DATA 624: PREDICTIVE ANALYTICS HW2

Gabriel Campos

Last edited February 10, 2024

```
library('fpp3')
library('tsibble')
library('ggplot2')
library('readr')
library('zoo')
library('cowplot')
library('ggfortify')
library('gridExtra')
library('latex2exp')
library('seasonal')
```

Instuctions

Do exercises 3.1, 3.2, 3.3, 3.4, 3.5, 3.7, 3.8 and 3.9 from the online Hyndman book. Please include your Rpubs link along with.pdf file of your run code

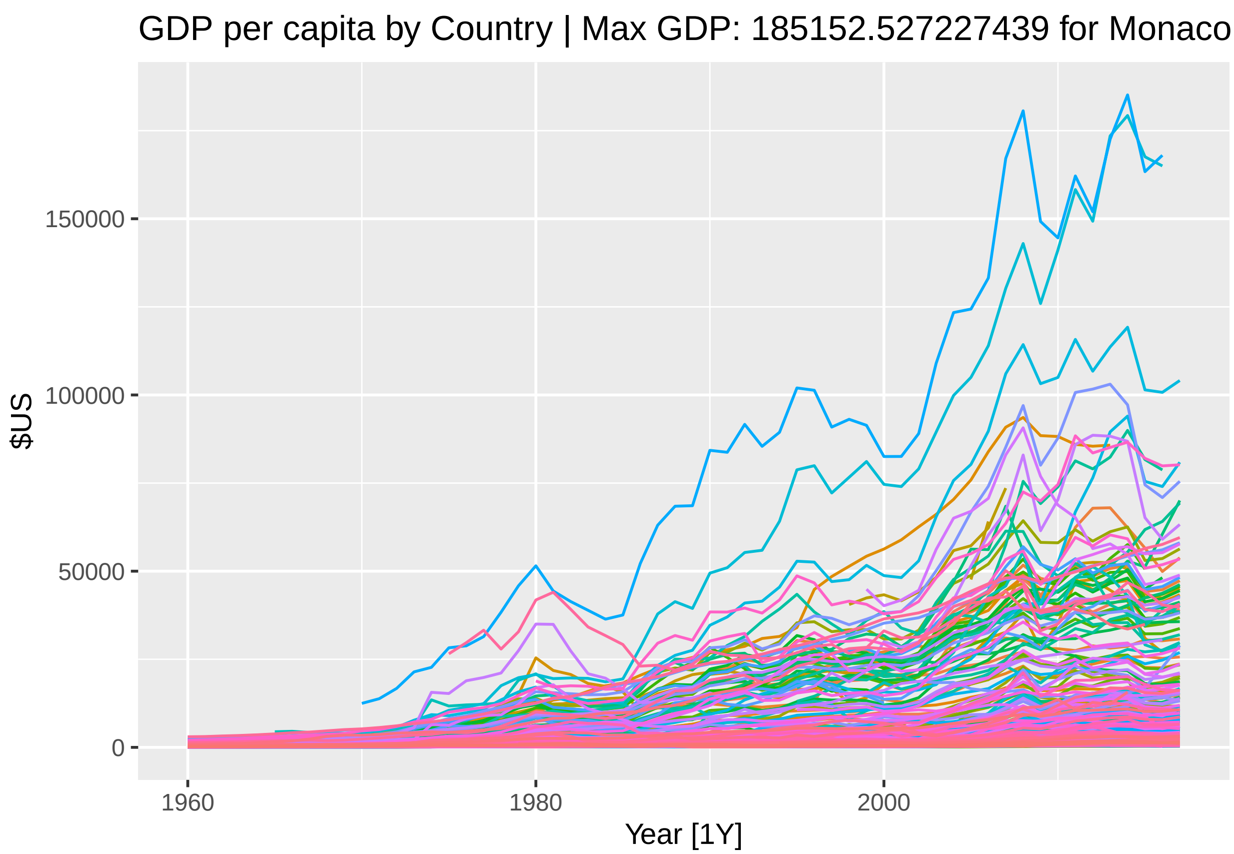
3.1

i

Consider the GDP information in `global_economy`. Plot the GDP per capita for each country over time. Which country has the highest GDP per capita?

```
# global_economy
```

```
# add GDP to the Data frame
global_economy <- global_economy %>%
  mutate(GDP_per_capita = GDP / Population)

# get the max values entire row for the title
max_gdp_row <- global_economy[which.max(global_economy$GDP_per_capita), ]


global_economy %>%
  autoplot(GDP_per_capita, show.legend = FALSE) +
  labs(title = paste("GDP per capita by Country | Max GDP:", max_gdp_row$GDP_per_capita, "for", max_gdp_
       y = "$US")
```




```
na_cnt<-sum(is.na(global_economy%>%
        filter(Country=="Monaco")%>%
          select(GDP_per_capita)))


paste("In the date range of 1960 to 2017 Monaco has had ",na_cnt," NA's")
```

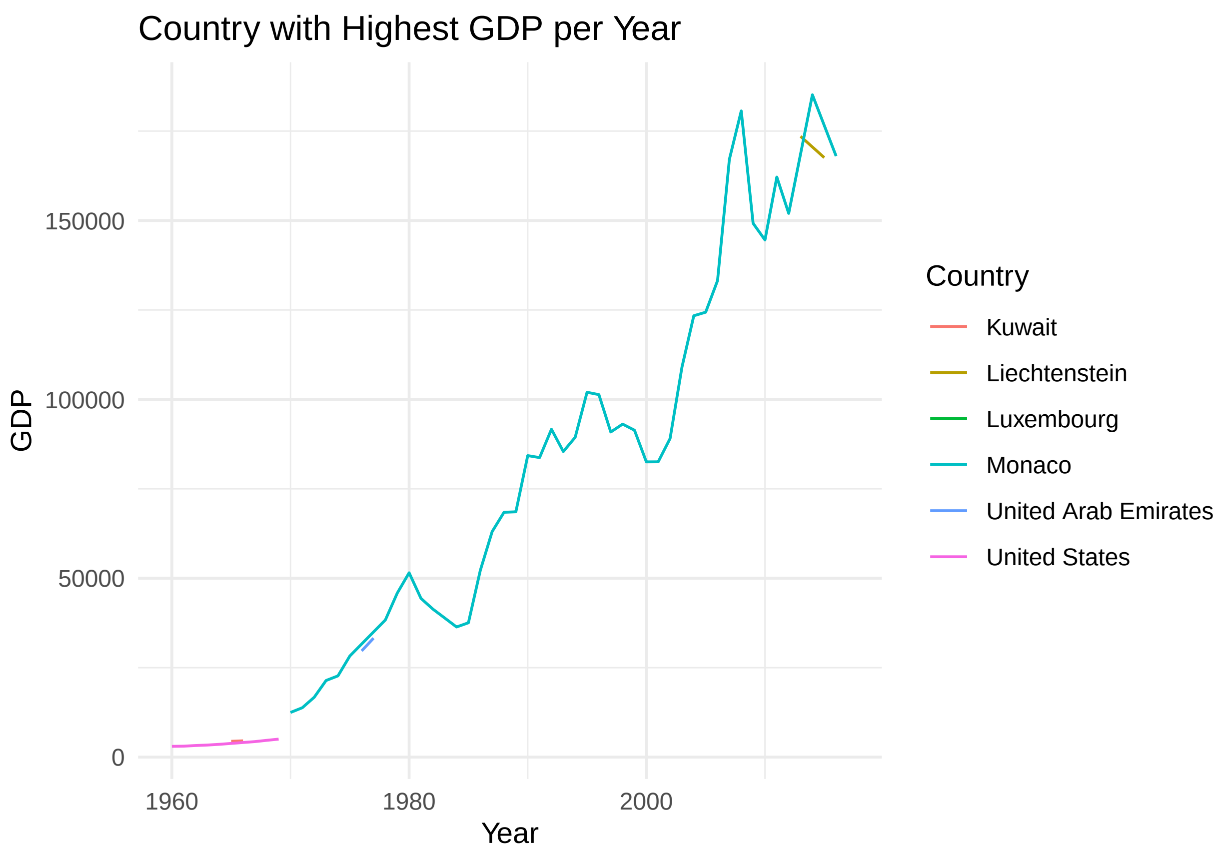
```
## [1] "In the date range of 1960 to 2017 Monaco has had  11  NA's"
```

ii

How has this changed over time?

```
global_economy <- index_by(global_economy, Year)

# store only rows with max gdp
max_gdp_annual <- global_economy %>%
  slice_max(GDP_per_capita) %>%
  ungroup()

# plot by rows selected
ggplot(max_gdp_annual, aes(x = Year, y = GDP_per_capita, color = Country)) +
  geom_line() +
  labs(title = "Country with Highest GDP per Year",
```

```r
      x = "Year",
      y = "GDP") +
  theme_minimal()
```



Looks like Luxembourg is using top in GDP making Monaco's 2014 GDP

3.2

For each of the following series, make a graph of the data. If transforming seems appropriate, do so and describe the effect.

i

United States `GDP from global_economy`.

Using GDP per Capita made the most sense. It was already transformed by the above manipulations.

