Data 524 Assignment 2

Warner Alexis

2025-02-16

Time Series Decomposition

Excercise 3.1

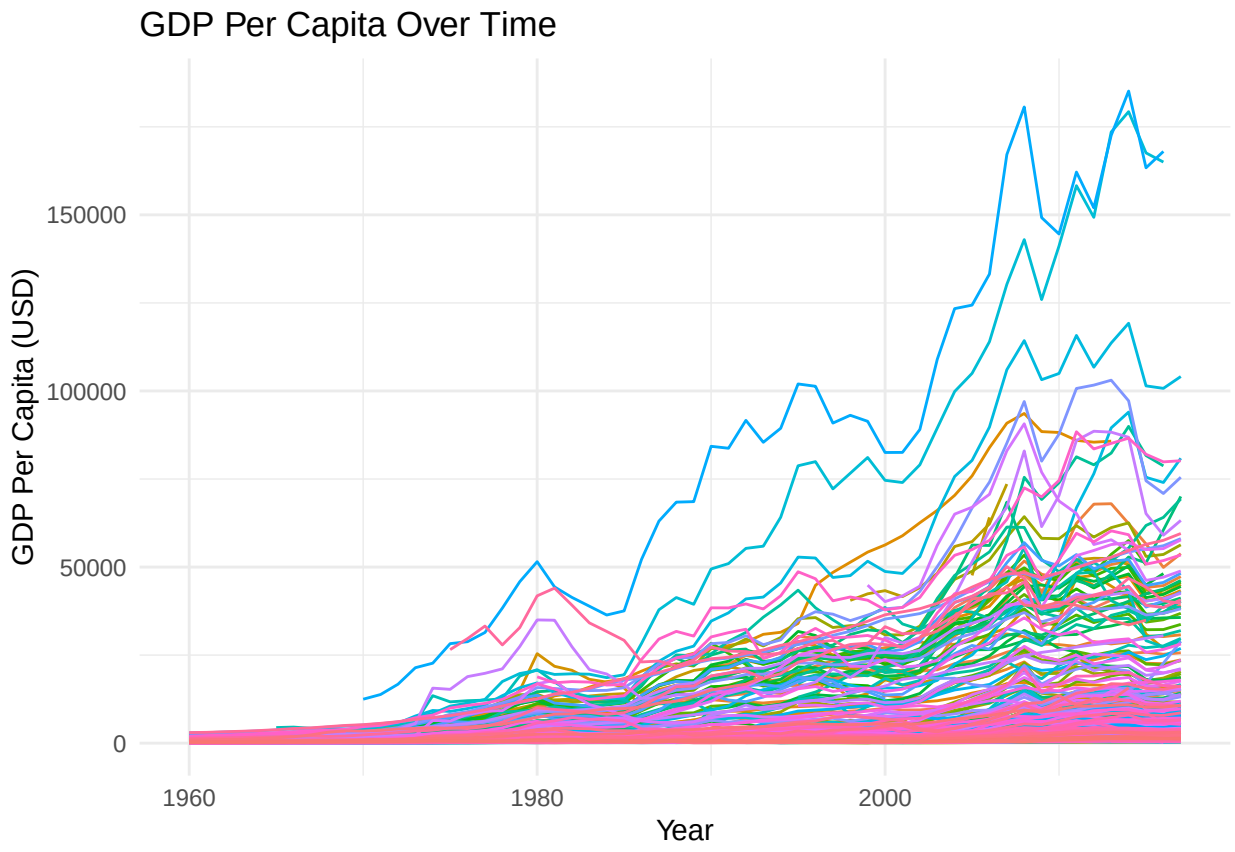
Consider the GDP information in global_economy. Plot the GDP per capita for each country over time. Which country has the highest GDP per capita? How has this changed over time?

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

##
## Attaching package: 'tsibble'

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, union

## -- Attaching packages --------------------------------------------- fpp3 1.0.1 --

## v tibble    3.2.1     v ggplot2   3.5.1
## v dplyr     1.1.4     v feasts    0.4.1
## v tidyr     1.3.1     v fable     0.4.1
## v lubridate 1.9.3

## -- Conflicts ------------------------------------------------ fpp3_conflicts --
## x lubridate::date()     masks base::date()
## x dplyr::filter()       masks stats::filter()
## x tsibble::intersect()  masks base::intersect()
## x lubridate::interval() masks tsibble::interval()
## x dplyr::lag()          masks stats::lag()
## x tsibble::setdiff()    masks base::setdiff()
## x tsibble::union()      masks base::union()

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

## -- Attaching packages --------------------------------------------- fpp2 2.5 --
```

```
## v forecast  8.23.0     v expsmooth 2.3
## v fma       2.5

##

##
## Attaching package: 'fpp2'

## The following object is masked from 'package:fpp3':
##
##     insurance
```

The plot shows GPD per capita trends over time and the country with the higehst GDP per Capita is Monaco (MCO)

```
## Warning: Removed 3242 rows containing missing values or values outside the scale range
## (`geom_line()`).
```

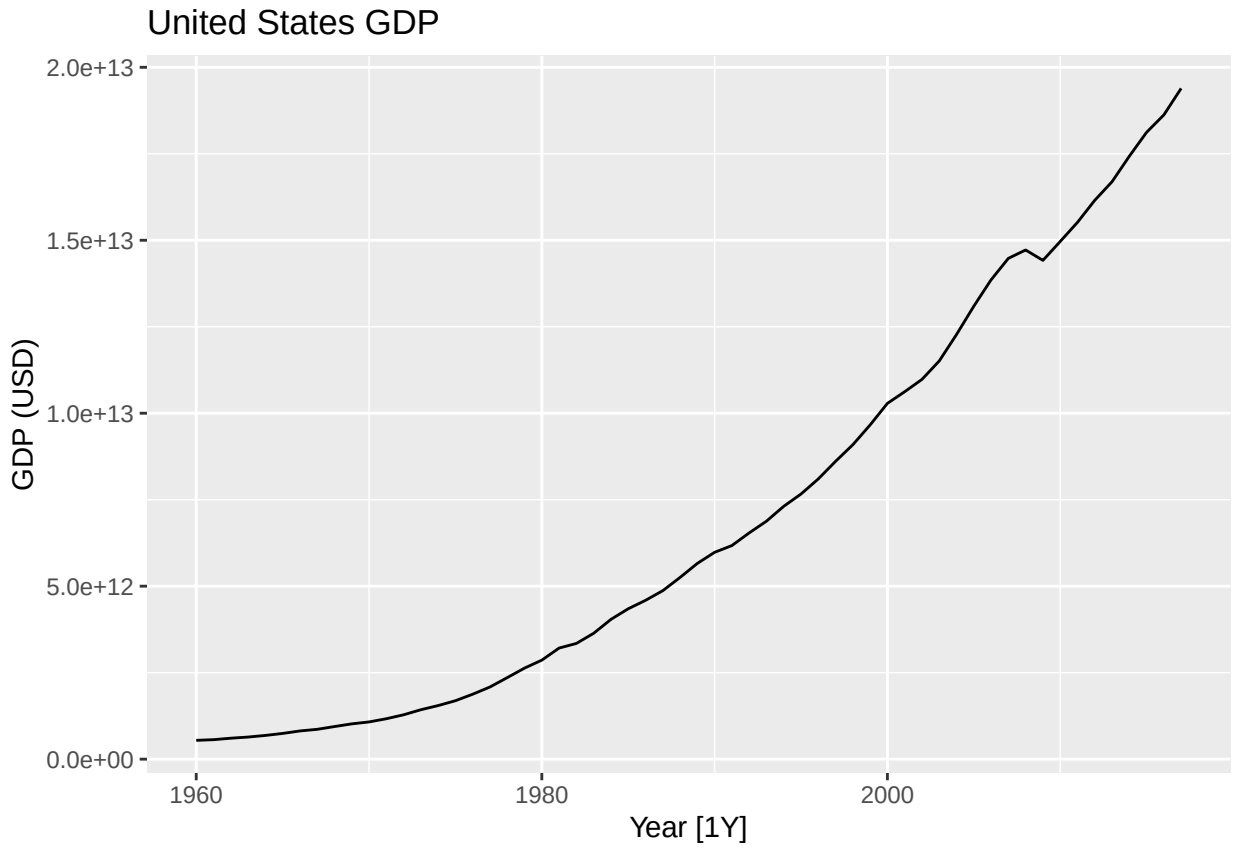



