Data 624 - HW1 (Fall 2024)

Khyati Naik

```
library(fpp3)
```

```
## -- Attaching packages -------------------------------------------- fpp3 1.0.0 --

## v tibble      3.2.1     v tsibble     1.1.3
## v dplyr       1.1.4     v tsibbledata 0.4.1
## v tidyr       1.3.0     v feasts      0.3.2
## v lubridate   1.9.2     v fable       0.3.4
## v ggplot2     3.5.1     v fabletools  0.4.2

## -- 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()
```

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

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

```
data("aus_production")
?aus_production
```

```
## starting httpd help server ... done
```

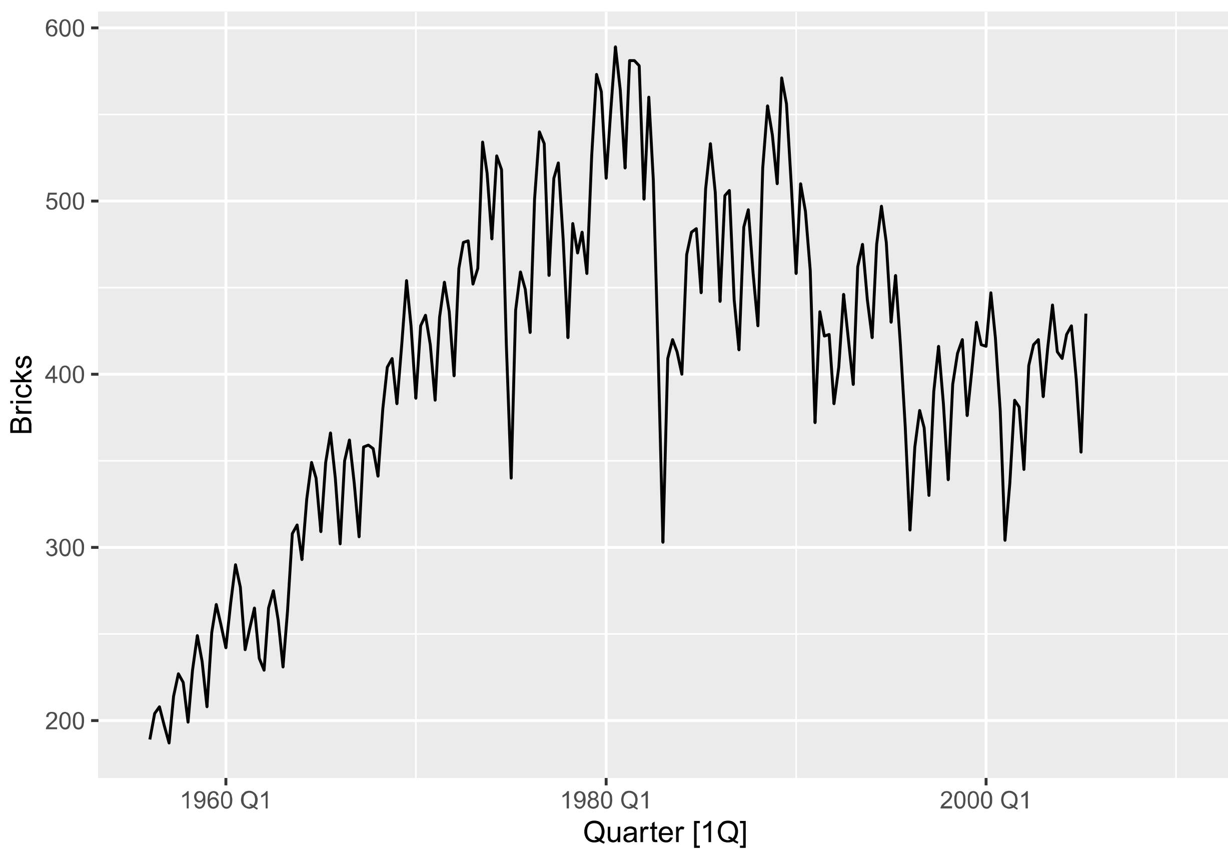
```
data("pelt")
?pelt

data("gafa_stock")
?gafa_stock

data("vic_elec")
?vic_elec
```

2.1.2 What is the time interval of each series?

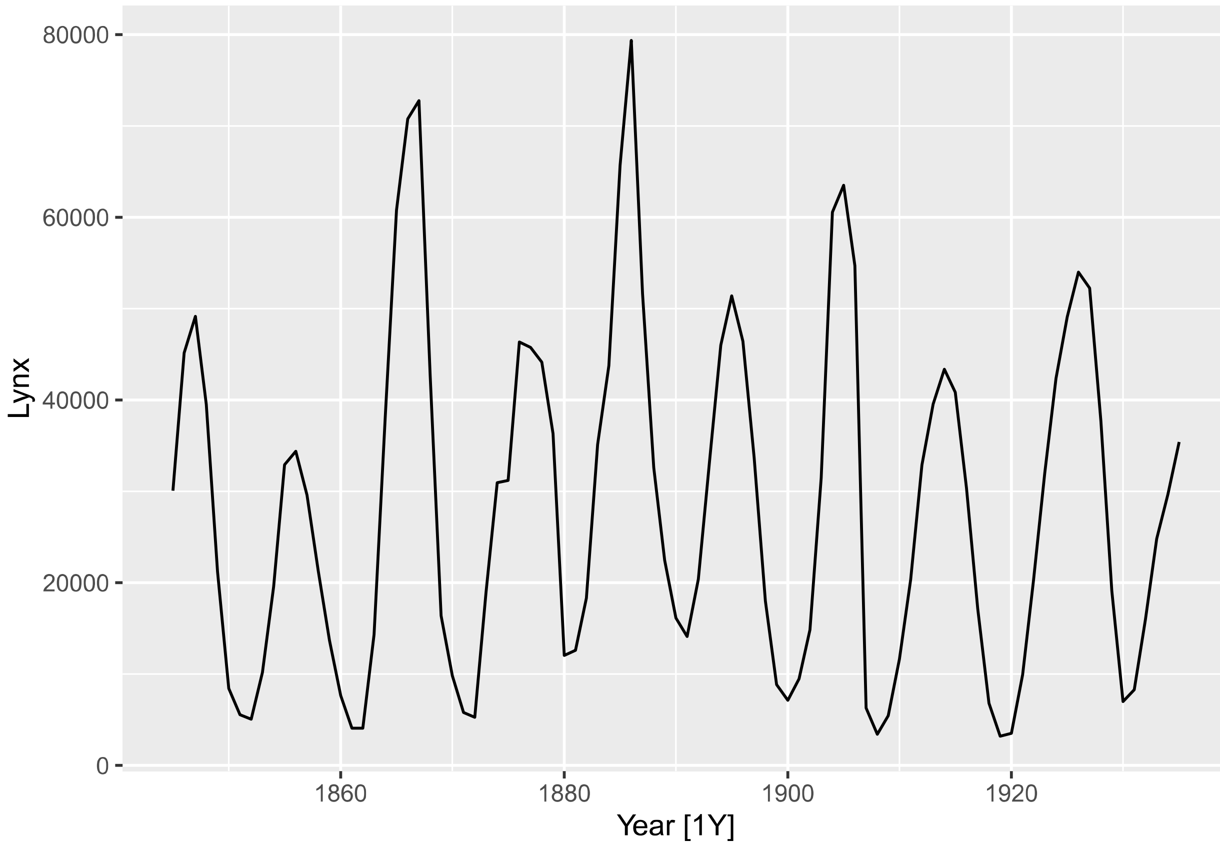
aus_production is quarterly data from 1956 to 2010.
pelt is yearly data from 1845 to 1935.
gafa_stock business day data when the Market is open from 2014 to 2018.
vic_elec is every 30 minutes data from 2012 to 2014.

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

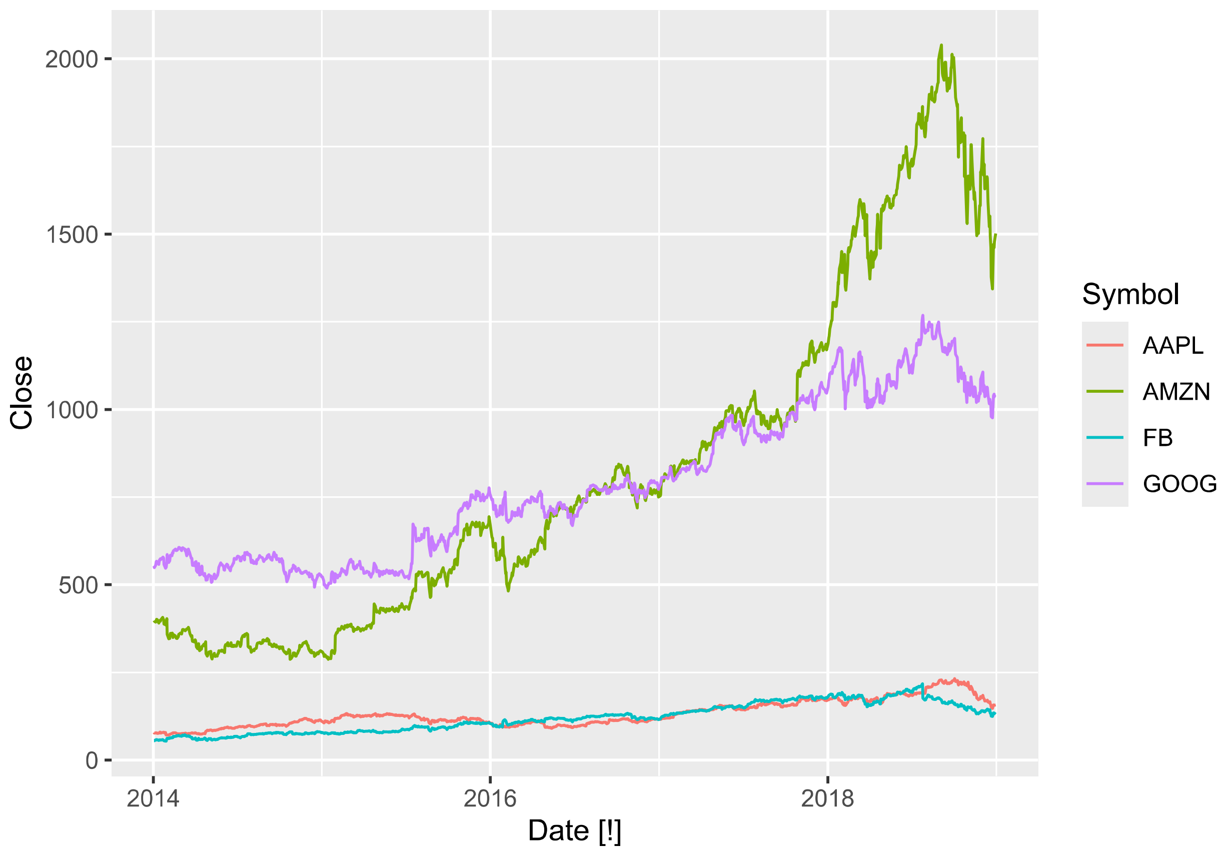
```
aus_production %>%  autoplot(Bricks)
```




```
pelt %>%  autoplot(Lynx)
```