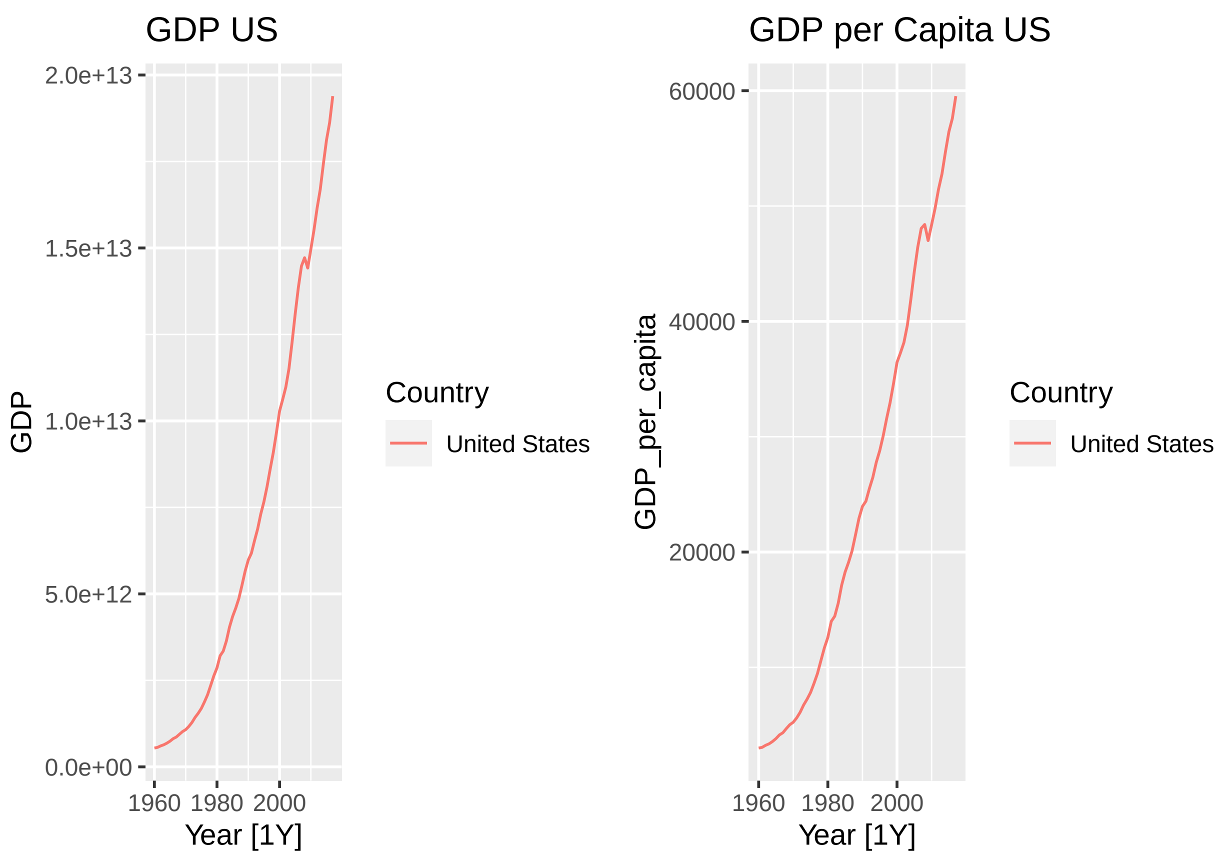
```r
fig1<-global_economy%>%
  filter(Country=="United States")%>%
    autoplot()+
  labs(title = "GDP US")
```

```
## Plot variable not specified, automatically selected `.vars = GDP`
## `mutate_if()` ignored the following grouping variables:
```

```
fig2<-global_economy%>%
  filter(Country=="United States")%>%
    autoplot(GDP_per_capita)+
  labs(title = "GDP per Capita US")
```

```
## `mutate_if()` ignored the following grouping variables:
## * Column `Year`
```

```
plot_grid(fig1,
          fig2, nrow = 1)
```



ii

Slaughter of Victorian "Bulls, bullocks and steers" in `aus_livestock`.

The data seemed clutter. By grouping it by Quarter the visual is clearer and the initial dip is apparent.

```
head(aus_livestock)
```

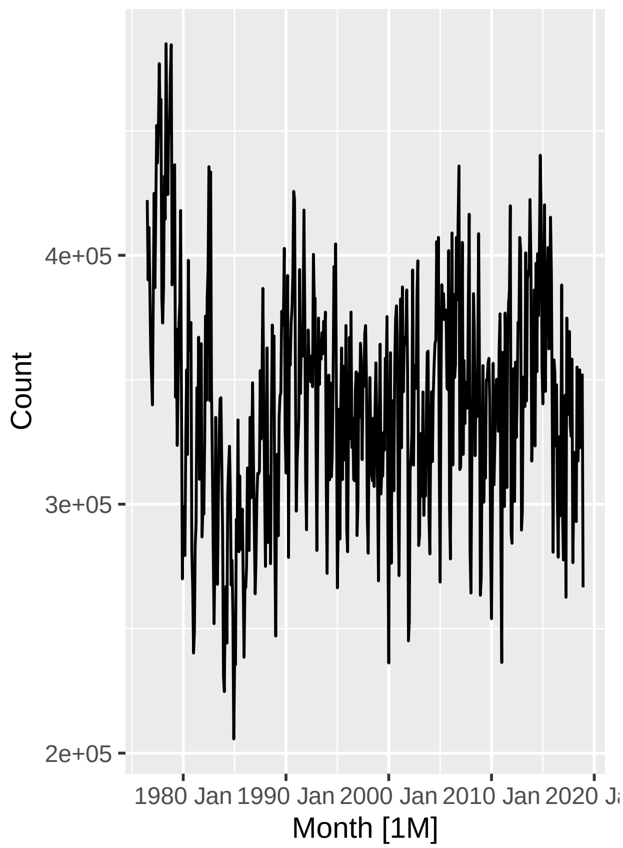
```
## # A tsibble: 6 x 4 [1M]
## # Key:       Animal, State [1]
##      Month Animal                    State                        Count
##      <mth> <fct>                     <fct>                        <dbl>
```

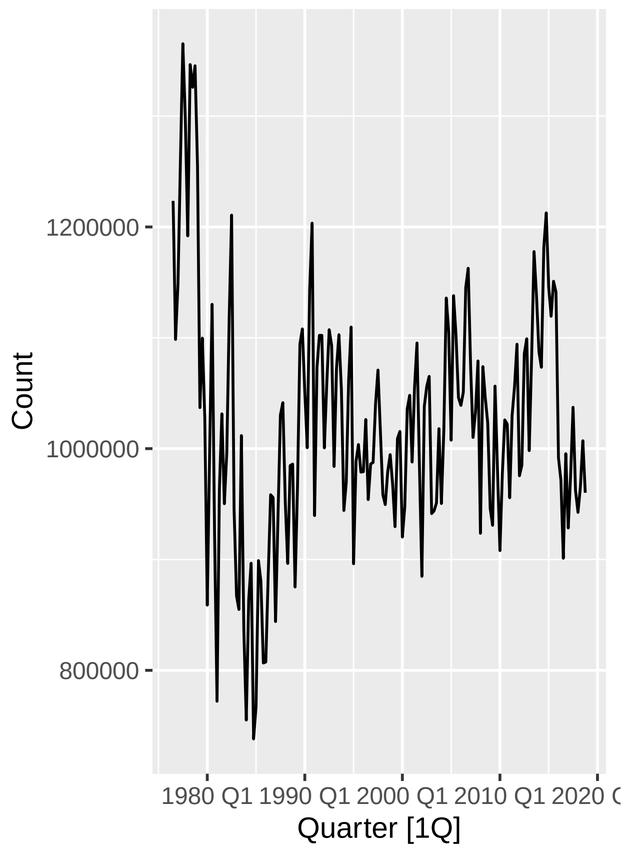
```
## 1 1976 Jul Bulls, bullocks and steers Australian Capital Territory  2300
## 2 1976 Aug Bulls, bullocks and steers Australian Capital Territory  2100
## 3 1976 Sep Bulls, bullocks and steers Australian Capital Territory  2100
## 4 1976 Oct Bulls, bullocks and steers Australian Capital Territory  1900
## 5 1976 Nov Bulls, bullocks and steers Australian Capital Territory  2100
## 6 1976 Dec Bulls, bullocks and steers Australian Capital Territory  1800
```

```
fig3 <- aus_livestock %>%
  filter(Animal == "Bulls, bullocks and steers") %>%
  summarise(Count = sum(Count)) %>%
  autoplot(show.legend=FALSE)
```

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

```
fig4 <- aus_livestock %>%
  filter(Animal == "Bulls, bullocks and steers") %>%
  mutate(Quarter = yearquarter(Month)) %>%
  index_by(Quarter) %>%
  summarise(Count = sum(Count)) %>%
  autoplot(Count, show.legend=FALSE)

plot_grid(fig3, fig4, nrow = 1)
```





iii

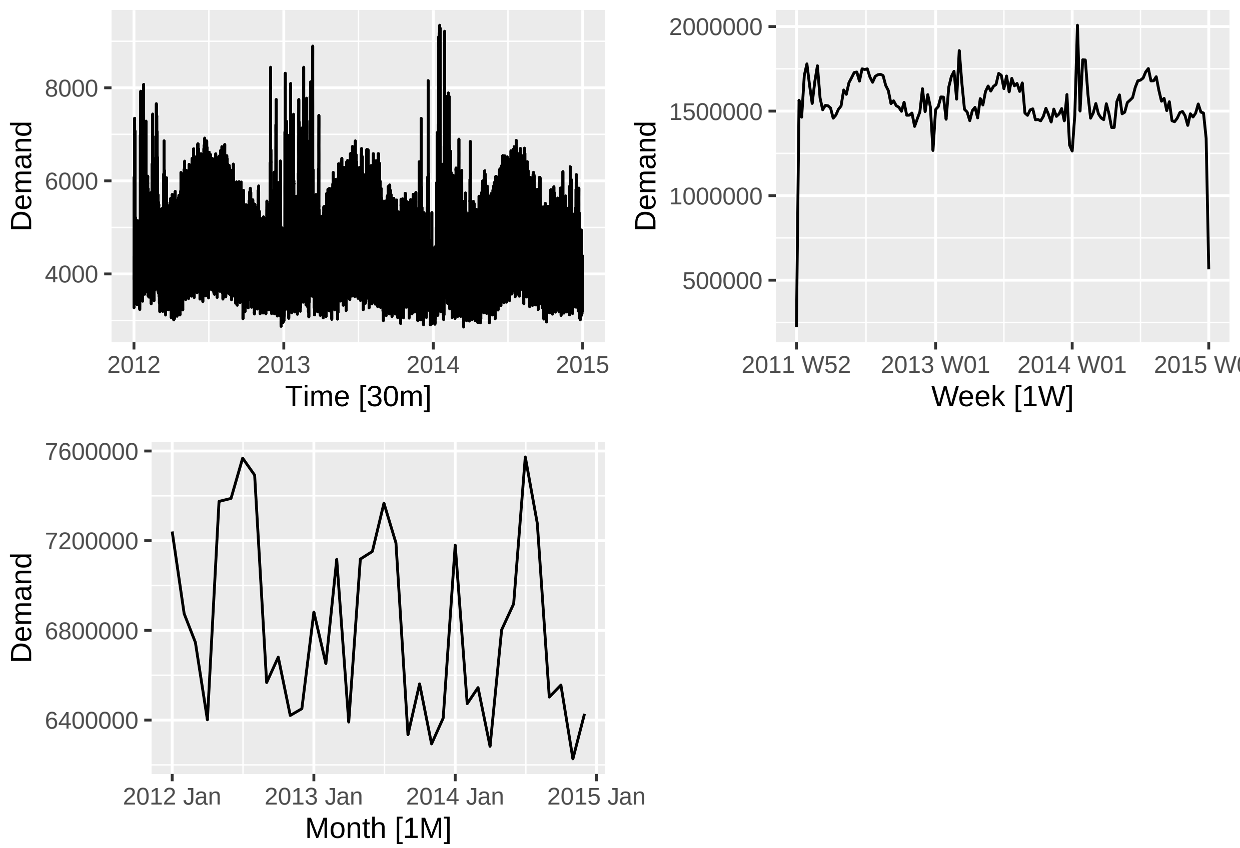
Victorian Electricity Demand from `vic_elec`.

