Homework 3

Brandon Leslie

Forecasting Principles and Practice

```r
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.4
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```r
library(tidymodels)
```

```
-- Attaching packages -------------------------------------- tidymodels 1.3.0 --
v broom        1.0.7     v rsample      1.2.1
v dials        1.4.0     v tune         1.3.0
v infer        1.0.7     v workflows    1.2.0
v modeldata    1.4.0     v workflowsets 1.1.0
v parsnip      1.3.0     v yardstick    1.3.2
v recipes      1.1.1
-- Conflicts ----------------------------------------- tidymodels_conflicts() --
x scales::discard() masks purrr::discard()
x dplyr::filter()   masks stats::filter()
x recipes::fixed()  masks stringr::fixed()
x dplyr::lag()      masks stats::lag()
```

```
x yardstick::spec() masks readr::spec()
x recipes::step()   masks stats::step()
```

```r
library(USgas)
library(scales)
library(fpp3)
```

```
Registered S3 method overwritten by 'tsibble':
  method               from
  as_tibble.grouped_df dplyr
-- Attaching packages -------------------------------------------- fpp3 1.0.1 --
v tsibble     1.1.6     v feasts      0.4.1
v tsibbledata 0.4.1     v fable       0.4.1
-- Conflicts ------------------------------------------------- fpp3_conflicts --
x fabletools::accuracy()    masks yardstick::accuracy()
x lubridate::date()         masks base::date()
x scales::discard()         masks purrr::discard()
x dplyr::filter()           masks stats::filter()
x fabletools::generate()    masks infer::generate()
x fabletools::hypothesize() masks infer::hypothesize()
x tsibble::intersect()      masks base::intersect()
x tsibble::interval()       masks lubridate::interval()
x dplyr::lag()              masks stats::lag()
x fabletools::null_model()  masks parsnip::null_model()
x tsibble::setdiff()        masks base::setdiff()
x tsibble::union()          masks base::union()
```

```r
library(tsibbledata)
```

Q4: US Gas

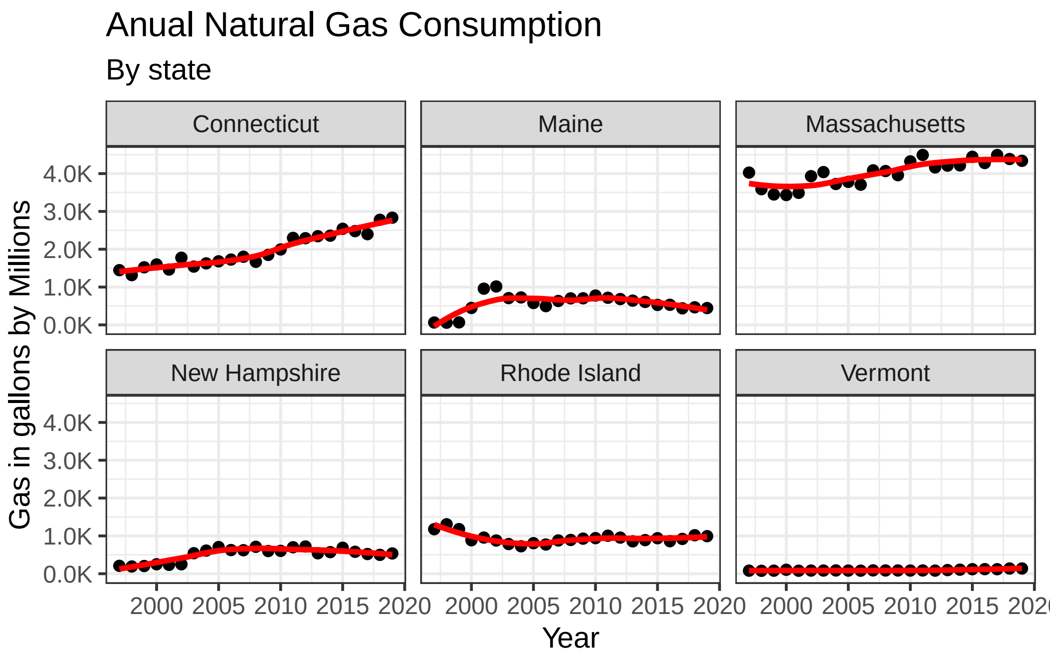
Tsibble

```r
tsibble1 <- us_total %>%
  as_tsibble(
    key = state,
    index = year
  )
glimpse(tsibble1)
```

```
Rows: 1,266
Columns: 3
Key: state [53]
$ year  <int> 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007~
$ state <chr> "Alabama", "Alabama", "Alabama", "Alabama", "Alabama", "Alabama"~
$ y     <int> 324158, 329134, 337270, 353614, 332693, 379343, 350345, 382367, ~
```

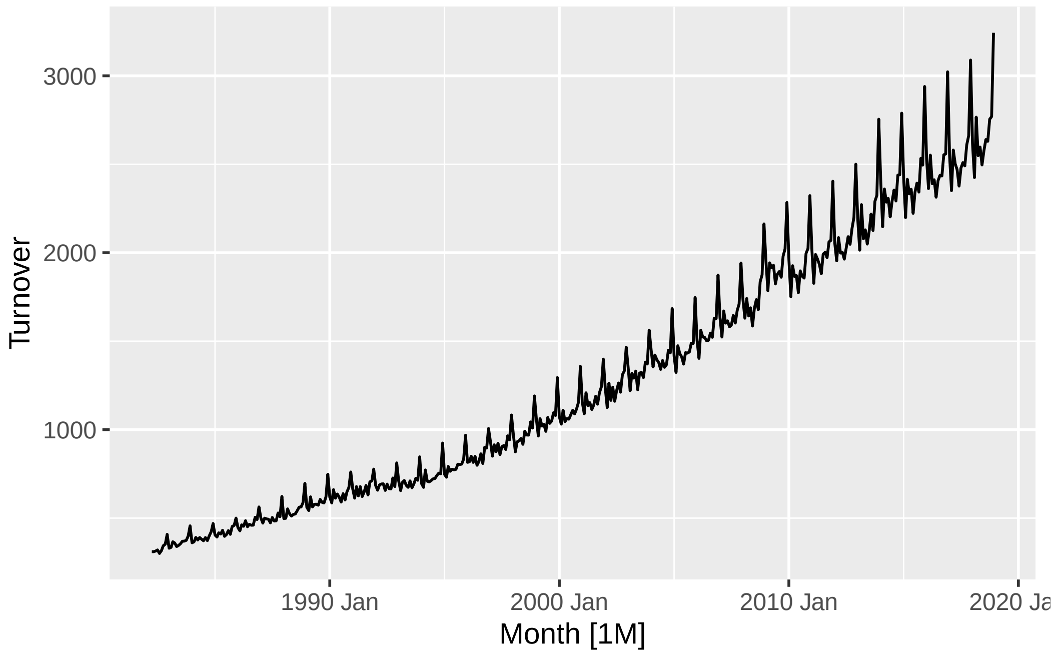
Graph

```r
NewEngland = c("Maine", "Vermont", "New Hampshire", "Massachusetts", "Connecticut", "Rhode Is
tsibble1 %>%
  filter(state %in% NewEngland) %>%
  ggplot(aes(year, y)) +
  geom_point() +
  geom_smooth(se = FALSE, method = "loess", color = "red") +
  scale_y_continuous(labels = label_number(scale = 0.00001, suffix = "K")) +
  scale_x_continuous(breaks = breaks_width(5)) +
  labs(
    title = "Anual Natural Gas Consumption",
    subtitle = "By state",
    y = "Gas in gallons by Millions",
    x = "Year"
  ) +
  facet_wrap(~state) +
  theme_bw()
```

