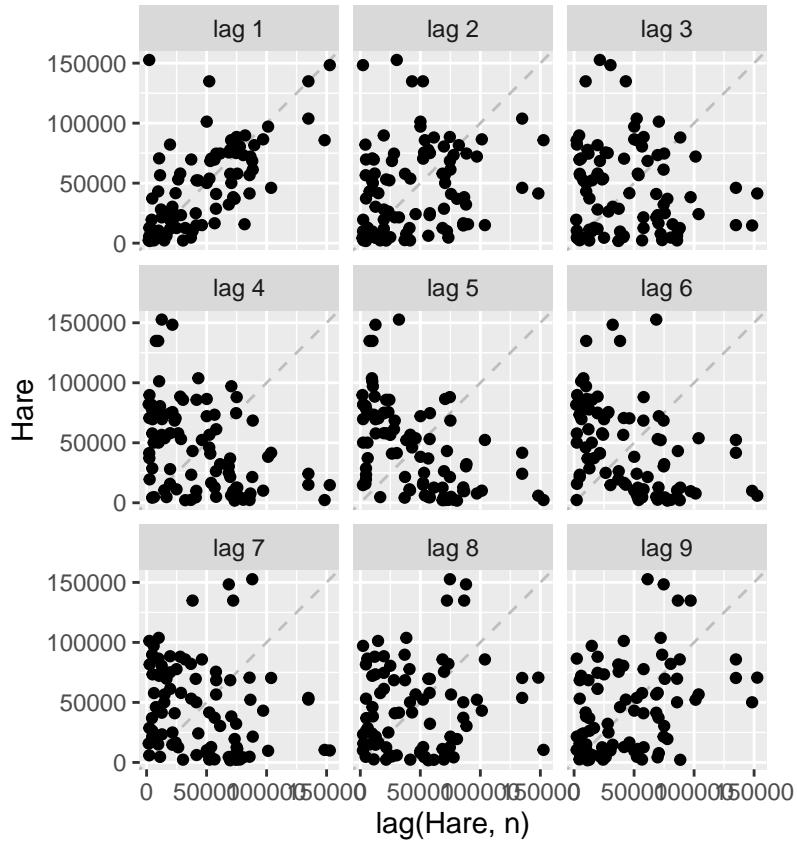


```
# 2. Seasonal Plot (SKIPPED - Annual Data)
# pelt |> gg_season(Hare)

# 3. Subseries Plot (SKIPPED - Annual Data)
```
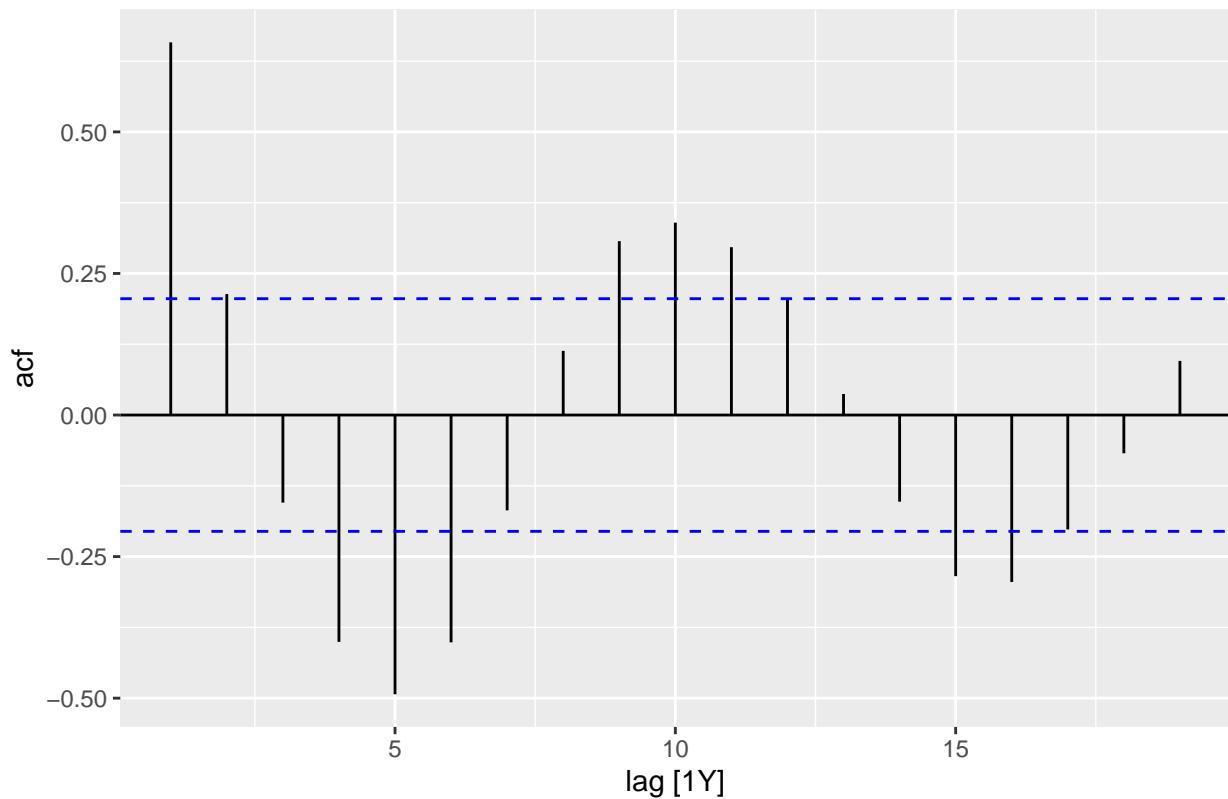
```r
# pelt |> gg_subseries(Hare)

# 4. Lag Plot
pelt |> gg_lag(Hare, geom="point")
```



```r
# 5. ACF Plot
pelt |> ACF(Hare) |> autoplot() +
  labs(title = "Hare ACF Plot")
```