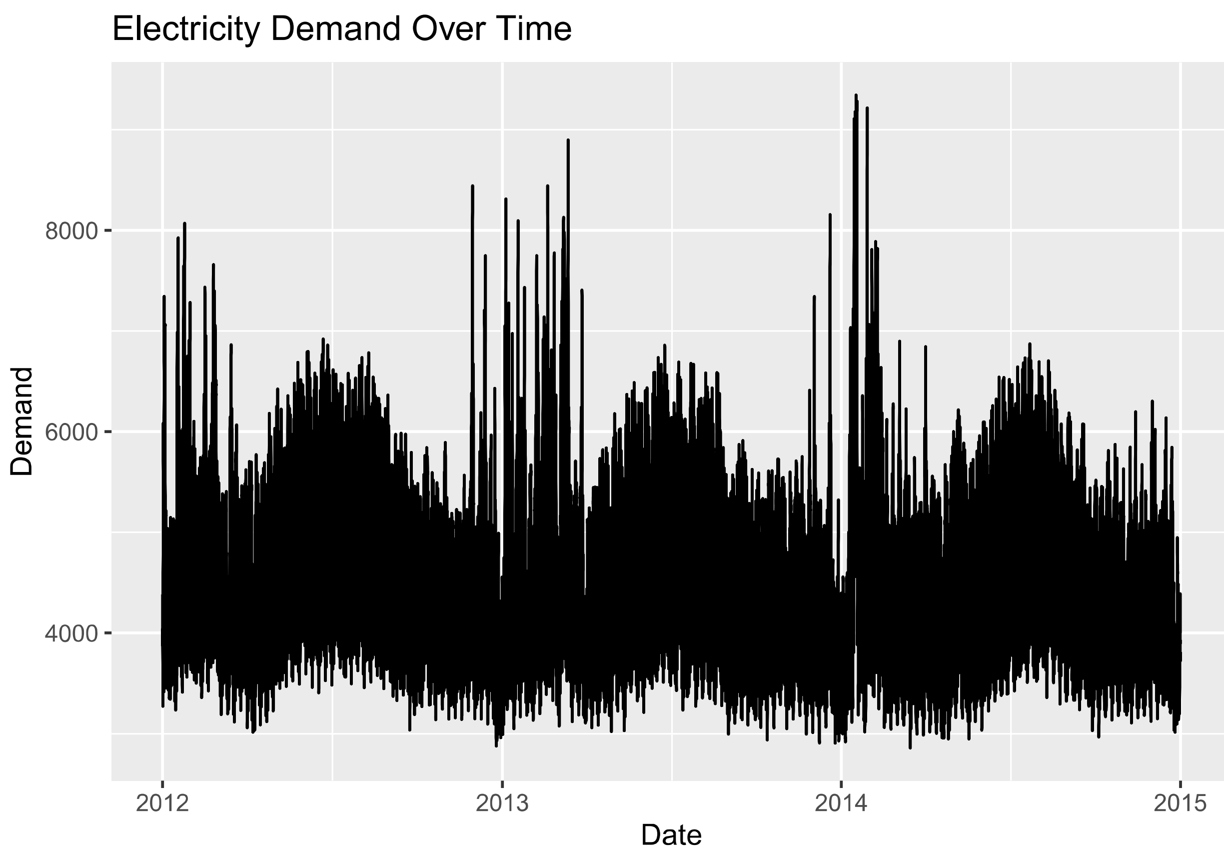



```
gafa_stock %>%   autoplot(Close)
```



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

```
vic_elec %>%   autoplot(Demand) +
  labs(x = "Date", y = "Demand") +
  ggtitle("Electricity Demand Over Time")
```



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

```
gafa_stock %>%
             group_by(Symbol) %>%
             filter(Close == max(Close))
```

```
## # A tsibble: 4 x 8 [!]
## # Key:       Symbol [4]
## # Groups:    Symbol [4]
##   Symbol Date        Open  High   Low Close Adj_Close   Volume
##   <chr>  <date>     <dbl> <dbl> <dbl> <dbl>     <dbl>    <dbl>
## 1 AAPL   2018-10-03  230.  233.  230.  232.      230. 28654800
## 2 AMZN   2018-09-04 2026. 2050. 2013  2040.     2040.  5721100
## 3 FB     2018-07-25  216.  219.  214.  218.      218. 58954200
## 4 GOOG   2018-07-26 1251  1270. 1249. 1268.     1268.  2405600
```

2.3 Download the file tute1.csv from the book website, open it in Excel (or some other spreadsheet application), and review its contents. You should find four columns of information. Columns B through D each contain a quarterly series, labelled Sales, AdBudget and GDP. Sales contains the quarterly sales for a small company over the period 1981-2005. AdBudget is the advertising budget and GDP is the gross domestic product. All series have been adjusted for inflation.

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

```
tute1 <- readr::read_csv("https://raw.githubusercontent.com/Naik-Khyati/data_624/main/hw1/tute1.csv")
```

```
## Rows: 100 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl  (3): Sales, AdBudget, GDP
## date (1): Quarter
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

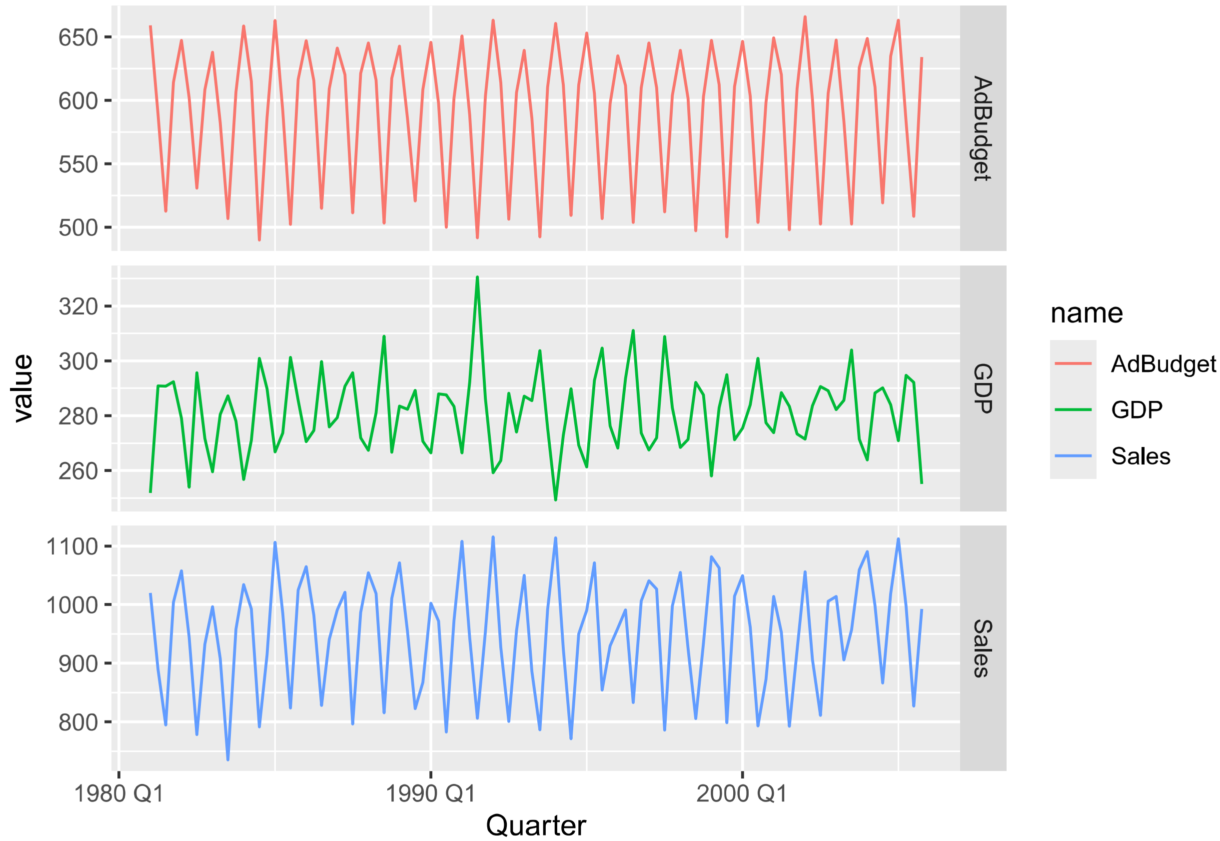
```
View(tute1)
```

2.3.b Convert the data to time series

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

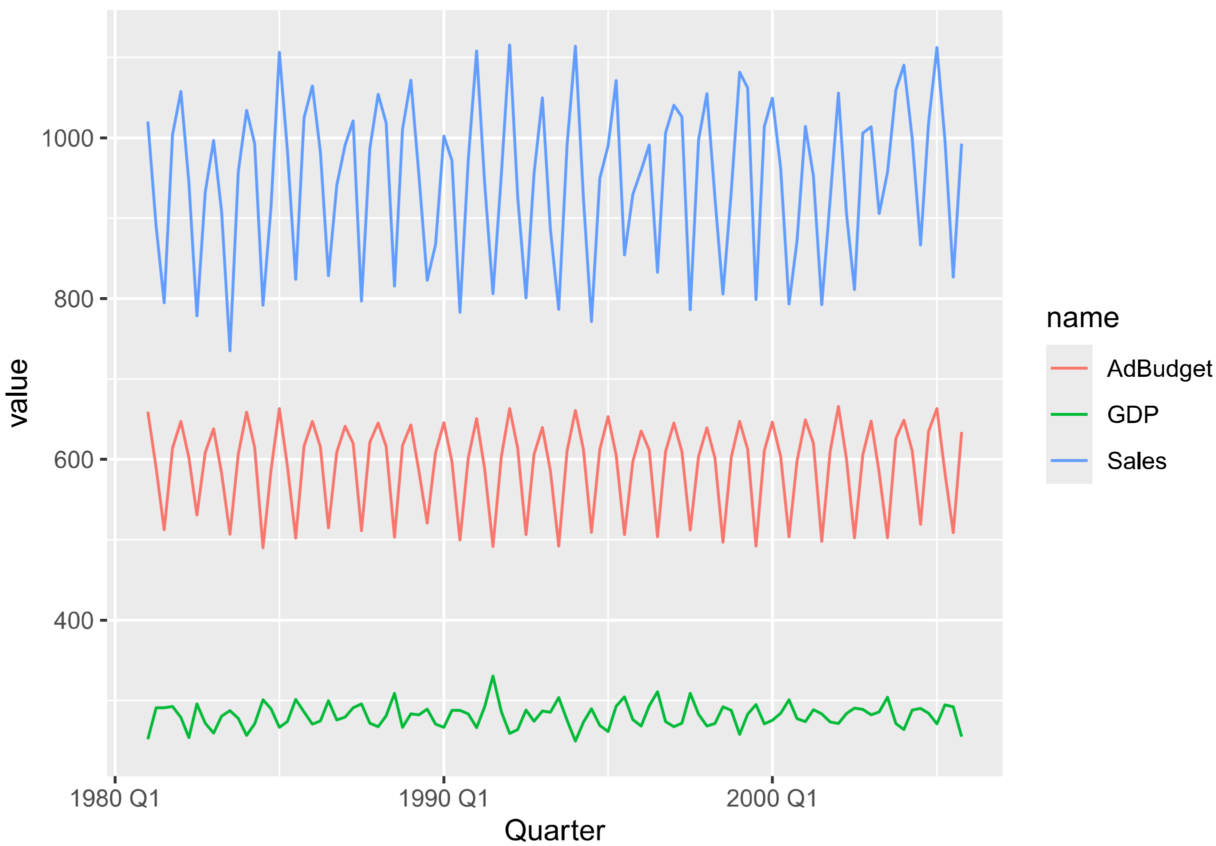
2.3.c Construct time series plots of each of the three series

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



2.3.c.1 Check what happens when you don't include facet_grid().

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



When facet_grid() is not included in the ggplot code, the result is a single plot where all the time series data (for different variables represented by name) are plotted together in one chart.

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

2.4.a Install the USgas package.

```r
library(USgas)
```

2.4.b Create a tsibble from us_total with year as the index and state as the key.

```r
ts <- us_total

ts <- ts %>%
  as_tsibble(index = year, key = state)

head(ts)
```

```
## # A tsibble: 6 x 3 [1Y]
## # Key:       state [1]
##    year state        y
##   <int> <chr>    <int>
```

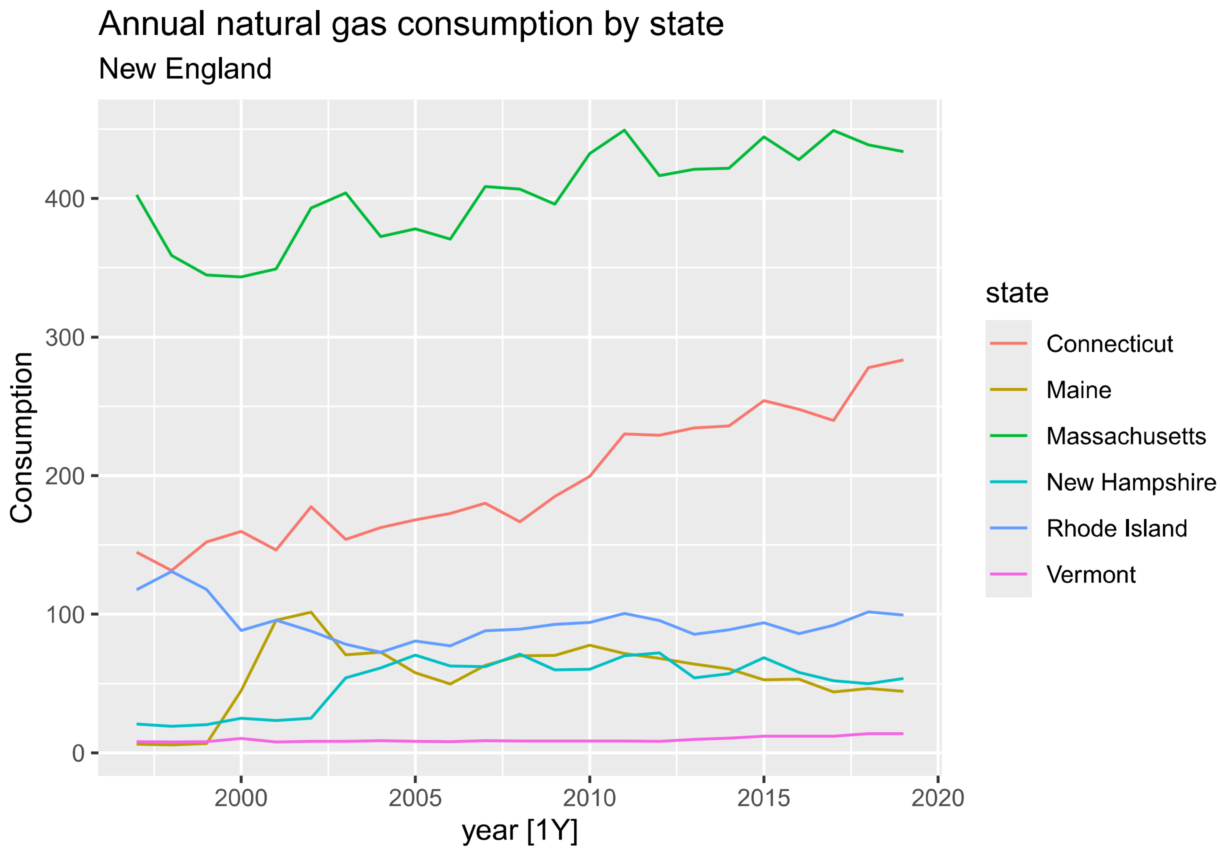
```
## 1  1997 Alabama 324158
## 2  1998 Alabama 329134
## 3  1999 Alabama 337270
## 4  2000 Alabama 353614
## 5  2001 Alabama 332693
## 6  2002 Alabama 379343
```

2.4.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).