```
## # A tsibble: 1 x 10 [1Y]
## # Key:       Country [1]
##   Country Code   Year         GDP Growth   CPI Imports Exports Population
##   <fct>   <fct> <dbl>       <dbl>  <dbl> <dbl>   <dbl>   <dbl>      <dbl>
## 1 Monaco  MCO    2014 7060236168.   7.18    NA      NA      NA      38132
## # i 1 more variable: GDPperCapita <dbl>
```

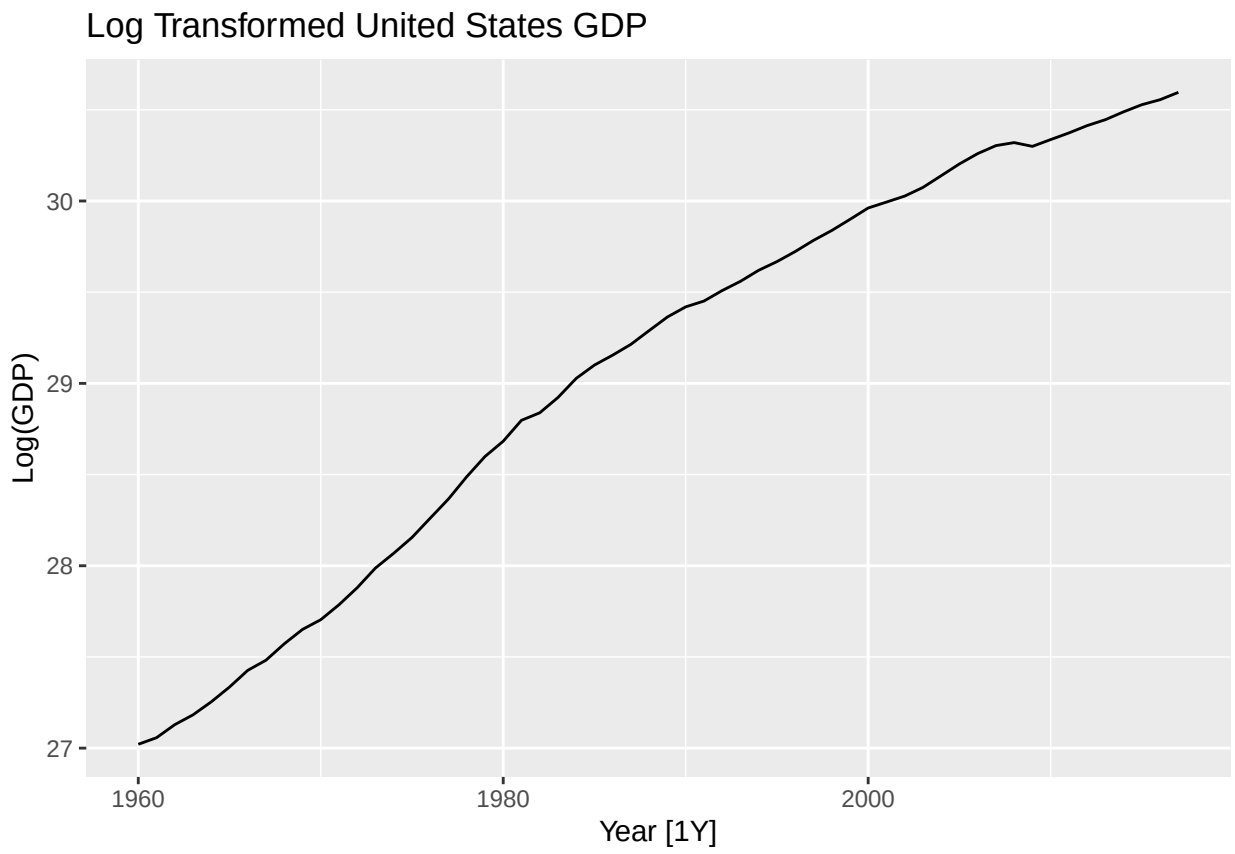
Excercise 3.2 For each of the following series, make a graph of the data. If transforming seems appropriate, do so and describe the effect.

- United States GDP from global_economy.
- Slaughter of Victorian "Bulls, bullocks and steers" in aus_livestock.
- Victorian Electricity Demand from vic_elec. = Gas production from aus_production.

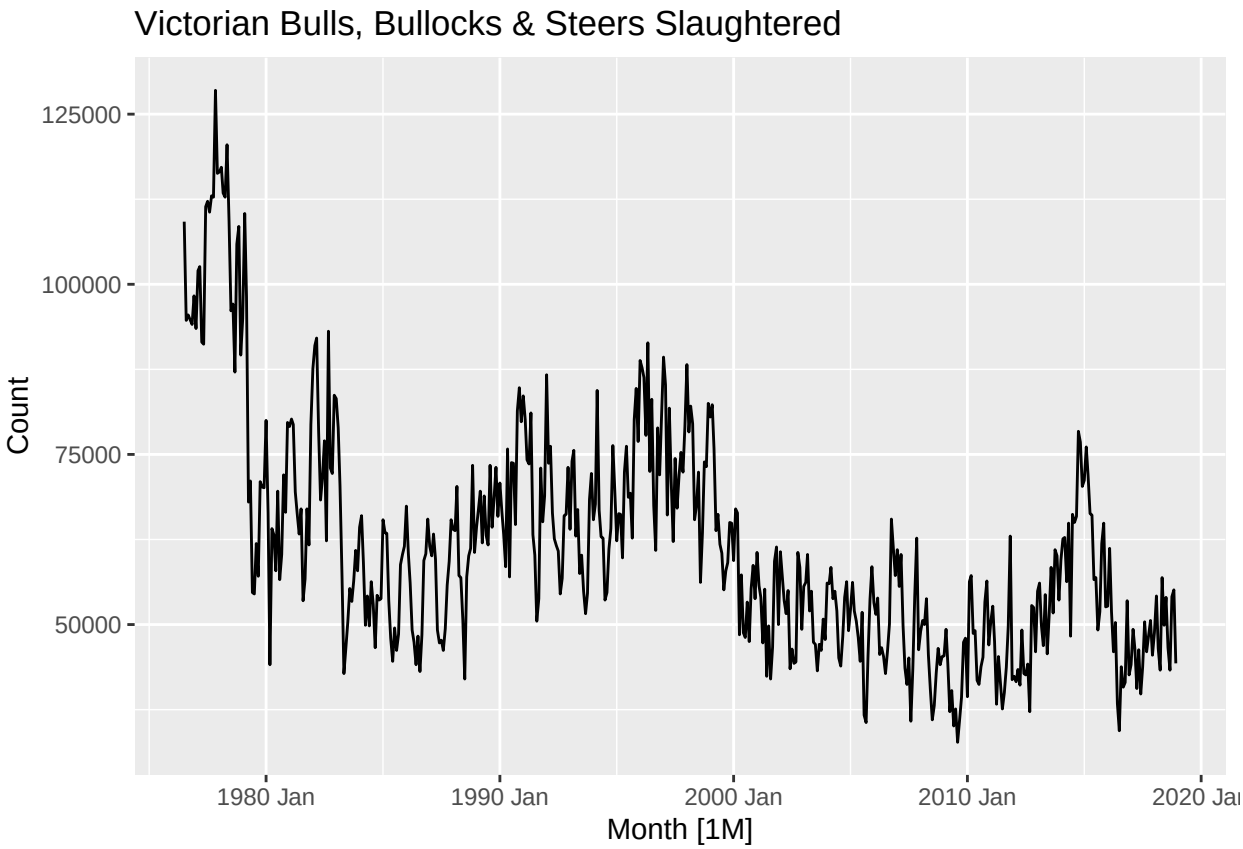
```
library(fpp3)

# 1. United States GDP from global_economy
us_gdp <- global_economy %>%
  filter(Country == "United States") %>%
  select(Year, GDP)

# Plot US GDP
us_gdp %>%
  autoplot(GDP) +
  labs(title = "United States GDP", y = "GDP (USD)")
```



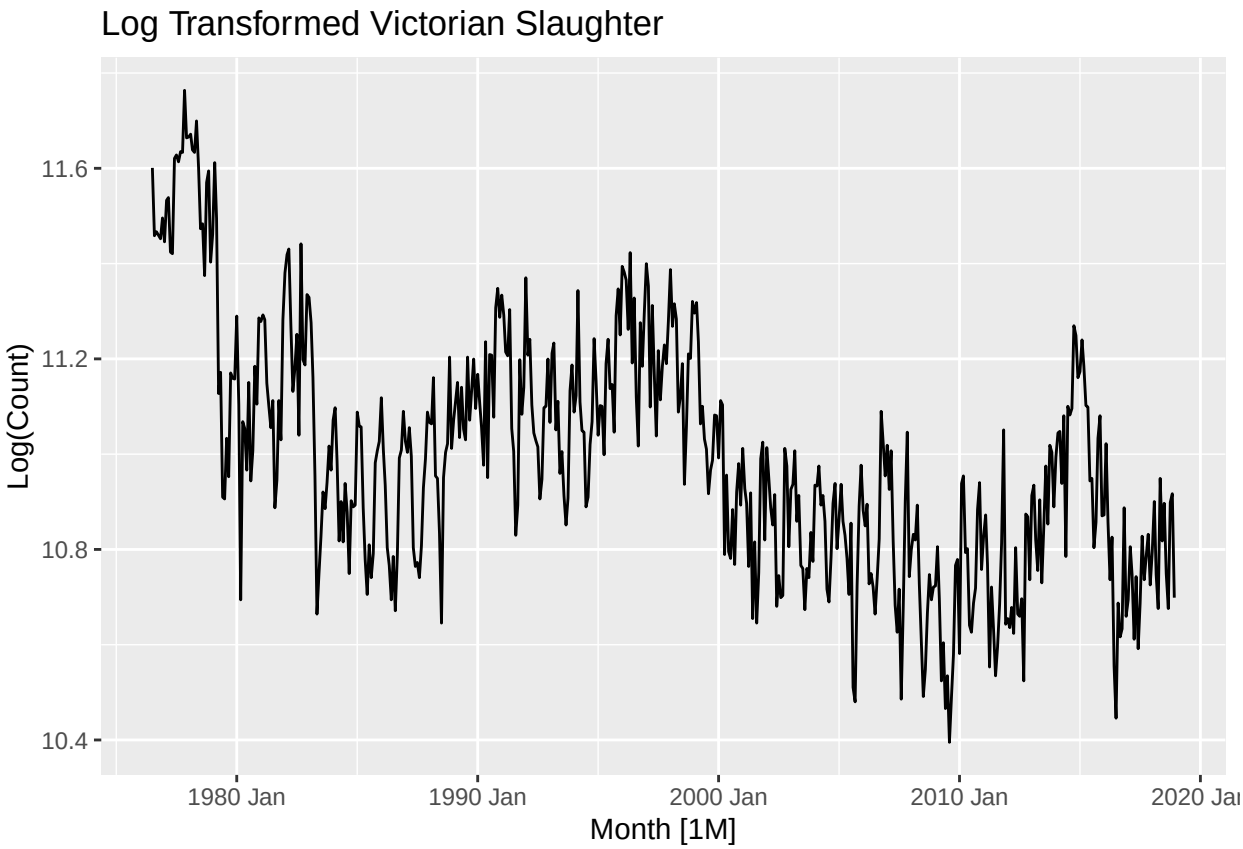