Hare ACF Plot

Seasonality, Cyclicity, and Trend:

Trend Analysis:

There is no long-term upward or downward trend.

Seasonality Analysis:

None. This is annual data, so it cannot have a seasonal pattern.

Cyclicity Analysis:

There is a very regular rise and fall every almost 10 years.

What do you learn about the series?

The population follows a cycle. The peaks and troughs are massive going from near zero to huge numbers.

What can you say about the seasonal patterns?

N/A (Annual data).
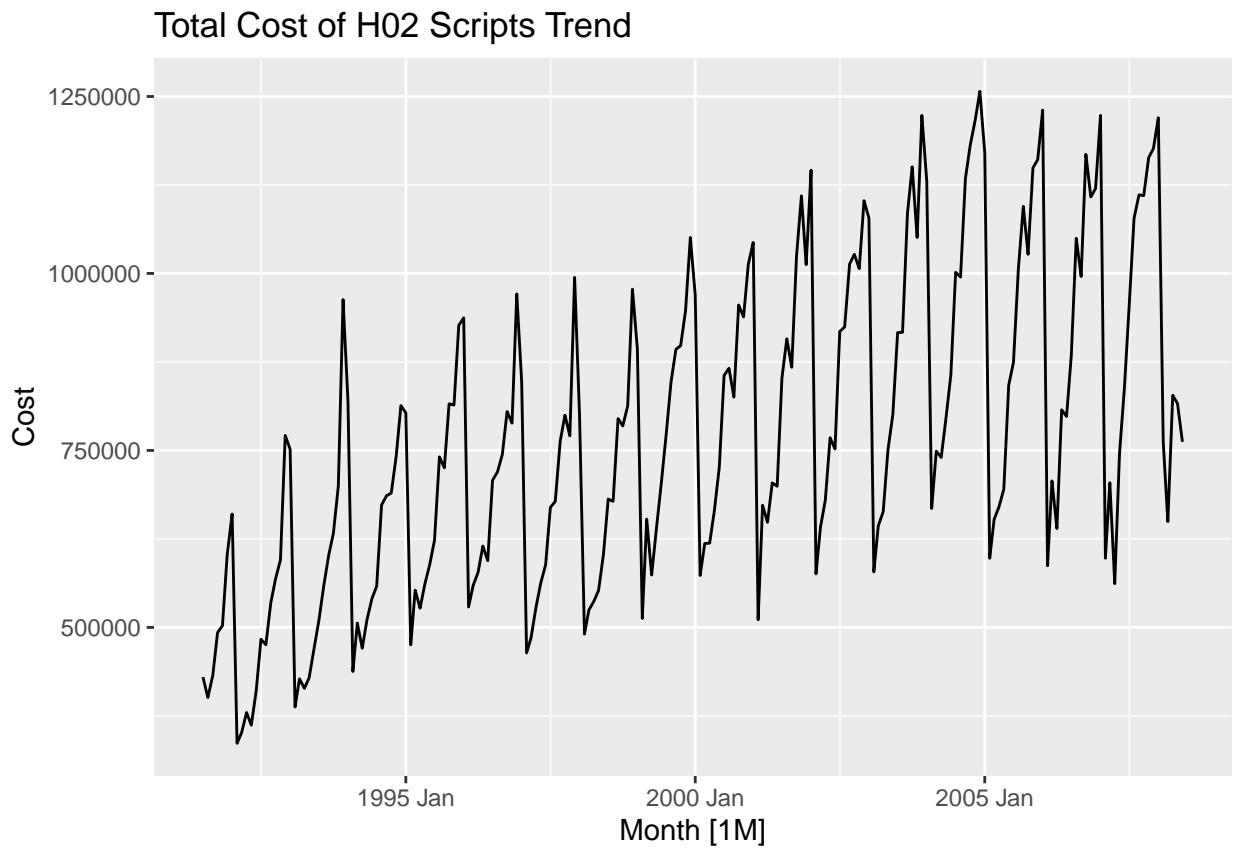
Can you identify any unusual years?

The cycles are fairly regular, but the magnitude of the peaks varies.
The peak around 1865 was exceptionally high as compare to others.