```
ne_ts <- ts %>%
filter(state %in% c('Maine', 'Vermont', 'New Hampshire', 'Massachusetts', 'Connecticut', 'Rhode Island'

head(ne_ts)
```

```
## # A tsibble: 6 x 3 [1Y]
## # Key:       state [1]
##    year state           y
##   <int> <chr>       <dbl>
## 1  1997 Connecticut  145.
## 2  1998 Connecticut  131.
## 3  1999 Connecticut  152.
## 4  2000 Connecticut  160.
## 5  2001 Connecticut  146.
## 6  2002 Connecticut  178.
```

```
autoplot(ne_ts, y) +
  labs(title = "Annual natural gas consumption by state",
       subtitle = "New England",
       y = "Consumption")
```



2.5 Tourism Data Analysis

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

```r
library(readxl)
library(httr)

# URL of the raw Excel file
url <- "https://raw.githubusercontent.com/Naik-Khyati/data_624/main/hw1/tourism.xlsx"

# Temporary file to store the downloaded Excel
temp_file <- tempfile(fileext = ".xlsx")

# Download the file
GET(url, write_disk(temp_file, overwrite = TRUE))
```

```
## Response [https://raw.githubusercontent.com/Naik-Khyati/data_624/main/hw1/tourism.xlsx]
##   Date: 2024-09-08 16:48
##   Status: 200
##   Content-Type: application/octet-stream
##   Size: 679 kB
## <ON DISK>  C:\Users\User\AppData\Local\Temp\RtmpiSI9C7\file22f02f826d13.xlsx
```

```r
# Read the Excel file from the temporary location
tourism_data <- read_excel(temp_file, sheet = "Sheet1")

# View the first few rows of the dataset
head(tourism_data)
```

```
## # A tibble: 6 x 5
##   Quarter    Region   State           Purpose  Trips
##   <chr>      <chr>    <chr>           <chr>    <dbl>
## 1 1998-01-01 Adelaide South Australia Business  135.
## 2 1998-04-01 Adelaide South Australia Business  110.
## 3 1998-07-01 Adelaide South Australia Business  166.
## 4 1998-10-01 Adelaide South Australia Business  127.
## 5 1999-01-01 Adelaide South Australia Business  137.
## 6 1999-04-01 Adelaide South Australia Business  200.
```

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

```r
# Convert tourism_data to tsibble
tourism_ts <- tourism_data %>%
  mutate(Quarter = yearquarter(Quarter)) %>%
  as_tsibble(index = Quarter, key = c(Region, State, Purpose))

head(tourism_ts)
```

```
## # A tsibble: 6 x 5 [1Q]
## # Key:       Region, State, Purpose [1]
##   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.
```

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

```r
reg_pur_max_on_trips <- tourism_data %>%
  group_by(Region, Purpose) %>%
  summarise(Trip_Avg = mean(Trips)) %>%
  filter(Trip_Avg == max(Trip_Avg)) %>%
  arrange(desc(Trip_Avg))
```

```
## `summarise()` has grouped output by 'Region'. You can override using the
## `.groups` argument.
```

```r
head(reg_pur_max_on_trips)
```

```
## # A tibble: 6 x 3
## # Groups:   Region [6]
##   Region          Purpose  Trip_Avg
##   <chr>           <chr>       <dbl>
## 1 Sydney          Visiting     747.
## 2 Melbourne       Visiting     619.
## 3 North Coast NSW Holiday      588.
## 4 Gold Coast      Holiday      528.
## 5 South Coast     Holiday      495.
## 6 Brisbane        Visiting     493.
```

The combination of the Sydney region and the purpose of Visiting has the highest average number of overnight trips per quarter, with 747 trips.

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

```r
reg_pur_tot_trips_state <- tourism_data %>%
  mutate(Quarter = as.Date(Quarter)) %>%
           group_by(State) %>%
           summarise(total_trips = sum(Trips)) %>%
           arrange(desc(total_trips))

head(reg_pur_tot_trips_state)
```

```
## # A tibble: 6 x 2
##   State              total_trips
##   <chr>                    <dbl>
## 1 New South Wales        557367.
## 2 Victoria               390463.
## 3 Queensland             386643.
## 4 Western Australia      147820.
## 5 South Australia        118151.
## 6 Tasmania                54137.
```

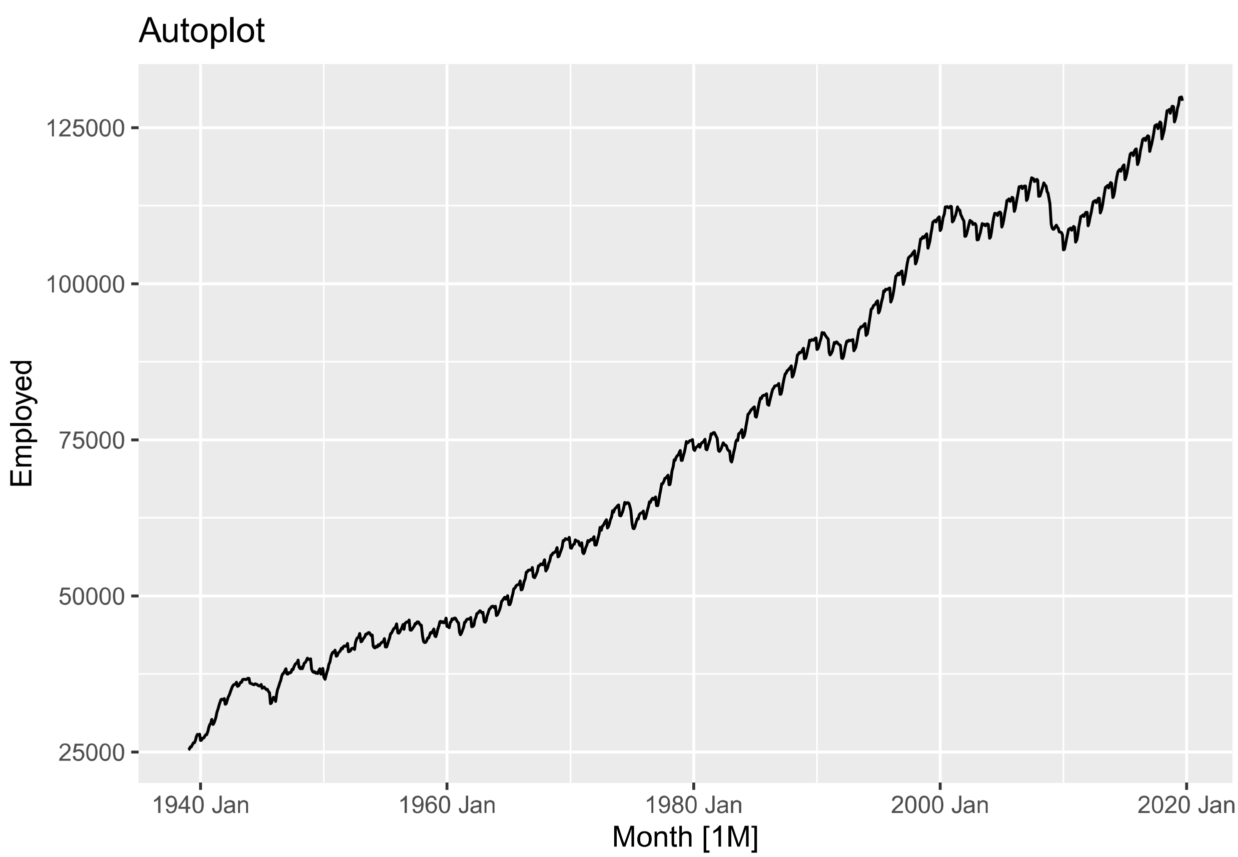
New South Wales, Victoria, and Queensland have a significant lead in total trips.

2.8. 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.

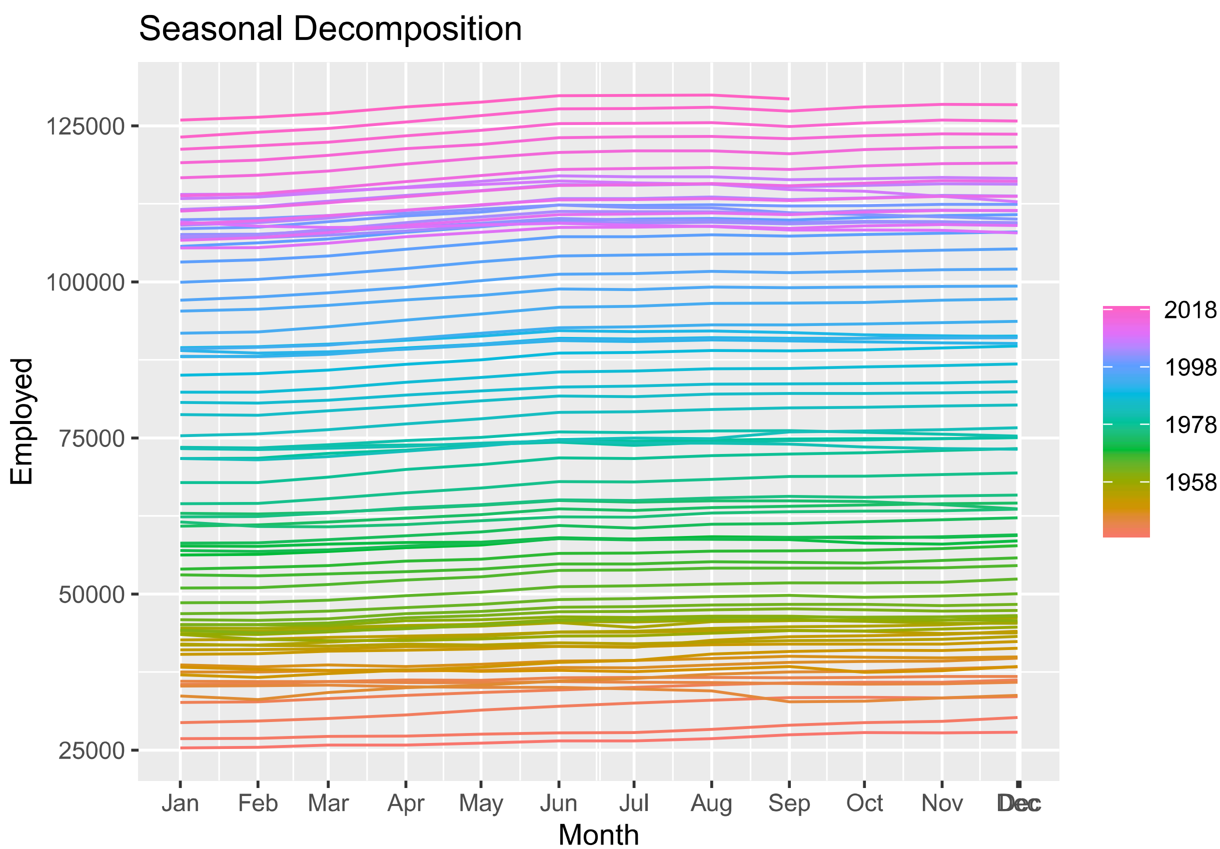
Can you spot any seasonality, cyclicity and trend? What do you learn about the series? What can you say about the seasonal patterns? Can you identify any unusual years?