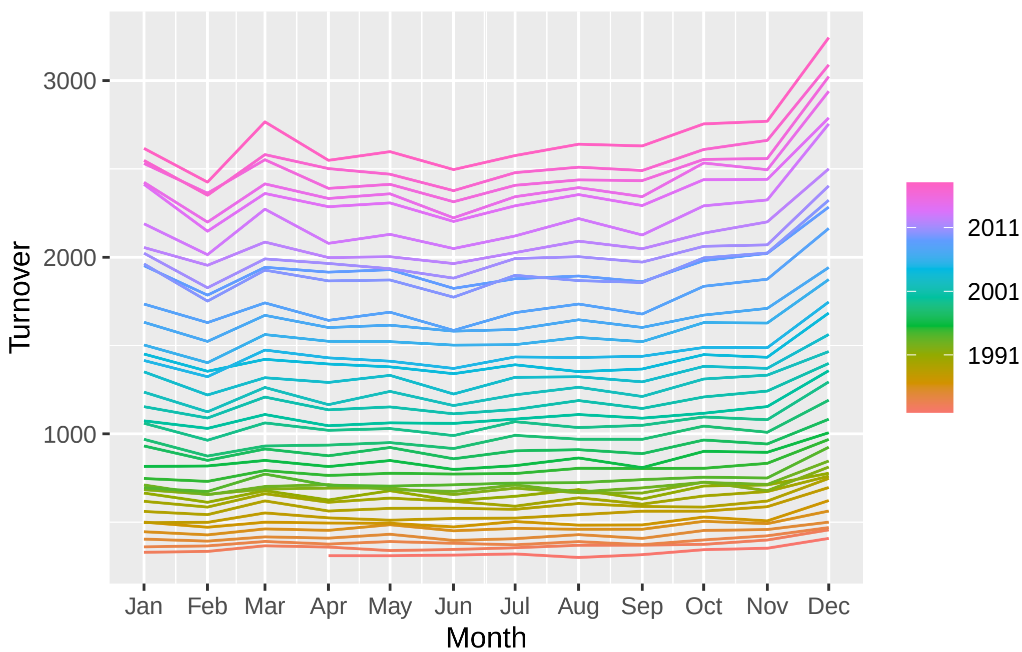


Q7: Australian Retail Data

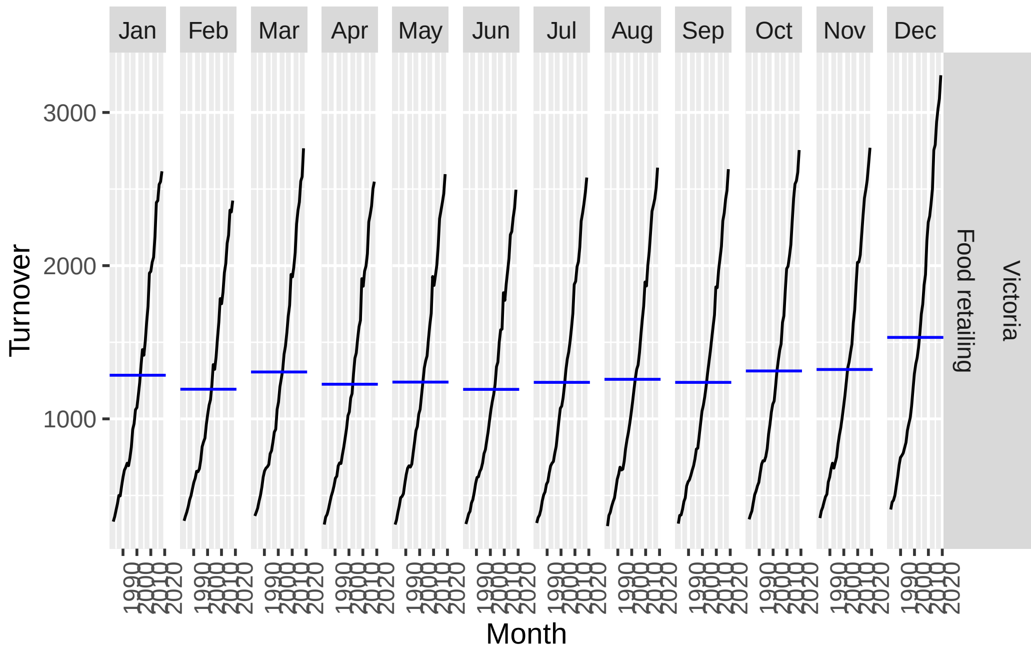
```r
set.seed(84625314)
myseries <- aus_retail %>%
  as_tsibble() %>%
  filter(`Series ID` == sample(unique(aus_retail$`Series ID`),1)) %>%
  as_tsibble(index = Month)


autoplot(myseries)
```



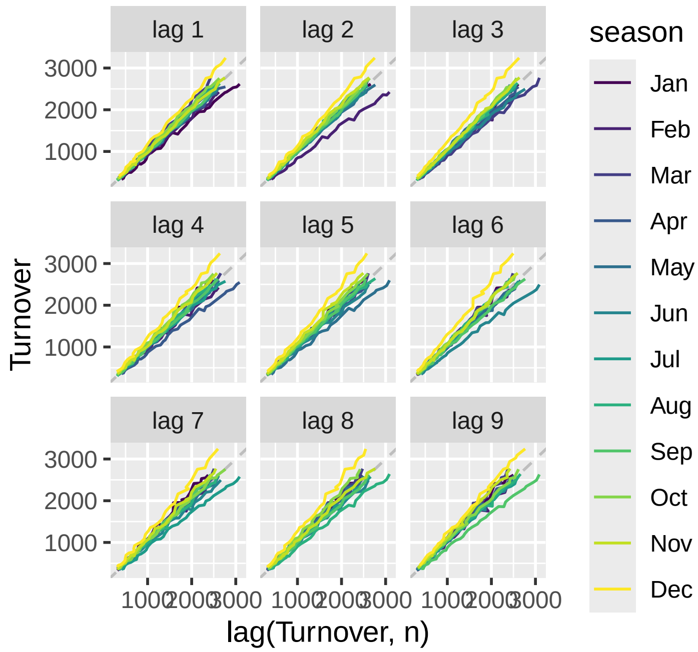

```r
gg_season(myseries)
```



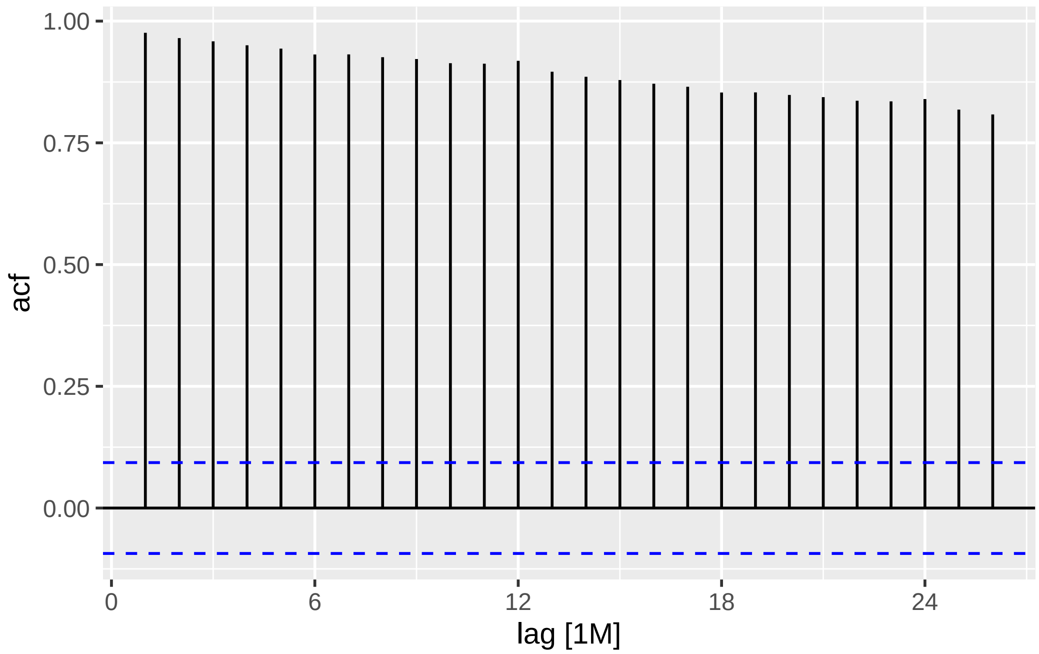

```r
gg_subseries(myseries)
```




```r
gg_lag(myseries)
```




```r
myseries %>%
  ACF(Turnover) %>%
  autoplot()
```



Can you spot any seasonality, cyclicity and trend? What do you learn about the series?