```
# Log transformation (if necessary)
us_gdp %>%
  mutate(log_GDP = log(GDP)) %>%
  autoplot(log_GDP) +
  labs(title = "Log Transformed United States GDP", y = "Log(GDP)")
```



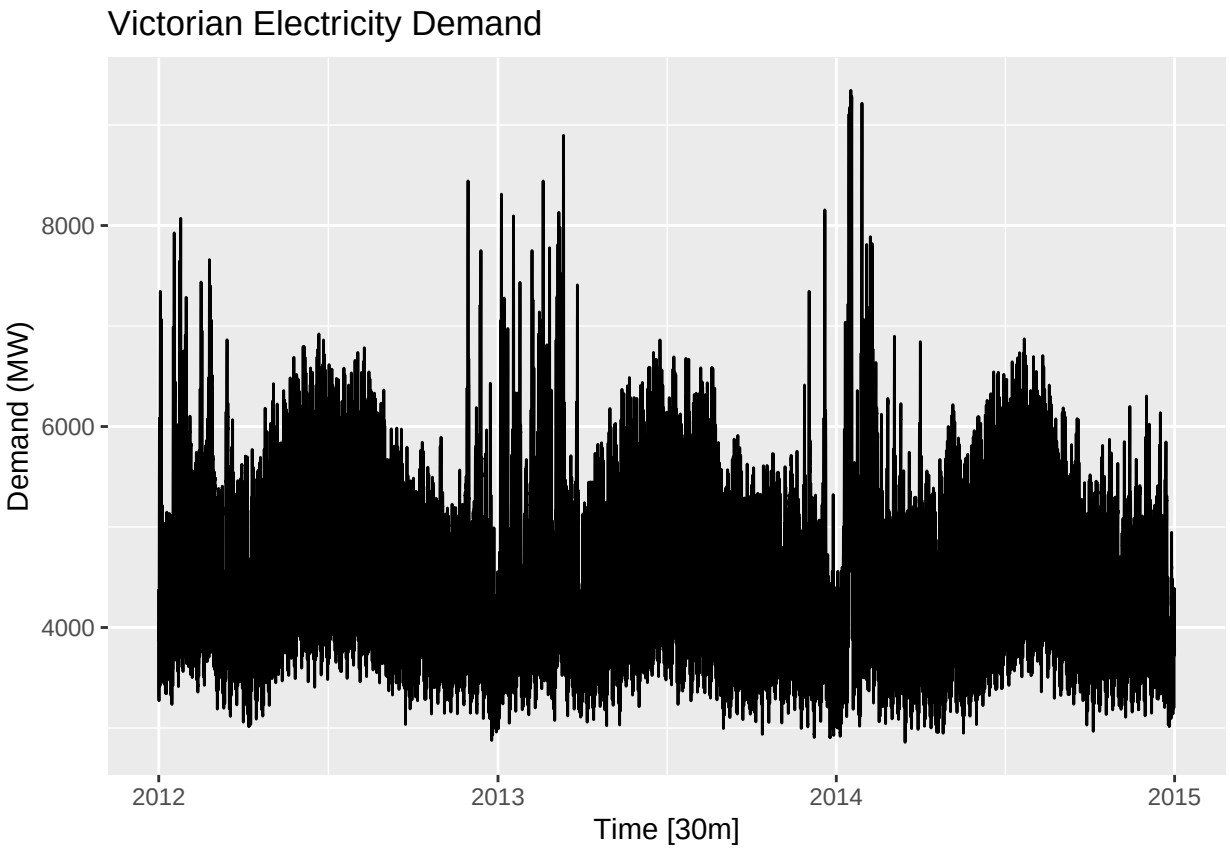

```
# 2. Slaughter of Victorian "Bulls, bullocks and steers" from aus_livestock
victorian_slaughter <- aus_livestock %>%
  filter(Animal == "Bulls, bullocks and steers", State == "Victoria")

# Plot raw data
victorian_slaughter %>%
  autoplot(Count) +
  labs(title = "Victorian Bulls, Bullocks & Steers Slaughtered", y = "Count")
```



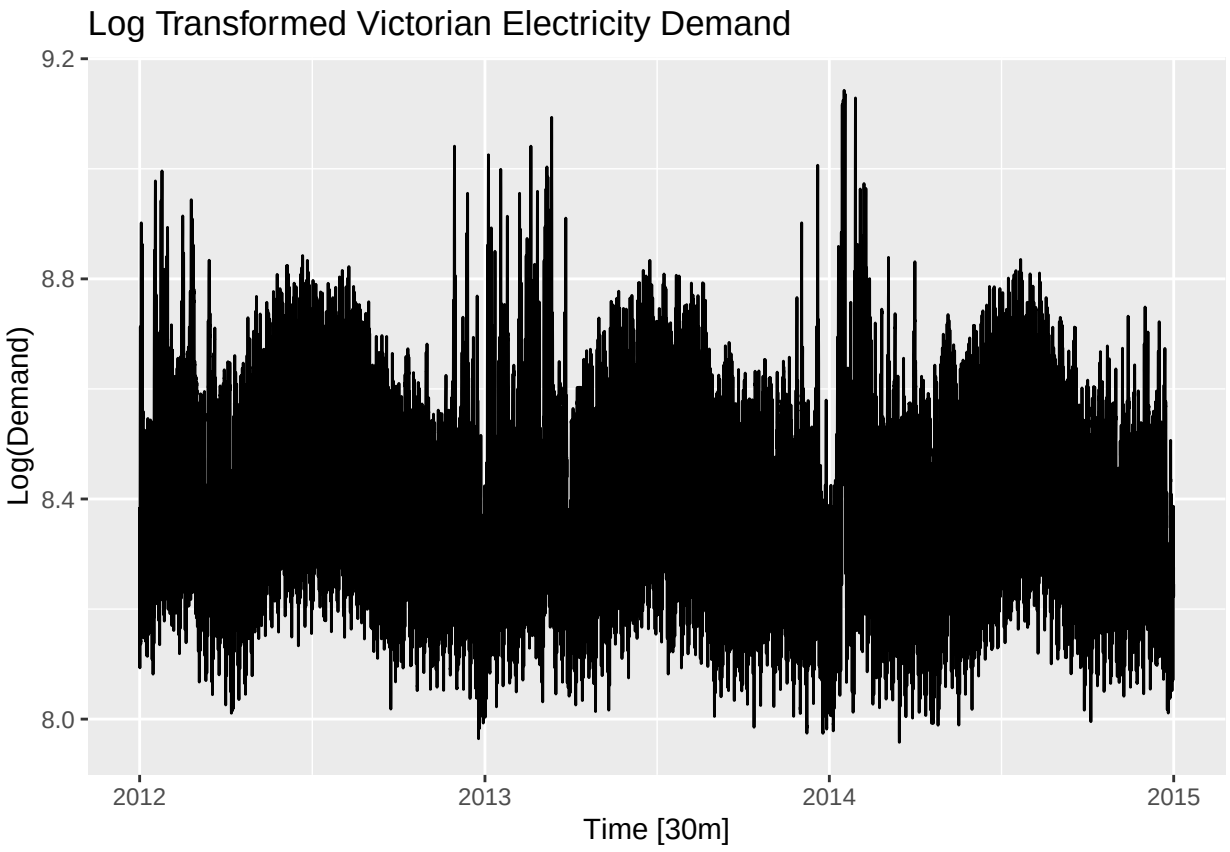

```
# Log transformation (if necessary)
victorian_slaughter %>%
  mutate(log_Count = log(Count)) %>%
  autoplot(log_Count) +
  labs(title = "Log Transformed Victorian Slaughter", y = "Log(Count)")
```



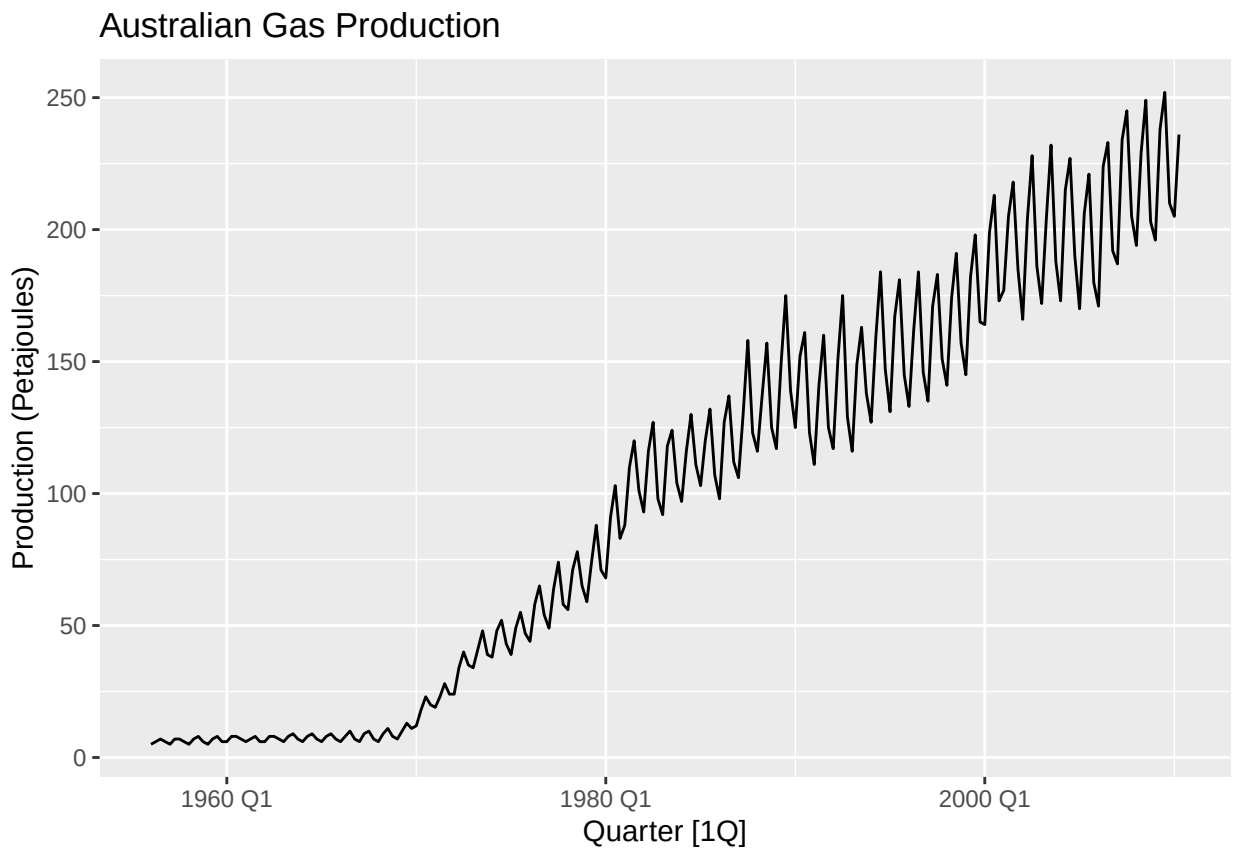

```
# 3. Victorian Electricity Demand from vic_elec
vic_electricity <- vic_elec %>%
  select(Time, Demand)

# Plot raw electricity demand
vic_electricity %>%
  autoplot(Demand) +
  labs(title = "Victorian Electricity Demand", y = "Demand (MW)")
```