```r
data("PBS")
data("us_employment")
data("us_gasoline")
```

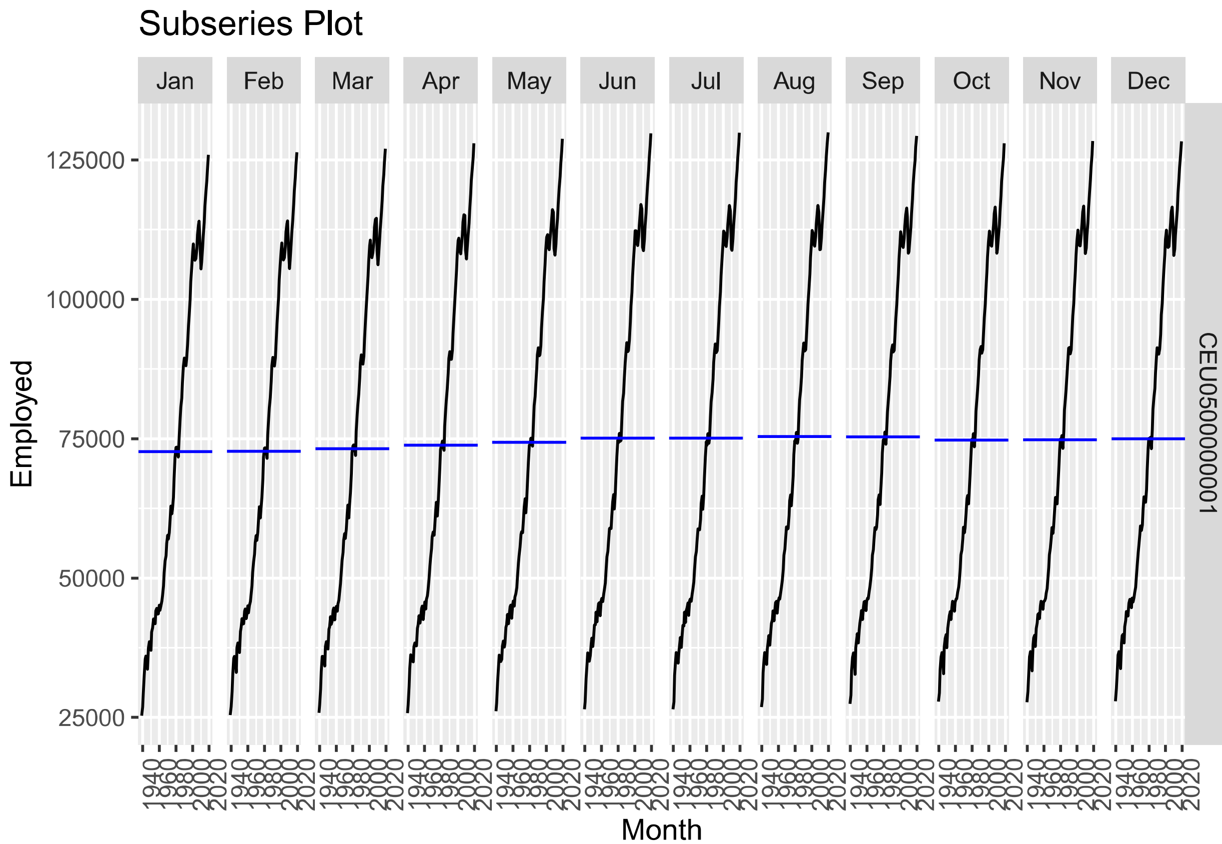
```r
us_employment %>%
  filter(Title == "Total Private") %>%
  autoplot(Employed) +
  ggtitle("Autoplot")
```



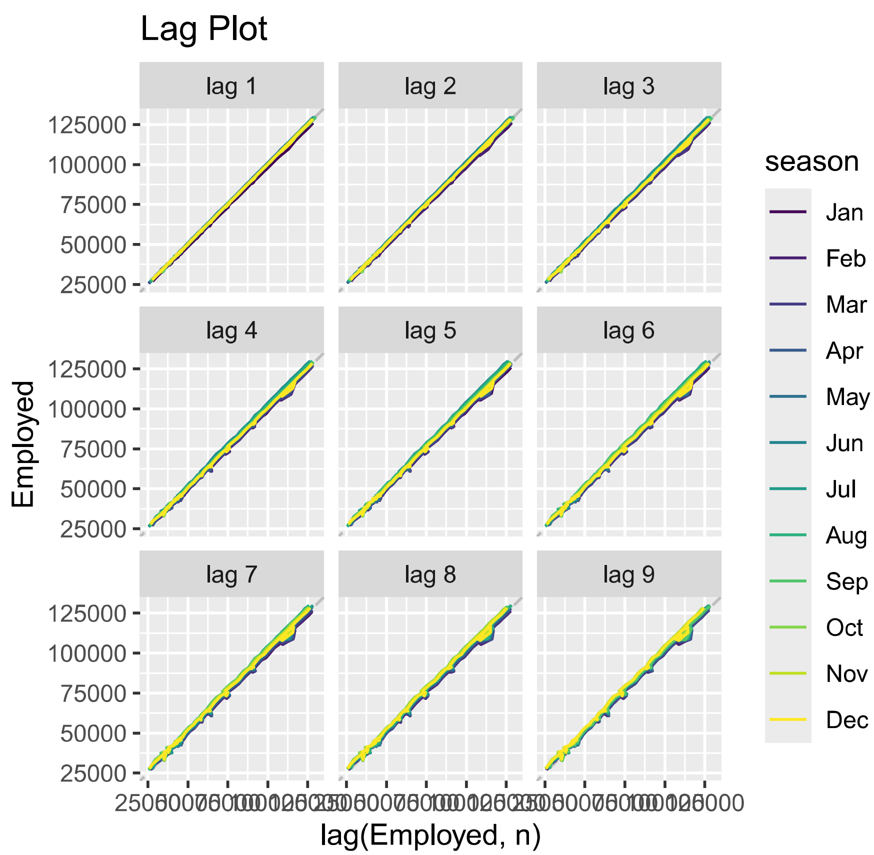

```r
us_employment %>%filter(Title == "Total Private") %>% gg_season(Employed) +
  ggtitle("Seasonal Decomposition")
```



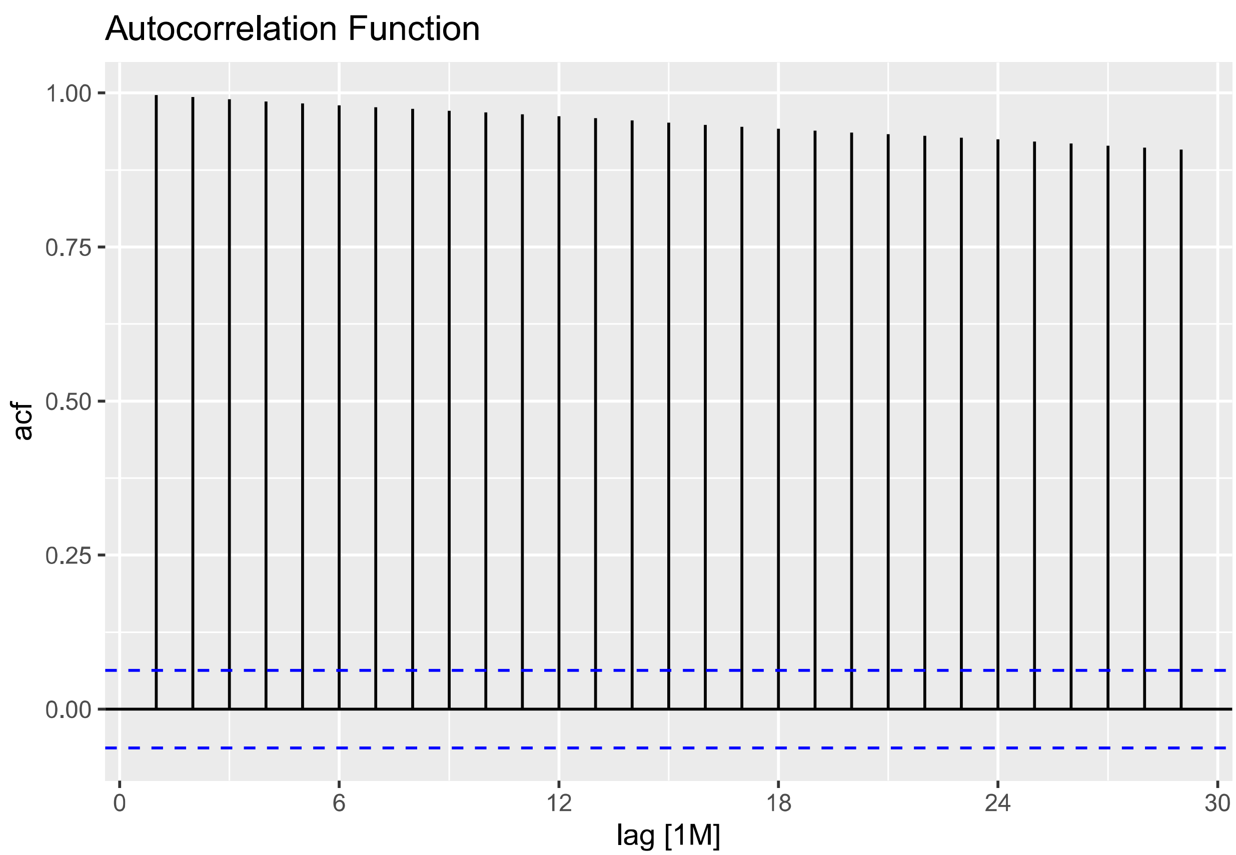

```
us_employment %>%
  filter(Title == "Total Private") %>%
  gg_subseries(Employed) +
  ggtitle("Subseries Plot")
```




```
us_employment %>%
  filter(Title == "Total Private") %>%
  gg_lag(Employed) +
  ggtitle("Lag Plot")
```

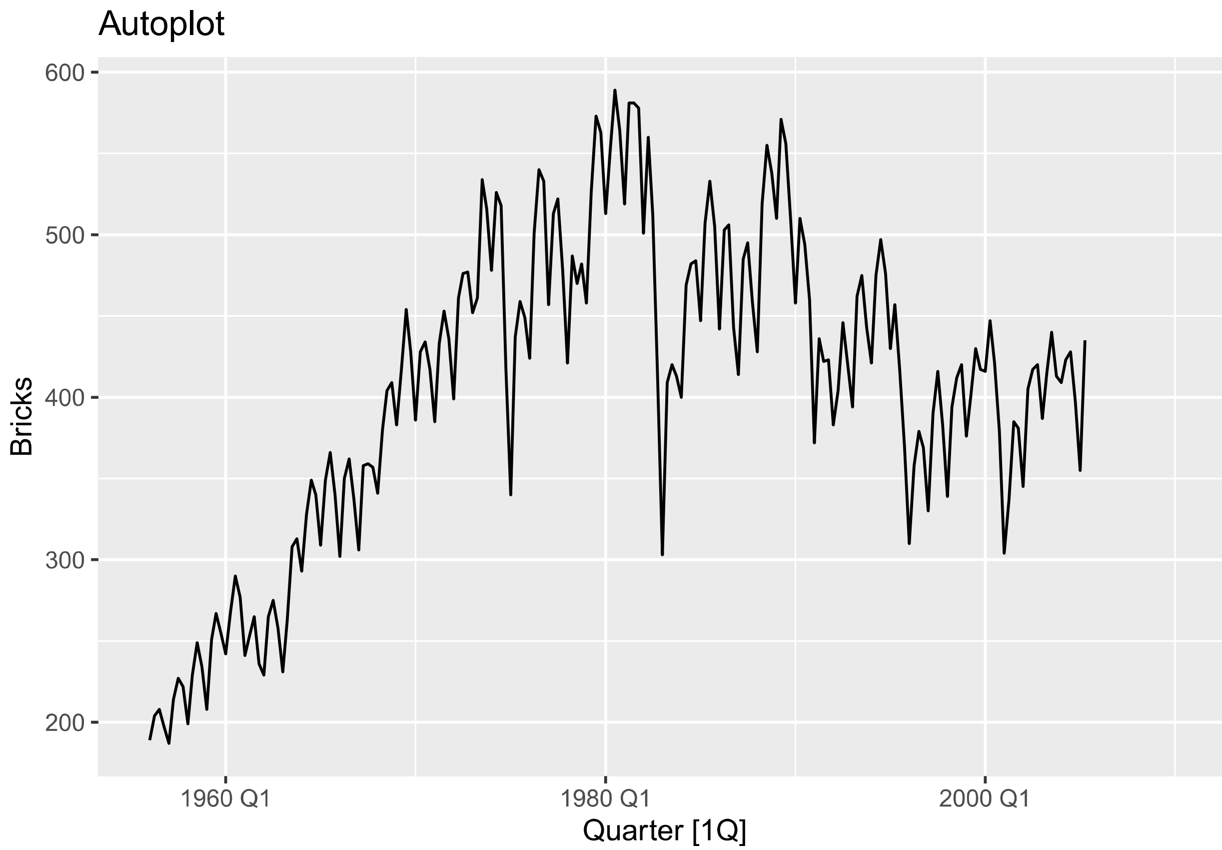