4. "H02" Cost from PBS
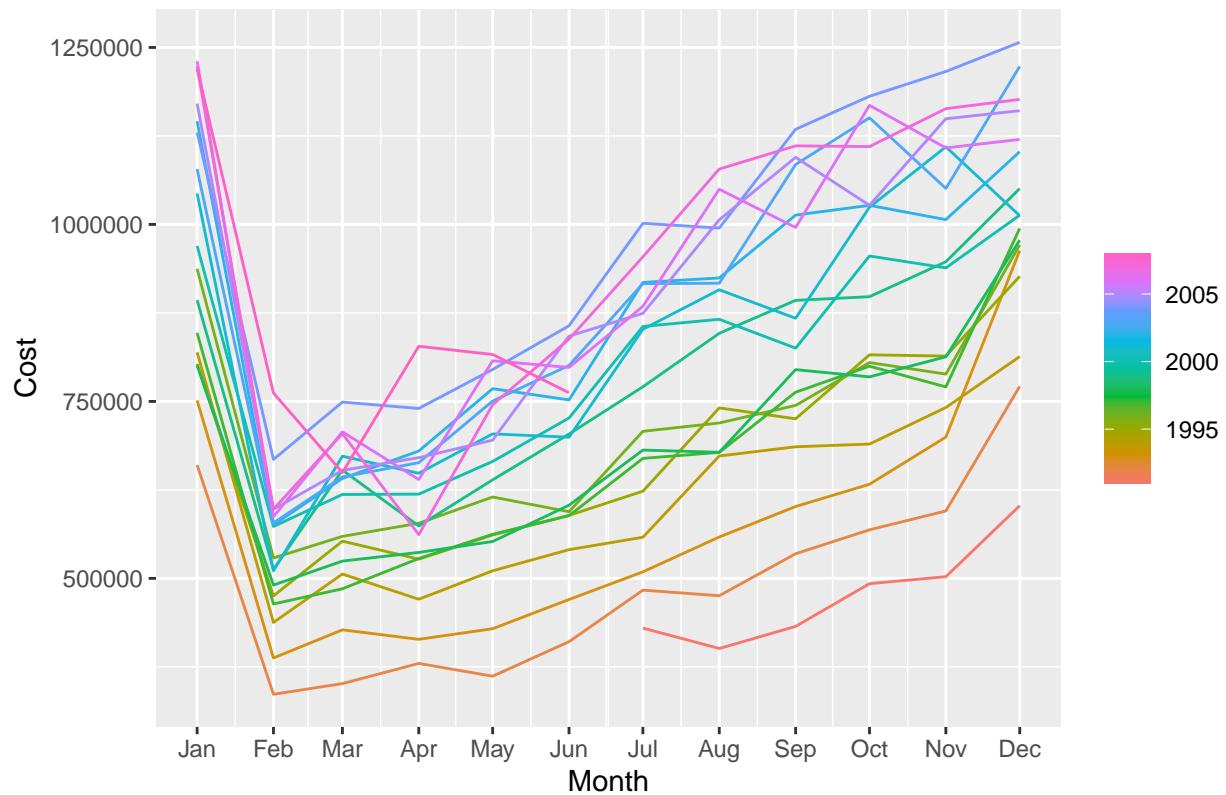
```
# Filter and Summarise the data since PBS contains many breakdowns (Concession/Type)
h02_cost <- PBS |>
  filter(ATC2 == "H02") |>
  summarise(Cost = sum(Cost))

# 1. Time Plot
h02_cost |> autoplot(Cost) +
  labs(title = "Total Cost of H02 Scripts Trend")
```
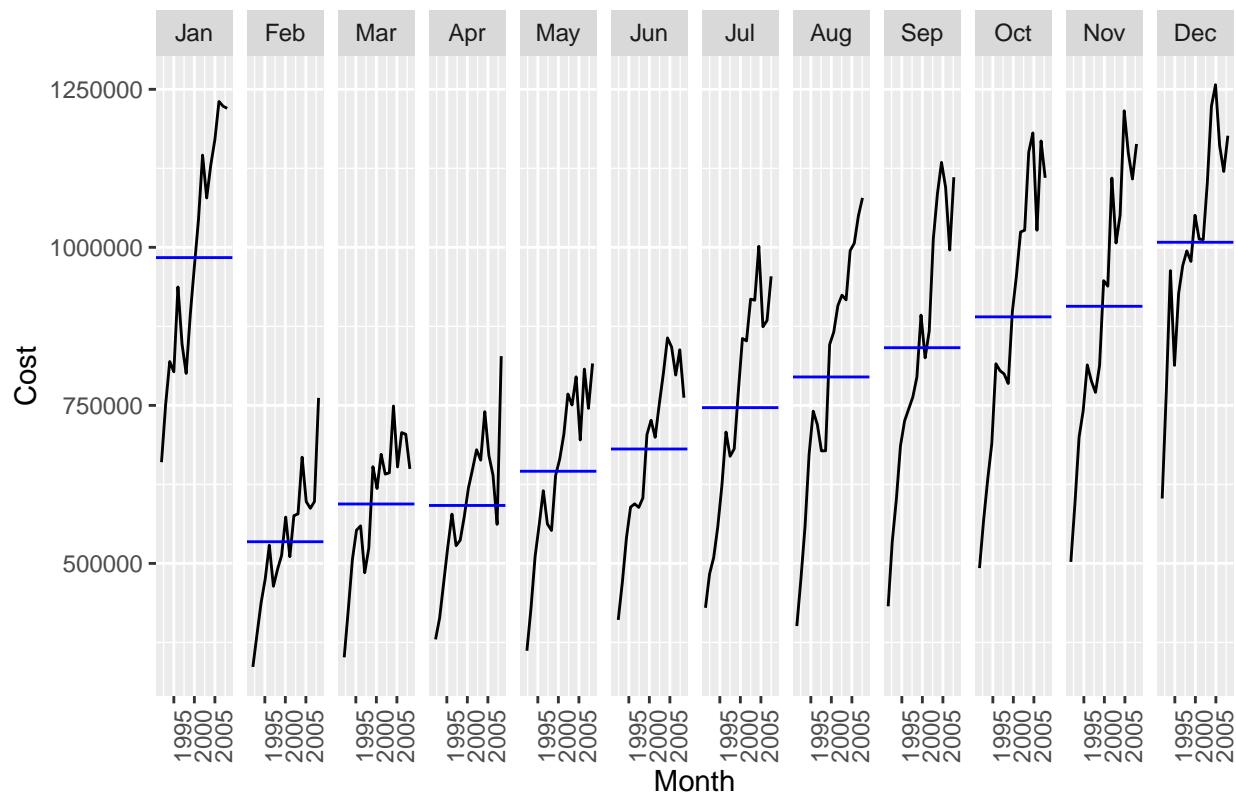


```
# 2. Seasonal Plot
h02_cost |> gg_season(Cost) +
  labs(title = "Seasonal Plot: H02 Cost")
```