This was a lot of data so dealers choice, but viewing the annual data in a weekly and monthly basis shows an initial spike the first week and month for the year.

```
head(vic_elec)
```

```
## # A tsibble: 6 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
```

```
fig5<-vic_elec%>%
        autoplot(Demand)

fig6<-vic_elec%>%
  mutate(Week = yearweek(Time)) %>%
  index_by(Week) %>%
  summarise(Demand = sum(Demand)) %>%
  autoplot(Demand, show.legend=FALSE)

fig7<-vic_elec%>%
  mutate(Month = yearmonth(Time)) %>%
  index_by(Month) %>%
  summarise(Demand = sum(Demand)) %>%
  autoplot(Demand, show.legend=FALSE)

# fig7<-vic_elec%>%
#   mutate(Day = as.Date(Time)) %>%
#   group_by(Day) %>%
#   summarise(Demand = sum(Demand)) %>%
#   autoplot(Demand, show.legend=FALSE)

plot_grid(fig5,
          fig6,
          fig7, nrow = 2)
```



iv

Gas production from `aus_production`.

```r
head(aus_production)
```

```
## # A tsibble: 6 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
## 6 1957 Q2   228    5651    214    604        4811     7
```

I played with all methods introduced in mathematical transformations considering the high variability between the start and end of the data, in an attempt to remove variability.

Than I applied *Box-Cox transformations:*

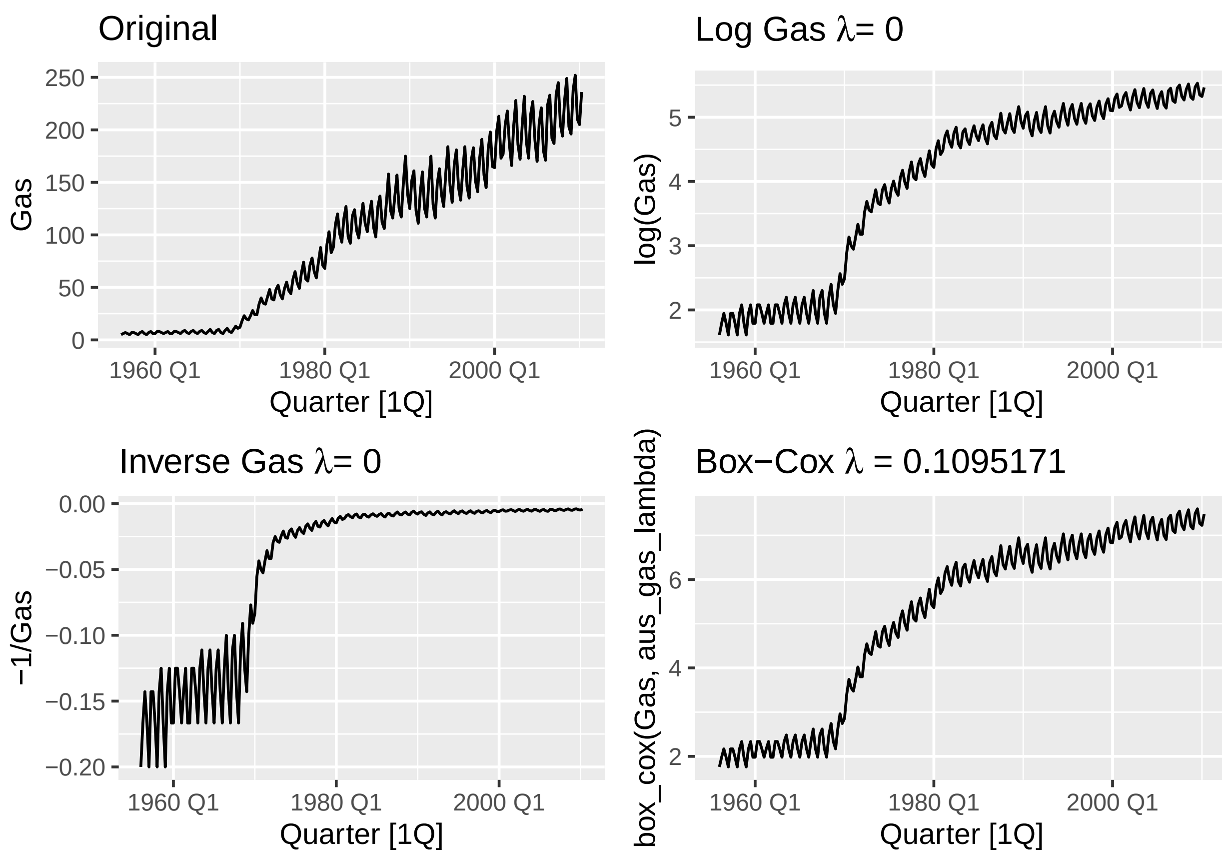
$$w_t = \begin{cases} log(y_t), & \lambda=0 \\ (sign(y_t)|y_t|^{\lambda}-1)/\lambda, & \lambda\neq0 \end{cases}$$

as explained in 3.1 video for transformations

```r
#example
# food |>
#   features(Turnover, features = guerrero())

(aus_gas_lambda<-aus_production%>%
  features(Gas, features = guerrero)%>%
   pull(lambda_guerrero))
```

```
## [1] 0.1095171
```

```r
fig8 <- aus_production %>%
  autoplot(Gas) +
  labs(title = "Original")

# Define fig9
fig9 <- aus_production %>%
  autoplot(log(Gas)) +
  labs(title = expression(paste("Log Gas ", lambda, "= 0")))

# Define fig10
fig10 <- aus_production %>%
  autoplot(-1/Gas) +
  labs(title = "Inverse Gas")

fig11 <- aus_production%>%
          autoplot(box_cox(Gas,aus_gas_lambda))

plot_grid(fig8 +
            labs(title = "Original"),
          fig9 +
            labs(title = expression(paste("Log Gas ",
                                          lambda, "= 0"))),
          fig10 +
            labs(title = expression(paste("Inverse Gas ",
                                          lambda, "= 0"))),
          fig11 +
            labs(title = expression(paste("Box-Cox ",
                                          lambda,
                                          " = 0.1095171"))),
          nrow = 2)
```



3.3

Why is a Box-Cox transformation unhelpful for the `canadian_gas` data?

as per the 3.1 Video for transformations.

A low value of λ can give extremely large prediction intervals and as we see in the plot below does not do much for transformation.

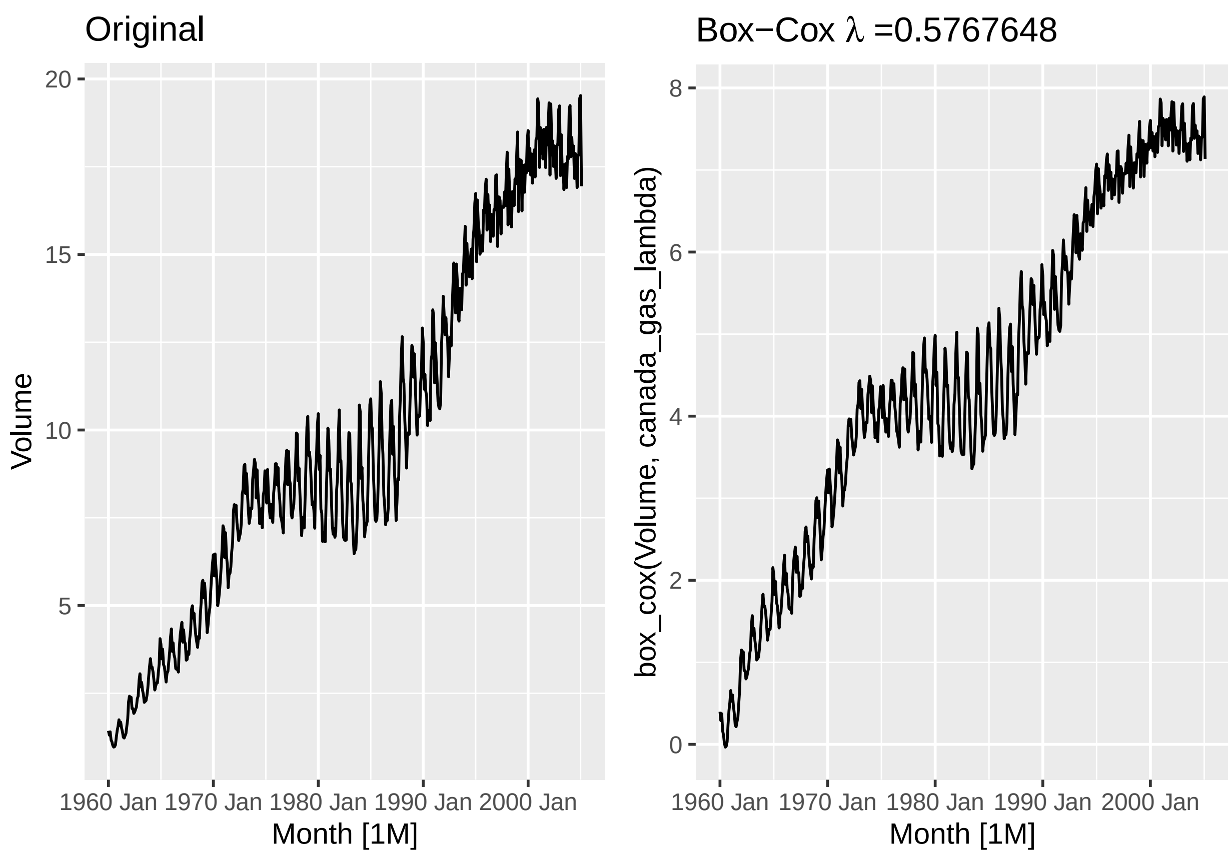
```
head(canadian_gas)
```

```
## # A tsibble: 6 x 2 [1M]
##      Month Volume
##      <mth>  <dbl>
## 1 1960 Jan   1.43
## 2 1960 Feb   1.31
## 3 1960 Mar   1.40
## 4 1960 Apr   1.17
## 5 1960 May   1.12
## 6 1960 Jun   1.01
```

```
(canada_gas_lambda<-canadian_gas%>%
    features(Volume, features = guerrero)%>%
    pull(lambda_guerrero)
)
```

```
## [1] 0.5767648
```

```r
fig12<-canadian_gas%>%
  autoplot(Volume)

fig13<-canadian_gas%>%
  autoplot(box_cox(Volume, canada_gas_lambda))

plot_grid(fig12+labs(title = "Original"),
          fig13+labs(title = expression(paste("Box-Cox ",
                                              lambda,
                                              " =0.5767648"))))
```




```r
rm(list = ls(pattern = "^fig"))
```