```r
us_employment %>%
  filter(Title == "Total Private") %>%
  ACF(Employed) %>%
  autoplot() +
  ggtitle("Autocorrelation Function")
```

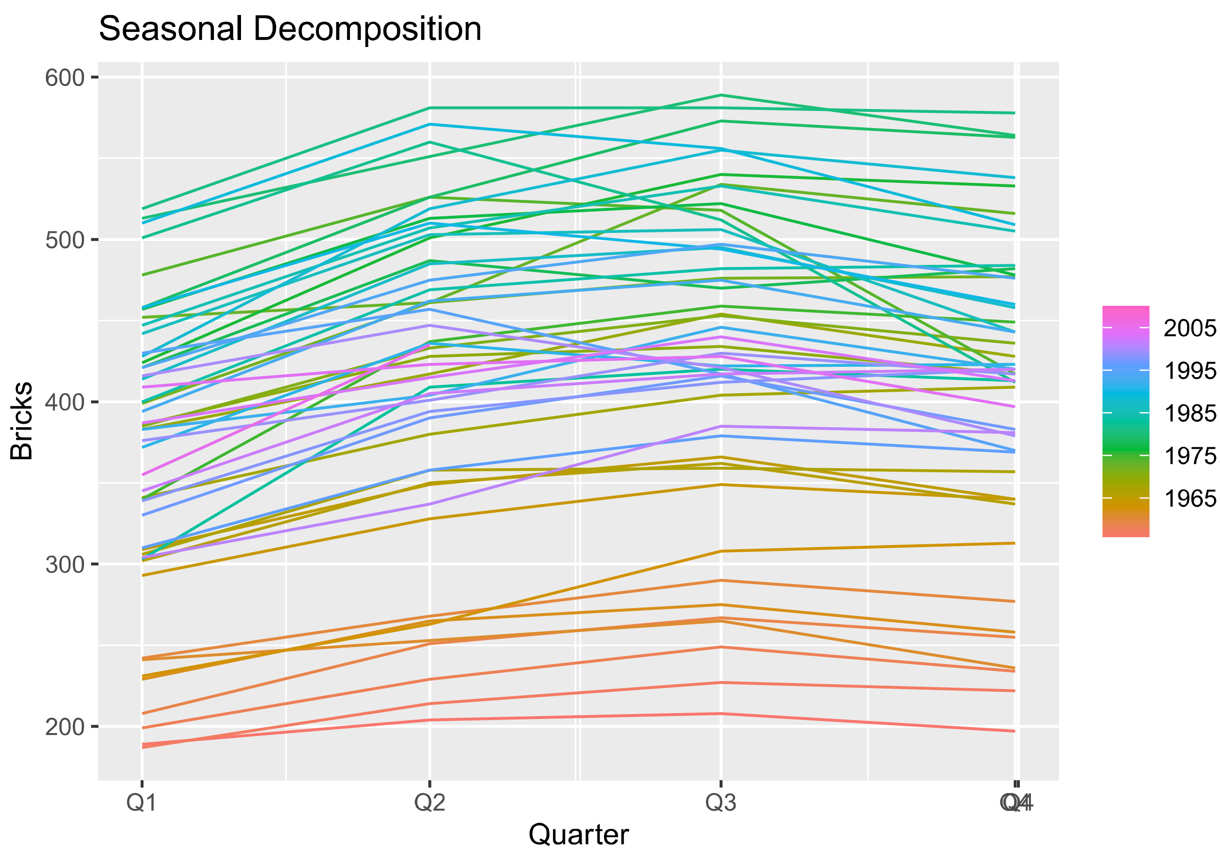


The US Employment dataset shows a general upward trend in Total Private employment over the years, with a notable dip around 2008 that aligns with the housing bubble crash. The data exhibits a seasonal pattern, with employment increasing in the first half of the year, decreasing afterward, and then rising again. The lag plot indicates a strong positive correlation at all lags. For a clearer seasonal decomposition, adjusting the employment numbers by the factor of population growth could be beneficial.

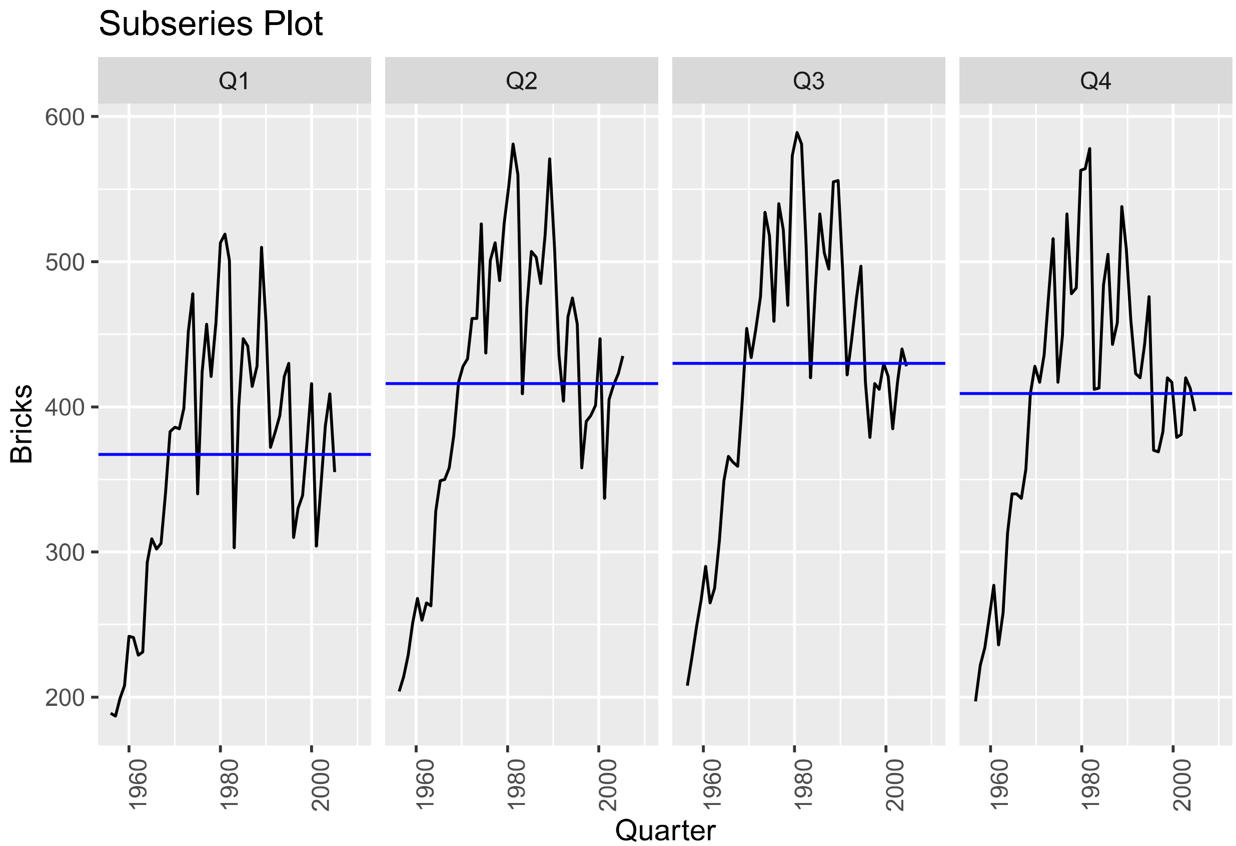
```
aus_production %>%
  autoplot(Bricks) +
  ggtitle("Autoplot")
```



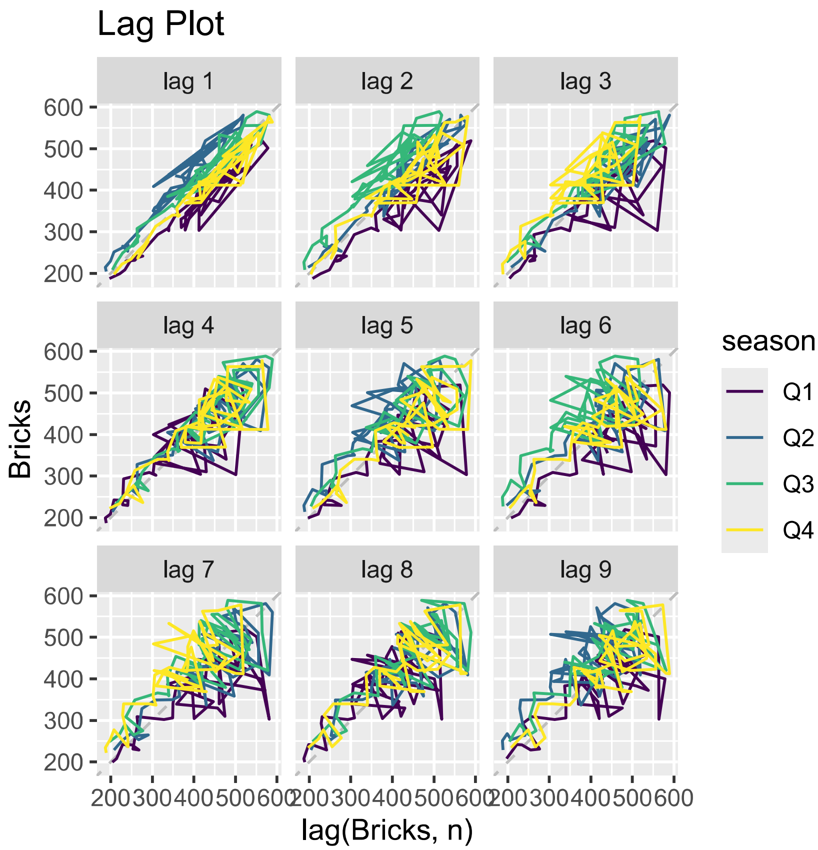

```
aus_production %>%
  gg_season(Bricks) +
  ggtitle("Seasonal Decomposition")
```



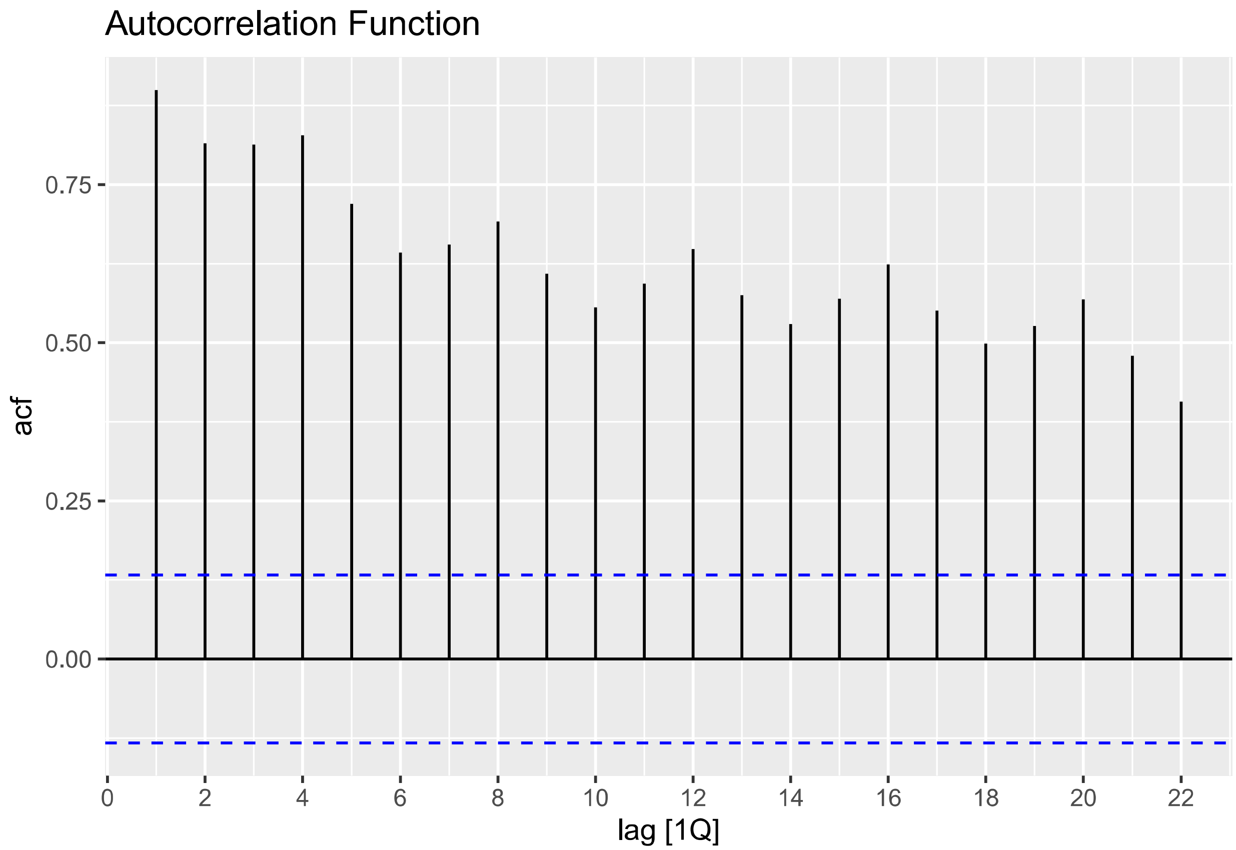