Seasonal Plot: H02 Cost
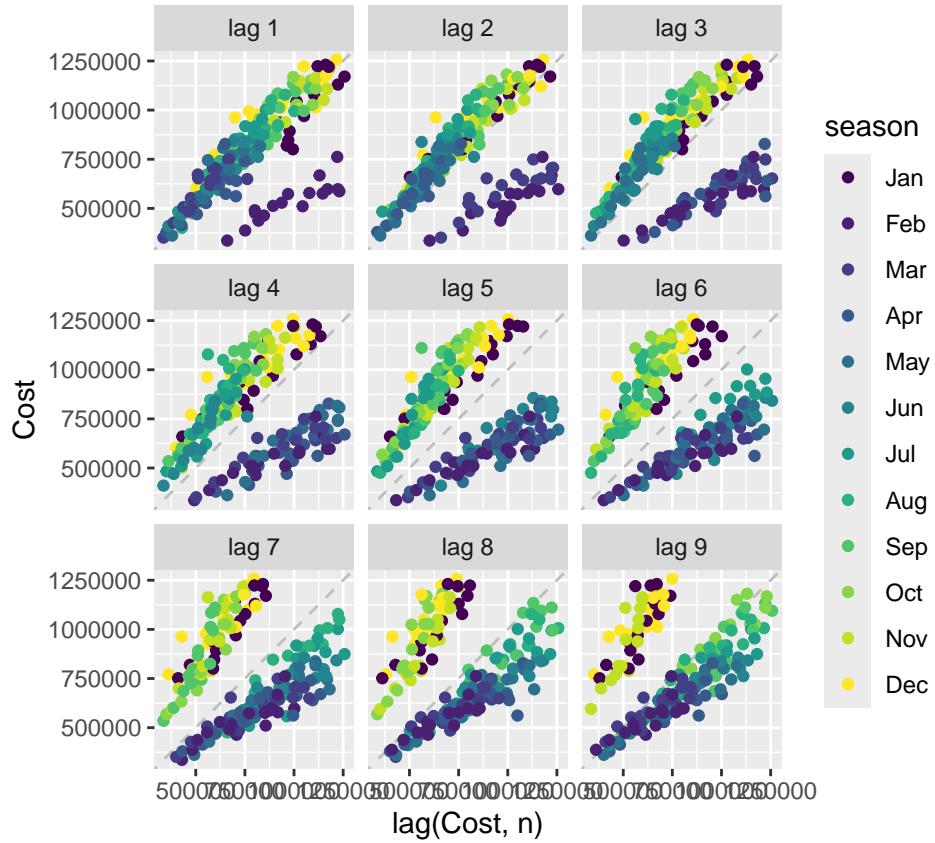

```
# 3. Subseries Plot
h02_cost |> gg_subseries(Cost) +
  labs(title = "Subseries Plot: H02 Cost")
```