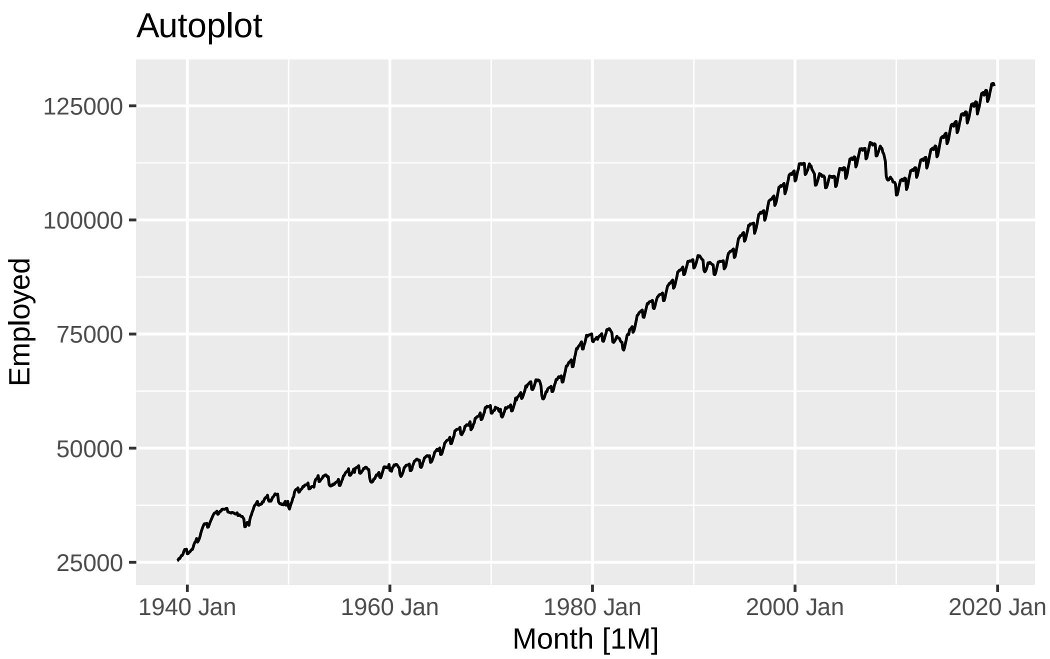
- **Seasonality**
  - There seems to be an increase in turnovers in late fall/early winter, as-well as a peak in the month of march.
- **Cyclicity**
  - In the autoplot graph, cyclicity is represented through repeated sudden peaks, then a dip, with.a short after-peak, followed by a slight increase, then this process repeats to another sudden peak.
- **Trends**
  - As we progress through time, we can see a positive trend as years & months go by.

Q8: "Total Private" Timeseries

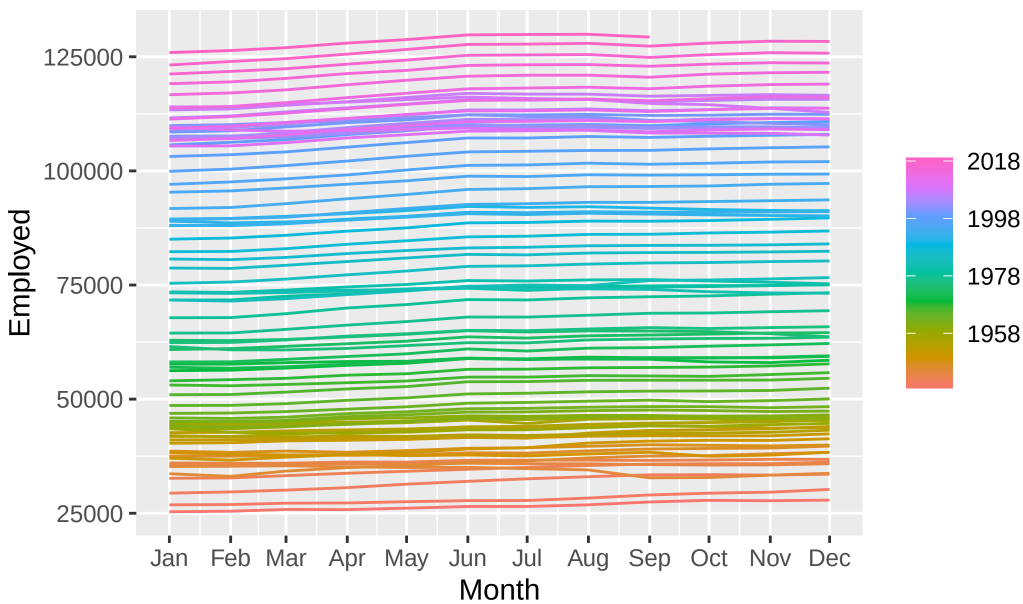
```
explore_ts <- function(data, index, value) {
  index_sym <- rlang::ensym(index)
  value_sym <- rlang::ensym(value)

  ts_data <- data %>%
```

```
    filter(!is.na(!!value_sym)) %>%
    as_tsibble(index = !!index_sym)

  print(autoplot(ts_data, !!value_sym) + ggtitle("Autoplot"))
  print(gg_season(ts_data, !!value_sym) + ggtitle("Seasonal Plot"))
  print(gg_subseries(ts_data, !!value_sym) + ggtitle("Subseries Plot"))
  print(gg_lag(ts_data, !!value_sym) + ggtitle("Lag Plot"))
  print(ts_data |> ACF(!!value_sym) |> autoplot() + ggtitle("ACF Plot"))
}
```

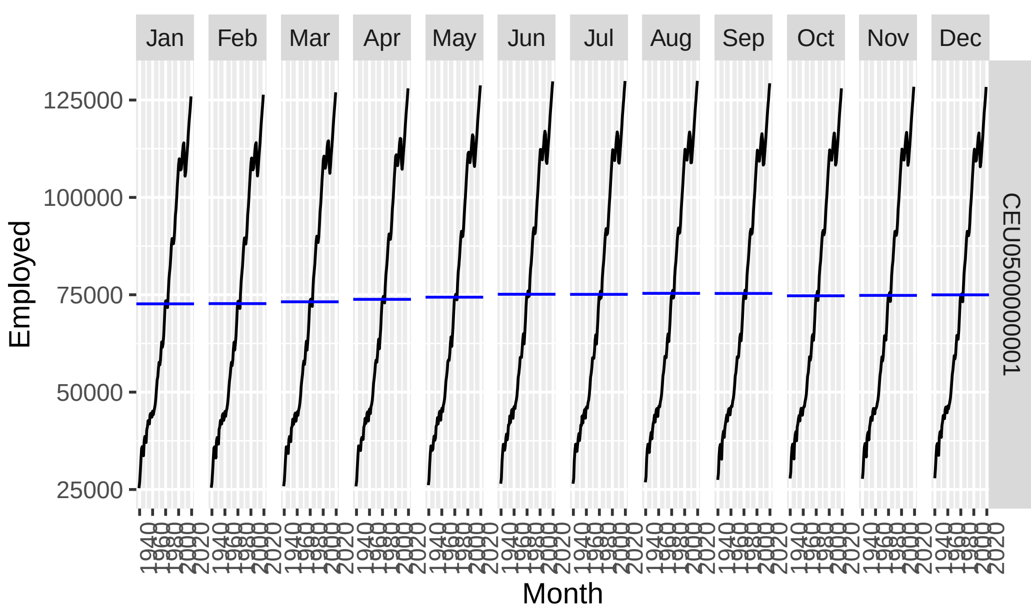
```
# 1. Total Private Employed
private_employed <- us_employment |> filter(Title == "Total Private")
explore_ts(private_employed, Month, Employed)
```

