FPP: Chapter 2 Exercises

Alex Ptacek

```r
library(fpp3)
library(tidyverse)
library(USgas)
library(readxl)
```

Chapter 2: Exercises

Question 1

1. Bricks from aus_production

Time interval is 1 Quarter

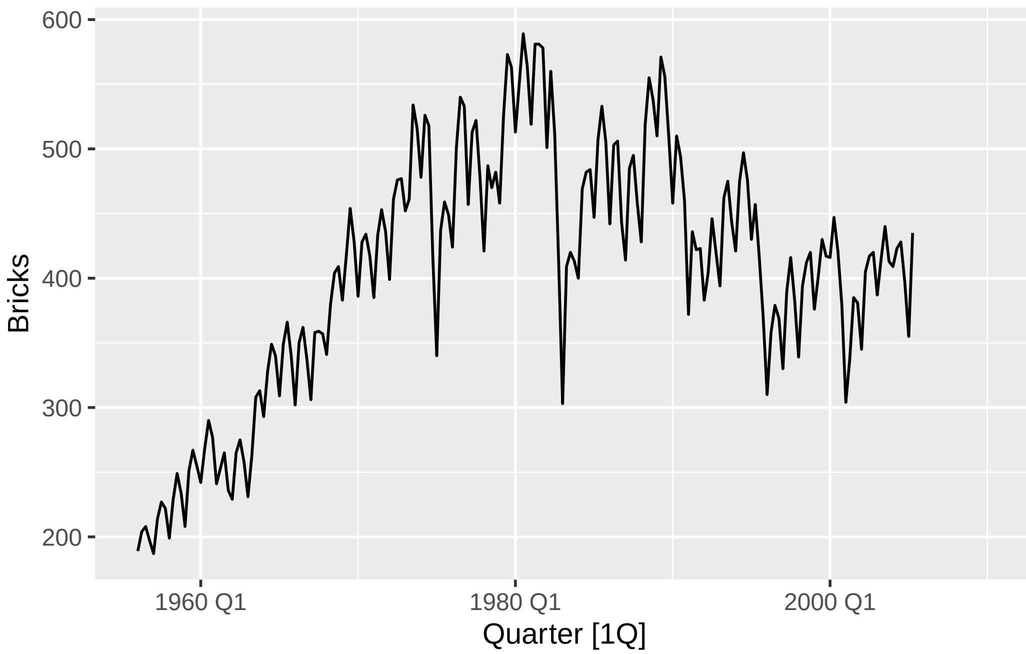
```r
#Using '?' gets meta-data from data that's within packages
?aus_production

#Calling the dataset allows us to observe features such as the time interval
#Limiting print to 5, to keep report more consise
aus_production |>
  print(n = 5)
```

```
# A tsibble: 218 x 7 [1Q]
  Quarter  Beer Tobacco Bricks Cement Electricity   Gas
    <qtr> <dbl>   <dbl>  <dbl>  <dbl>       <dbl> <dbl>
1 1956 Q1   284    5225    189    465        3923     5
2 1956 Q2   213    5178    204    532        4436     6
3 1956 Q3   227    5297    208    561        4806     7
4 1956 Q4   308    5681    197    570        4418     6
5 1957 Q1   262    5577    187    529        4339     5
# i 213 more rows
```

```r
#autoplot creates a time series when we input a tsibble
aus_production |>
  autoplot(Bricks)
```

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



2. Lynx from `pelt`

Time interval is 1 Year

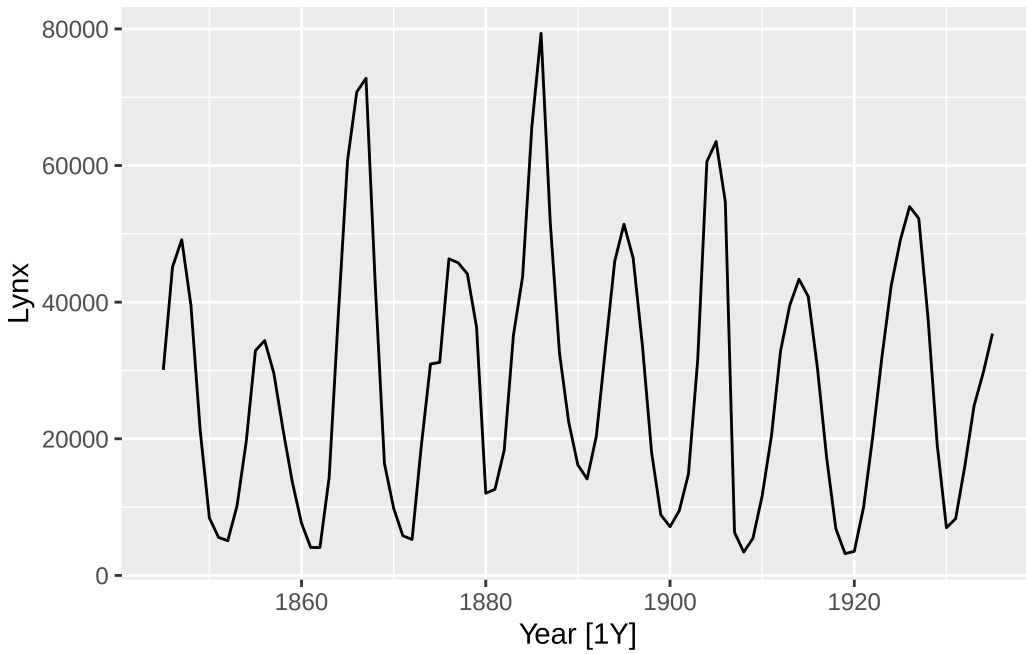
```r
?pelt

pelt |>
  print(n = 5)
```

```
# A tsibble: 91 x 3 [1Y]
   Year  Hare  Lynx
  <dbl> <dbl> <dbl>
1  1845 19580 30090
2  1846 19600 45150
```

```
3  1847 19610 49150
4  1848 11990 39520
5  1849 28040 21230
# i 86 more rows
```

```r
pelt |>
  autoplot(Lynx)
```



3. `Close` from `gafa_stock`

Time interval varies/undetermined

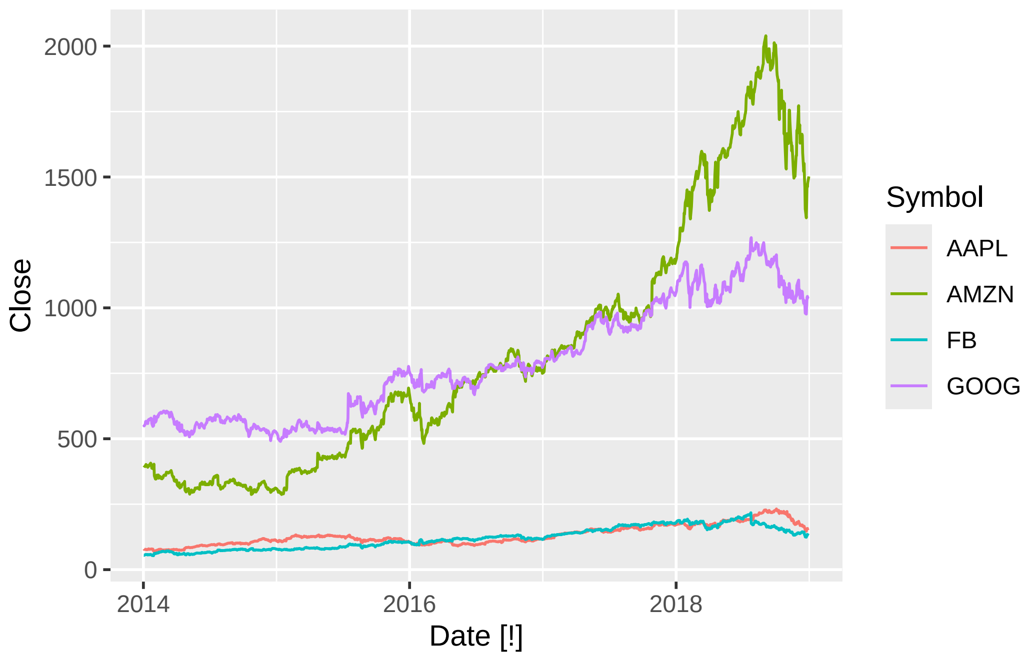
```r
?gafa_stock

gafa_stock |>
  print(n = 5)
```

```
# A tsibble: 5,032 x 8 [!]
# Key:       Symbol [4]
  Symbol Date        Open  High   Low Close Adj_Close    Volume
  <chr>  <date>     <dbl> <dbl> <dbl> <dbl>     <dbl>     <dbl>
1 AAPL   2014-01-02  79.4  79.6  78.9  79.0      67.0  58671200
```

```
2 AAPL   2014-01-03  79.0  79.1  77.2  77.3      65.5  98116900
3 AAPL   2014-01-06  76.8  78.1  76.2  77.7      65.9 103152700
4 AAPL   2014-01-07  77.8  78.0  76.8  77.1      65.4  79302300
5 AAPL   2014-01-08  77.0  77.9  77.0  77.6      65.8  64632400
# i 5,027 more rows
```

```
gafa_stock |>
  autoplot(Close)
```