## Subseries Plot: H02 Cost
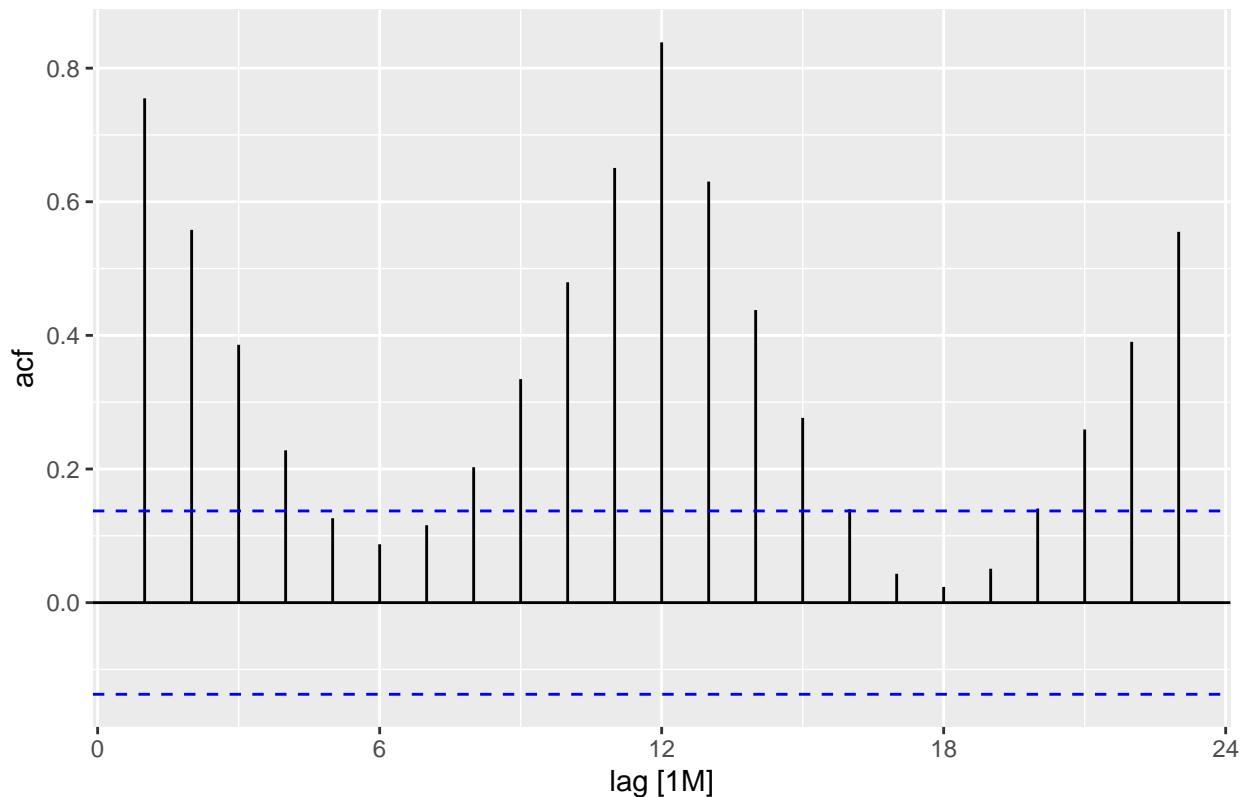

```
# 4. Lag Plot
h02_cost |> gg_lag(Cost, geom="point")
```

```r
# 5. ACF Plot
h02_cost |> ACF(Cost) |> autoplot() +
  labs(title = "H02 Cost ACF Plot")
```

H02 Cost ACF Plot

Seasonality, Cyclicity, and Trend:

Trend Analysis:

A steady upward trend in cost.

Seasonality Analysis:

There is a very strong annual seasonality.

Cyclicity Analysis:

Not clearly visible; the series is dominated by the trend and seasonality.

What do you learn about the series?

The cost of H02 drugs is growing, and the seasonal swings are getting larger.

What can you say about the seasonal patterns?

There is a sharp drop at the start of every year in January.
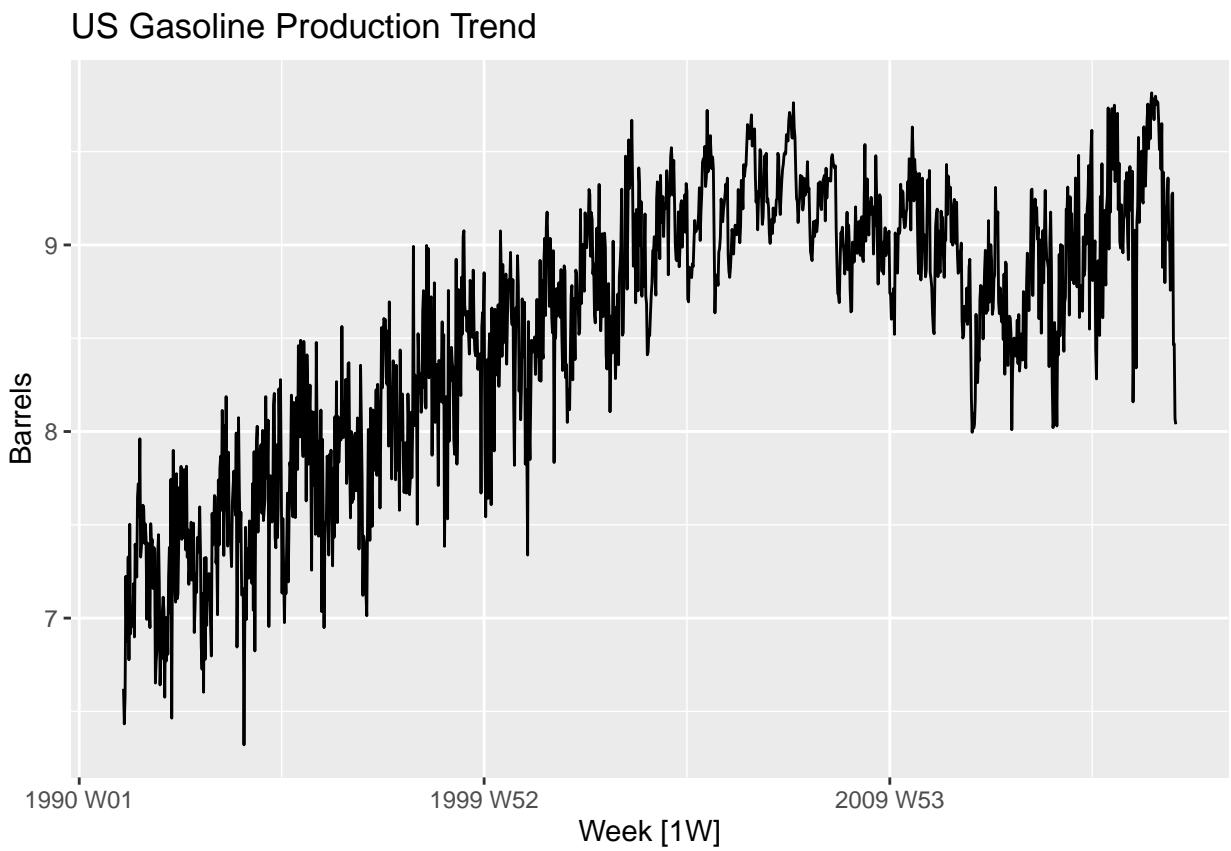
Costs rise throughout the year and peak in December.