```
# Apply log transformation if variance is unstable
vic_electricity %>%
  mutate(log_Demand = log(Demand)) %>%
  autoplot(log_Demand) +
  labs(title = "Log Transformed Victorian Electricity Demand", y = "Log(Demand)")
```

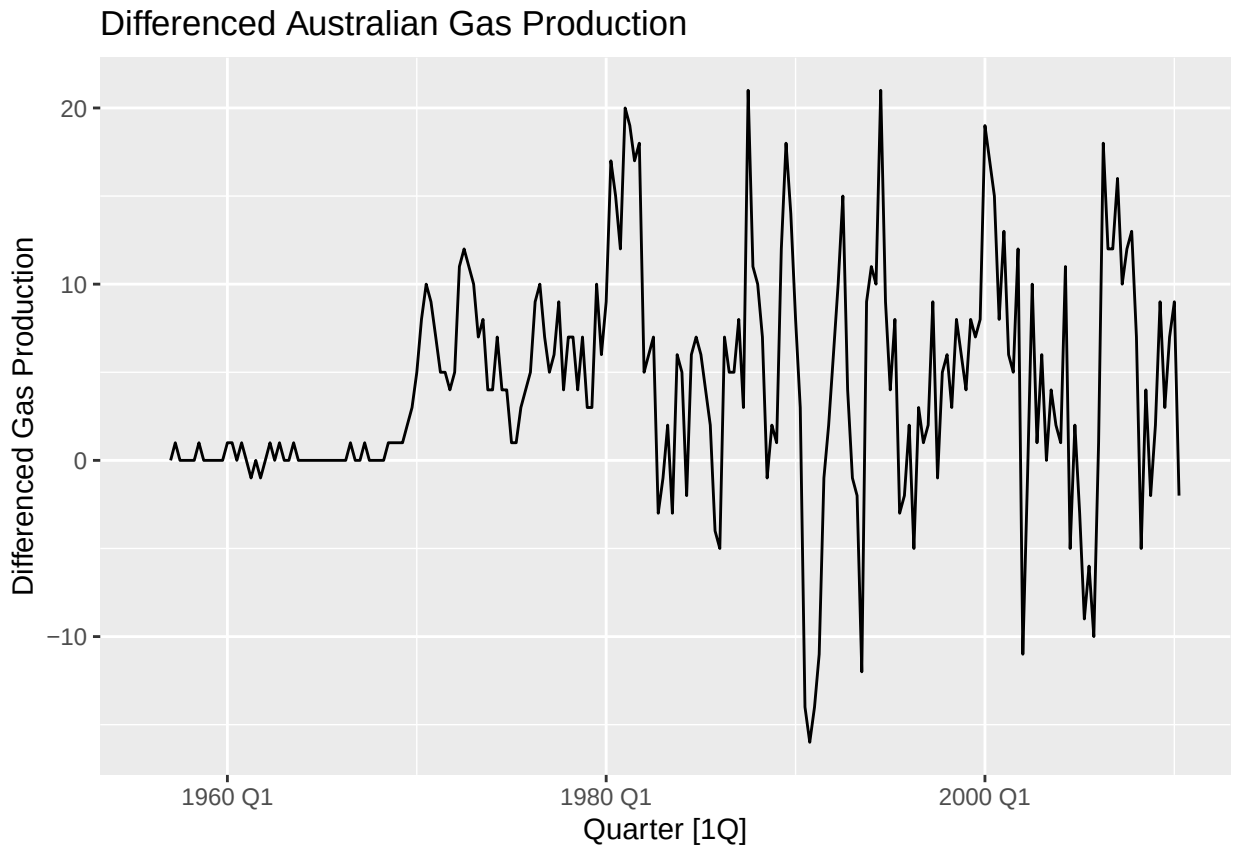



```
# 4. Gas production from aus_production
gas_production <- aus_production %>%
  select(Quarter, Gas)

# Plot raw gas production
gas_production %>%
  autoplot(Gas) +
  labs(title = "Australian Gas Production", y = "Production (Petajoules)")
```




```
# Seasonal difference transformation (if necessary)
gas_production %>%
  mutate(diff_Gas = difference(Gas, lag = 4)) %>%
  autoplot(diff_Gas) +
  labs(title = "Differenced Australian Gas Production", y = "Differenced Gas Production")
```

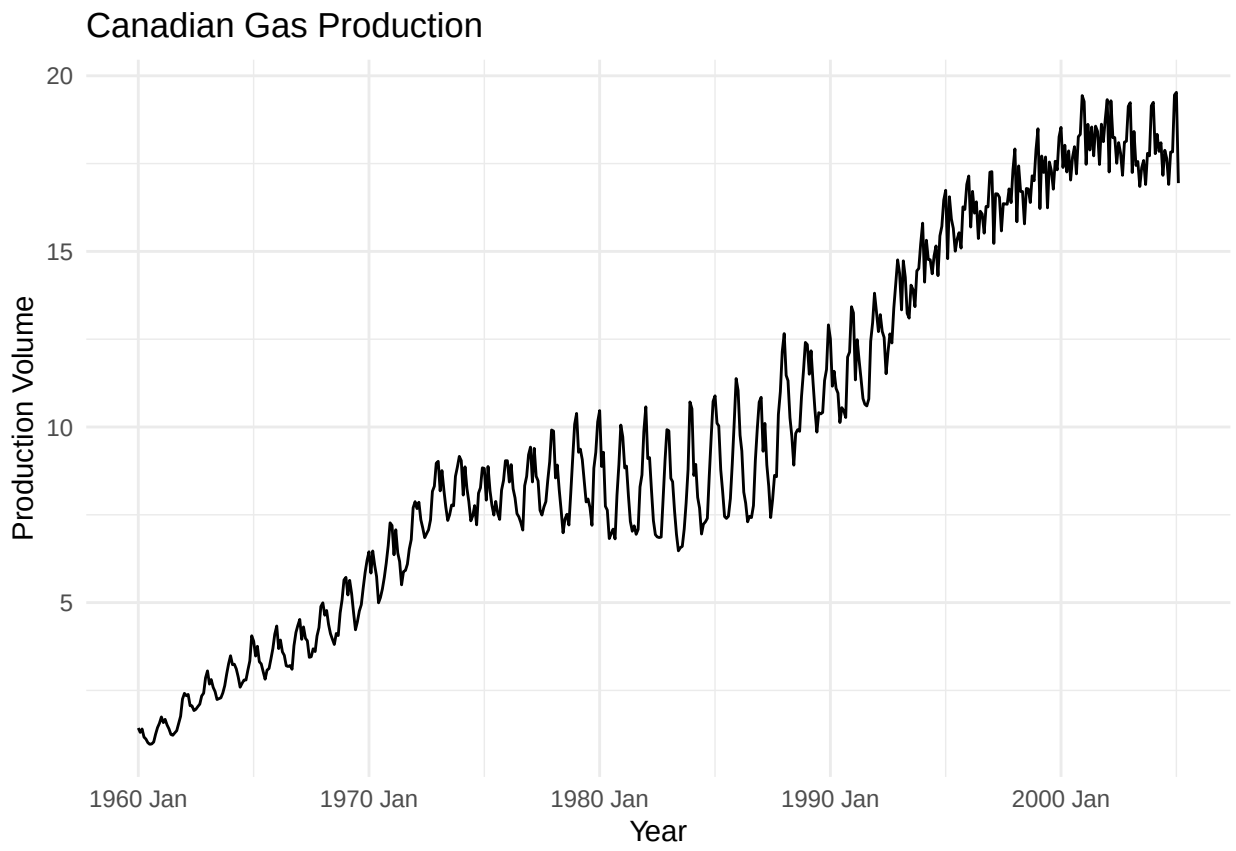
```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## (`geom_line()`).
```



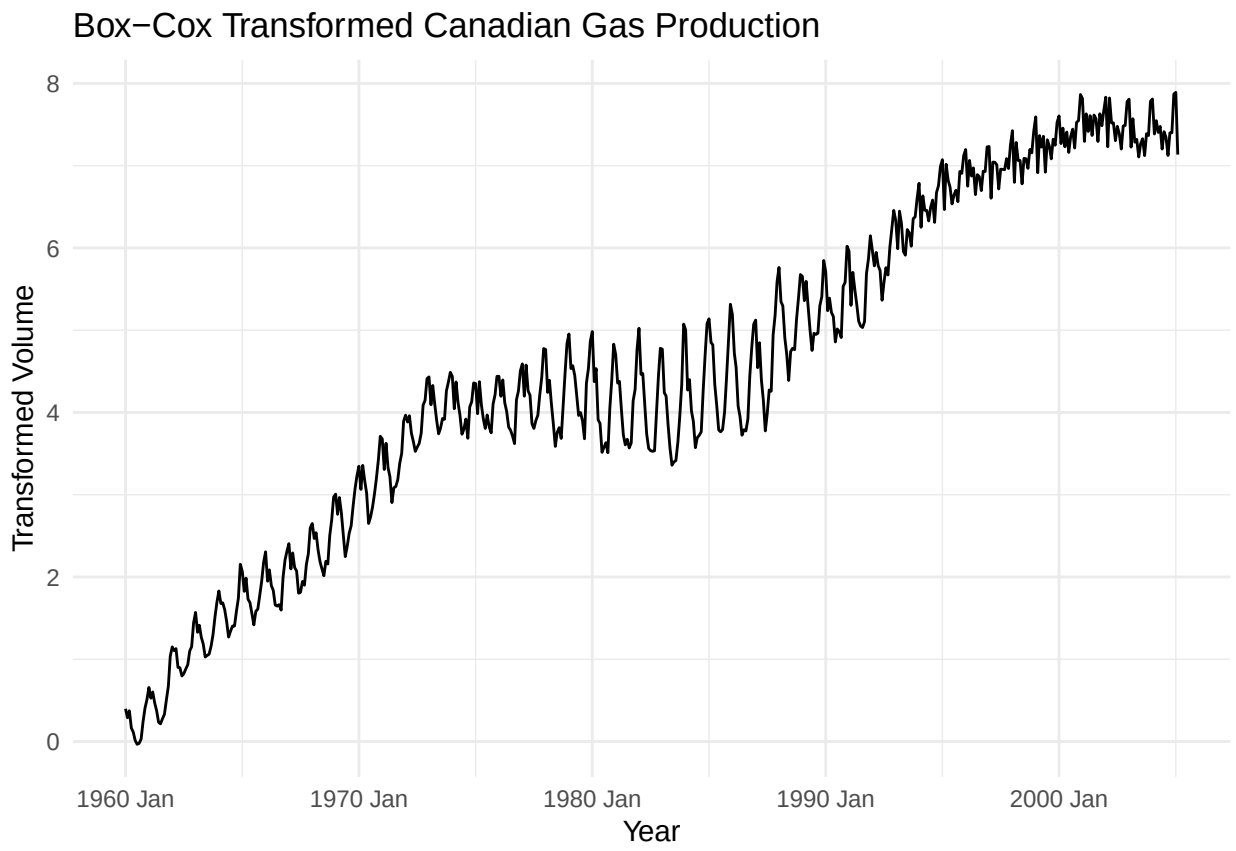
The transformation analysis for the selected time series highlights different patterns and necessary adjustments to improve interpretability. The **United States GDP** exhibits an exponential growth trend, making a **log transformation** effective in stabilizing variance and linearizing the trend over time. Similarly, the **Victorian slaughter data** for bulls, bullocks, and steers may show increasing fluctuations, where a **log transformation** helps reduce heteroscedasticity. The **Victorian electricity demand** often displays strong seasonal variation, and applying a **log transformation** can help stabilize variance while preserving seasonal patterns. Lastly, the **Australian gas production** series has noticeable seasonal fluctuations, making a **seasonal differencing transformation (lag = 4 for quarterly data)** useful in removing seasonal effects and revealing underlying trends. These transformations improve model performance and make trends and patterns more interpretable for further time series analysis.

Exercise 3.3 What Box-Cox transformation would you select for your canadian gas data ?