4. `Demand` from `vic_elec`

Time interval is 30 minutes

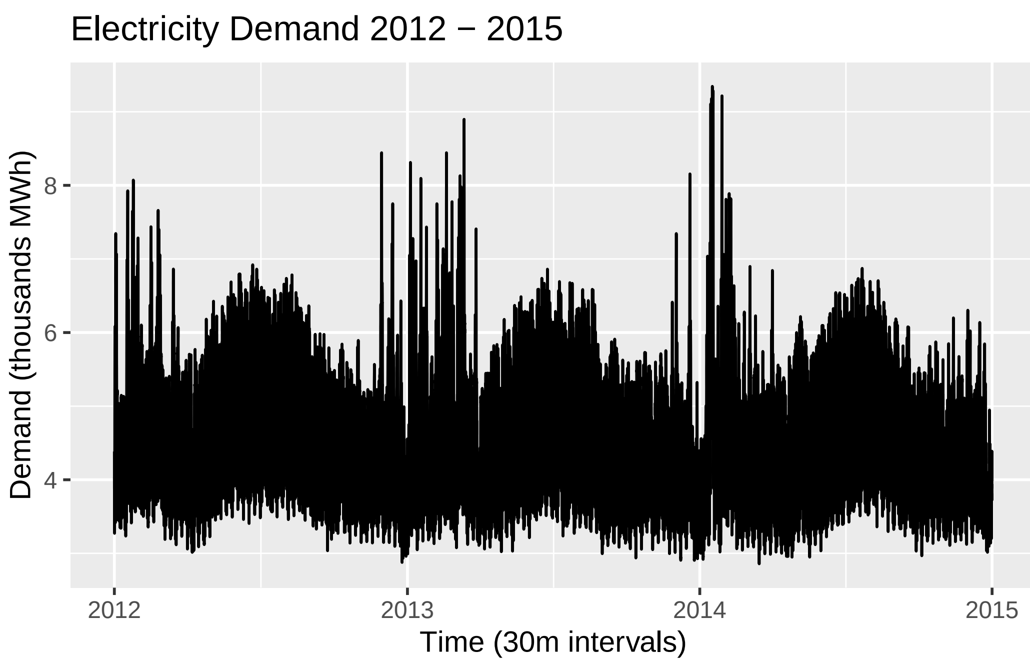
```
?vic_elec

vic_elec |>
  print(n = 5)
```

```
# 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
# i 52,603 more rows
```

```
#Use labs functions to add labels to ggplots
vic_elec |>
  mutate(Demand = Demand / 1e3) |>
  autoplot(Demand) +
  labs(title = "Electricity Demand 2012 - 2015") +
  xlab("Time (30m intervals)") +
  ylab("Demand (thousands MWh)")
```



Question 2

```
#max() function finds the max value of numeric variable
#and .by groups by selected variable
gafa_stock |>
  filter(Close == max(Close), .by = Symbol)
```

```
# A tsibble: 4 x 8 [!]
# Key:       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
```

Question 3

a.

```r
tute1 <- read_csv("~/Downloads/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.
```

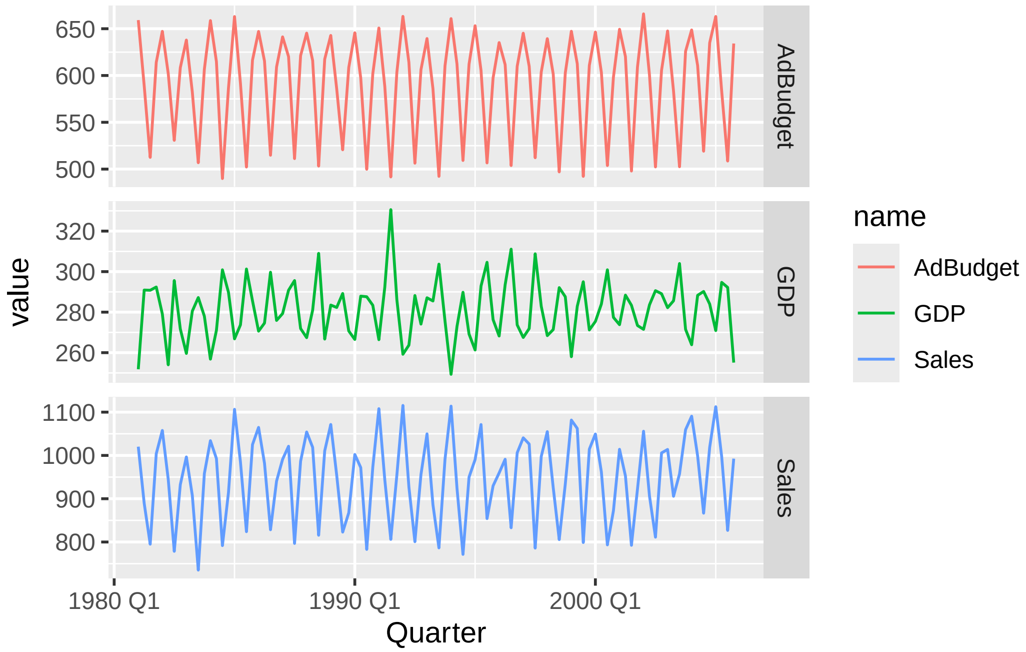
```r
view(tute1)
```

b.

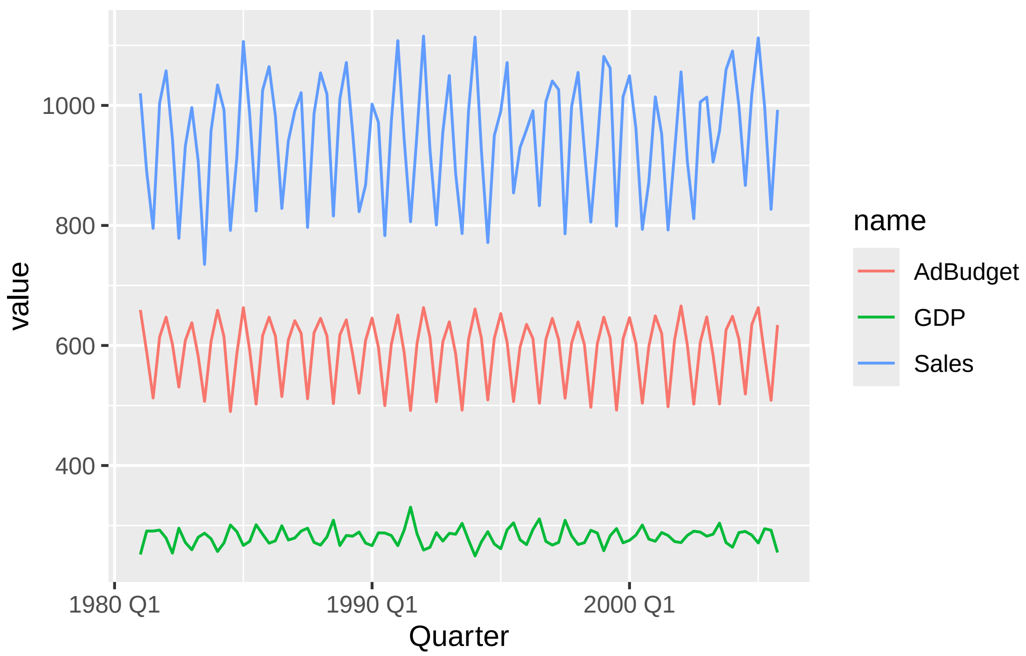
```r
mytimeseries <- tute1 |>
  mutate(Quarter = yearquarter(Quarter)) |>
  as_tsibble(index = Quarter)
```

c. Removing the facet_grid function leaves us with one plot with incorrect y-axis labels

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




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



Question 4

a.

```r
library(USgas)
```

b.

```r
us_total_timeseries <- tsibble(us_total, key = state, index = year)
```

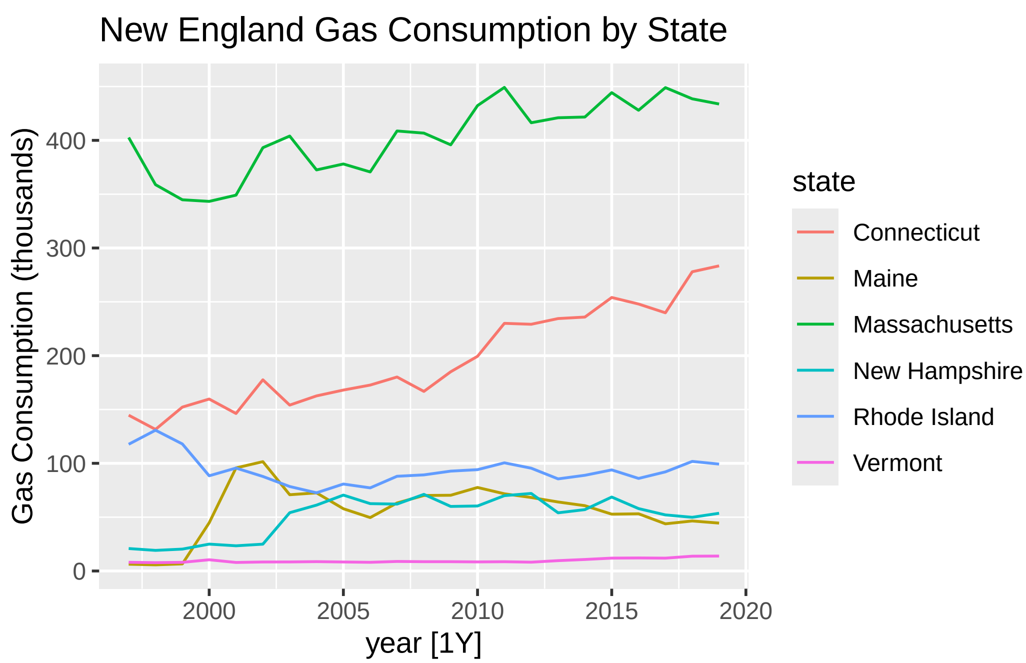
c.

```r
#Observe the format for stat names
us_total_timeseries |>
  count(state)
```

```
# A tibble: 53 x 2
   state                                        n
   <chr>                                    <int>
 1 Alabama                                     23
 2 Alaska                                      23
 3 Arizona                                     23
```

```
 4 Arkansas                              23
 5 California                            23
 6 Colorado                              23
 7 Connecticut                           23
 8 Delaware                              23
 9 District of Columbia                  23
10 Federal Offshore -- Gulf of Mexico    21
# i 43 more rows
```

```
#Use grepl in filter to shorthand type state names
us_total_timeseries |>
  filter(grepl("main|verm|new hamp|massac|connec|rhode",
               state, ignore.case = TRUE)) |>
  mutate(y = y / 1e3) |>
  autoplot(y) +
  labs(title = "New England Gas Consumption by State") +
  ylab("Gas Consumption (thousands)")
```