3.4

What Box-Cox transformation would you select for your retail data (from Exercise 7 in Section 2.10)?

I would rely on `features=guerrero` since its designed to be the best fit.

```r
# data provided
set.seed(123)
myseries <- aus_retail |>
```

```
  filter(`Series ID` == sample(aus_retail$`Series ID`,1))
head(myseries)
```

```
## # A tsibble: 6 x 5 [1M]
## # Key:       State, Industry [1]
##   State    Industry                  `Series ID`    Month Turnover
##   <chr>    <chr>                     <chr>          <mth>    <dbl>
## 1 Victoria Household goods retailing A3349643V   1982 Apr     173.
## 2 Victoria Household goods retailing A3349643V   1982 May     180.
## 3 Victoria Household goods retailing A3349643V   1982 Jun     167.
## 4 Victoria Household goods retailing A3349643V   1982 Jul     174.
## 5 Victoria Household goods retailing A3349643V   1982 Aug     178.
## 6 Victoria Household goods retailing A3349643V   1982 Sep     180.
```

```
(myseries_lambda <- myseries%>%
        features(Turnover,features = guerrero)%>%
          pull(lambda_guerrero))
```

```
## [1] 0.2151641
```

```
fig1 <- myseries%>%
          autoplot(Turnover)

fig2 <- myseries%>%
        autoplot(box_cox(Turnover,myseries_lambda))
fig2<-fig2+ylab("Turnover")

plot_grid(fig1+labs(title = "Original"),
          fig2+labs(title = paste("Box-cox   =",myseries_lambda)), nrow=2)
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox   = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <ce>
```

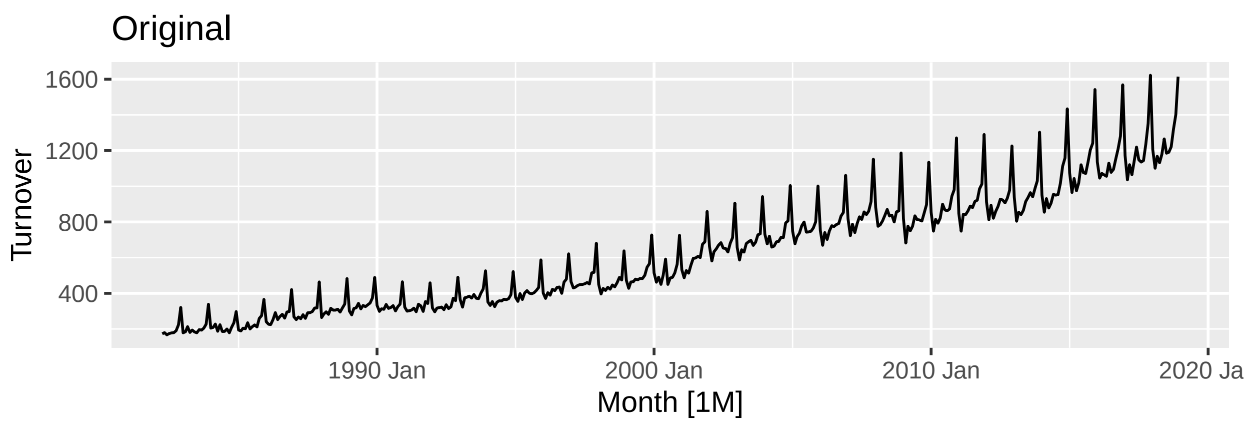
```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox   = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <bb>
```

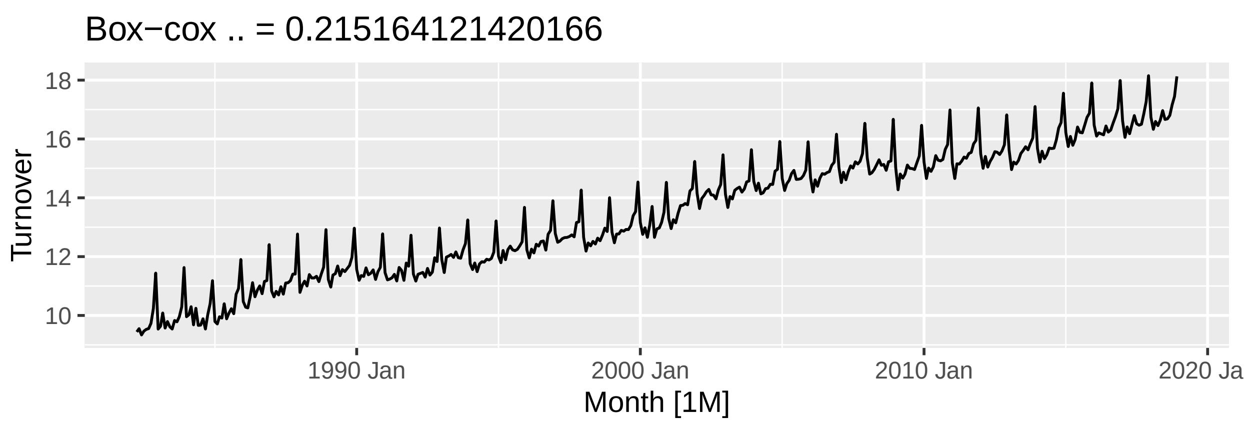
```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox   = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <ce>
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox   = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <bb>
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox   = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <ce>
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox  = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <bb>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox  = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox  = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <bb>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox  = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox  = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <bb>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox  = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox  = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <bb>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox  = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox  = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <bb>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox  = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <ce>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Box-cox  = 0.215164121420166' in 'mbcsToSbcs': dot
## substituted for <bb>
```





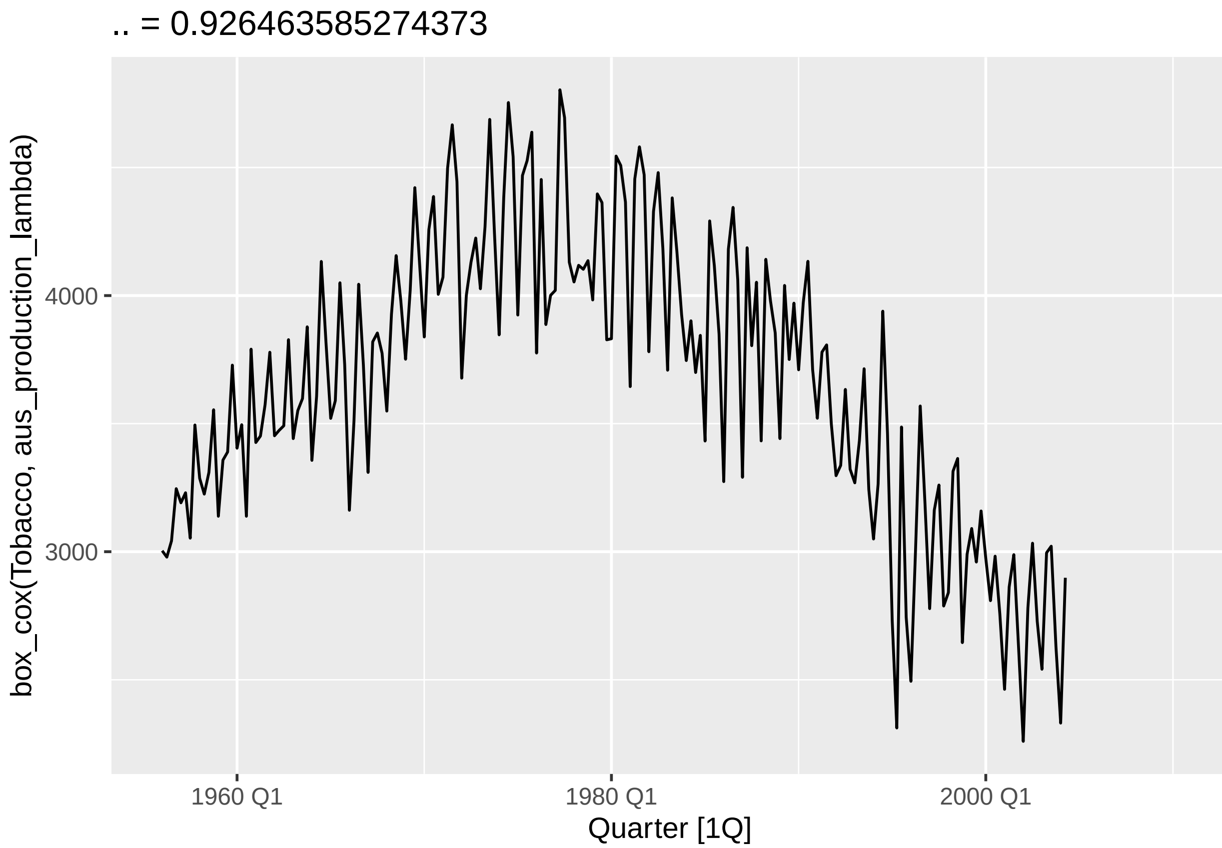
3.5

For the following series, find an appropriate Box-Cox transformation in order to stabilize the variance.