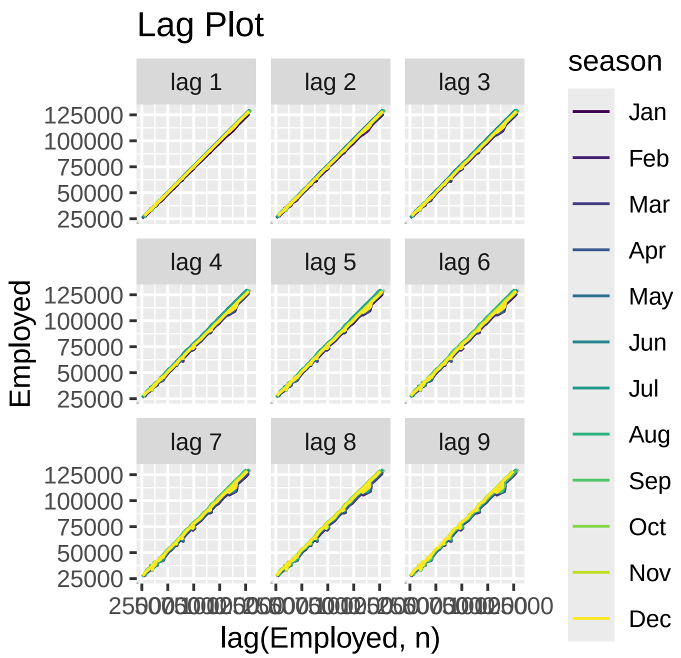


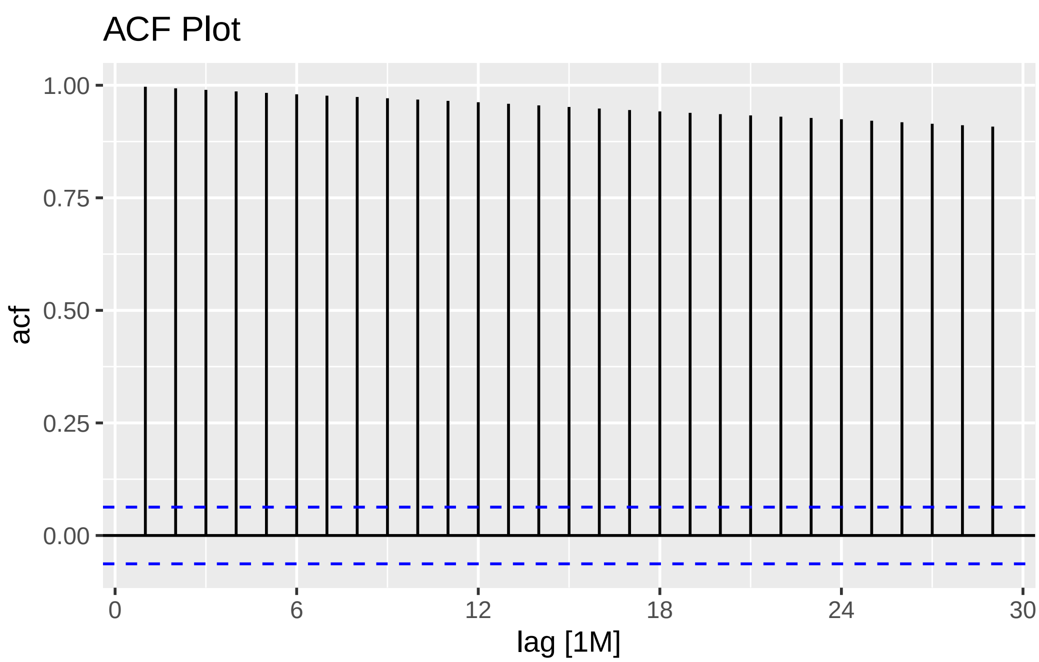
Seasonal Plot



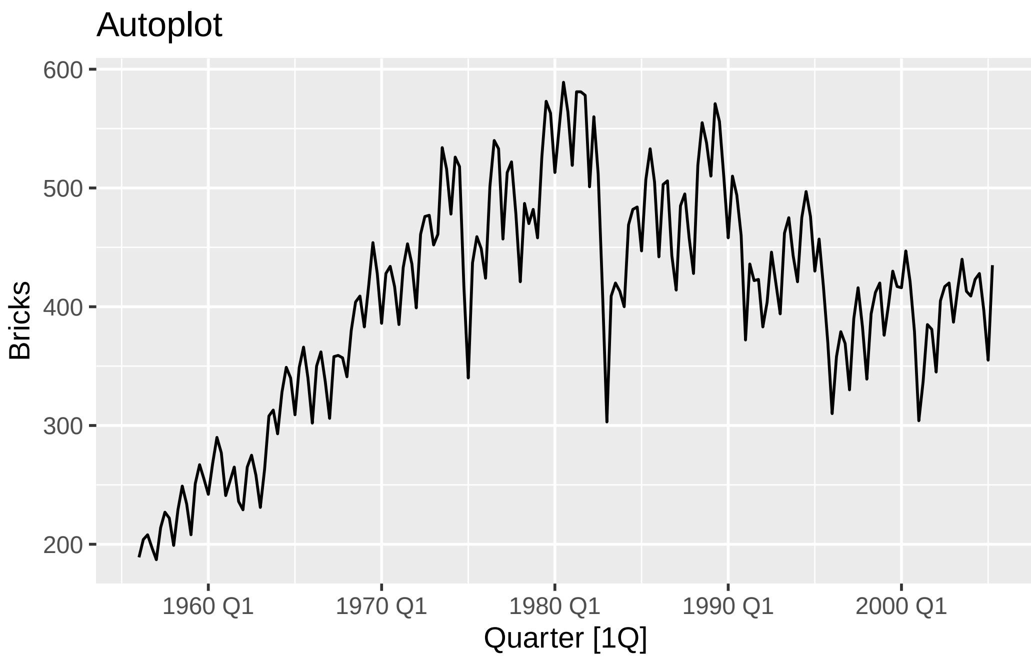
Subseries Plot

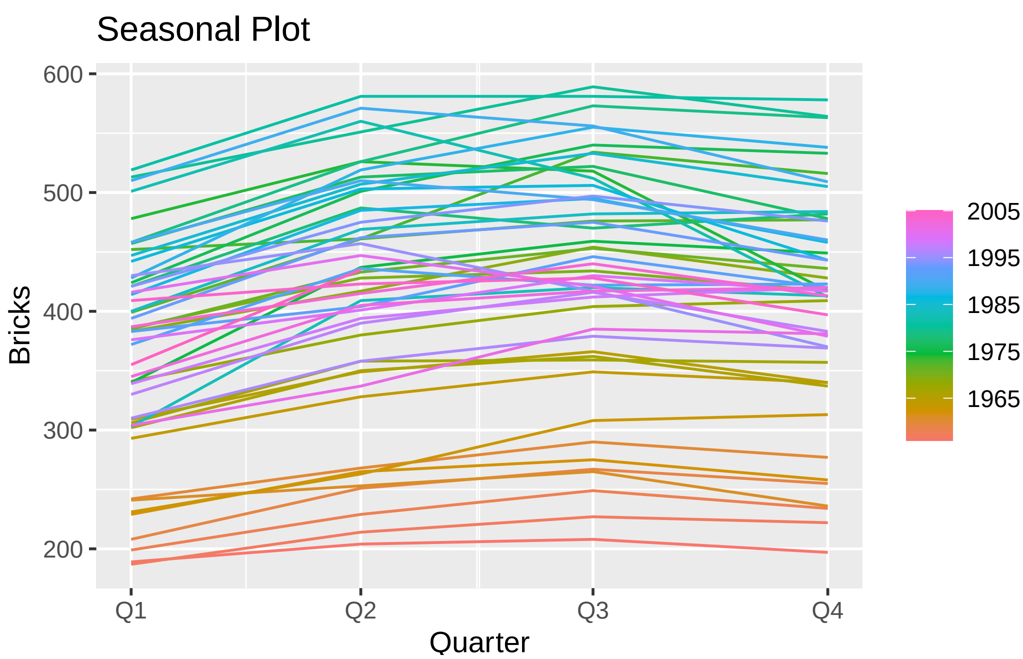




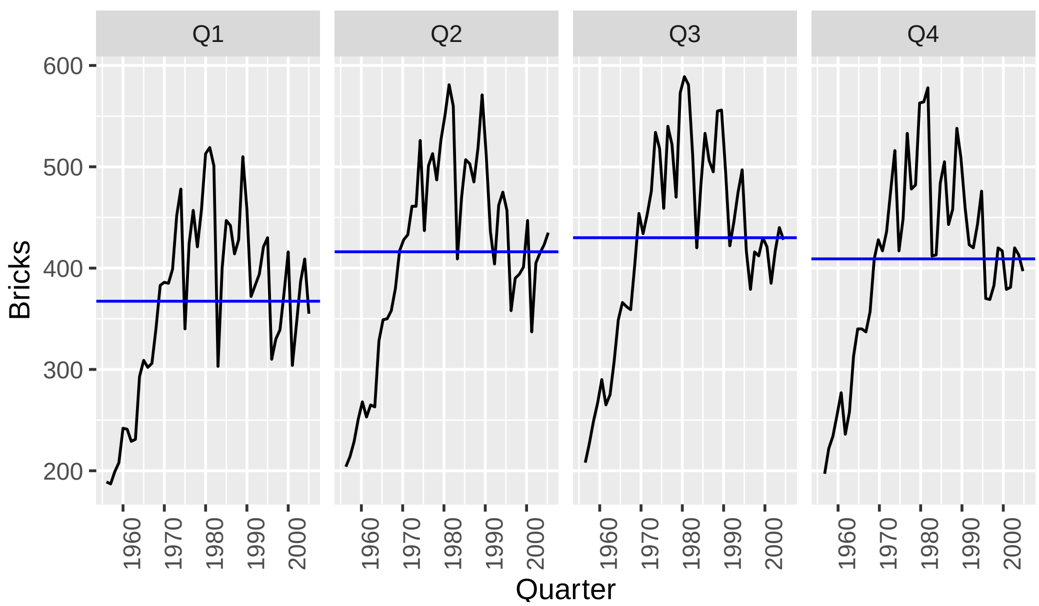



```
# 2. Bricks
explore_ts(aus_production, Quarter, Bricks)