Question 5

a.

```r
textbook_tourism <- read_excel("~/Downloads/tourism.xlsx")
```

b.

```r
#Observe time interval and key
tourism |>
  print(n = 5)
```

```
# 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.
# i 24,315 more rows
```

```r
#Convert downloaded dataset to a tsibble with same index and key as tourism
#df from tsibble package
textbook_tourism <- textbook_tourism |>
  mutate(Quarter = yearquarter(Quarter)) |>
  tsibble(key = c(Region, State, Purpose), index = Quarter)

#Checks that our datasets are nearly identical
all.equal(tourism, textbook_tourism)
```

```
[1] TRUE
```

c.

```r
#Find the avg trips by region and purpose and filter to the highest value
tourism |>
  group_by(Region, Purpose) |>
  summarise(avg_trips = mean(Trips)) |>
  arrange(desc(avg_trips)) |>
  head(1)
```

```
# A tsibble: 1 x 4 [1Q]
# Key:       Region, Purpose [1]
# Groups:    Region [1]
  Region    Purpose  Quarter avg_trips
  <chr>     <chr>      <qtr>     <dbl>
1 Melbourne Visiting 2017 Q4      985.
```

d.

```
#Concatenate Region and Purpose
#Calculate trips
trips_by_state <- tourism |>
  mutate(region_purpose = str_c(Region, Purpose, sep = "_")) |>
  group_by(State, region_purpose) |>
  summarise(trips = sum(Trips)) |>
  relocate(Quarter, .before = 1)

trips_by_state |>
  print(n = 5)
```

```
# A tsibble: 24,320 x 4 [1Q]
# Key:       State, region_purpose [304]
# Groups:    State [8]
  Quarter State region_purpose    trips
    <qtr> <chr> <chr>             <dbl>
1 1998 Q1 ACT   Canberra_Business 150.
2 1998 Q2 ACT   Canberra_Business  99.9
3 1998 Q3 ACT   Canberra_Business 130.
4 1998 Q4 ACT   Canberra_Business 102.
5 1999 Q1 ACT   Canberra_Business  95.5
# i 24,315 more rows
```

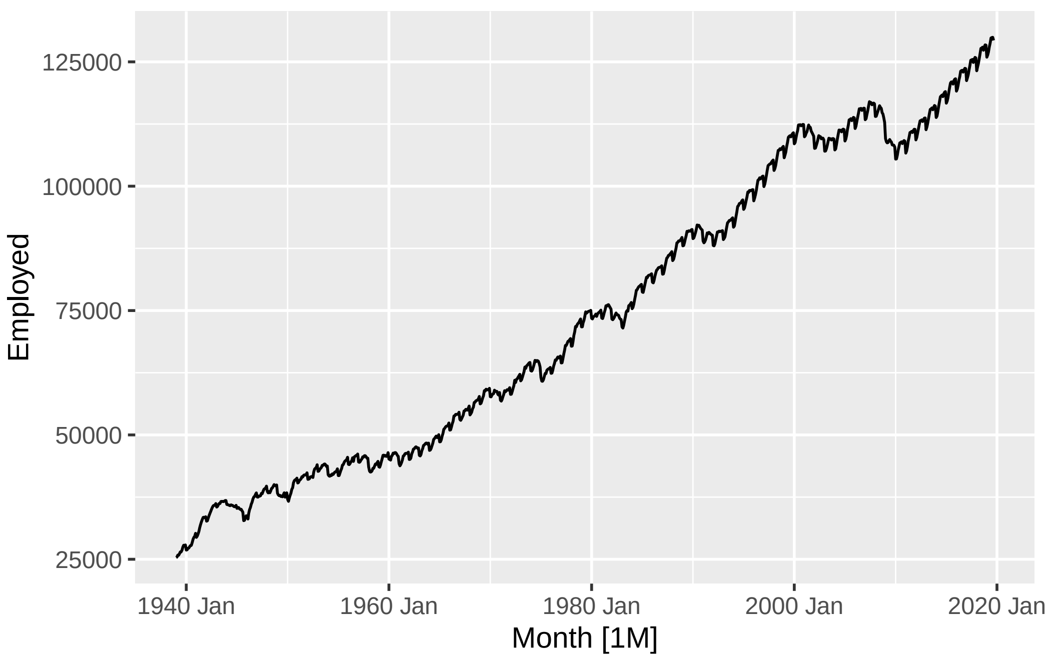
Question 8

1. `Employed` from `us_employment`

a. time plot Based on the time plot, we can see a clear positive trend in this time series, as well as seasonality. There also appears to be a cycle of steady rising followed by short periods of decline.

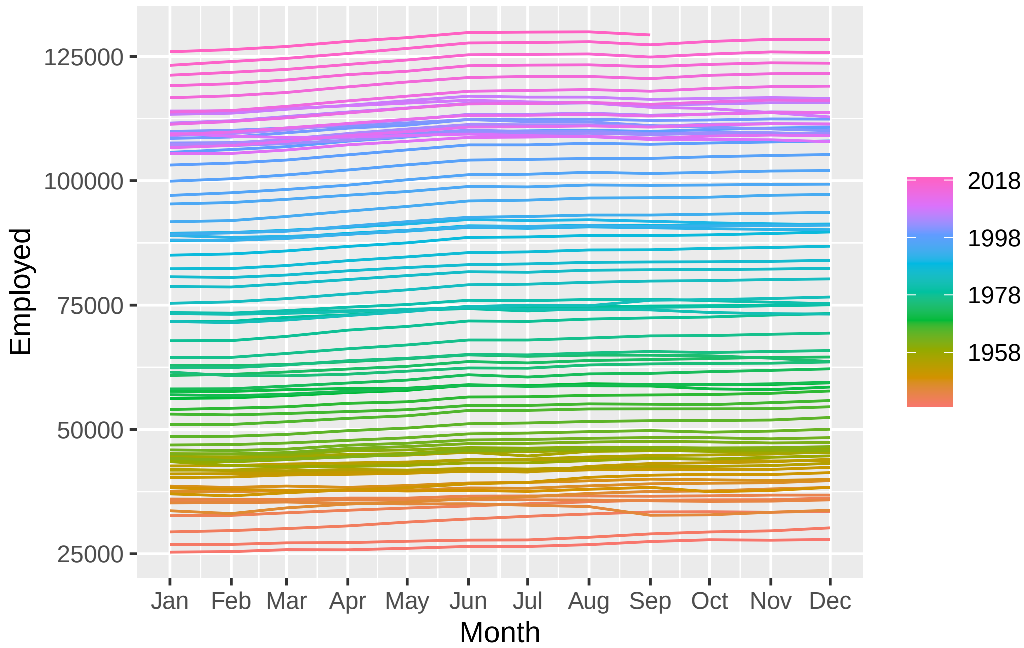
```r
total_private <- us_employment |>
  filter(Title == "Total Private")

total_private |>
  autoplot(Employed)
```



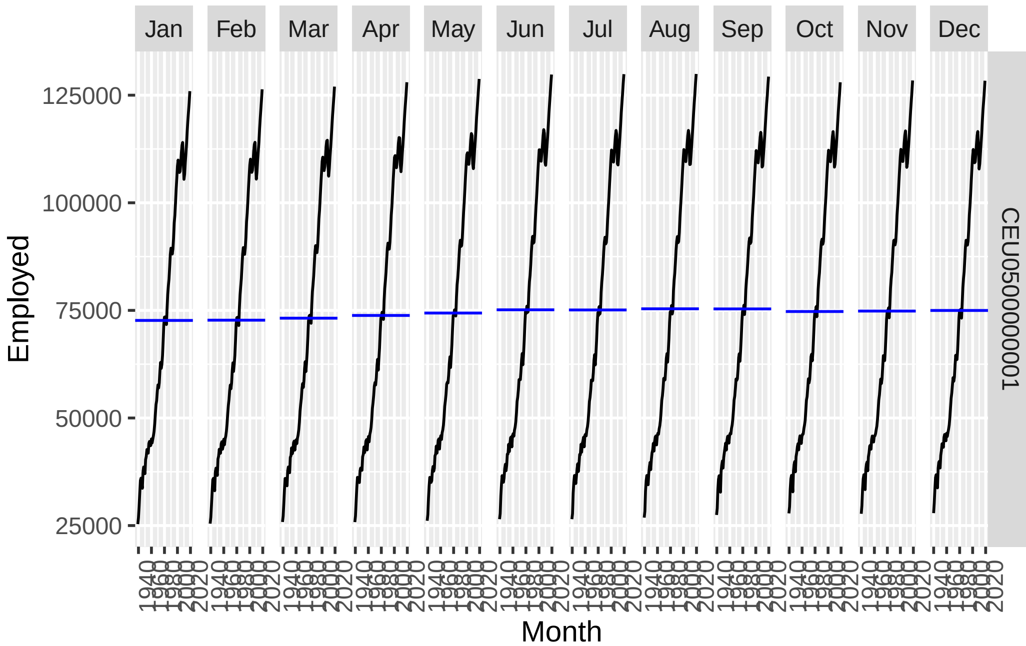
b. seasonal plot Based on the seasonal plot, we can see that there is a positive trend in this time series, because the years are descending downwards in the plot. It's hard to see a clear seasonal trend, because the chart is quite busy, and also there may not be any.

```r
total_private |> gg_season(Employed)
```



c. seasonal subseries plot Based on the seasonal subseries plot, we a consistent positive trend in all months. The averages are fairly similar across all months, furthering our suspicion that there isn't actually seasonality.