Can you identify any unusual years?

The seasonal jumps become much larger in later years but the pattern itself is consistent.
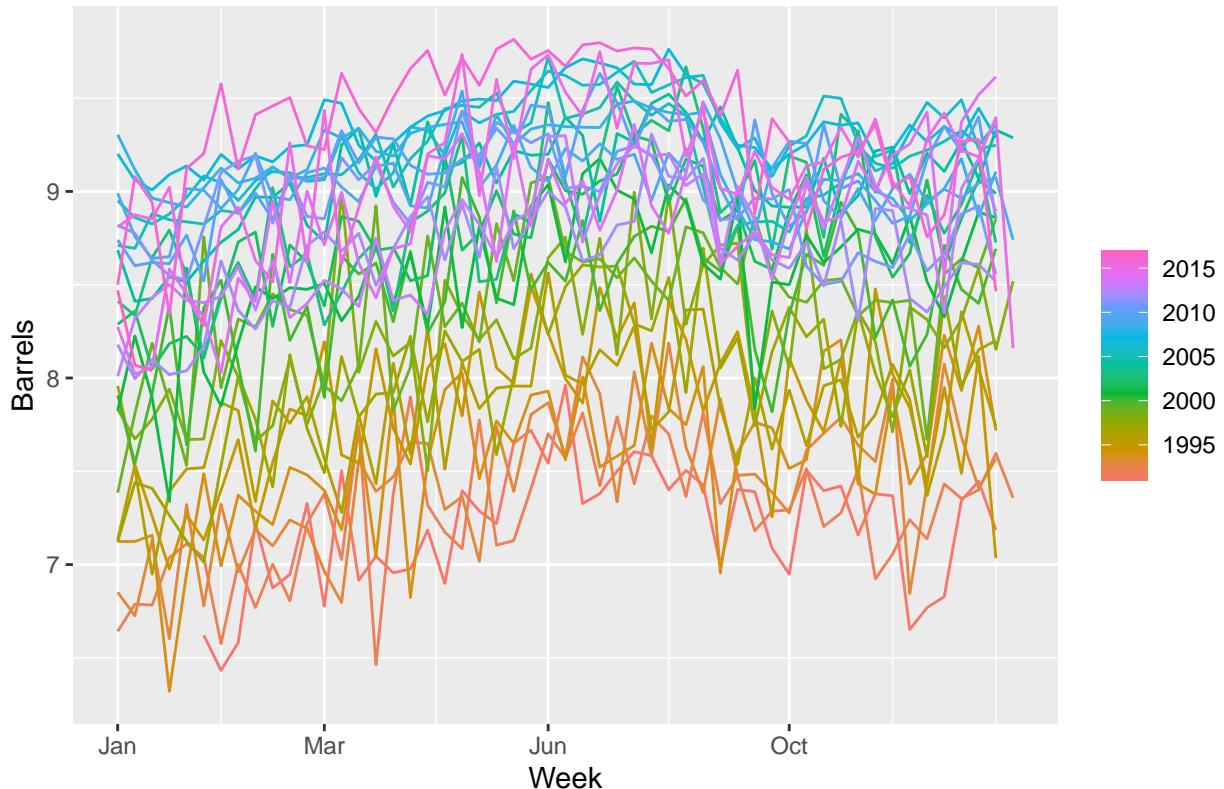
5. Barrels from us_gasoline

```
# 1. Time Plot
us_gasoline |> autoplot(Barrels) +
  labs(title = "US Gasoline Production Trend")
```