```

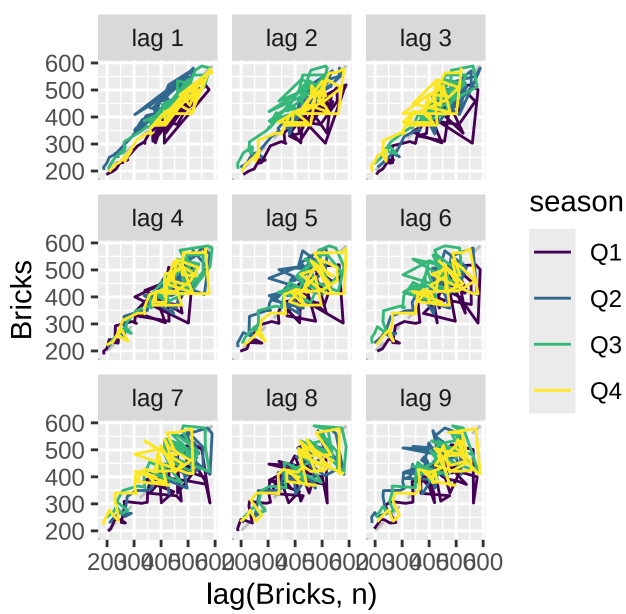

Autoplot
600
500
400
300
200
Bricks
1960 Q1
1970 Q1
1980 Q1
1990 Q1
2000 Q1
Quarter [1Q]

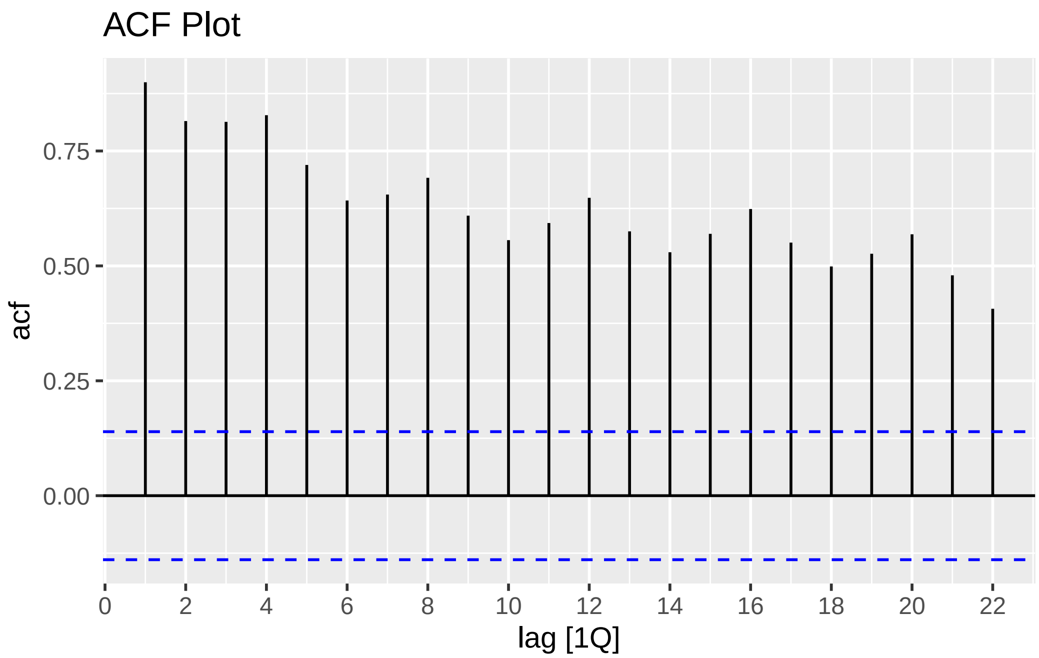

Seasonal Plot
600
500
400
300
200
Bricks
Q1
Q2
Q3
Q4
Quarter
2005
1995
1985
1975
1965