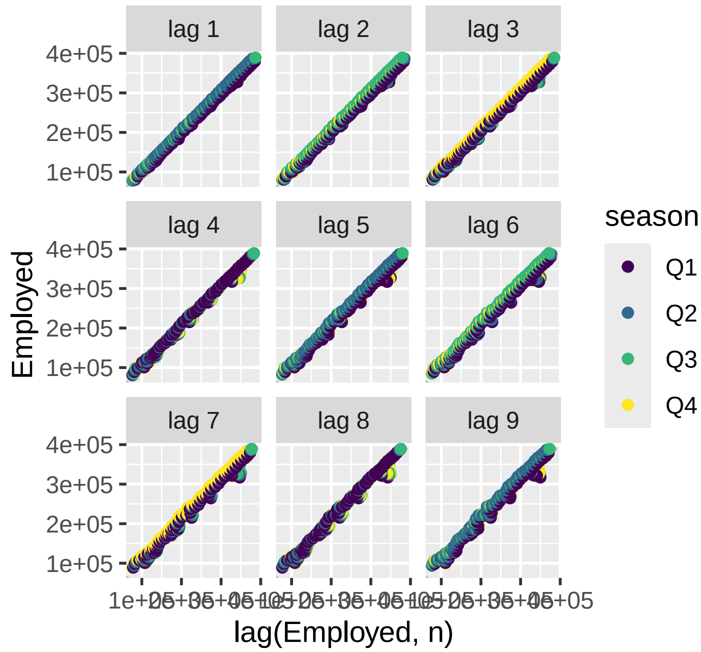
```
total_private |> gg_subseries(Employed)
```



d. lag plot Every lag plot is nearly perfectly linear, further proving that there is no seasonality. I made the choice to transform the tsibble, changing the index from 1M to 1Q, so that I could see seasonal multiples in a 9 panel grid.

```
quarterly_employment <- total_private |>
  mutate(Quarter = yearquarter(Month)) |>
  as_tibble() |>
  select(-Month) |>
  group_by(Quarter, Series_ID, Title) |>
  summarise(Employed = sum(Employed)) |>
  ungroup() |>
  as_tsibble(index = Quarter, key = Series_ID)

quarterly_employment |>
  gg_lag(Employed, geom = "point")
```

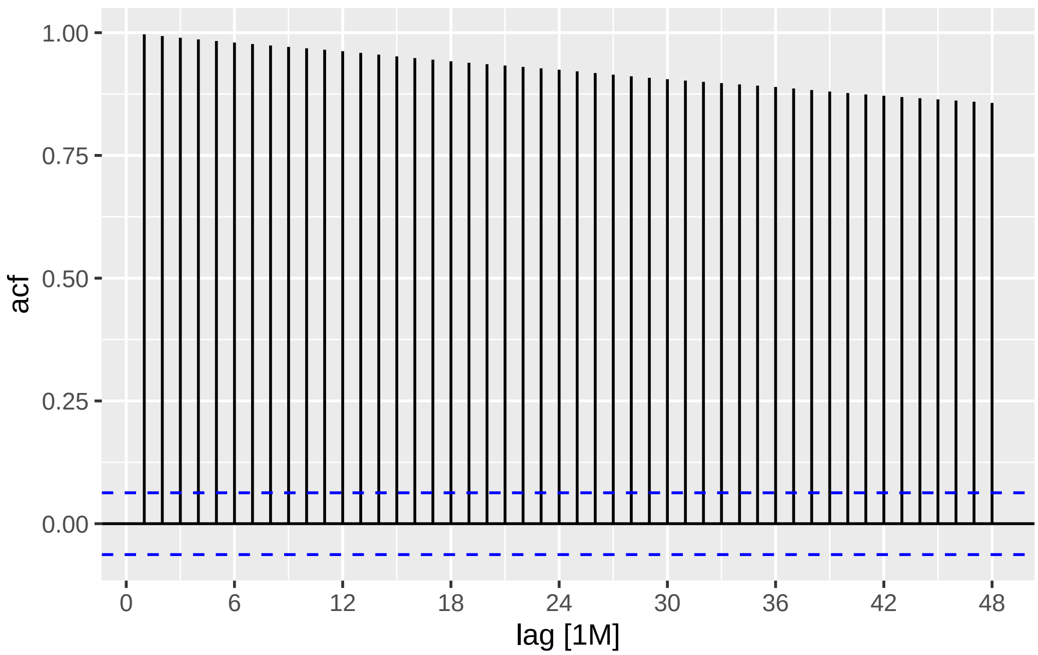


e. ACF and ACF plot Based on the ACF coefficients, we can see that there is a strong positive correlation between all of the lagged values. Once again, it is hard to see any seasonality, but there is a clear trend.

```
total_private |> ACF(Employed, lag_max = 12)
```

```
# A tsibble: 12 x 3 [1M]
# Key:       Series_ID [1]
   Series_ID          lag   acf
   <chr>          <cf_lag> <dbl>
 1 CEU0500000001       1M 0.997
 2 CEU0500000001       2M 0.993
 3 CEU0500000001       3M 0.990
 4 CEU0500000001       4M 0.986
 5 CEU0500000001       5M 0.983
 6 CEU0500000001       6M 0.980
 7 CEU0500000001       7M 0.977
 8 CEU0500000001       8M 0.974
 9 CEU0500000001       9M 0.971
10 CEU0500000001      10M 0.968
11 CEU0500000001      11M 0.965
12 CEU0500000001      12M 0.962
```

```r
total_private |> ACF(Employed, lag_max = 48) |> autoplot()
```

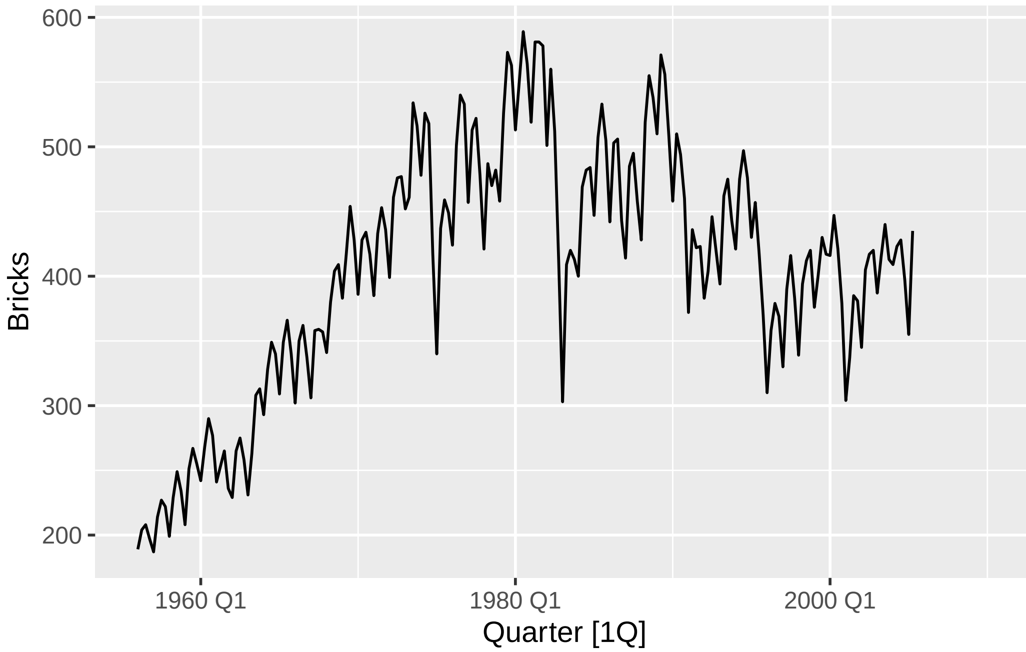


2. `Bricks` from `aus_production`

a. time plot The time plot shows us that this time series has a steep upward trend up to half-way point, then a mild downward trend. There appears to be strong seasonality, as well as cyclic deep depressions. This cycle seems to start prominently in 1975, and reoccurs about every 5-10 years from then on. Notably, in about 1983, production has its biggest fall.

```r
aus_production |> autoplot(Bricks)
```

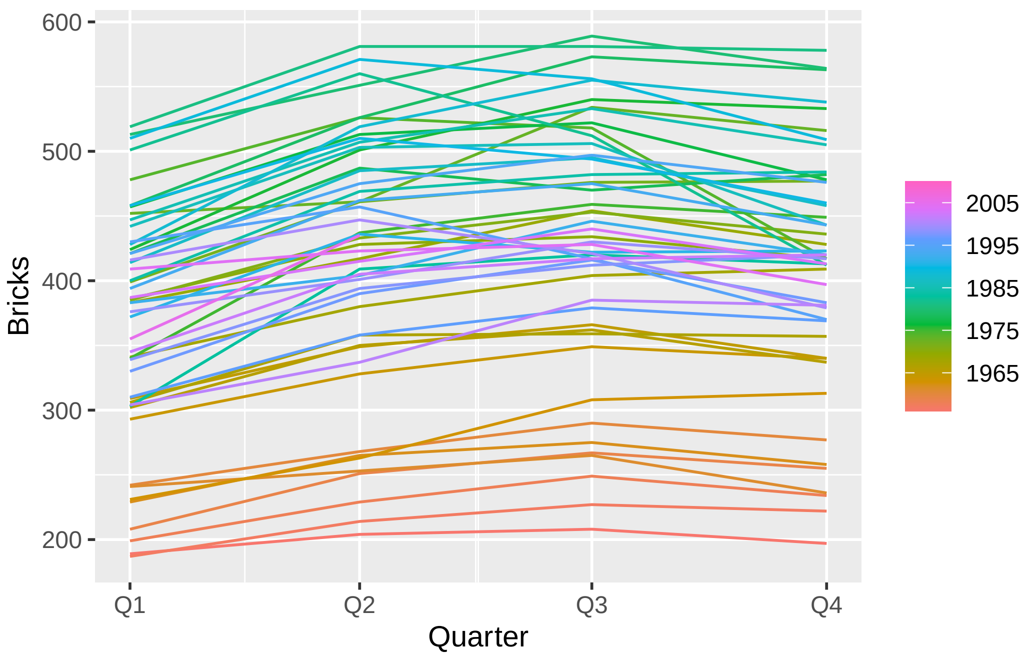
```
Warning: Removed 20 rows containing missing values or values outside the scale range
(`geom_line()`).
```



b. seasonal plot Based on the seasonal plot, production seems to peak in Q2 and Q3, especially in Q3 (except some exceptions). There's also several years where there is a sharp decline in production in Q3 and Q4.