US Gasoline Production Trend
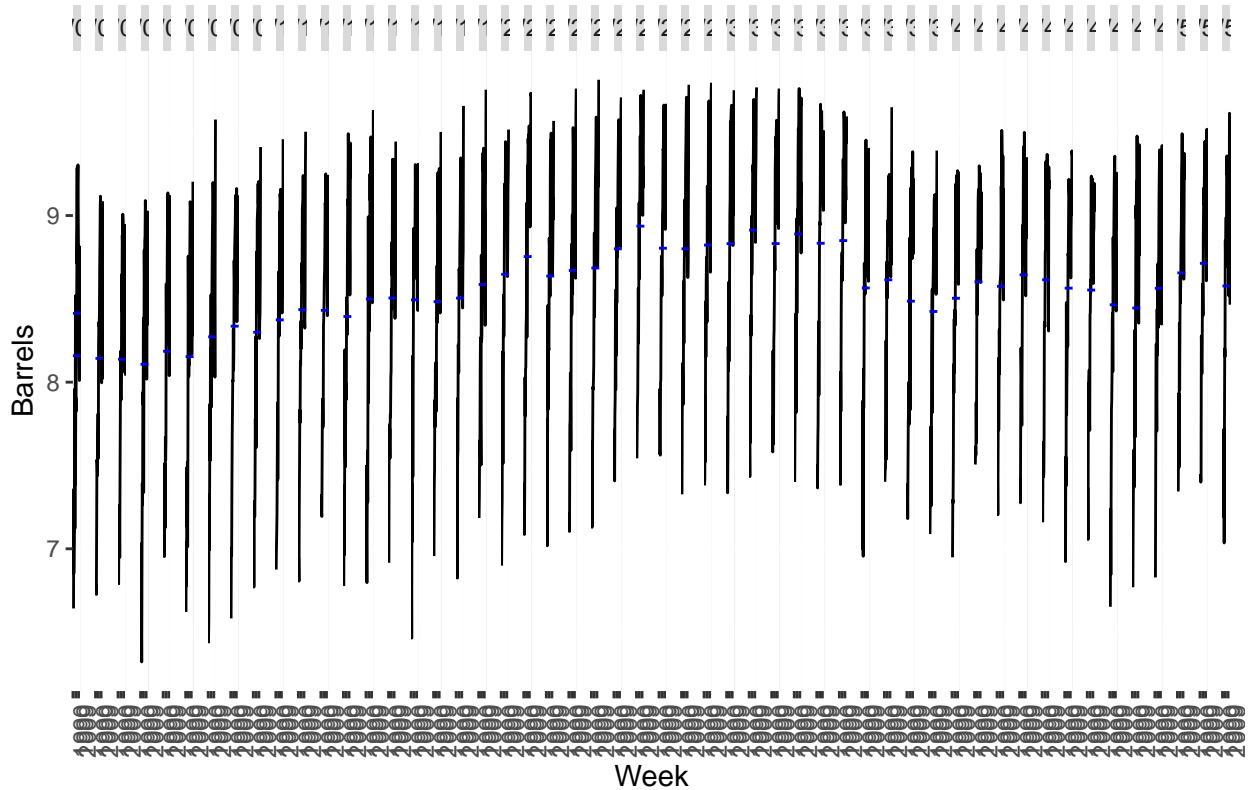

```
# 2. Seasonal Plot
# This will make visual dense because it is Weekly data
us_gasoline |> gg_season(Barrels) +
  labs(title = "Gasoline Seasonal Plot")
```

## Gasoline Seasonal Plot
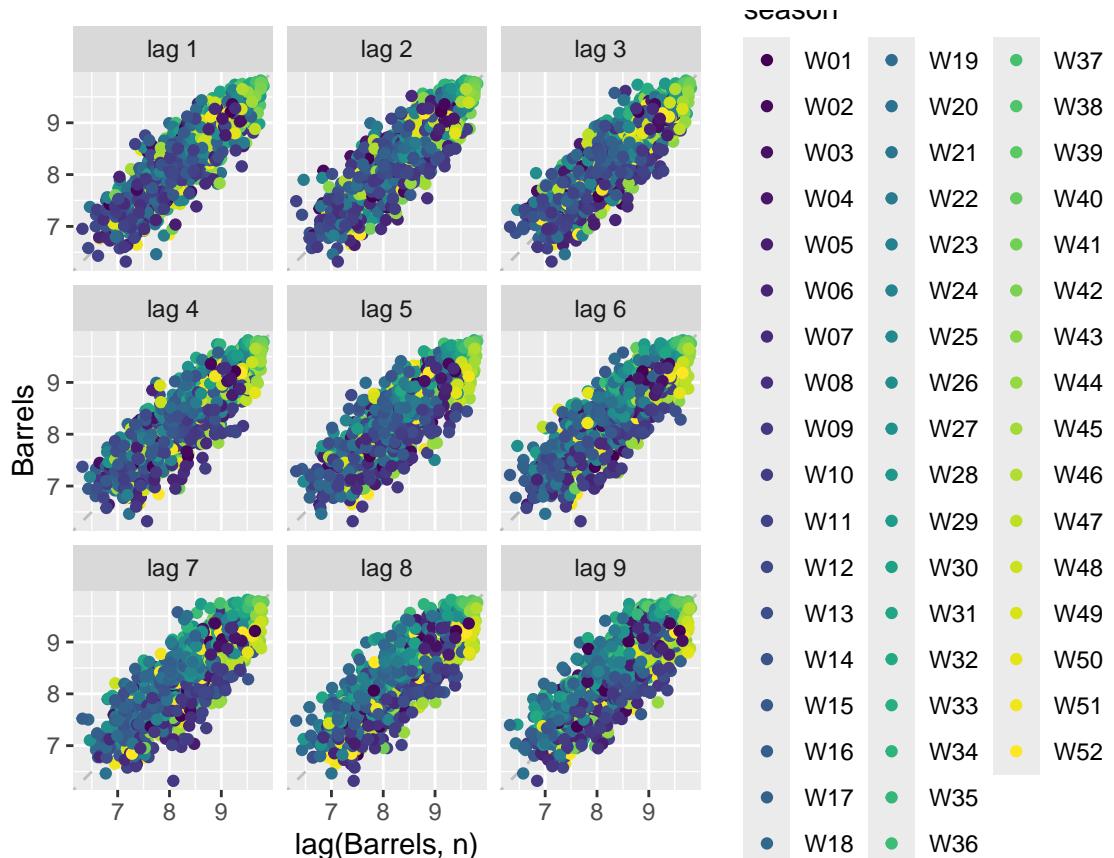

```r
# 3. Subseries Plot
us_gasoline |> gg_subseries(Barrels) +
  labs(title = "Gasoline Subseries Plot")
```