```r
aus_production %>%
  gg_subseries(Bricks) +
  ggtitle("Subseries Plot")
```




```r
aus_production %>%
  gg_lag(Bricks) +
  ggtitle("Lag Plot")
```

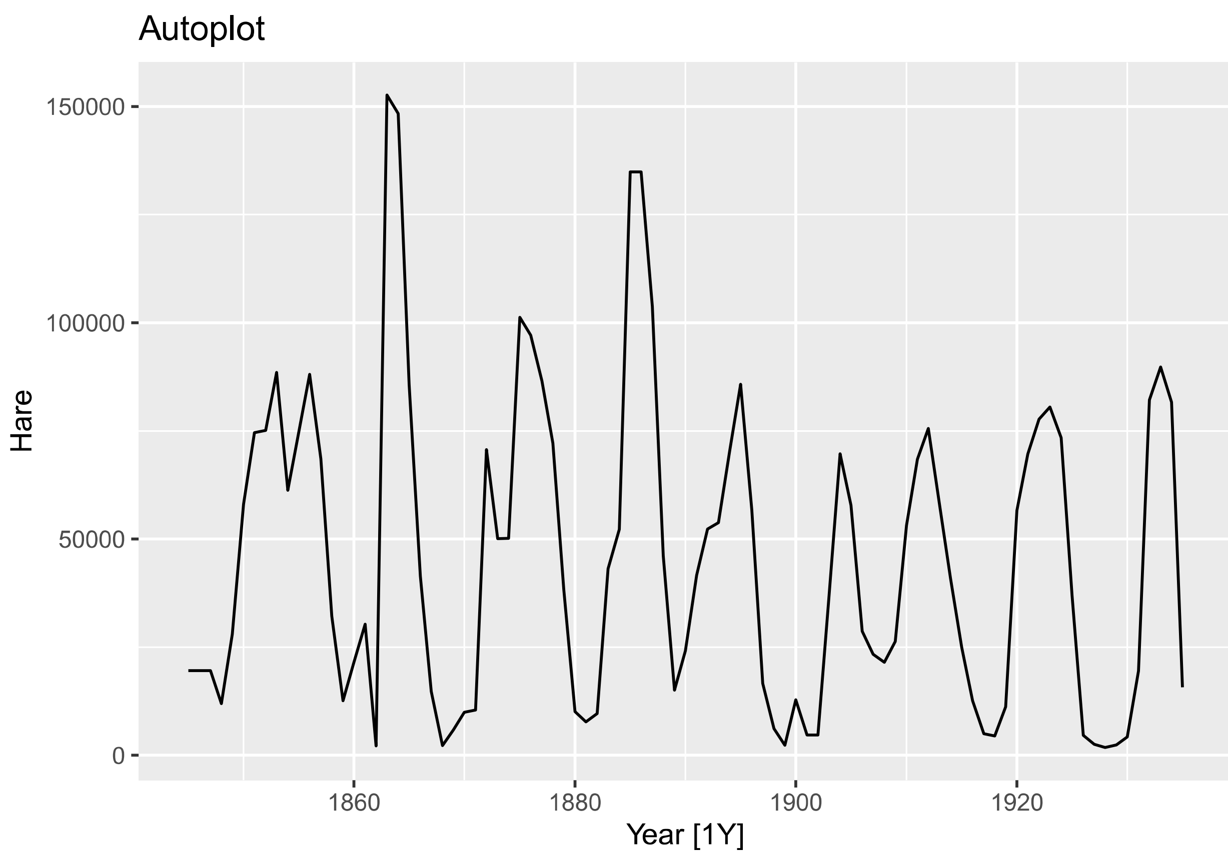



```
aus_production %>%
  ACF(Bricks) %>%
  autoplot() +
  ggtitle("Autocorrelation Function")
```

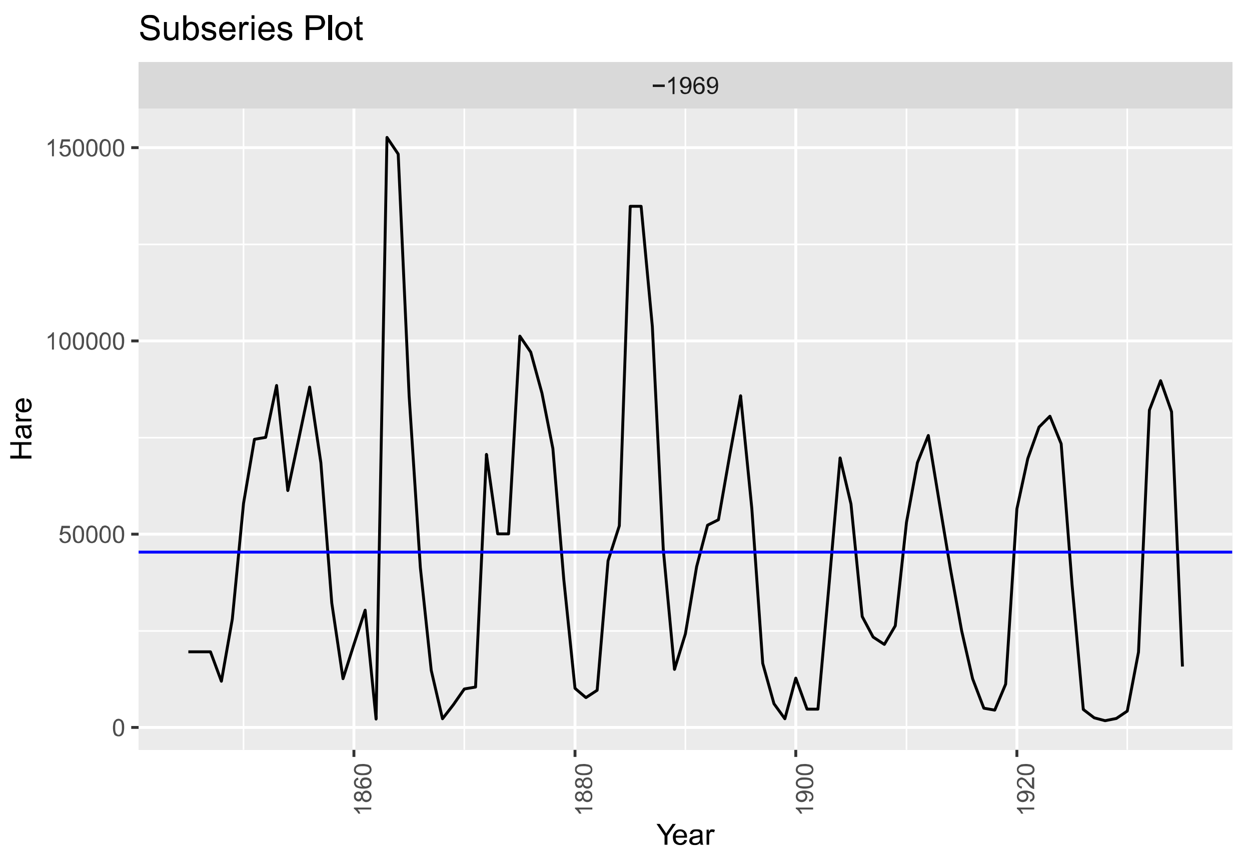


For the AUS Production dataset, brick production lacks a clear trend but displays strong annual seasonality with a cyclical pattern. Production notably dropped in the early 1980s. The seasonal plot shows increases in Q1 and Q3, with a decline in Q4. The lag plot reveals consistent positive season-to-season correlation.

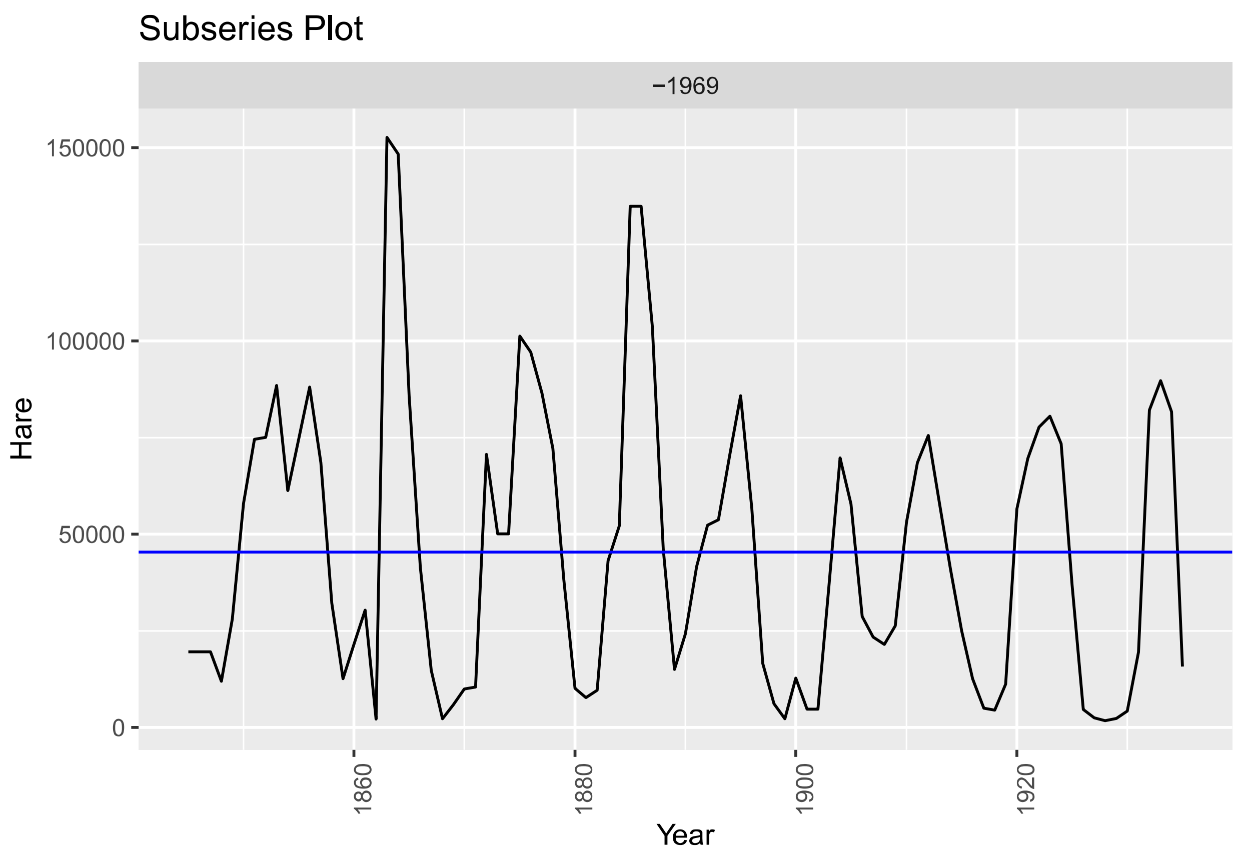
```r
pelt %>%
  autoplot(Hare) +
  ggtitle("Autoplot")
```



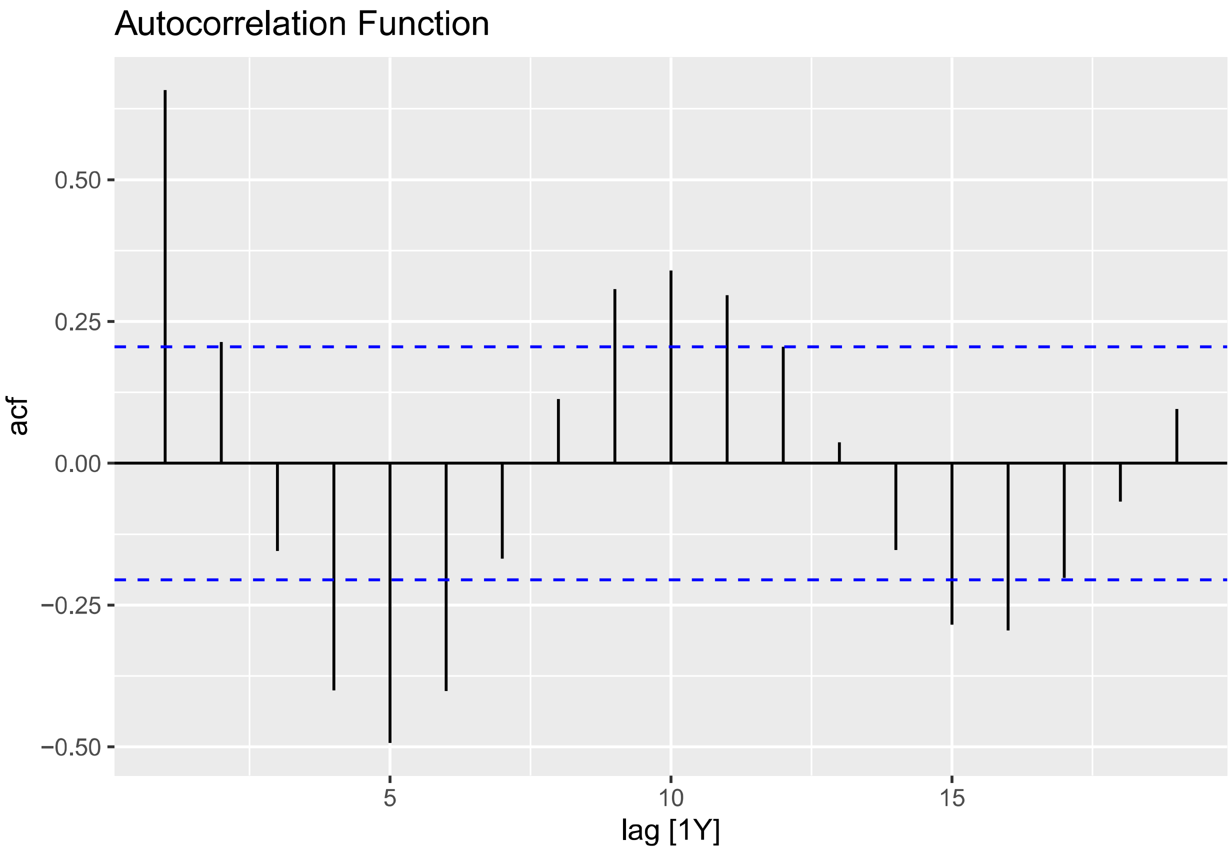