Subseries Plot

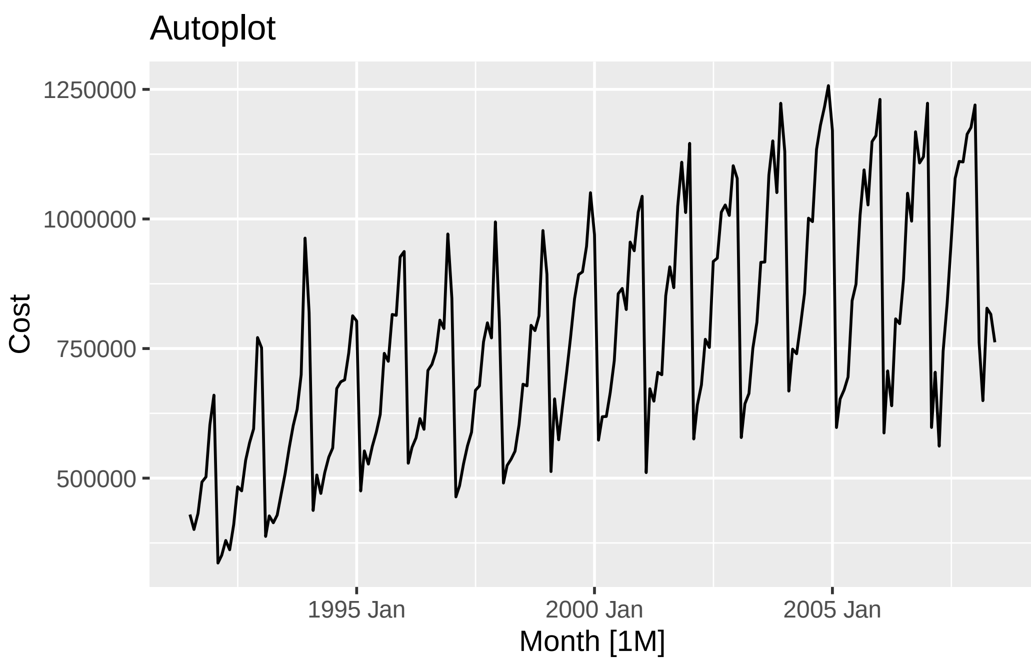


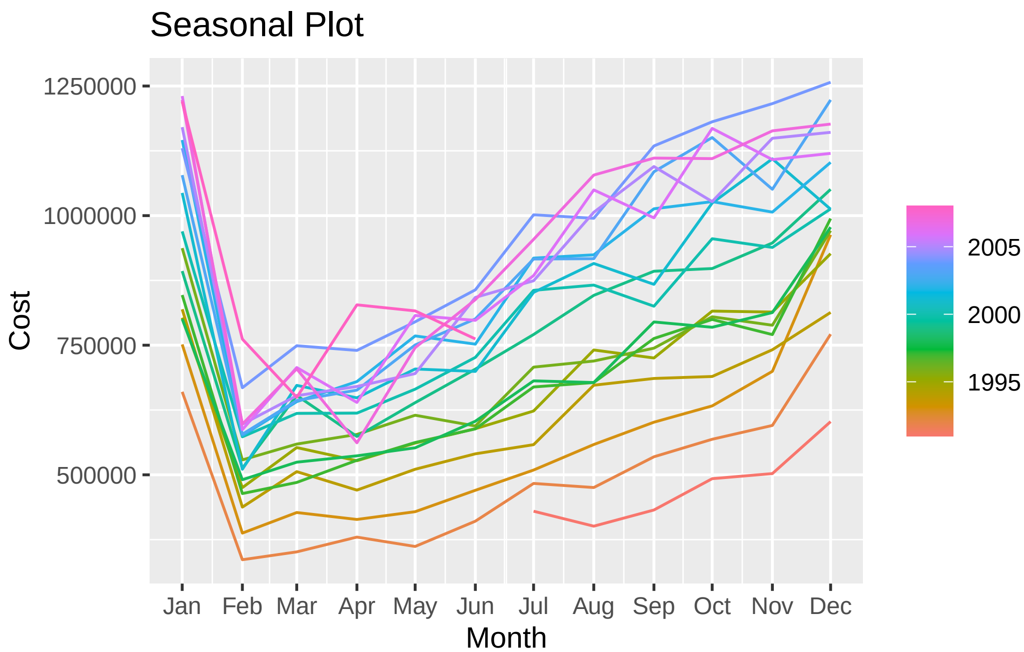
Lag Plot

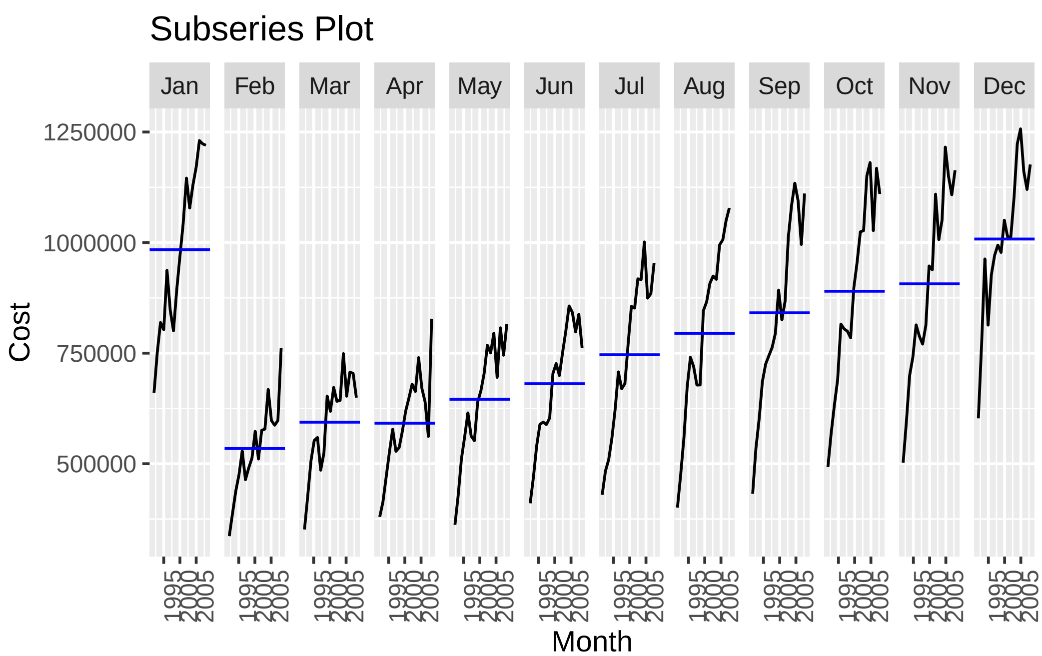


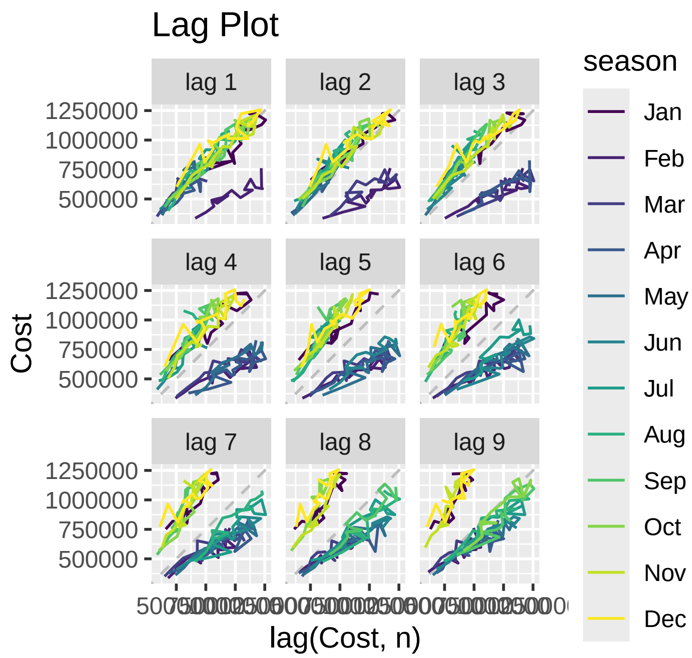


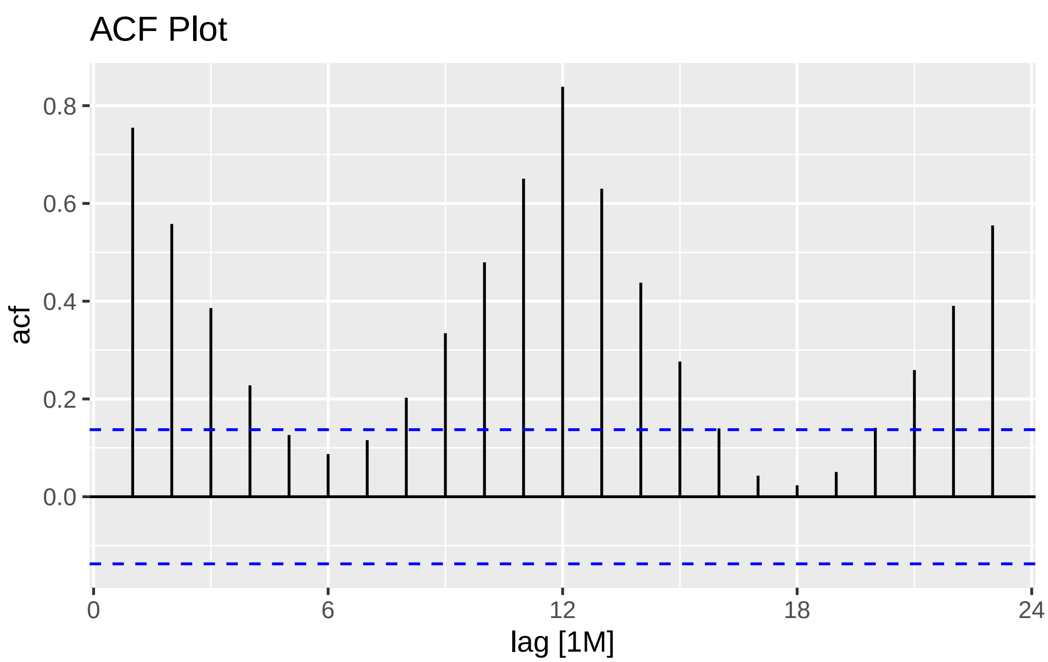

```
# 4. H02 Cost
h02 <- PBS |> filter(ATC2 == "H02") |> summarise(Cost = sum(Cost))
explore_ts(h02, Month, Cost)
```