```
# Plot raw Canadian gas production data
canadian_gas %>%
  autoplot(Volume) +
  labs(title = "Canadian Gas Production",
       y = "Production Volume",
       x = "Year") +
  theme_minimal()
```

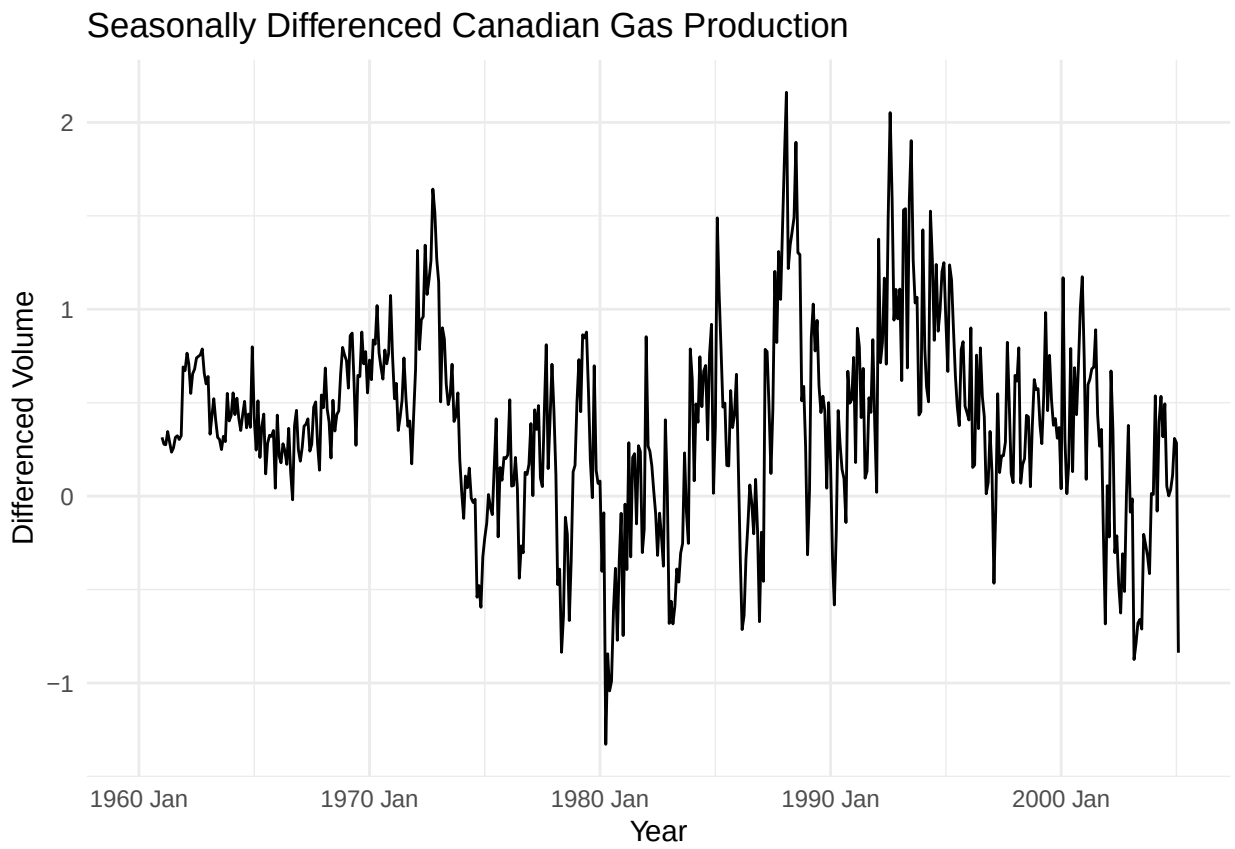



```
# Estimate optimal lambda for Box-Cox transformation
lambda <- canadian_gas %>% features(Volume, features = guerrero) %>% pull(lambda_guerrero)

# Apply the Box-Cox transformation and plot
canadian_gas %>%
  mutate(BoxCox_Volume = box_cox(Volume, lambda)) %>%
  autoplot(BoxCox_Volume) +
  labs(title = "Box-Cox Transformed Canadian Gas Production",
       y = "Transformed Volume",
       x = "Year") +
  theme_minimal()
```




```
# Apply seasonal differencing (lag = 12 for monthly data)
canadian_gas %>%
  mutate(Seasonally_Differenced = difference(Volume, lag = 12)) %>%
  autoplot(Seasonally_Differenced) +
  labs(title = "Seasonally Differenced Canadian Gas Production",
       y = "Differenced Volume",
       x = "Year") +
  theme_minimal()
```

```
## Warning: Removed 12 rows containing missing values or values outside the scale range
## (`geom_line()`).
```



A **Box-Cox transformation** is unhelpful for the **Canadian gas production** data because the transformed series still exhibits **strong seasonality**, and the transformation does not remove the **repeating annual pattern**. Since the variance in the data is already stable, the Box-Cox transformation does not provide significant improvement. In contrast, **seasonal differencing** (lag = 12 for monthly data) is a more effective approach. It removes the **seasonal pattern**, making the series more **stationary** and easier to model using forecasting methods like **ARIMA**. Unlike the Box-Cox transformation, which primarily addresses variance instability, seasonal differencing effectively eliminates **seasonal fluctuations**, allowing for better trend analysis and prediction. Since seasonality is the main issue rather than variance instability, **seasonal differencing** is the appropriate transformation to prepare the data for accurate forecasting.

Exercise 3.4 For each of the following series, make a graph of the data. If transforming seems appropriate, do so and describe the effect. dole, usdeaths, bricksq.

```
library(readxl)
library(httr)
library(openxlsx)
url <- 'https://raw.githubusercontent.com/joewarner89/Data-624-Predictive-Anaytics/main/workspace/retail
temp_file <- tempfile(fileext = ".xlsx")  # Create a temporary file

download.file(url, temp_file, mode = "wb")  # Download
retail <- read_excel(temp_file, skip = 1)  # Read the Excel file

head(retail)
```

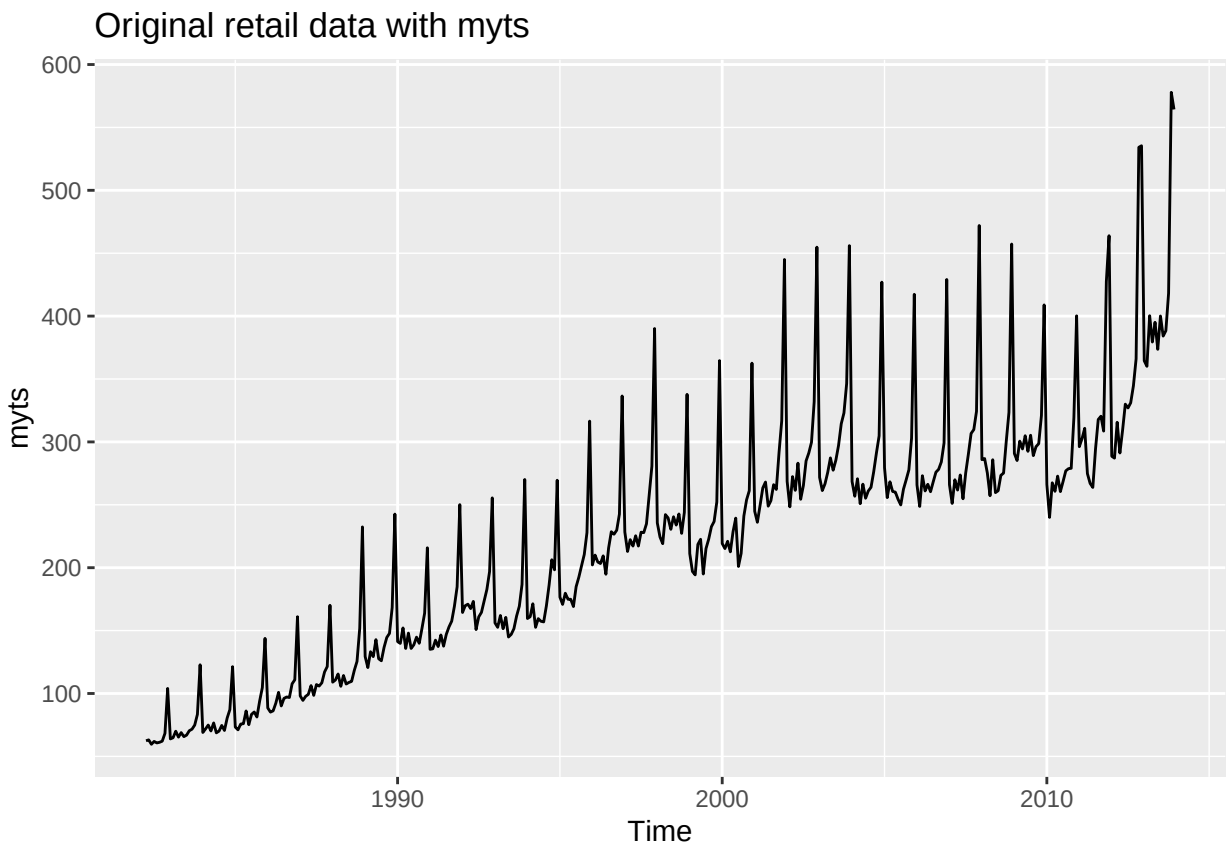
```
## # A tibble: 6 x 190
```