i

Tobacco from `aus_production`

```
aus_production_lambda<-aus_production%>%
  features(Tobacco,features=guerrero)%>%
  pull(lambda_guerrero)

aus_production%>%
  autoplot(box_cox(Tobacco,aus_production_lambda))+
  labs(title =paste("  =",aus_production_lambda),ylab="" )
```



ii

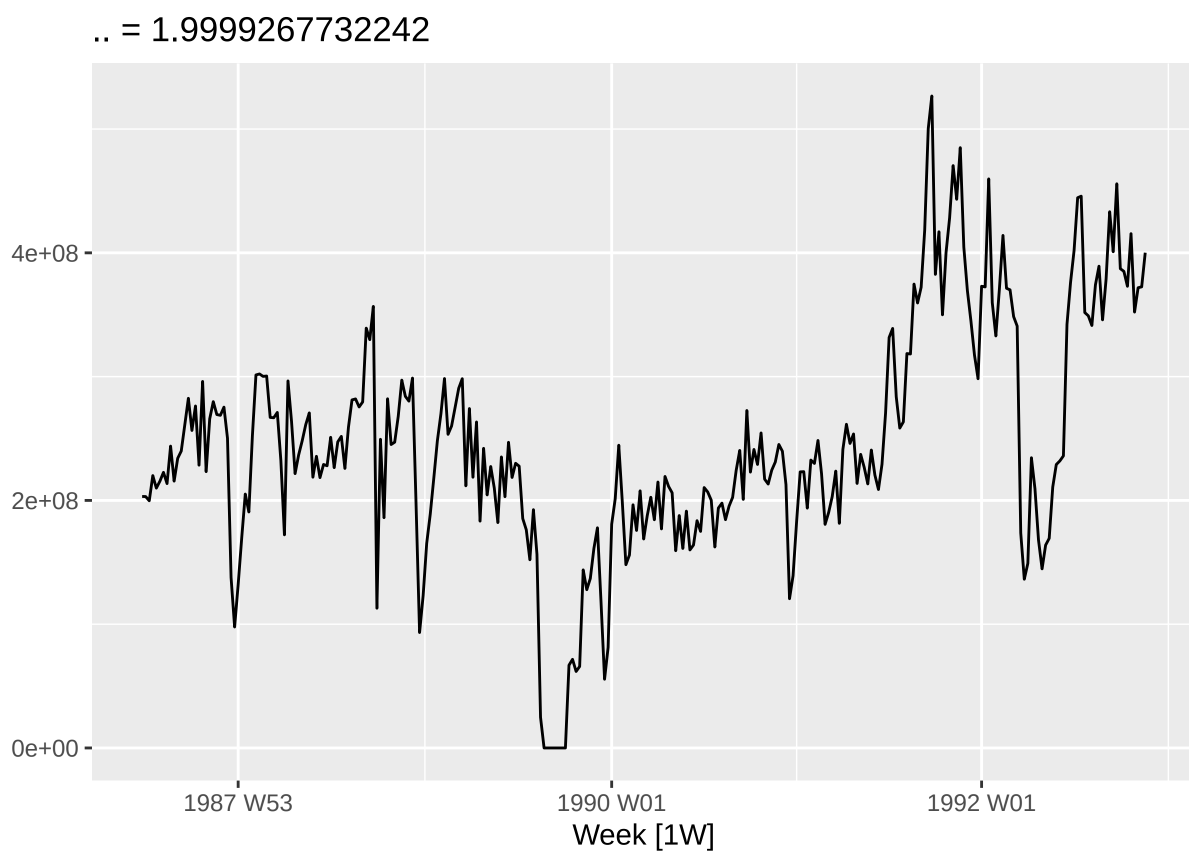
Economy class passengers between Melbourne and Sydney from `ansett`

```
head(ansett)
```

```
## # A tsibble: 6 x 4 [1W]
## # Key:       Airports, Class [1]
##        Week Airports Class    Passengers
##      <week> <chr>    <chr>         <dbl>
## 1 1989 W28 ADL-PER  Business        193
## 2 1989 W29 ADL-PER  Business        254
## 3 1989 W30 ADL-PER  Business        185
## 4 1989 W31 ADL-PER  Business        254
## 5 1989 W32 ADL-PER  Business        191
## 6 1989 W33 ADL-PER  Business        136
```

```
ansett_lambda<-ansett%>%
  filter(Class=="Economy"& Airports=='MEL-SYD')%>%
  features(Passengers,features=guerrero)%>%
  pull(lambda_guerrero)

ansett%>%
```

```r
  filter(Class=="Economy"& Airports=='MEL-SYD')%>%
  autoplot(box_cox(Passengers,ansett_lambda))+
  labs(title =paste("  =",ansett_lambda), y=" " )
```



iii

Pedestrian counts at Southern Cross Station from `pedestrian`.

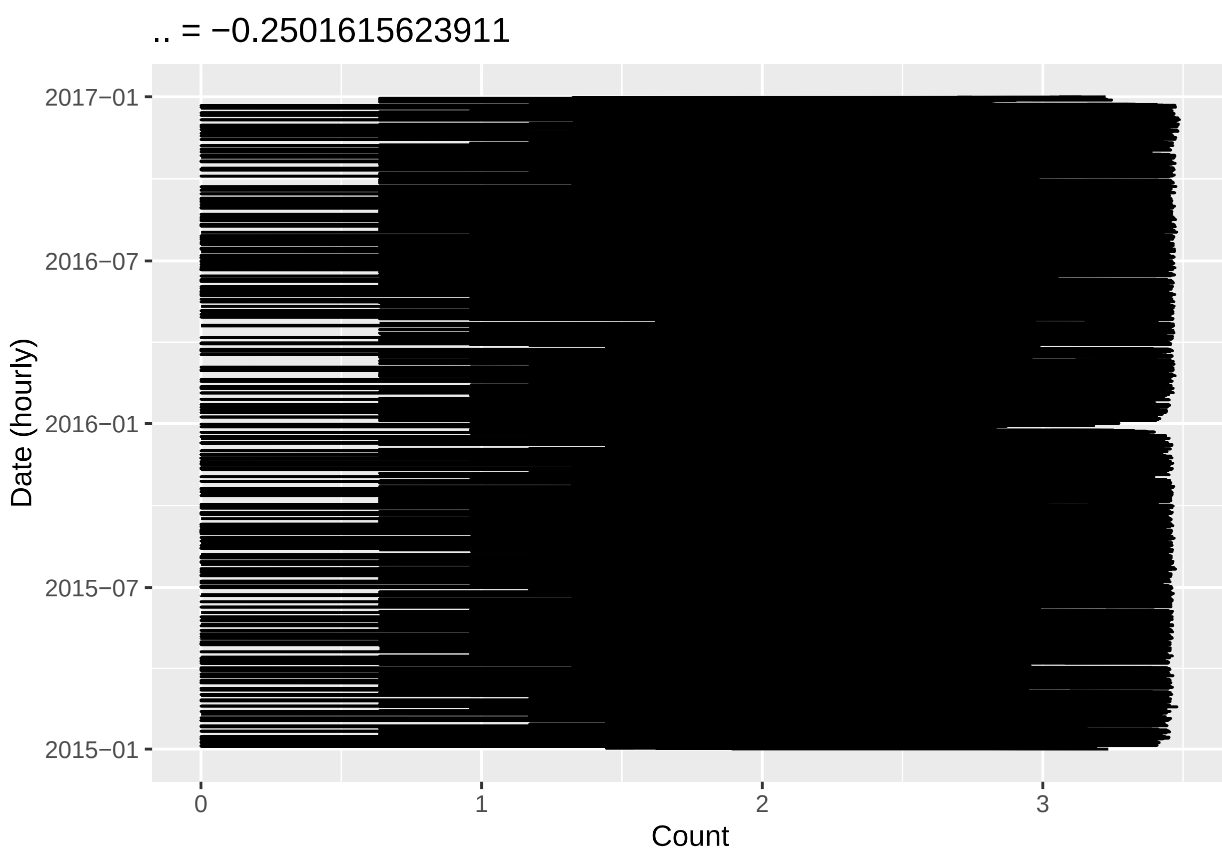
```r
head(pedestrian)
```

```
## # A tsibble: 6 x 5 [1h] <Australia/Melbourne>
## # Key:       Sensor [1]
##   Sensor         Date_Time           Date        Time Count
##   <chr>          <dttm>              <date>     <int> <int>
## 1 Birrarung Marr 2015-01-01 00:00:00 2015-01-01     0  1630
## 2 Birrarung Marr 2015-01-01 01:00:00 2015-01-01     1   826
## 3 Birrarung Marr 2015-01-01 02:00:00 2015-01-01     2   567
## 4 Birrarung Marr 2015-01-01 03:00:00 2015-01-01     3   264
## 5 Birrarung Marr 2015-01-01 04:00:00 2015-01-01     4   139
## 6 Birrarung Marr 2015-01-01 05:00:00 2015-01-01     5    77
```

```r
pedestrian_lambda<-pedestrian%>%
  filter(Sensor=="Southern Cross Station")%>%
  features(Count,features=guerrero)%>%
  pull(lambda_guerrero)

pedestrian%>%
  filter(Sensor=="Southern Cross Station")%>%
  autoplot(box_cox(Count,pedestrian_lambda))+
  coord_flip()+
  labs(title =paste("  =",pedestrian_lambda),x = "Date (hourly)",
       y="Count" )
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '  = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '  = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <bb>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '  = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '  = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <bb>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '  = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '  = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <bb>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '  = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '  = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <bb>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '  = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '  = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <bb>
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <bb>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <bb>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <ce>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' = -0.2501615623911' in 'mbcsToSbcs': dot substituted
## for <bb>
```