```
pelt %>%
  gg_subseries(Hare)+
  ggtitle("Subseries Plot")
```




```r
pelt %>%
  gg_subseries(Hare)+
  ggtitle("Subseries Plot")
```

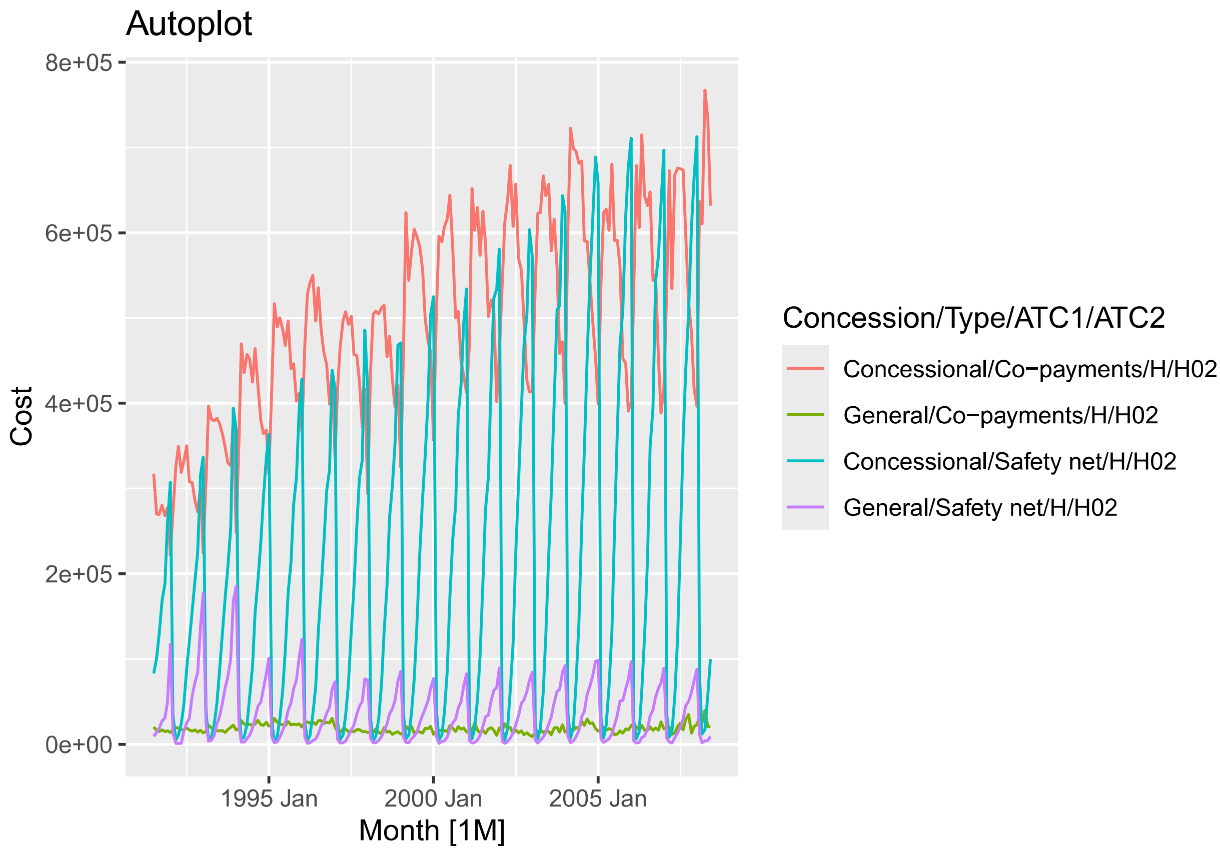



```
pelt %>%
  ACF(Hare) %>%
  autoplot() +
  ggtitle("Autocorrelation Function")
```



For the Pelts data set for Hare, I notice that there isn't a distinct trend, but it is shown a potential seasonal pattern accompanied by some cyclic behavior.There seem to be sharp fluctuations in the number of traded Hare pelts through a few year periods, with a general decrease as the decade comes to an end. The lag plot illustrates a moderate positive correlation particularly in lag 1.

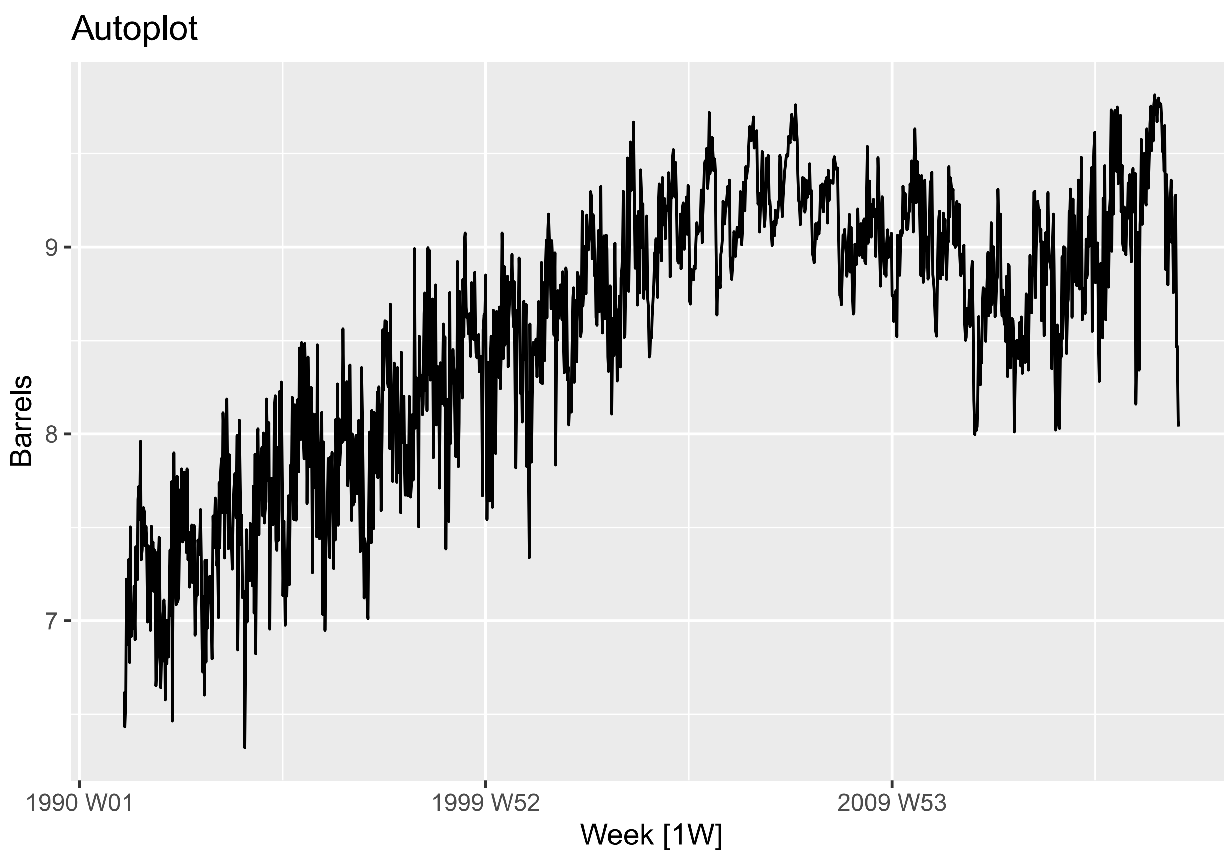
```
PBS %>%
  filter(ATC2 == "H02")  %>%
  autoplot(Cost) +
  ggtitle("Autoplot")
```



The Hare pelts dataset shows no clear trend but hints at seasonality and cyclic behavior. There are sharp fluctuations in traded pelts, with a general decline towards the end of the decade. The lag plot shows moderate positive correlation, especially at lag 1.

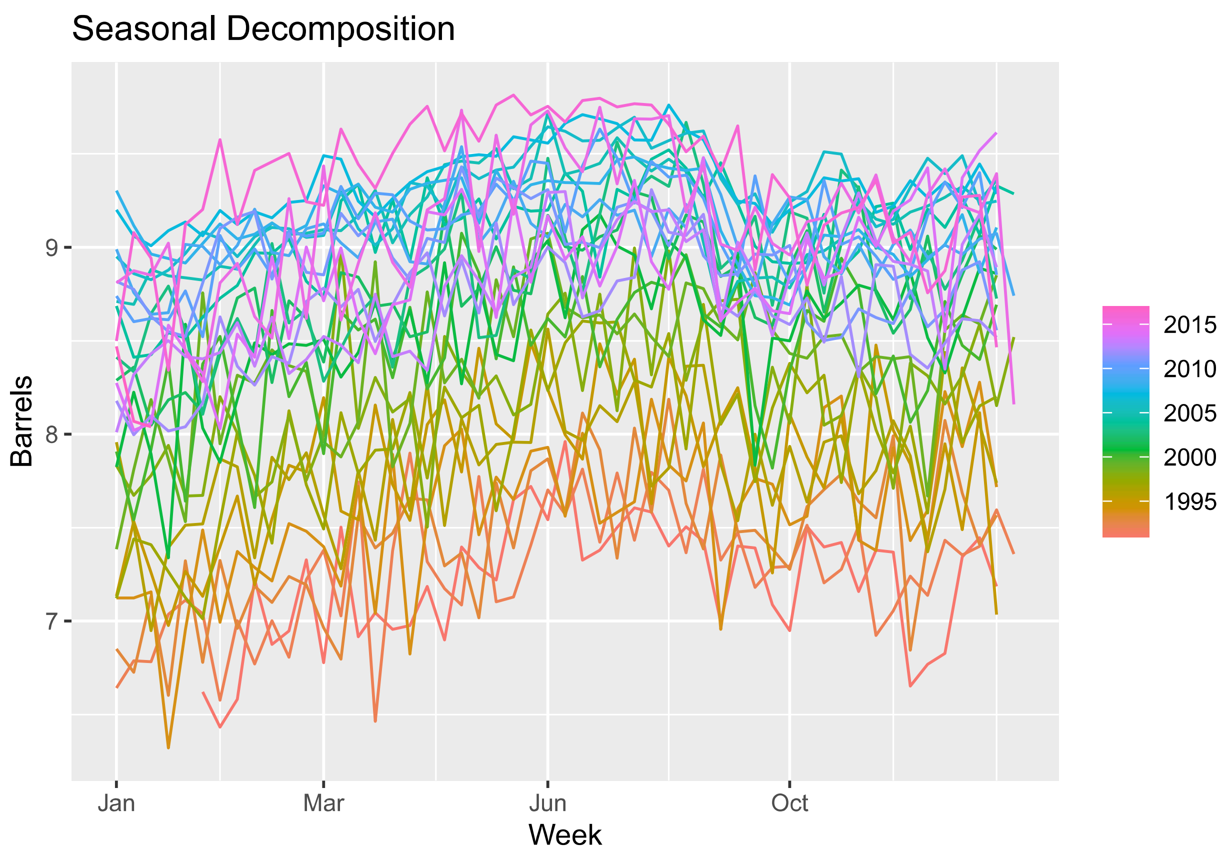
```
us_gasoline %>%
  autoplot() +
  ggtitle("Autoplot")
```

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



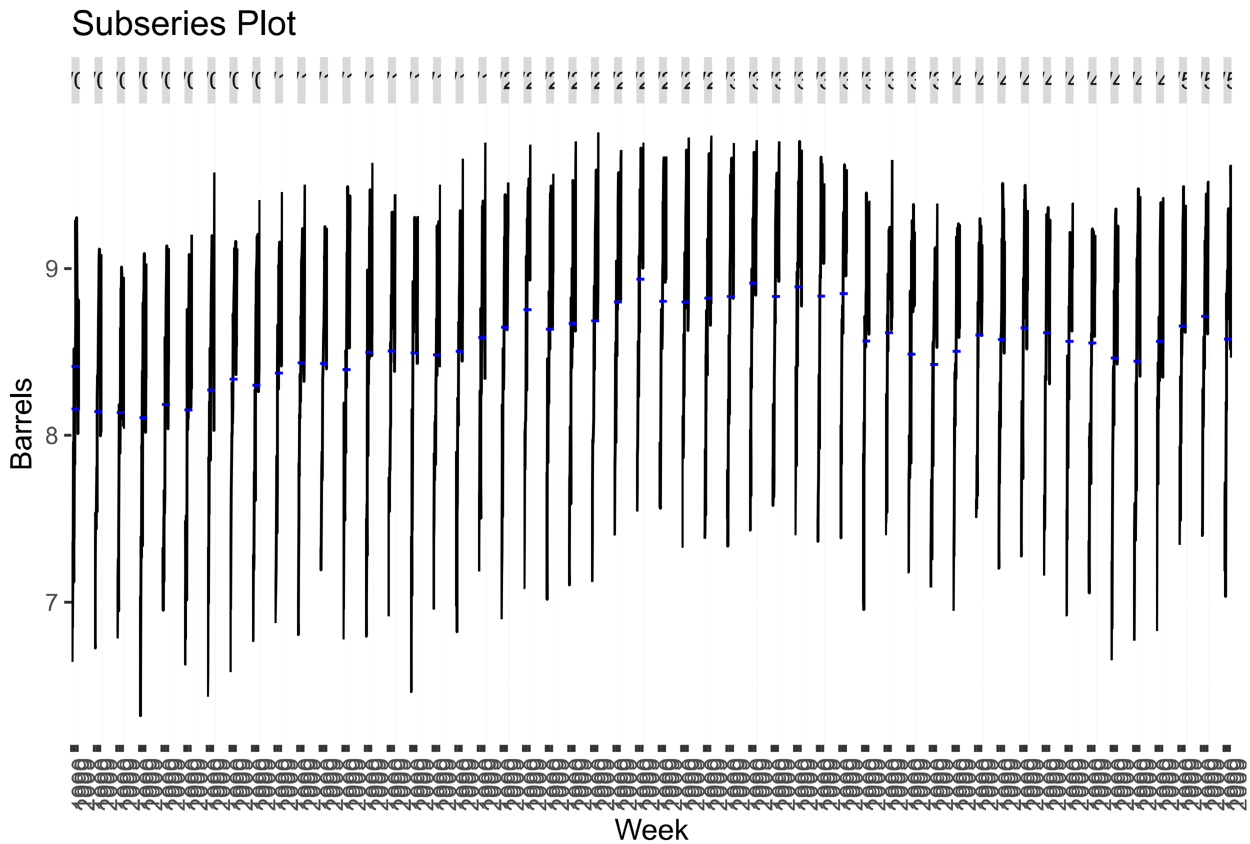