```
##   `Series ID`         A3349335T A3349627V A3349338X A3349398A A3349468W
##   <dttm>                  <dbl>     <dbl>     <dbl>     <dbl>     <dbl>
## 1 1982-04-01 00:00:00      303.      41.7      63.9      409.      65.8
## 2 1982-05-01 00:00:00      298.      43.1      64        405.      65.8
## 3 1982-06-01 00:00:00      298       40.3      62.7      401       62.3
## 4 1982-07-01 00:00:00      308.      40.9      65.6      414.      68.2
## 5 1982-08-01 00:00:00      299.      42.1      62.6      404.      66
## 6 1982-09-01 00:00:00      305.      42        64.4      412.      62.3
## # i 184 more variables: A3349336V <dbl>, A3349337W <dbl>, A3349397X <dbl>,
## #   A3349399C <dbl>, A3349874C <dbl>, A3349871W <dbl>, A3349790V <dbl>,
## #   A3349556W <dbl>, A3349791W <dbl>, A3349401C <dbl>, A3349873A <dbl>,
## #   A3349872X <dbl>, A3349709X <dbl>, A3349792X <dbl>, A3349789K <dbl>,
## #   A3349555V <dbl>, A3349565X <dbl>, A3349414R <dbl>, A3349799R <dbl>,
## #   A3349642T <dbl>, A3349413L <dbl>, A3349564W <dbl>, A3349416V <dbl>,
## #   A3349643V <dbl>, A3349483V <dbl>, A3349722T <dbl>, A3349727C <dbl>, ...
```

```
myts <- ts(retail[,"A3349873A"],
           frequency=12, start=c(1982,4))

# Find lambda for each series
lambda_myts <- find_lambda(myts)
```

```
## [1] "Optimal Lambda for series: 0.127636859661548"
```

```
myts_transformed <- BoxCox(myts,lambda_myts)

autoplot(myts) + ggtitle("Original retail data with myts")
```

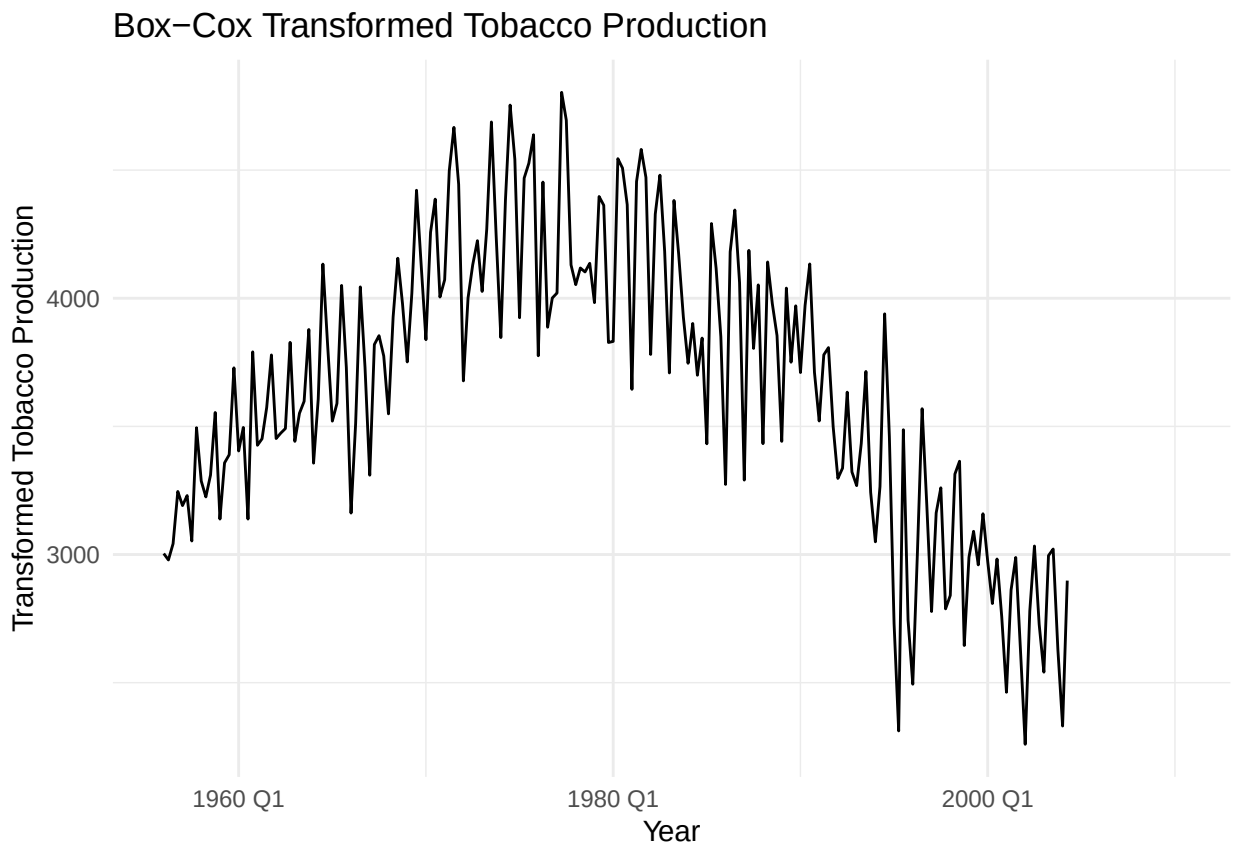


Exercise 3.5

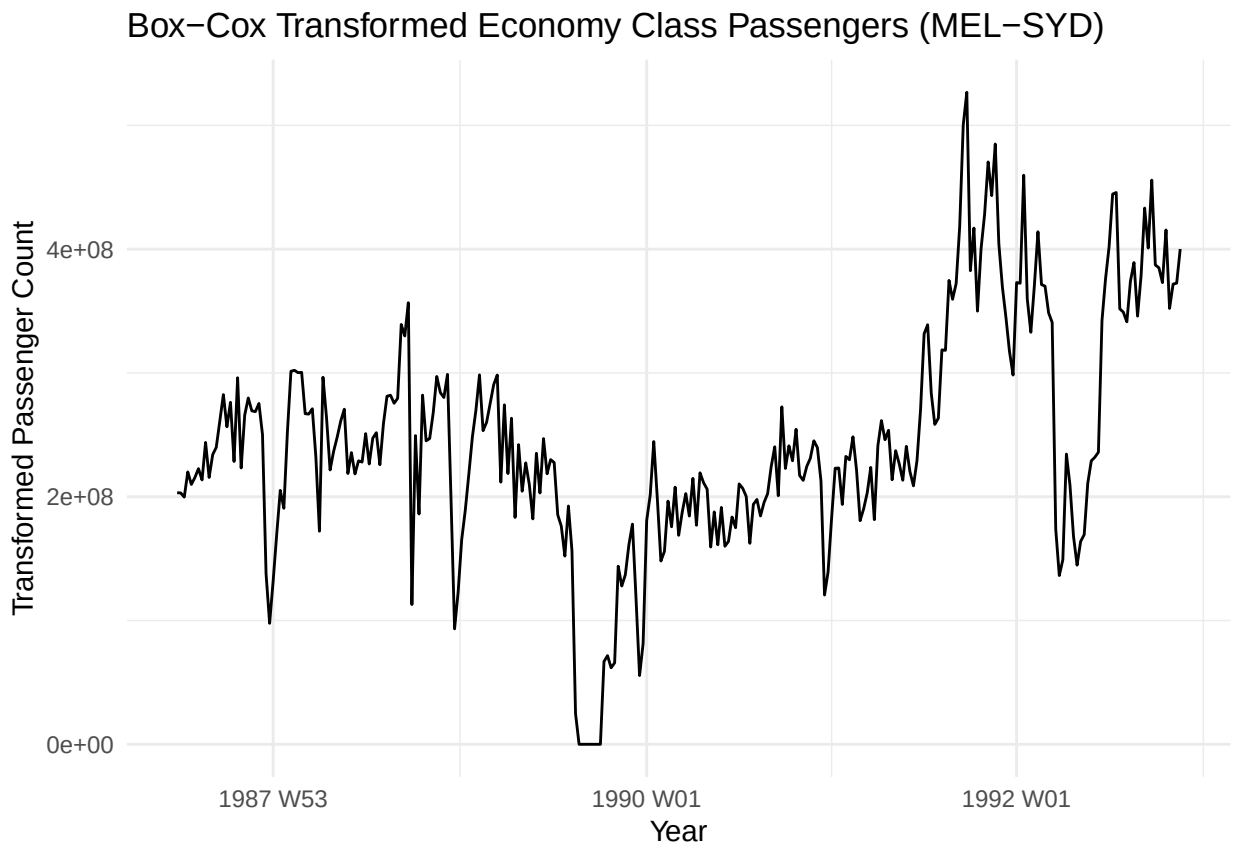
For the following series, find an appropriate Box-Cox transformation in order to stabilise the variance. Tobacco from aus_production, Economy class passengers between Melbourne and Sydney from ansett, and Pedestrian counts at Southern Cross Station from pedestrian.

```
# Load necessary libraries


# Estimate optimal Box-Cox lambda for Tobacco
lambda_tobacco <- aus_production %>%
  features(Tobacco, features = guerrero) %>%
  pull(lambda_guerrero)