3.7

Consider the last five years of the Gas data from `aus_production`.

```
gas <- tail(aus_production, 5*4) |> select(Gas)

head(gas)
```

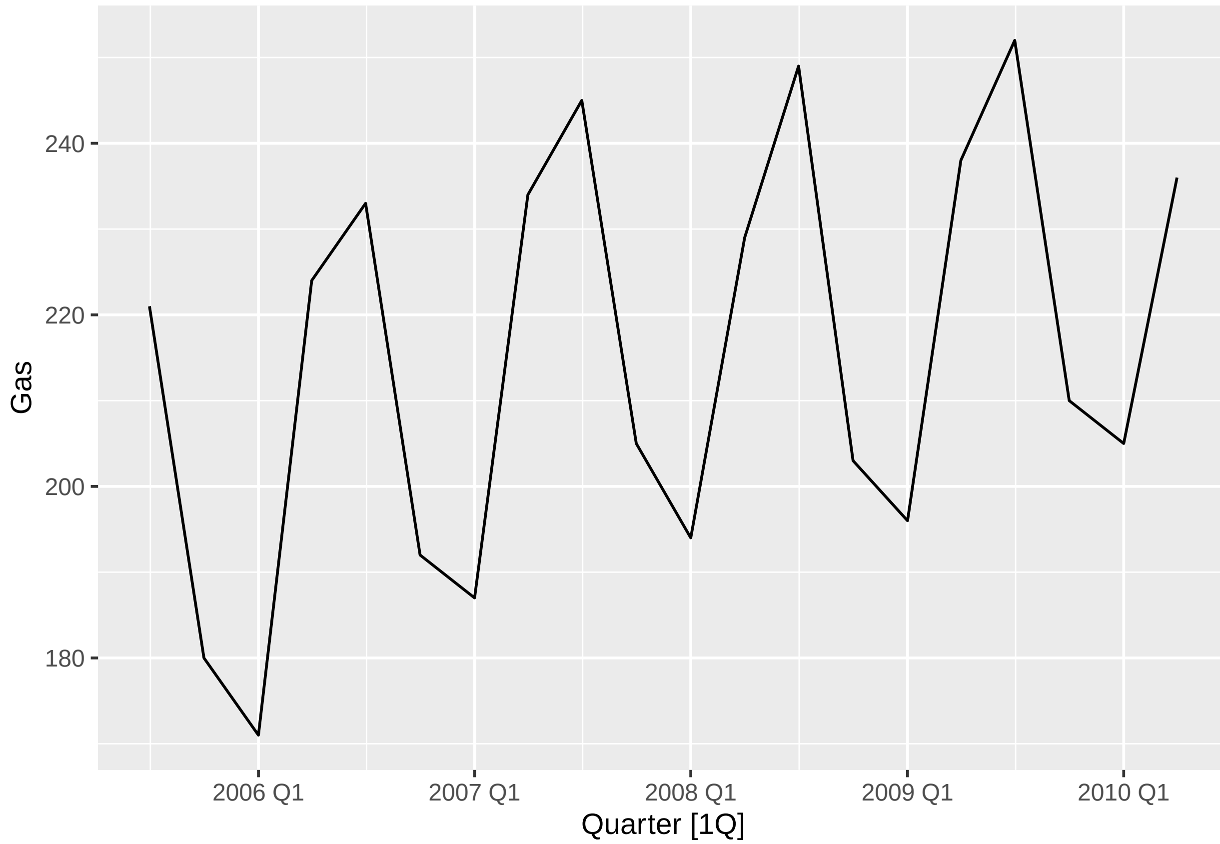
```
## # A tsibble: 6 x 2 [1Q]
##     Gas Quarter
##   <dbl>   <qtr>
## 1   221 2005 Q3
## 2   180 2005 Q4
## 3   171 2006 Q1
## 4   224 2006 Q2
## 5   233 2006 Q3
## 6   192 2006 Q4
```

a.

Plot the time series. Can you identify seasonal fluctuations and/or a trend-cycle?

Looks like the trend from 2006-2010 is upwards. Seasonality is the high every 3rd quarter and low every 1rst Quarter

```
gas%>%
  autoplot(Gas)
```



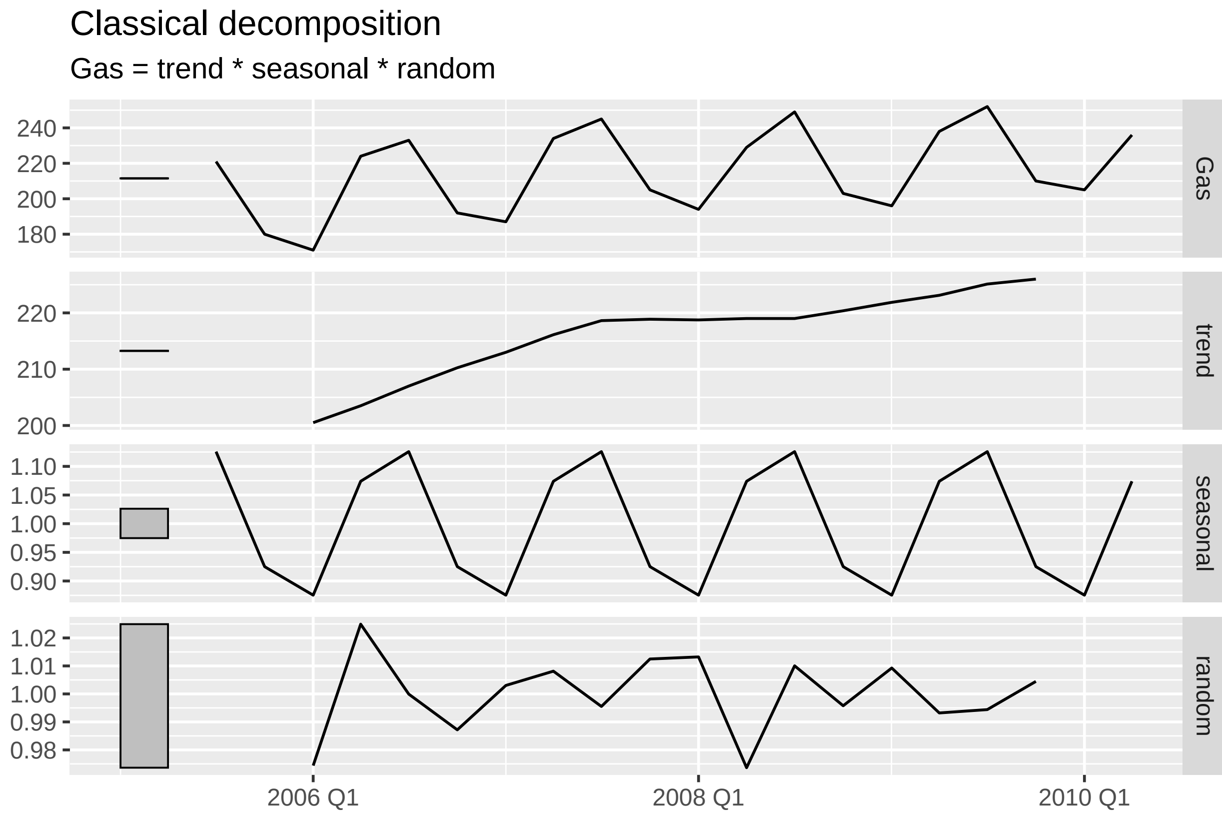
b.

Use `classical_decomposition` with `type=multiplicative` to calculate the trend-cycle and seasonal indices.

Ref 3.4 video

```
#example
# us_retail_employment |>
#   model(classical_decomposition(Employed, type = "additive")) |>
#   components()|>
#   autoplot()+xlab("Year")+
#   ggtitle("Classical additive decomposition of total US retail employment")
```

```
gas%>%
  model(classical_decomposition(Gas,type= "multiplicative"))%>%
  components()%>%
  autoplot+xlab("")
```



c.

Do the results support the graphical interpretation from part a?

The results show a positive trend with quarterly seasonality so yes it does.

d.

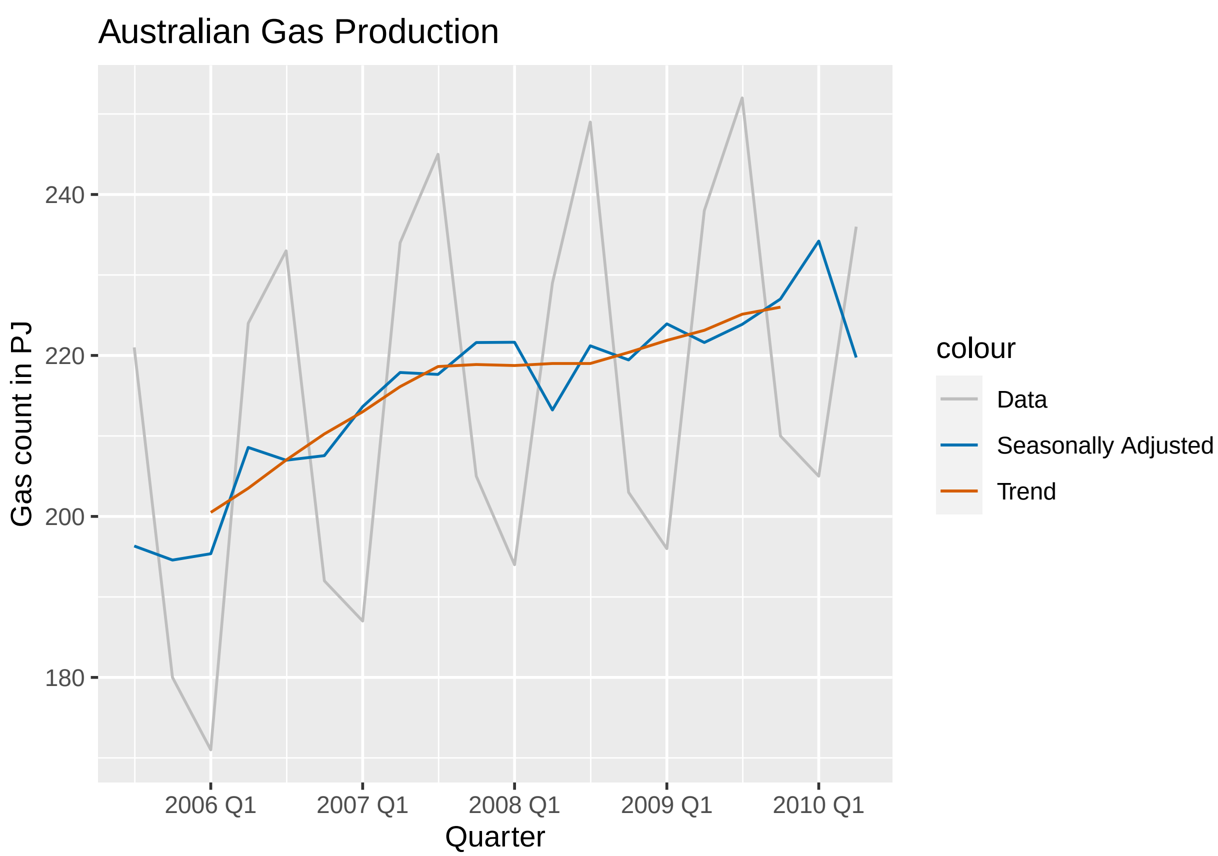
Compute and plot the seasonally adjusted data.