Seasonal Plot
Cost
1250000
1000000
750000
500000
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
Month
2005
2000
1995

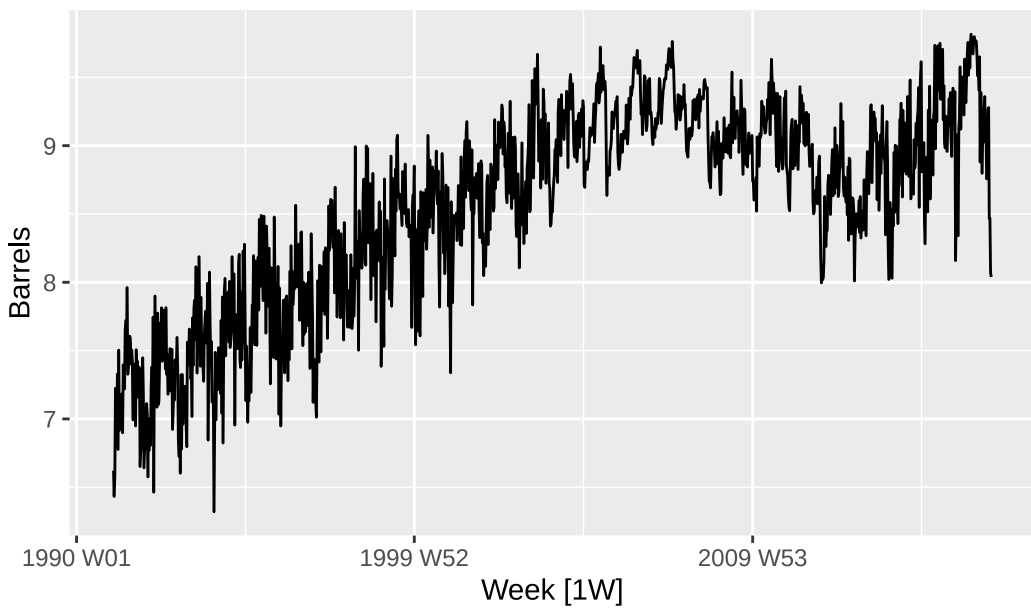

Subseries Plot
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
Cost
1250000
1000000
750000
500000
1995
2000
2005
Month



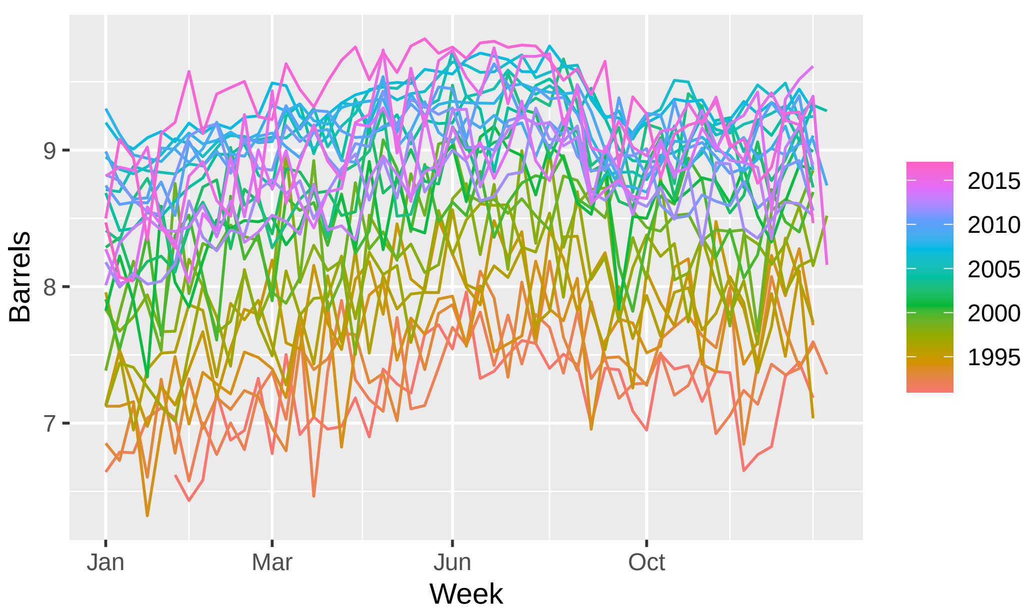


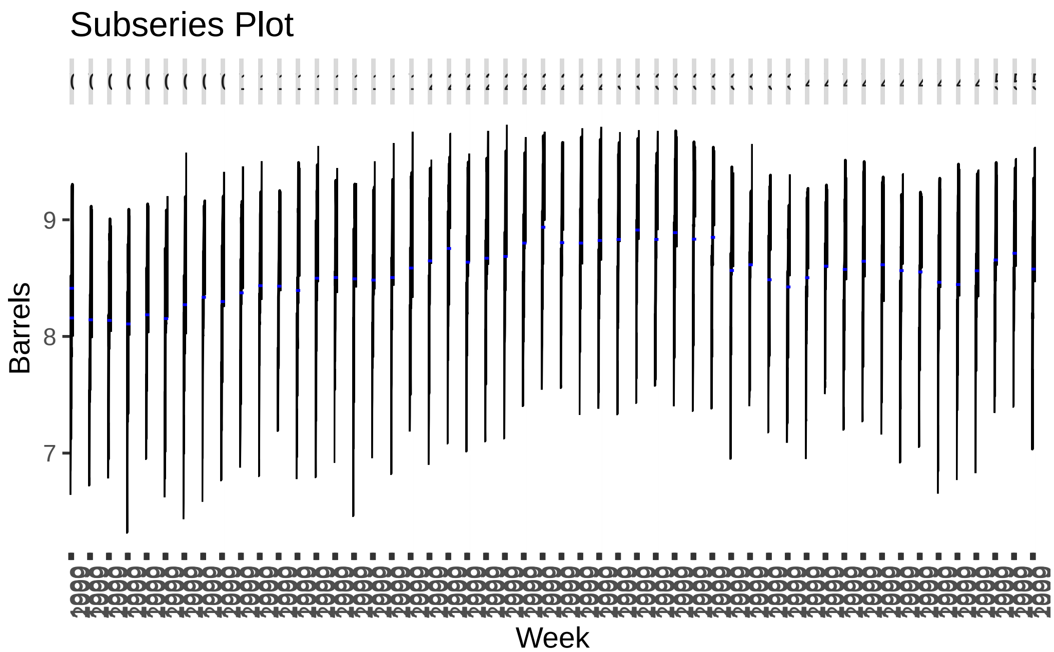

```
# 5. Barrels
explore_ts(us_gasoline, Week, Barrels)
```