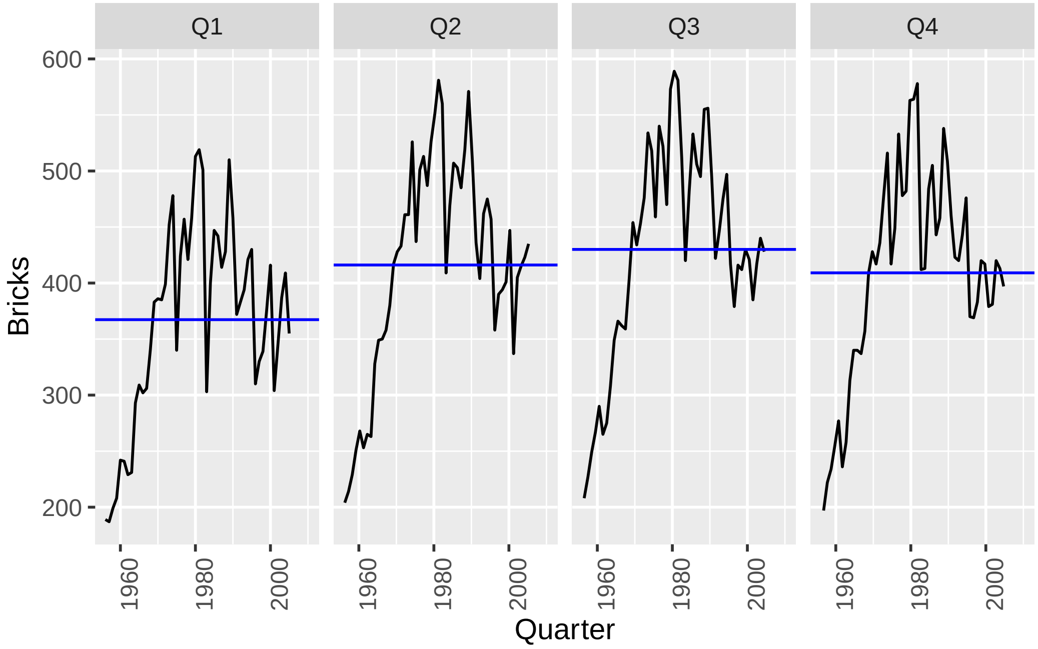
```
aus_production |> gg_season(Bricks)
```

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



c. seasonal subseries plot The subseries plots shows us what the seasonality looks like. Production tends to increase from Q1-Q3 and then decrease from Q3-Q1.

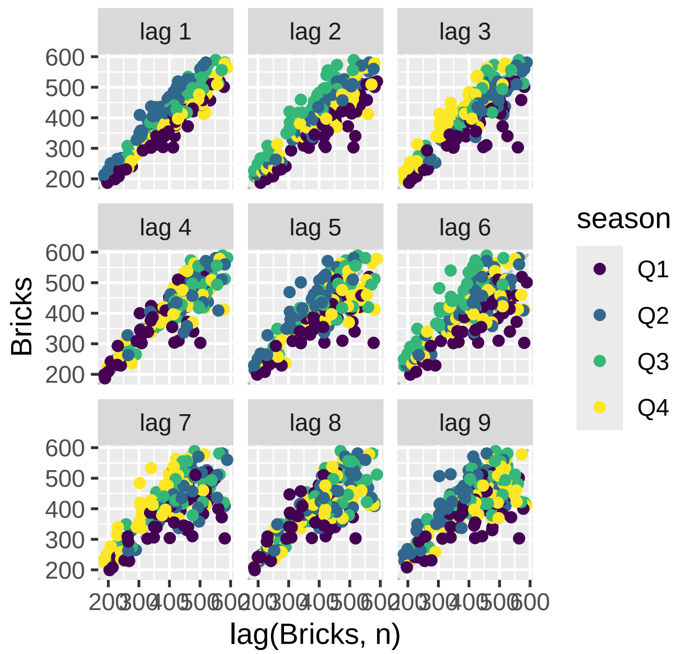
```
aus_production |> gg_subseries(Bricks)
```



d. lag plot

```
aus_production |> gg_lag(Bricks, geom = "point")
```

```
Warning: Removed 20 rows containing missing values (gg_lag).
```



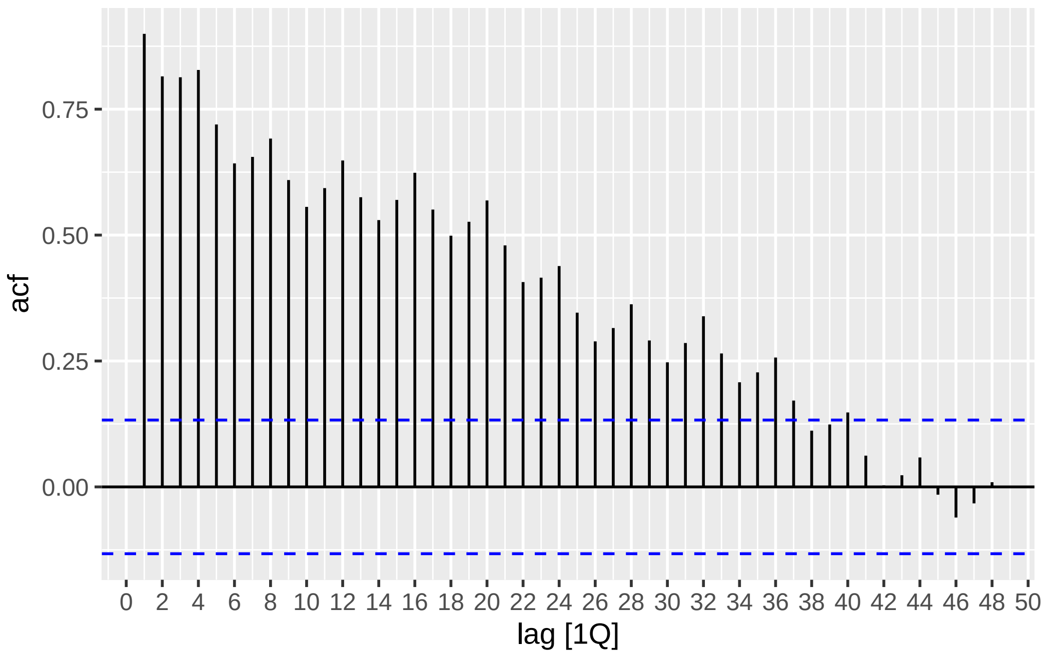
e. ACF and ACF plot Based on the lag plot and ACF, we can see there is a strong positive correlation in all lags, which provides further evidence for a trend in the time series. But, the autocorrelation coefficient decreases greatly with each lag, until it eventually dips below the significance level 38 quarters (9.5 years) into the time series. This tells us that past data may not be a good predictor of values at 10+ years into the future. The ACF plot also displays peaks at seasonal intervals (multiples of 4), providing more evidence for seasonality.

```
aus_production |> ACF(Bricks)
```

```
# A tsibble: 22 x 2 [1Q]
        lag   acf
   <cf_lag> <dbl>
 1       1Q 0.900
 2       2Q 0.815
```

```
 3        3Q 0.813
 4        4Q 0.828
 5        5Q 0.720
 6        6Q 0.642
 7        7Q 0.655
 8        8Q 0.692
 9        9Q 0.609
10       10Q 0.556
# i 12 more rows
```

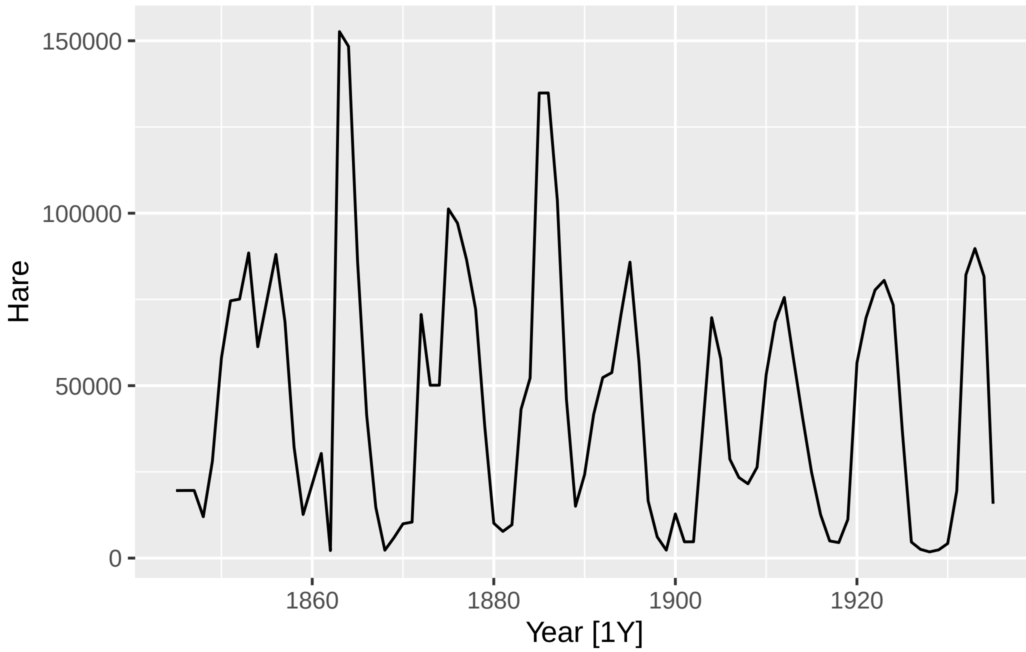
```
aus_production |> ACF(Bricks, lag_max = 48) |> autoplot()
```



3. **Hare** from **pelt**

a. time plot Based on the time plot, we can see a strong cyclical pattern in this time series. There appears to be lull periods of about 3-5 years, then trading will shoot up and stay around there over the course of a few years.