```
us_gasoline %>%
  gg_season() +
  ggtitle("Seasonal Decomposition")
```

```
## Plot variable not specified, automatically selected `y = Barrels`
```



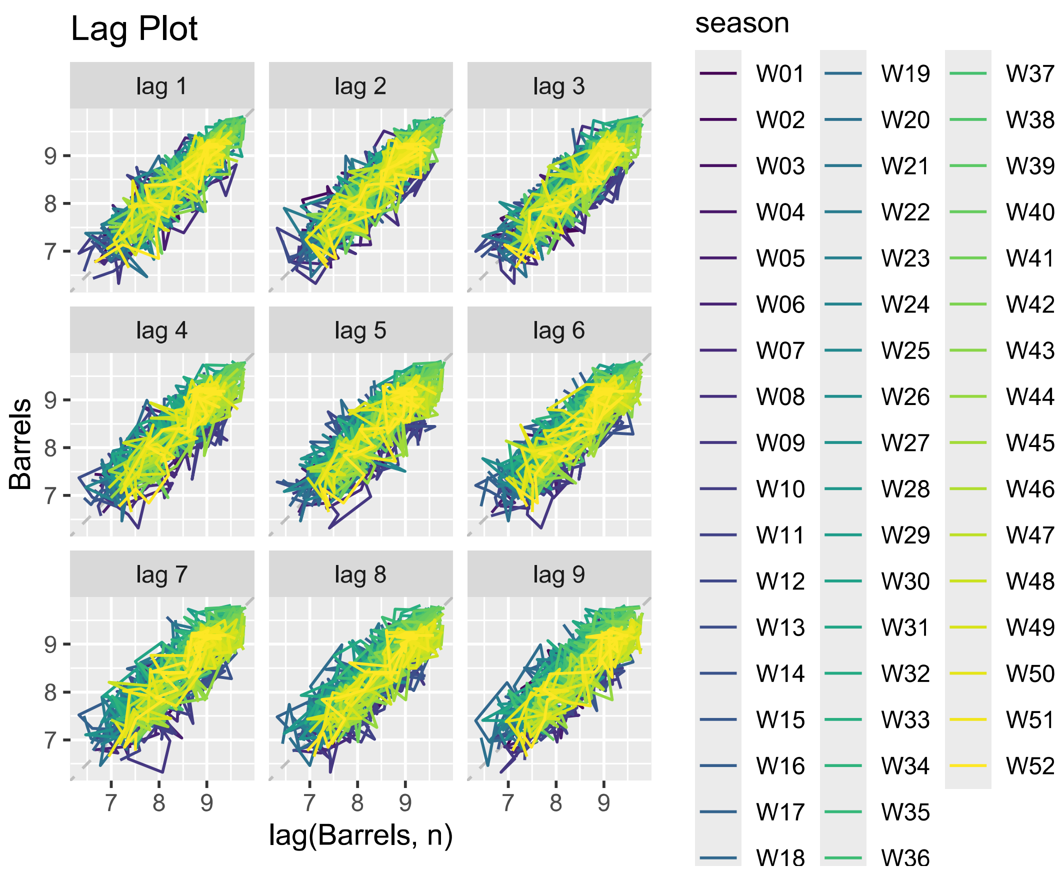

```
us_gasoline %>%
  gg_subseries()+
  ggtitle("Subseries Plot")
```

```
## Plot variable not specified, automatically selected `y = Barrels`
```



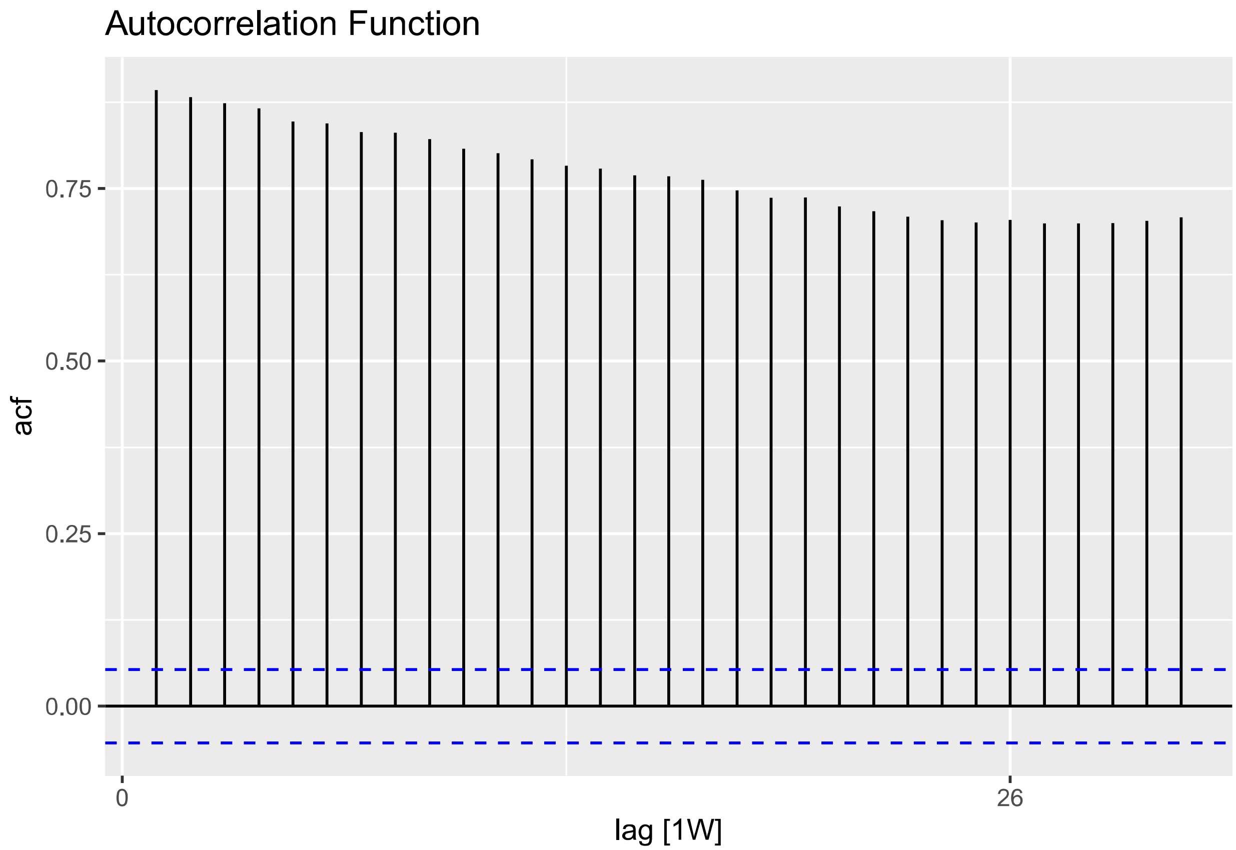

```r
us_gasoline %>%
  gg_lag() +
  ggtitle("Lag Plot")
```

```
## Plot variable not specified, automatically selected `y = Barrels`
```




```
us_gasoline %>%
  ACF() %>%
  autoplot() +
  ggtitle("Autocorrelation Function")
```

```
## Response variable not specified, automatically selected `var = Barrels`
```



The Gasoline Barrels series shows a positive trend with seasonal patterns but is quite noisy, with peaks and declines at specific times of the month. The lag plot reveals a positive correlation with some overplotting. No unusual years are evident, though overplotting may obscure such trends.