# Apply Box-Cox transformation
aus_production %>%
  mutate(Tobacco_BoxCox = box_cox(Tobacco, lambda_tobacco)) %>%
  autoplot(Tobacco_BoxCox) +
  labs(title = "Box-Cox Transformed Tobacco Production",
       y = "Transformed Tobacco Production",
       x = "Year") +
  theme_minimal()
```

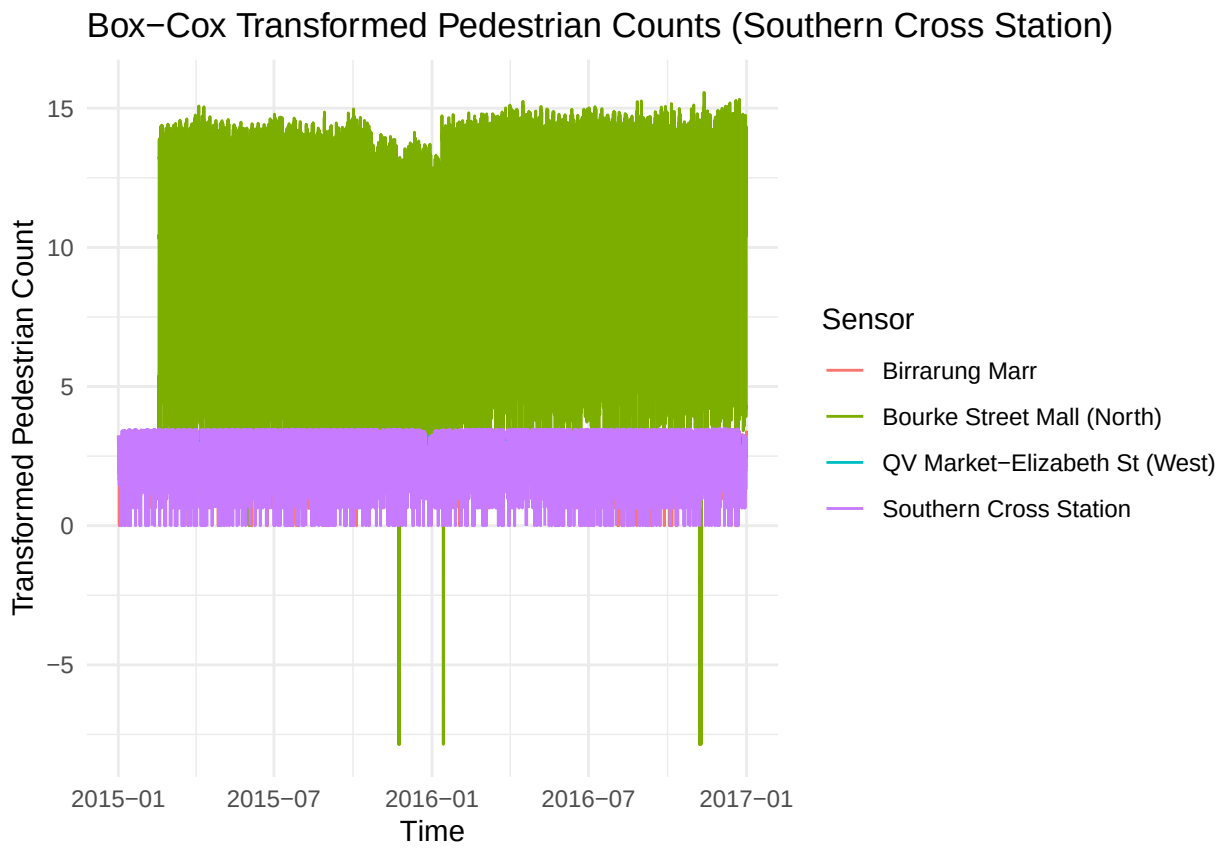
```
## Warning: Removed 24 rows containing missing values or values outside the scale range
## (`geom_line()`).
```




```
# Estimate optimal Box-Cox lambda for Economy class passengers
lambda_passengers <- ansett %>%
  filter(Class == "Economy", Airports == "MEL-SYD") %>%
  features(Passengers, features = guerrero) %>%
  pull(lambda_guerrero)

# Apply Box-Cox transformation
ansett %>%
  filter(Class == "Economy", Airports == "MEL-SYD") %>%
  mutate(Passengers_BoxCox = box_cox(Passengers, lambda_passengers)) %>%
  autoplot(Passengers_BoxCox) +
  labs(title = "Box-Cox Transformed Economy Class Passengers (MEL-SYD)",
       y = "Transformed Passenger Count",
       x = "Year") +
  theme_minimal()
```



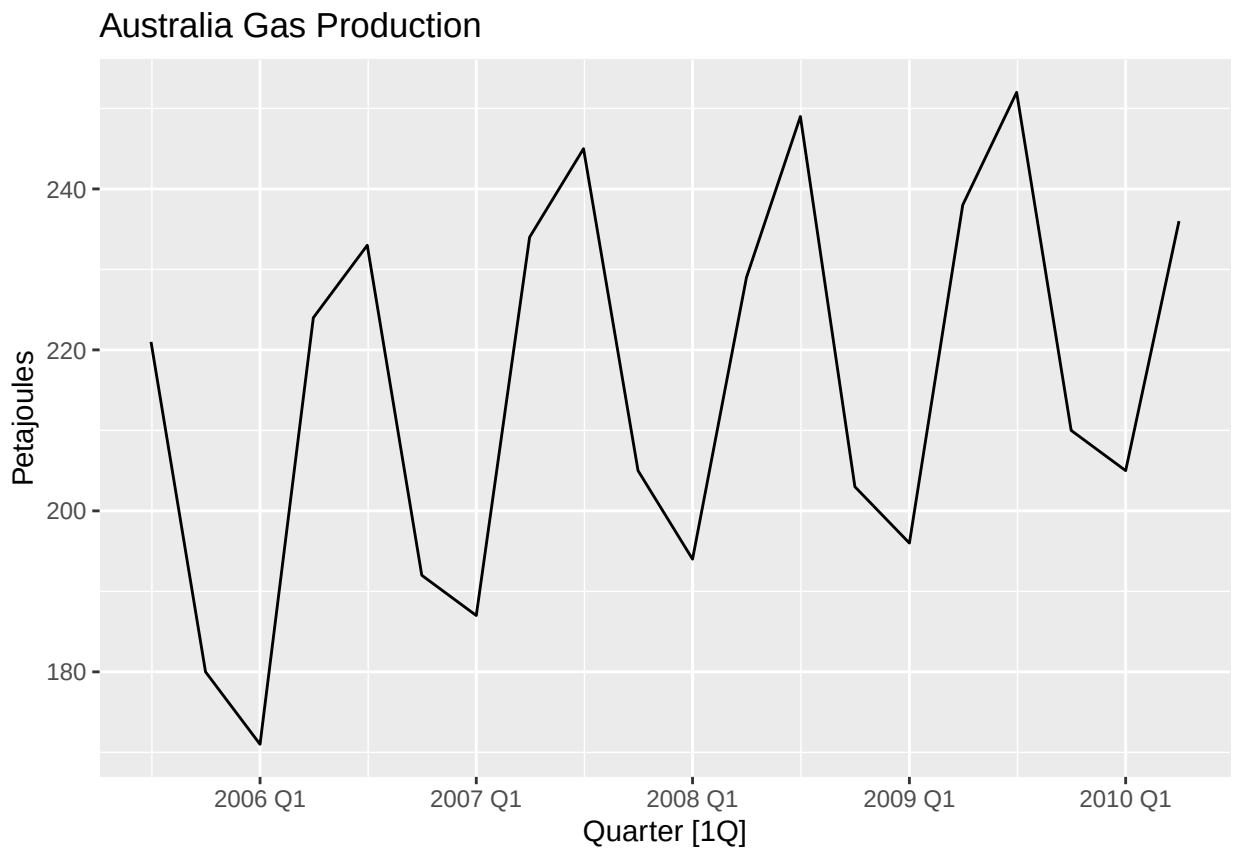

```
# Estimate optimal Box-Cox lambda for Pedestrian counts
lambda_pedestrian <- pedestrian %>%
  features(Count, features = guerrero) %>%
  pull(lambda_guerrero)


# Estimate optimal Box-Cox lambda for each group (location)
lambda_pedestrian <- pedestrian %>%
  group_by(Sensor) %>%
  features(Count, features = guerrero)

# Apply Box-Cox transformation within each group
pedestrian_transformed <- pedestrian %>%
  left_join(lambda_pedestrian, by = "Sensor") %>%  # Merge lambda values with data
  mutate(Count_BoxCox = box_cox(Count, lambda_guerrero))

# Plot transformed pedestrian counts
pedestrian_transformed %>%
  ggplot(aes(x = Date_Time, y = Count_BoxCox, color = Sensor)) +
  geom_line() +
  labs(title = "Box-Cox Transformed Pedestrian Counts (Southern Cross Station)",
       y = "Transformed Pedestrian Count",
       x = "Time") +
  theme_minimal()
```



A **Box-Cox transformation** is applied to stabilize variance and improve interpretability across different time series datasets. In the case of **tobacco production**, the raw series may show **increasing variance over time**, and applying the transformation helps to **stabilize fluctuations**, making the trend easier to analyze. For **economy class passengers** between Melbourne and Sydney, the number of passengers may exhibit **increasing variance**, making trends less clear. The Box-Cox transformation helps to **stabilize these changes in variance**, ensuring a more consistent pattern for forecasting. Similarly, for **pedestrian counts at Southern Cross Station**, fluctuations in data may vary significantly due to events, seasonal patterns, and external factors. The transformation is useful in **stabilizing variance**, making trend analysis and forecasting more reliable. By applying Box-Cox in these cases, we improve data consistency and enhance the effectiveness of time series models.

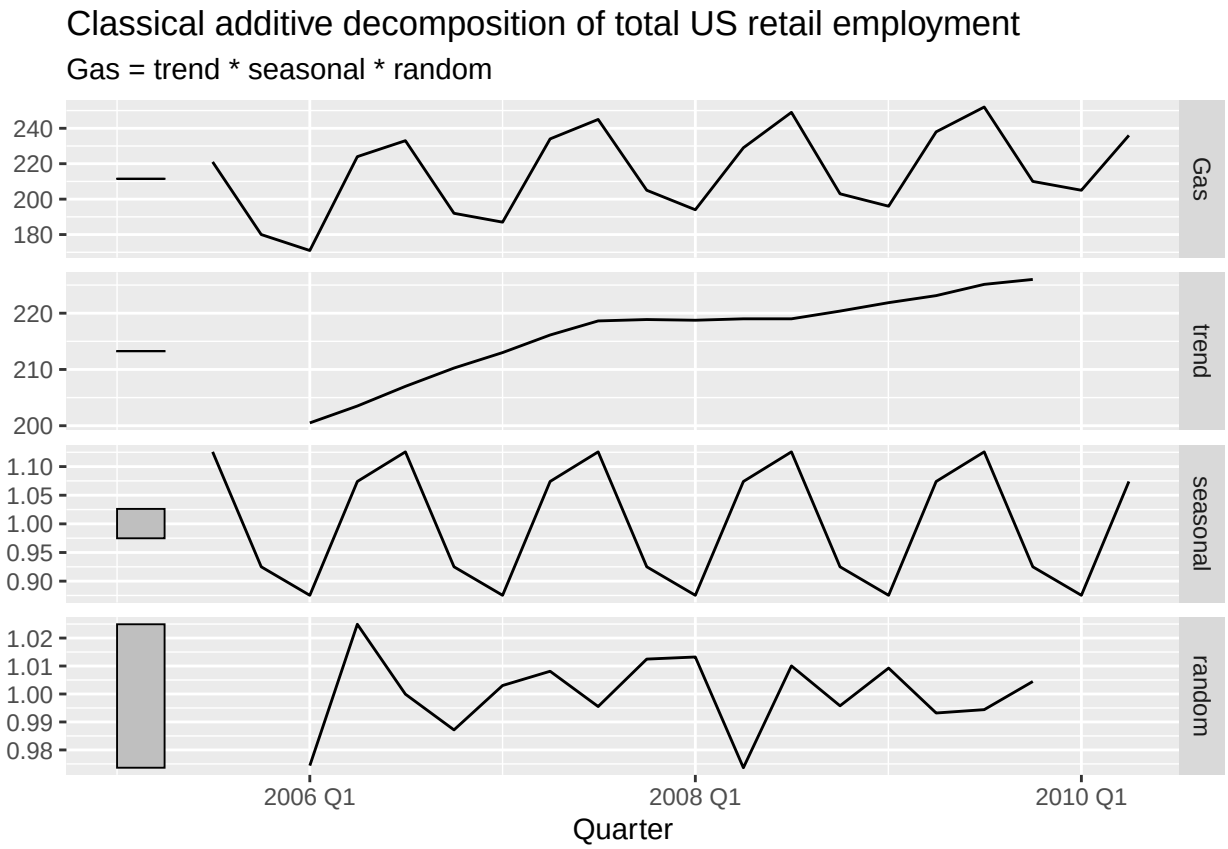
Exercise 3.7 Consider the last five years of the Gas data from aus_production

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

gas %>% autoplot(Gas) +
  labs(title = "Australia Gas Production", y = "Petajoules")
```