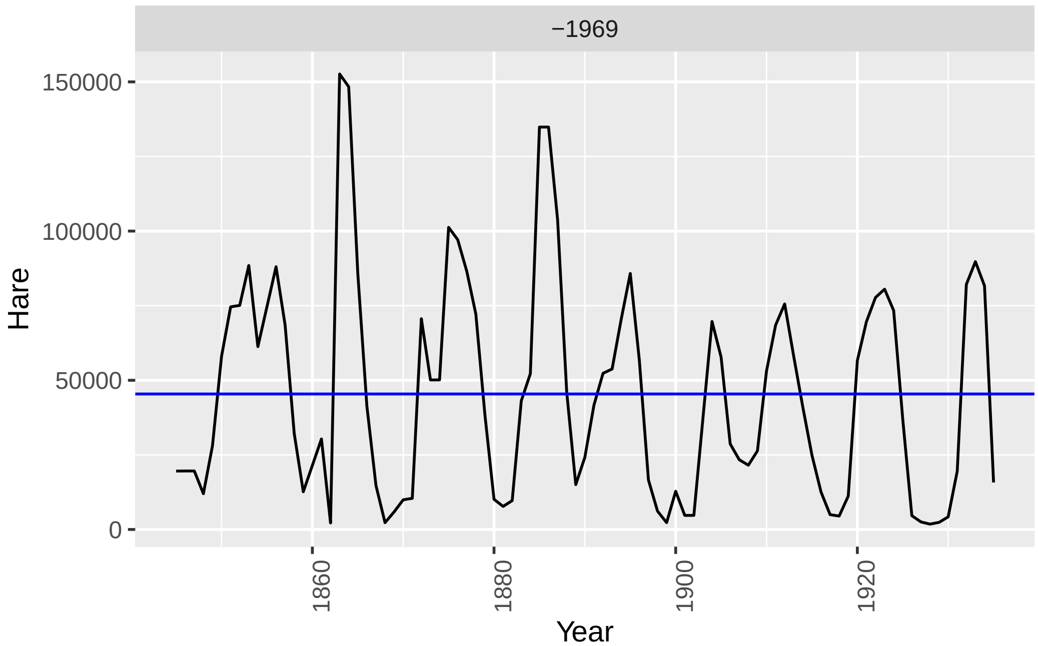
```
pelt |> autoplot(Hare)
```



b. seasonal plot

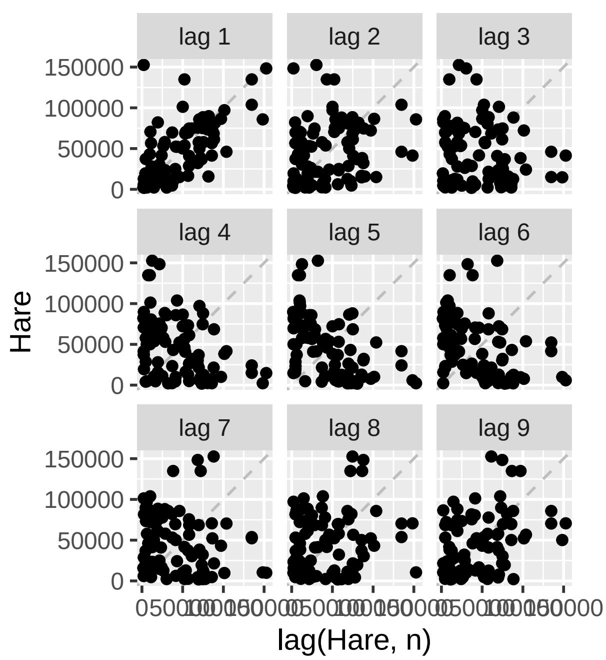
c. subseries plot The seasonal and subseries plots don't appear to work with the pelt data, because the index is 1 year. I couldn't find a solution to this

```
pelt |> gg_subseries(Hare)
```



d. lag plot The lag plots show us that there is a strong positive correlation in lag 1. That correlation diminishes with each lag, but seemingly retursn a bit after lag 8

```
pelt |> gg_lag(Hare, geom = "point")
```

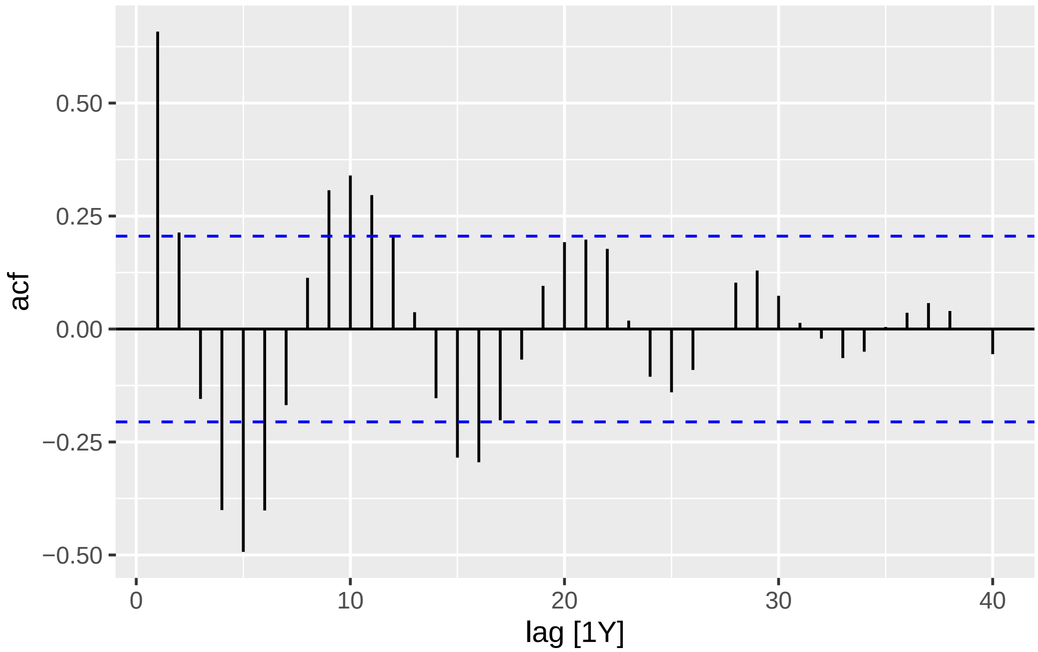


e. ACF and ACF plot ACF reveals that the autocorrelation coefficient ebbs and flows between positive and negative at an interval of about 3-5 lags. I'm not entirely sure, but I think this is evidence of a cyclic effect.

```
pelt |> ACF(Hare)
```

```
# A tsibble: 19 x 2 [1Y]
        lag     acf
   <cf_lag>   <dbl>
 1       1Y  0.658
 2       2Y  0.214
 3       3Y -0.155
 4       4Y -0.401
 5       5Y -0.493
 6       6Y -0.401
 7       7Y -0.168
 8       8Y  0.113
 9       9Y  0.307
10      10Y  0.340
11      11Y  0.296
12      12Y  0.206
13      13Y  0.0372
14      14Y -0.153
15      15Y -0.285
16      16Y -0.295
17      17Y -0.202
18      18Y -0.0676
19      19Y  0.0956
```

```
pelt |> ACF(Hare, lag_max = 40) |> autoplot()
```

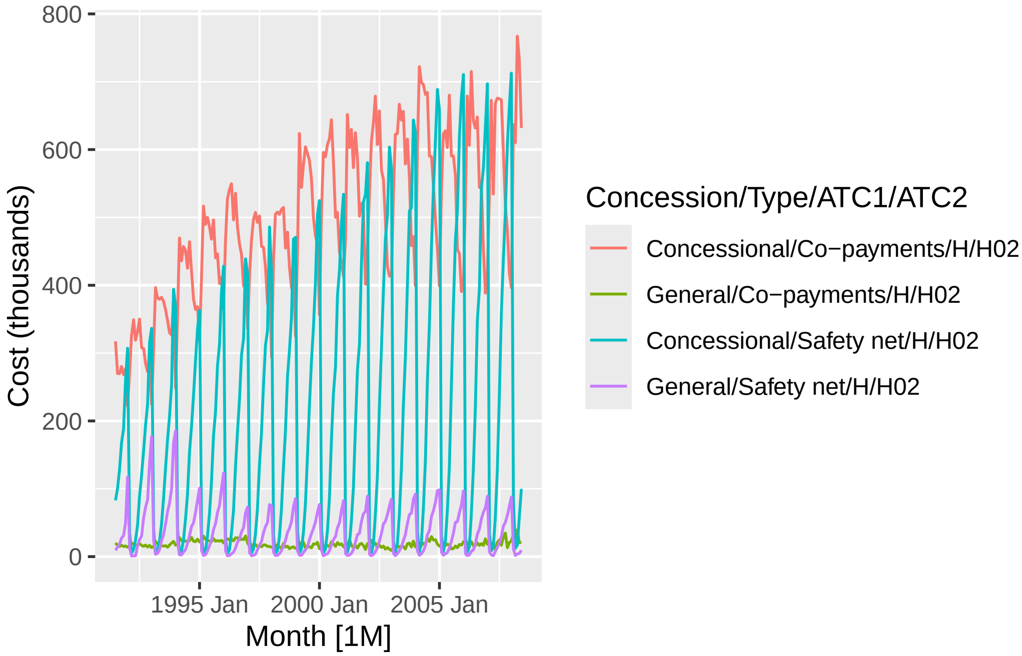


4. `Cost` from `PBS`

a. time plot In this time plot, we see a variety of behaviors depending on the group. Most of the groups don't appear to have a trend, but most appear to have seasonality and/or cyclic pattern.

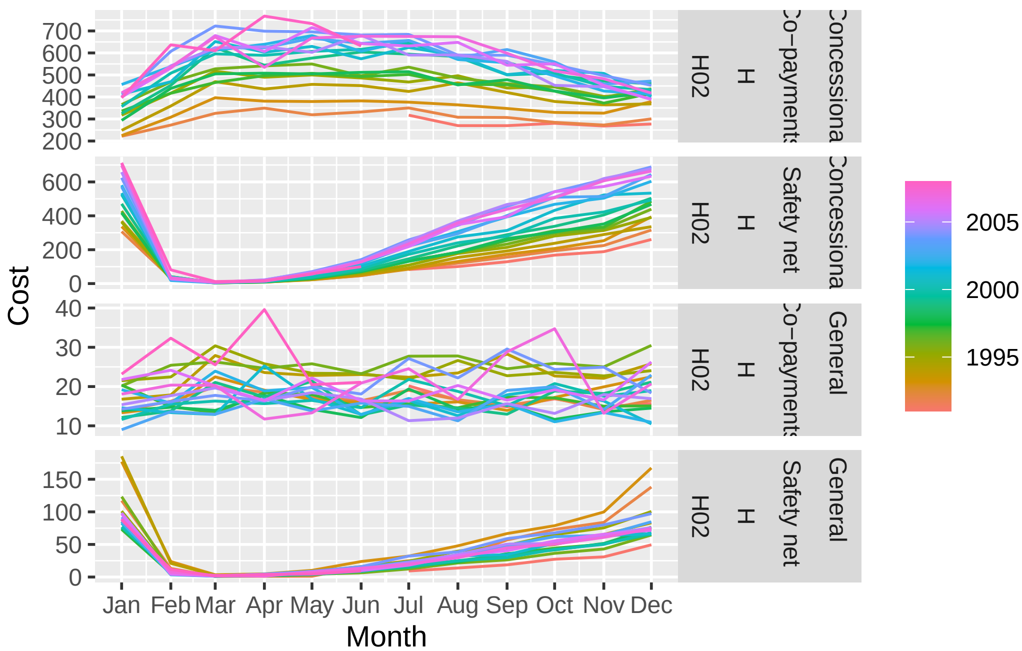
```
ho2 <- PBS |>
  filter(ATC2 == "H02")

ho2 |>
  mutate(Cost = Cost / 1e3) |>
  autoplot(Cost) +
  ylab("Cost (thousands)")
```