As shown in 3.5

```
#Example
# x11_dcmp |>
#   ggplot(aes(x = Month)) +
#   geom_line(aes(y = Employed, colour = "Data")) +
#   geom_line(aes(y = season_adjust,
#               colour = "Seasonally Adjusted")) +
#   geom_line(aes(y = trend, colour = "Trend")) +
#   labs(y = "Persons (thousands)",
#        title = "Total employment in US retail") +
#   scale_colour_manual(
#     values = c("gray", "#0072B2", "#D55E00"),
```

```
#      breaks = c("Data", "Seasonally Adjusted", "Trend")
#    )
```

```
gas%>%
  model(classical_decomposition(Gas,type= "multiplicative"))%>%
  components()%>%
  ggplot(aes(x = Quarter)) +
  geom_line(aes(y = Gas, colour = "Data")) +
  geom_line(aes(y = season_adjust,
                colour = "Seasonally Adjusted")) +
  geom_line(aes(y = trend, colour = "Trend")) +
  labs(y = "Gas count in PJ",
       title = "Australian Gas Production") +
  scale_colour_manual(
    values = c("gray", "#0072B2", "#D55E00"),
    breaks = c("Data", "Seasonally Adjusted", "Trend")
  )
```



e.

Change one observation to be an outlier (e.g., add 300 to one observation), and recompute the seasonally adjusted data. What is the effect of the outlier?

```r
rm(list = ls(pattern = "^fig"))
```

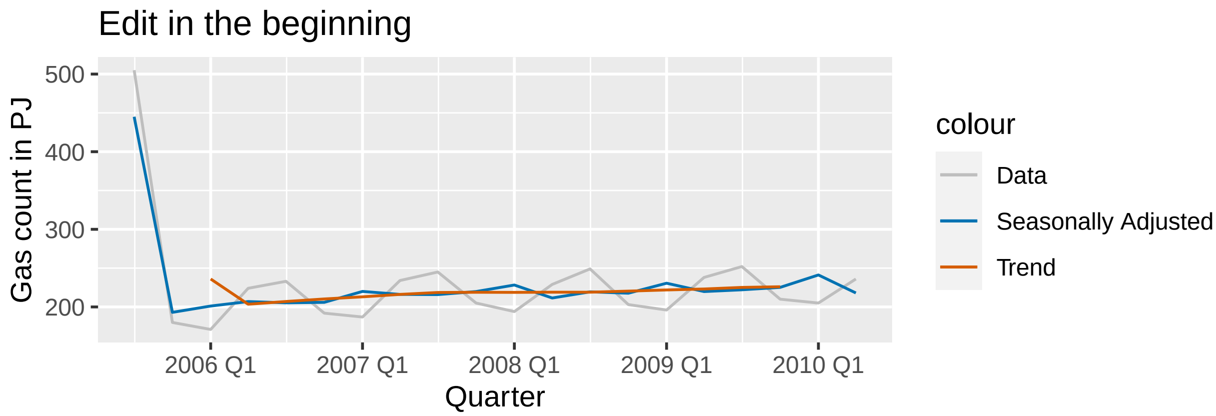
```r
print(nrow(gas))
```

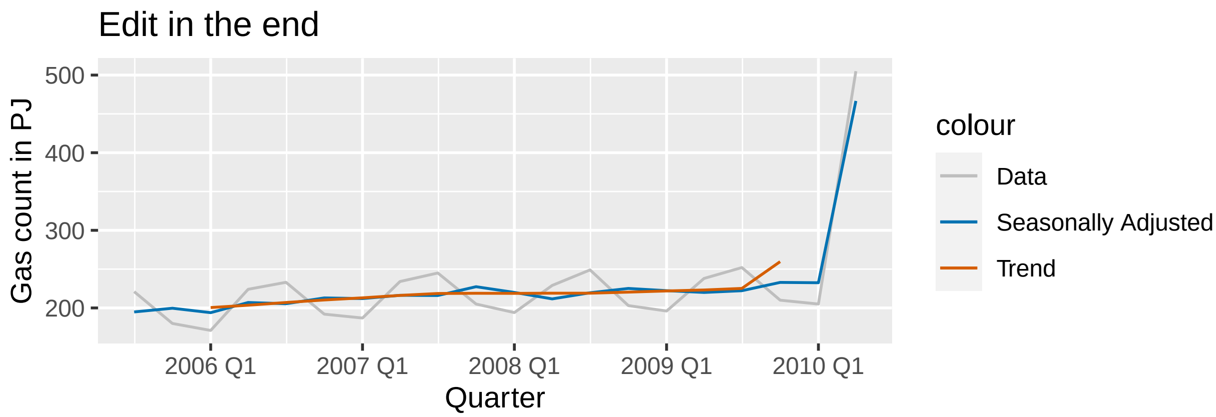
```
## [1] 20
```

```r
gas_begin_edit <- gas
gas_end_edit <- gas
gas_begin_edit$Gas[1] <- gas_begin_edit$Gas[10] + 300
gas_end_edit$Gas[20] <- gas_begin_edit$Gas[10] + 300
```

```r
fig1<-gas_begin_edit%>%
  model(classical_decomposition(Gas,type= "multiplicative"))%>%
  components()%>%
  ggplot(aes(x = Quarter)) +
  geom_line(aes(y = Gas, colour = "Data")) +
  geom_line(aes(y = season_adjust,
                colour = "Seasonally Adjusted")) +
  geom_line(aes(y = trend, colour = "Trend")) +
  labs(y = "Gas count in PJ",
       title = "Australian Gas Production") +
  scale_colour_manual(
    values = c("gray", "#0072B2", "#D55E00"),
    breaks = c("Data", "Seasonally Adjusted", "Trend")
  )

fig2<-gas_end_edit%>%
  model(classical_decomposition(Gas,type= "multiplicative"))%>%
  components()%>%
  ggplot(aes(x = Quarter)) +
  geom_line(aes(y = Gas, colour = "Data")) +
  geom_line(aes(y = season_adjust,
                colour = "Seasonally Adjusted")) +
  geom_line(aes(y = trend, colour = "Trend")) +
  labs(y = "Gas count in PJ",
       title = "Australian Gas Production") +
  scale_colour_manual(
    values = c("gray", "#0072B2", "#D55E00"),
    breaks = c("Data", "Seasonally Adjusted", "Trend")
  )

plot_grid(fig1+labs(title = "Edit in the beginning"),
          fig2+labs(title = "Edit in the end"), nrow=2)
```

```
## Warning: Removed 4 rows containing missing values (`geom_line()`).
## Removed 4 rows containing missing values (`geom_line()`).
```





It spikes the data

f.

Does it make any difference if the outlier is near the end rather than in the middle of the time series?