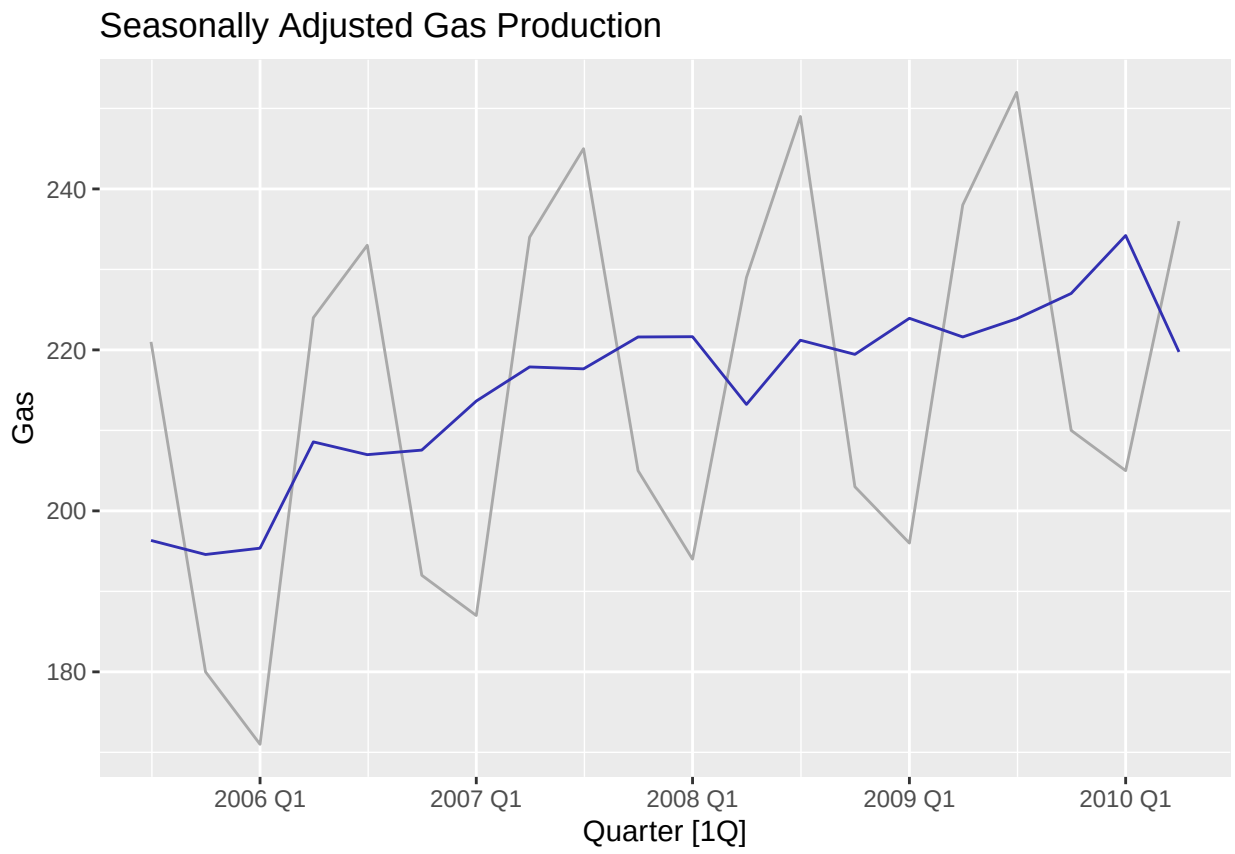



```
# B
gas %>% model(classical_decomposition(Gas, type = "multiplicative")) %>%
  components() %>%
  autoplot() +
  labs(title = "Classical additive decomposition of total US retail employment")
```

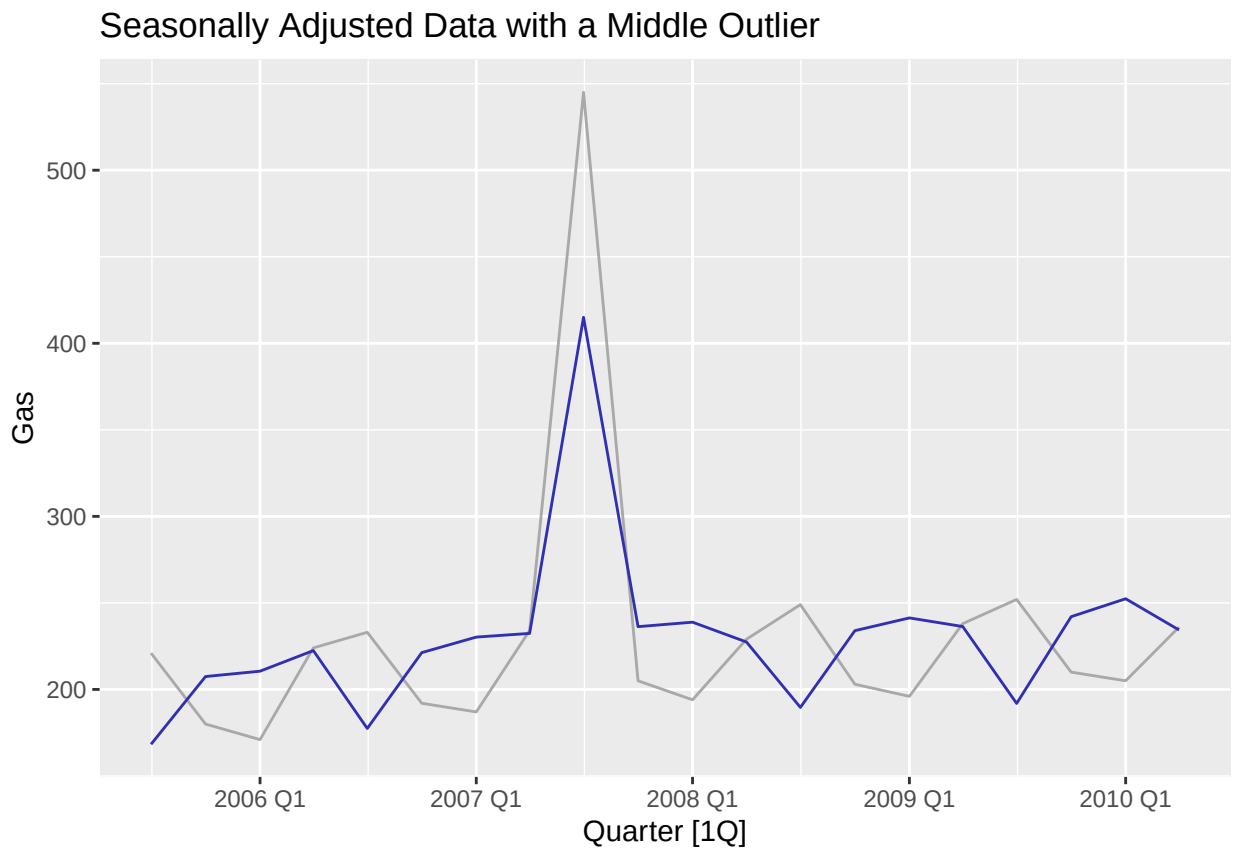
```
## Warning: Removed 2 rows containing missing values or values outside the scale range
## (`geom_line()`).
```




```
# C
gas_season <- gas %>% model(classical_decomposition(Gas, type = "multiplicative"))
components(gas_season) %>%
  as_tsibble() %>%
  autoplot(Gas, colour = "darkgray") +
  geom_line(aes(y=season_adjust), colour = "#3230B2") +
  labs(title = "Seasonally Adjusted Gas Production")
```




```
# F
gas$Gas[gas$Gas == 471] <- gas$Gas[gas$Gas == 471] - 300
gas$Gas[gas$Gas == 245] <- gas$Gas[gas$Gas == 245] + 300

gas %>%
  model(classical_decomposition(Gas, type = "multiplicative")) %>%
  components() %>%
  as_tsibble() %>%
  autoplot(Gas, colour = "darkgray") +
  geom_line(aes(y=season_adjust), colour = "#3230B2") +
  labs(title = "Seasonally Adjusted Data with a Middle Outlier")
```



The results align with Part A. The seasonal line shows a trough at the beginning of the year, followed by peaks around mid-year, and then a gradual decline toward the end of the year.

The first graph displays **Australia's gas production** over time, revealing a clear **seasonal pattern** with peaks and troughs recurring each year. The second graph shows the **classical multiplicative decomposition**, breaking the time series into its **trend-cycle, seasonal, and random components**. The trend component indicates an **upward trend**, while the seasonal component confirms the **recurring fluctuations** observed in the first graph. These results support the initial graphical interpretation. The third graph represents the **seasonally adjusted gas production**, where the seasonal component has been removed, making the underlying trend more visible. The final graph introduces an **outlier in the middle of the time series**, which significantly disrupts the seasonally adjusted data, causing a sharp spike. This highlights the **sensitivity of time series decomposition to outliers**. If the outlier were placed at the end instead of the middle, its effect might be less noticeable on the overall trend estimation but would still distort the recent data, affecting short-term forecasts.

Exercise 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?

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

Exercise 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 illustrates a **positive and increasing trend** over time. The **seasonal pattern** reveals three peaks per year in the hiring of the civilian labor force, occurring around **March, September, and December**. Throughout the dataset, there is minimal noise until the **late 1980s through to approximately 1993**, with a noticeable trough around **late 1990 and early 1991**.

In Figure 3.20, the **seasonal component from the decomposition** shows a **sharp decline from March to August in the early 1990s**. This aligns with the **STL decomposition overview in Figure 3.19**, where a significant decrease is observed in the **remainder (noise) component** between **1990 and 1991**, further supporting the indication of a **recession** during that period.