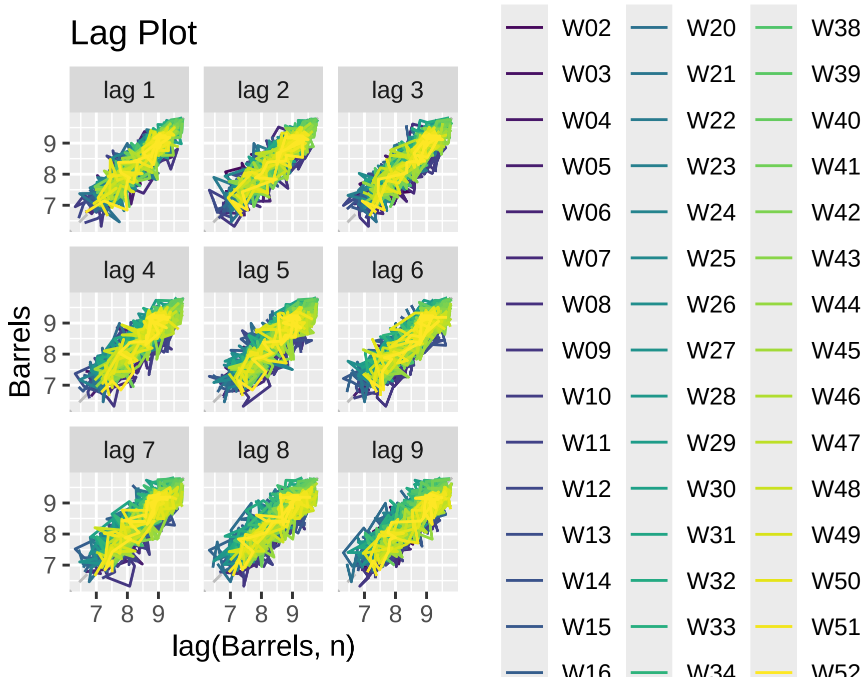
Autoplot

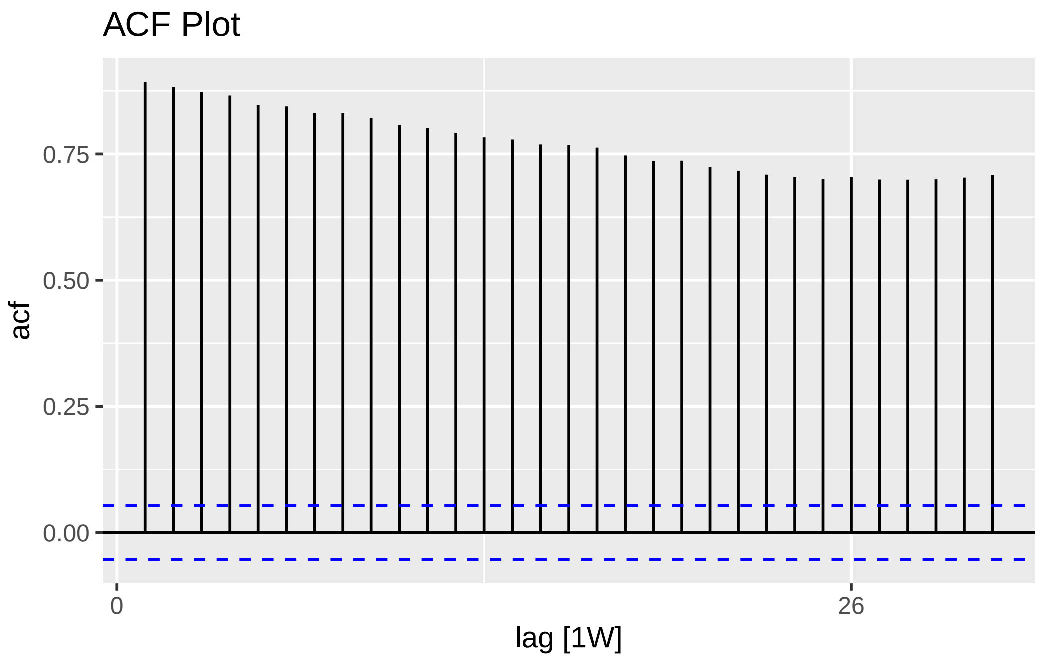


Seasonal Plot




Subseries Plot
Barrels
9
8
7
Week


Lag Plot
lag 1
lag 2
lag 3
lag 4
lag 5
lag 6
lag 7
lag 8
lag 9
Barrels
9
8
7
7 8 9
lag(Barrels, n)
W02
W03
W04
W05
W06
W07
W08
W09
W10
W11
W12
W13
W14
W15
W16
W20
W21
W22
W23
W24
W25
W26
W27
W28
W29
W30
W31
W32
W33
W34
W38
W39
W40
W41
W42
W43
W44
W45
W46
W47
W48
W49
W50
W51
W52



Summary of Time Series Features

Series	Seasonality	Cyclicity	Trend	Unusual Years
Retail (AU)	Strong	Light	Upward	2020
Employment	Mild	Clear	Upward	2009, 2020
Bricks	Strong	Some	Downward	2008–2010
Hare	None	Strong	Cyclical	Every 10 years
H02 Cost	Mild	Mild	Upward	Occasional spikes
Gasoline	Strong	Some	Stable	2020

Q9: Time Plot Matching

Daily Temperature of a Cow = C

Monthly Accidental Deaths = A

Monthly Air Passengers = D

Annual Mink Trappings = B

Q11: Google Closing Stock Prices

```
dgoog <- gafa_stock |>
  filter(Symbol == "GOOG", year(Date) >= 2018) |>
  mutate(trading_day = row_number()) |>
  update_tsibble(index = trading_day, regular = TRUE) |>
  mutate(diff = difference(Close))

dgoog
```

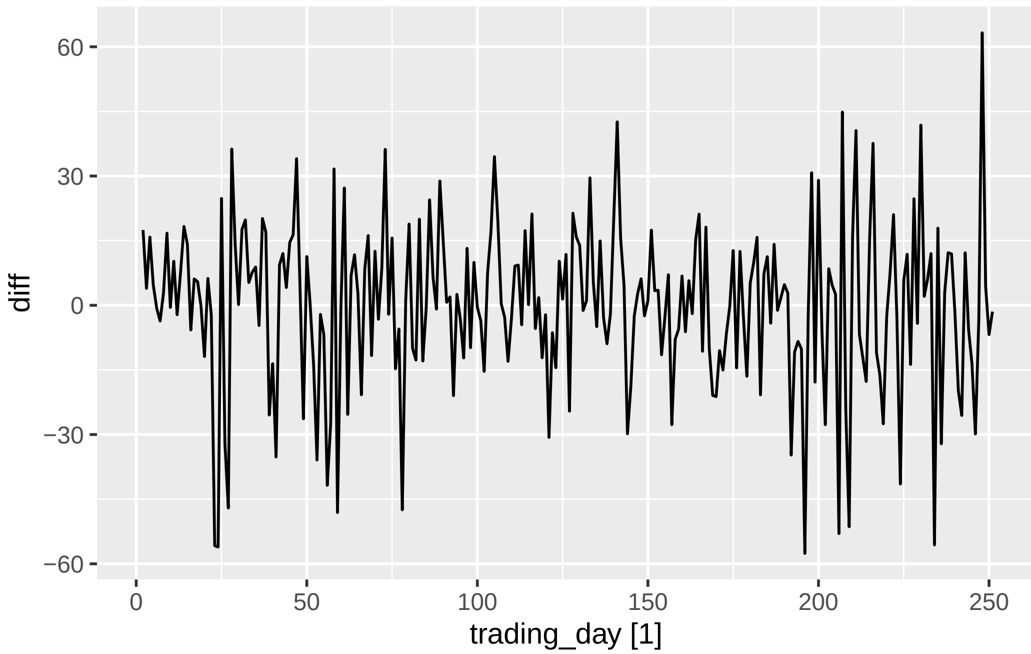
```
# A tsibble: 251 x 10 [1]
# Key:       Symbol [1]
   Symbol Date        Open  High   Low Close Adj_Close Volume trading_day   diff
   <chr>  <date>     <dbl> <dbl> <dbl> <dbl>     <dbl>  <dbl>       <int>  <dbl>
 1 GOOG   2018-01-02 1048. 1067. 1045. 1065       1065  1.24e6           1 NA
 2 GOOG   2018-01-03 1064. 1086. 1063. 1082.      1082. 1.43e6           2 17.5
 3 GOOG   2018-01-04 1088  1094. 1084. 1086.      1086. 1.00e6           3  3.92
 4 GOOG   2018-01-05 1094  1104. 1092  1102.      1102. 1.28e6           4 15.8
 5 GOOG   2018-01-08 1102. 1111. 1102. 1107.      1107. 1.05e6           5  4.71
 6 GOOG   2018-01-09 1109. 1111. 1101. 1106.      1106. 9.02e5           6 -0.680
 7 GOOG   2018-01-10 1097. 1105. 1096. 1103.      1103. 1.04e6           7 -3.65
 8 GOOG   2018-01-11 1106. 1107. 1100. 1106.      1106. 9.78e5           8  2.91
 9 GOOG   2018-01-12 1102. 1124. 1101. 1122.      1122. 1.72e6           9 16.7
10 GOOG   2018-01-16 1133. 1140. 1118. 1122.      1122. 1.58e6          10 -0.5
# i 241 more rows
```

- **Why was it necessary to re-index the tsibble?**

  1. We had to re-index the *tsibble* because we wanted the **Daily** changes

- **Plot these differences and their ACF.**

```
autoplot(dgoog, diff)
```



1. **Do the changes in the stock prices look like white noise?**

    1. Yes, the changes in stock prices look like white noise.