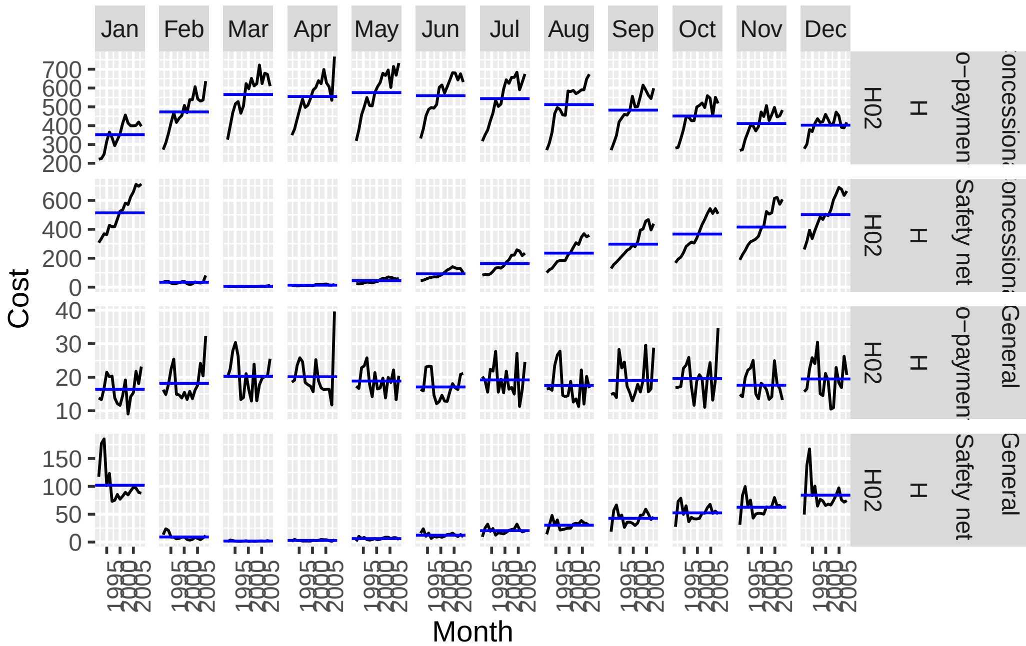
b. seasonal plot Again, we're seeing a variety of seasonal patterns. The Safety net group are fairly similar in that they peak in around Q3/Q4, but there yearly trends are different. the Co-payments group is more variant.

```
ho2 |>
  mutate(Cost = Cost / 1e3) |>
  gg_season(Cost)
```



c. subseries plot The subseries plots further my previous analyses. Most of the plots display seasonality, albeit in a variety of ways.

```
ho2 |>
  mutate(Cost = Cost / 1e3) |>
  gg_subseries(Cost)
```

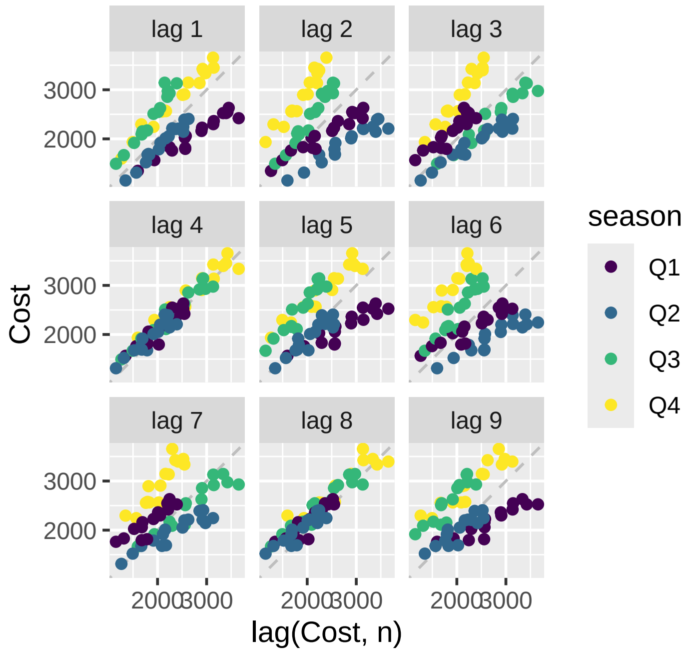


d. lag plot I realized I needed to adjust the dataset so that the key was just ATC2. After running the lag plot, I can see a strong positive trends for every lag. The correlation looks extra strong at the seasonal multiples.

```
grouped_ho2 <- ho2 |>
  group_by(ATC2) |>
  summarise(Cost = sum(Cost)) |>
  ungroup()

grouped_ho2 |>
  mutate(Cost = Cost / 1e3,
         Quarter = yearquarter(Month)) |>
  as_tibble() |>
  group_by(Quarter, ATC2) |>
  summarise(Cost = sum(Cost)) |>
  ungroup() |>
  as_tsibble(index = Quarter) |>
  gg_lag(Cost, geom = "point")
```

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

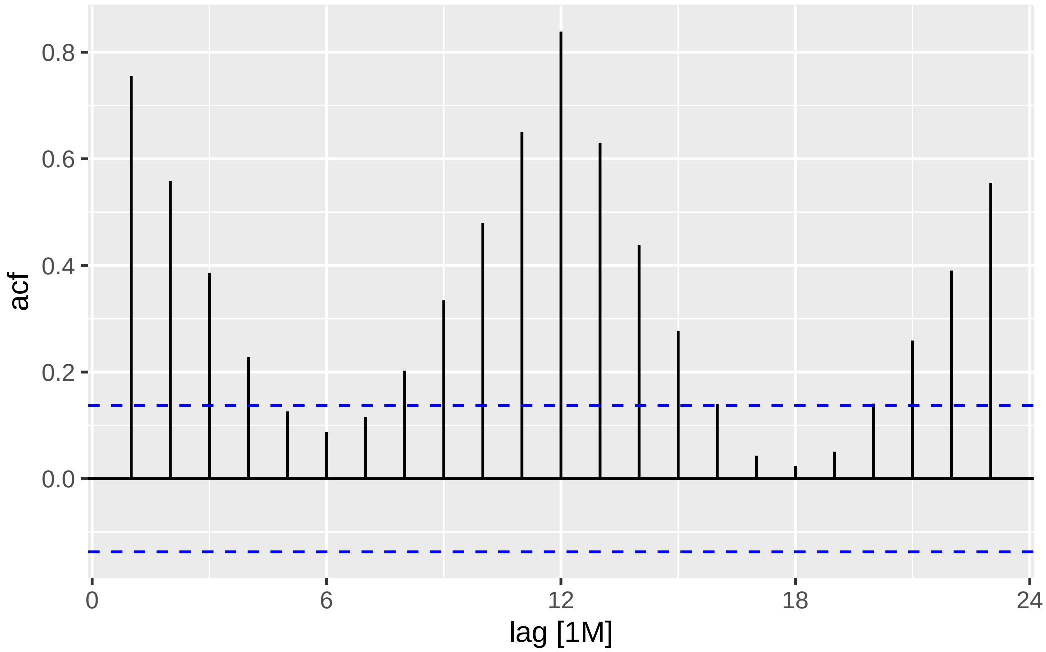


e. ACF and ACF plot The ACF plot shows us that the autocorrelation coefficient ebbs and flows from really high to really low positive correlation. I think this suggests a cyclic pattern.

```
grouped_ho2 |> ACF(Cost)
```

```
# A tsibble: 23 x 3 [1M]
# Key:       ATC2 [1]
   ATC2       lag    acf
   <chr> <cf_lag>  <dbl>
 1 H02         1M 0.755
 2 H02         2M 0.558
 3 H02         3M 0.386
 4 H02         4M 0.228
 5 H02         5M 0.126
 6 H02         6M 0.0874
 7 H02         7M 0.116
 8 H02         8M 0.203
 9 H02         9M 0.335
10 H02        10M 0.479
# i 13 more rows
```

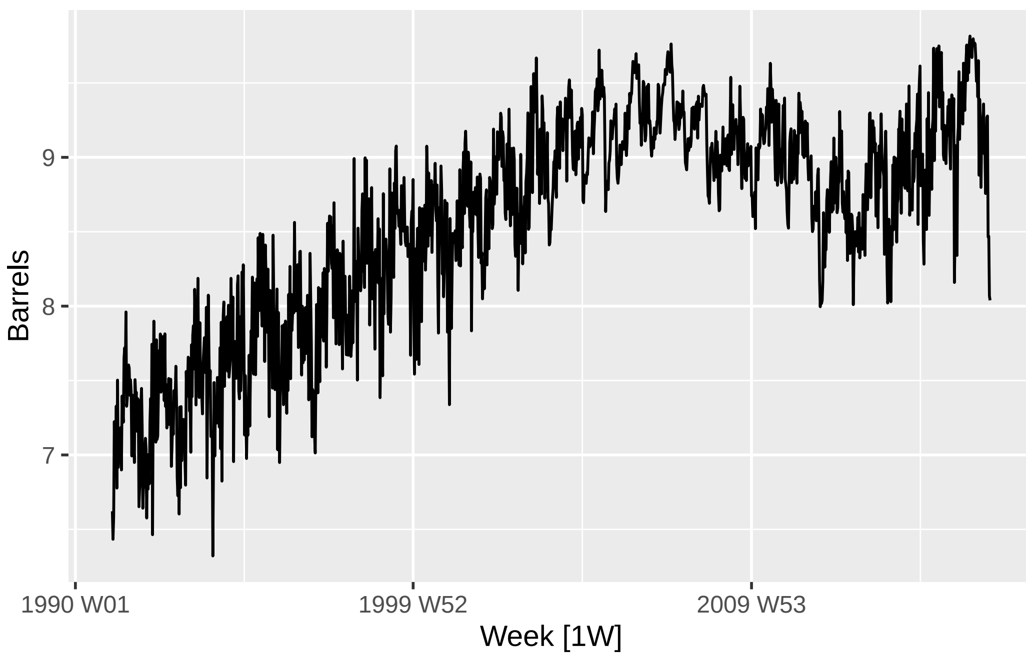
```r
grouped_ho2 |> ACF(Cost) |> autoplot()
```