Just placement, but the effect is the same

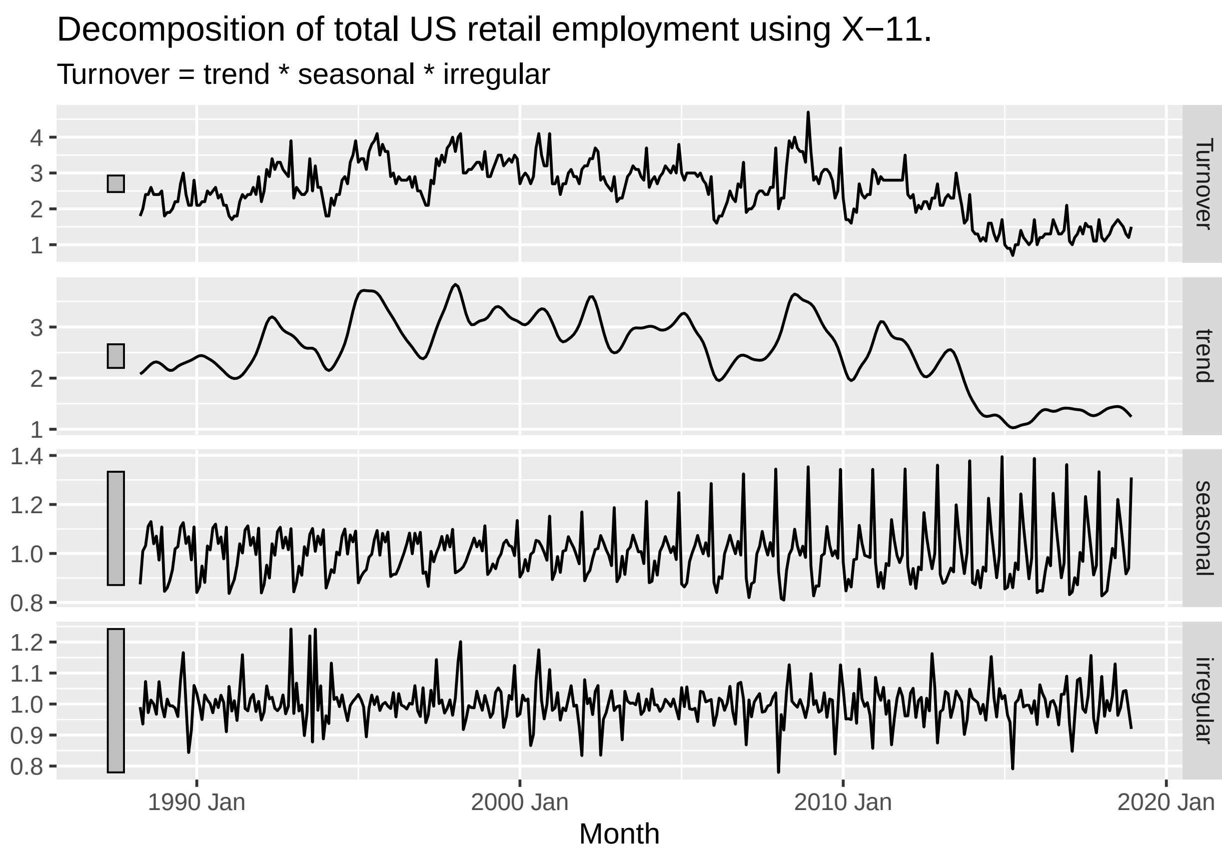
3.8

Recall your retail time series data (from Exercise 7 in Section 2.10). Decompose the series using X-11. Does it reveal any outliers, or unusual features that you had not noticed previously?

As per 3.5 Methods used by official statistics agencies

```
#Example
# x11_dcmp <- us_retail_employment |>
#   model(x11 = X_13ARIMA_SEATS(Employed ~ x11())) |>
#   components()
# autoplot(x11_dcmp) +
#   labs(title =
#     "Decomposition of total US retail employment using X-11.")
```

```r
set.seed(241)

myseries <- aus_retail %>%
  filter(`Series ID` == sample(aus_retail$`Series ID`,1))

x11_dcmp <- myseries %>%
  model(x11 = X_13ARIMA_SEATS(Turnover ~ x11())) %>%
  components()

autoplot(x11_dcmp) +
  labs(title =
    "Decomposition of total US retail employment using X-11.")
```



**I noted a long-term downward trend and greater volatility with the seasonal spikes

3.9

Figures 3.19 and 3.20 show the result of decomposing the number of persons in the civilian labour force in Australia each month from February 1978 to August 1995.

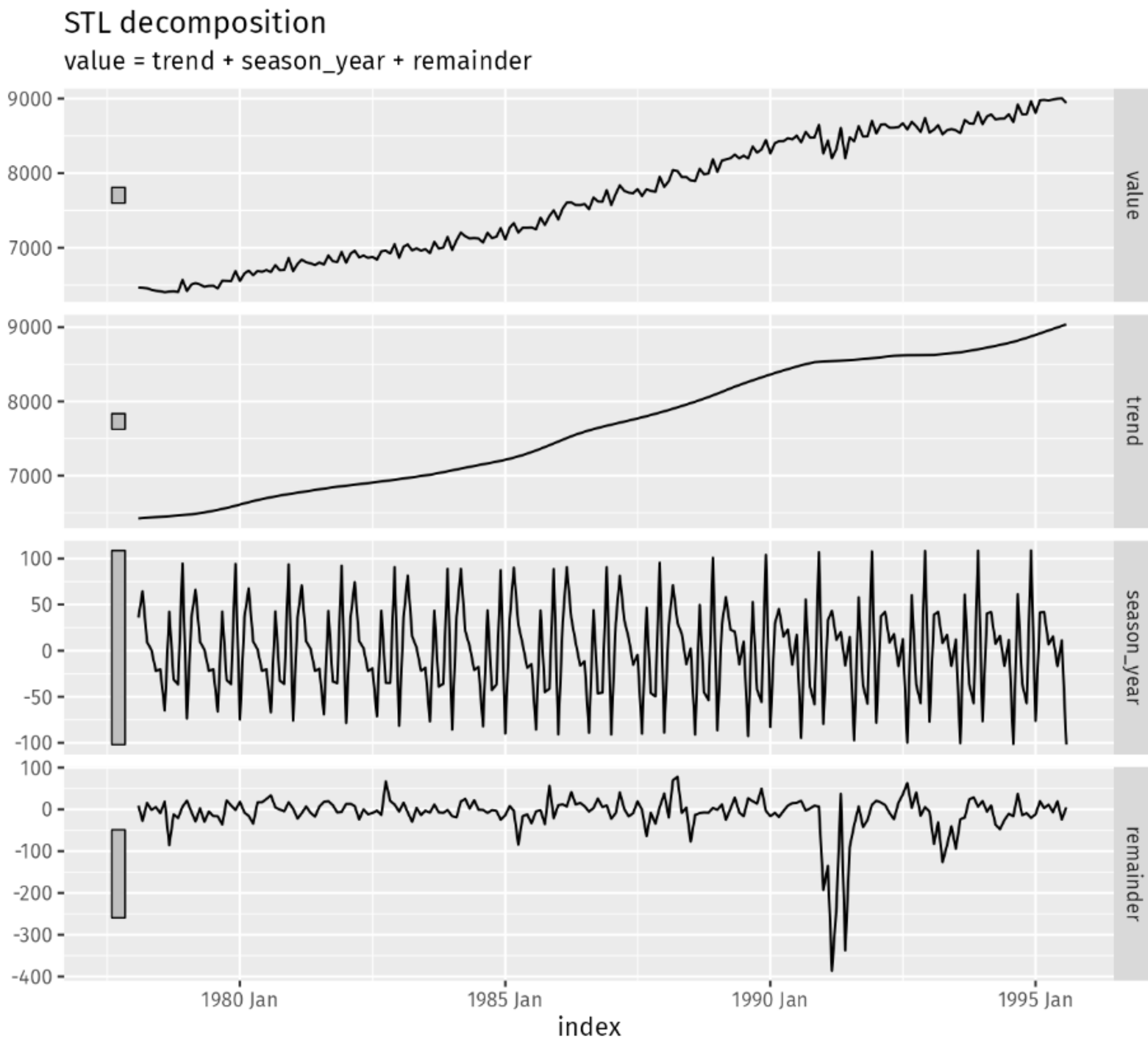


Figure 3.19: Decomposition of the number of persons in the civilian labour force in Australia each month from February 1978 to August 1995.

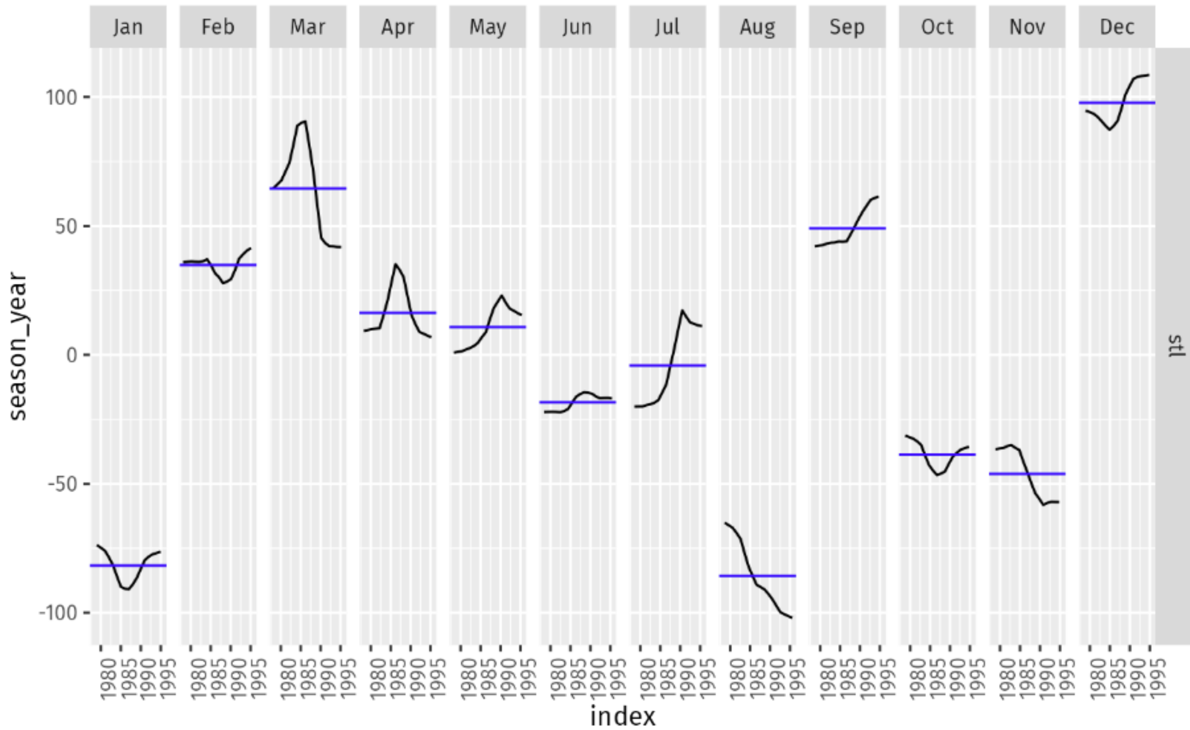


Figure 3.20: Seasonal component from the decomposition shown in the previous figure.

a.

Write about 3–5 sentences describing the results of the decomposition. Pay particular attention to the scales of the graphs in making your interpretation.

The trend is clearly a upward trend from 1978-1995, and what appears to be some seasonality. This makes sense to me as the spikes are common, with students leaving school for the holiday and summer, and seasonal work occurring throughout the year. I'm curious what caused the dip in 1992 which is a pretty clear outlier.

b.

Is the recession of 1991/1992 visible in the estimated components?

Very much. Its easily observed in the "remainder" plot.