Gasoline Subseries Plot
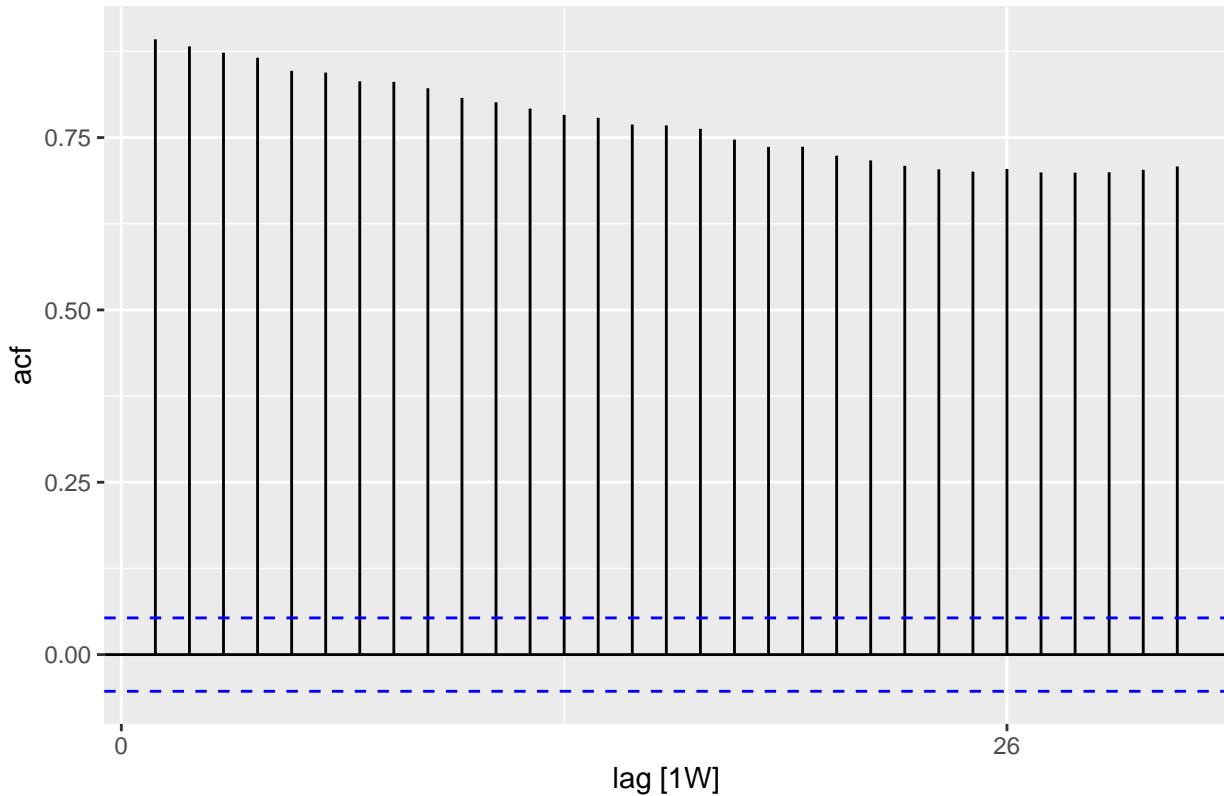

```
# 4. Lag Plot
us_gasoline |> gg_lag(Barrels, geom="point")
```

```
# 5. ACF Plot
us_gasoline |> ACF(Barrels) |> autoplot() +
  labs(title = "Gasoline ACF Plot")
```

Gasoline ACF Plot

Seasonality, Cyclicity, and Trend:

Trend Analysis:

Upward trend until about 2008, then it flattens out.

Seasonality Analysis:

Clear annual seasonality visible as a wave in the weekly data.

Cyclicity Analysis:

Minimal.

What do you learn about the series?

Gasoline production grew steadily but plateaued after 2008.

What can you say about the seasonal patterns?

Production is highest in the middle of the year.

Production drops in the winter months.

Can you identify any unusual years?

If you look closely at late 2005, you might see a sharp, unusual drop in production.
In 2008 the trend clearly changes.