5. `Barrels` from `us_gasoline`

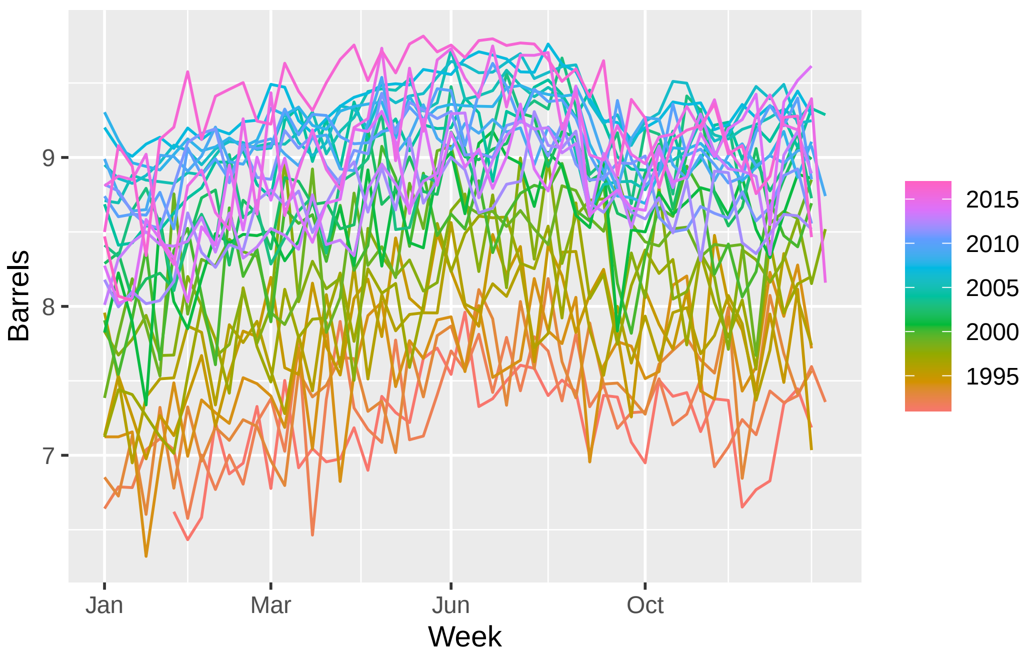
a. time plot Based on the time plot, we can see the time series starts with an upward trend, then eventually plateaus. There also appears to be seasonality.

```r
us_gasoline |> autoplot(Barrels)
```



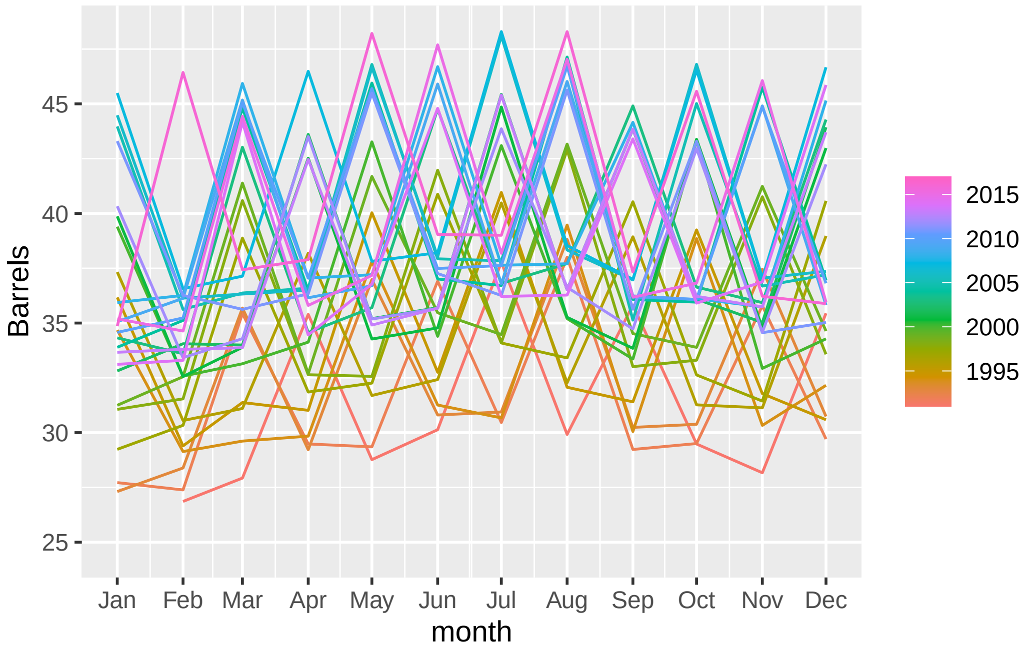
b. seasonality plot The seasonality plot looks like the barrels are lower in the fall/winter months and then build higher into the summer months. I decided to look at a monthly view to clear up some of the noise. This view is quite interesting, as there seems to be a consistent seasonality pattern of 2 month intervals.

```
us_gasoline |> gg_season(Barrels)
```



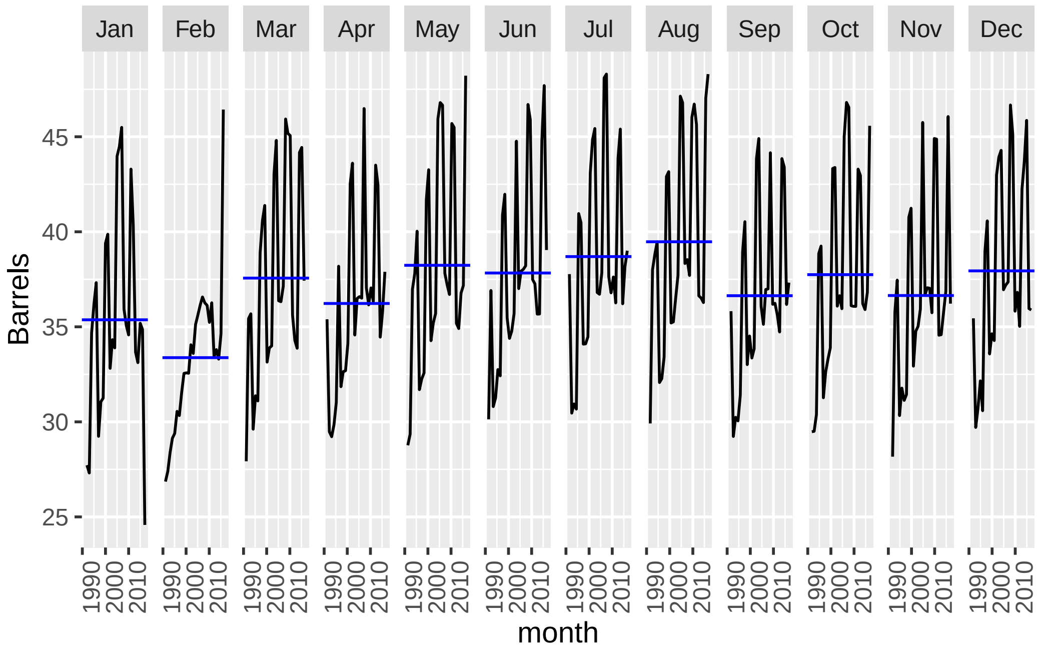

```
gas_month <- us_gasoline |>
  mutate(month = yearmonth(Week)) |>
  as_tibble() |>
  group_by(month) |>
  summarise(Barrels = sum(Barrels)) |>
  ungroup() |>
  as_tsibble(index = month)

gas_month |> gg_season(Barrels)
```



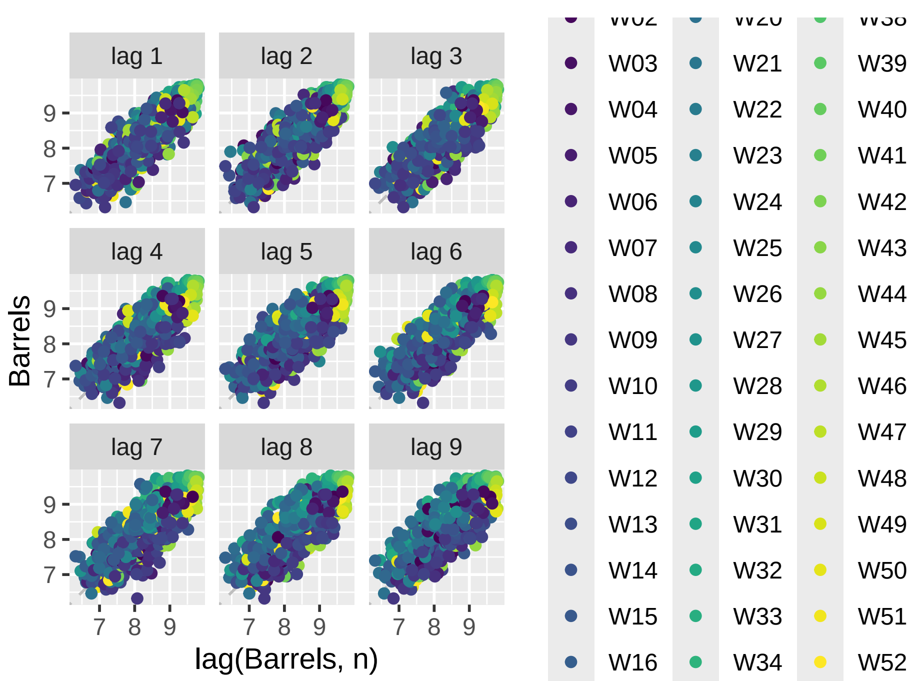
c. subseries plot The subseries plot shows further evidence for seasonality. February looks interesting because it has a very different pattern from the other months. It mostly shoots up over the span of the time series and has several lull years where `Barrels` growth slows, then sharply drops before rising again.

```
gas_month |> gg_subseries(Barrels)
```



d. lag plot

```
us_gasoline |> gg_lag(Barrels, geom = "point")
```



e. ACF and ACF plot The lag plot and ACF show us that there is strong positive correlation across all the lags.

```r
us_gasoline |> ACF(Barrels)
```

```
# A tsibble: 31 x 2 [1W]
        lag   acf
   <cf_lag> <dbl>
 1       1W 0.893
 2       2W 0.882
 3       3W 0.873
 4       4W 0.866
 5       5W 0.847
 6       6W 0.844
 7       7W 0.832
 8       8W 0.831
 9       9W 0.822
10      10W 0.808
# i 21 more rows
```

```r
us_gasoline |> ACF(Barrels) |> autoplot()
```