Assignment 1

Jaewoo Cho

FPP3 7.10 Problems: 1 (a-e), 2 (a-d), 4 (a-c)

1.

Half-hourly electricity demand for Victoria, Australia is contained in `vic_elec`. Extract the January 2014 electricity demand, and aggregate this data to daily with daily total demands and maximum temperatures.

```
# Code block found in book
jan_vic_elec <- vic_elec %>%
     filter(yearmonth(Time) == yearmonth("2014 Jan")) %>%
     index_by(Date = as_date(Time)) %>%
     summarise(Demand = sum(Demand), Temperature = max(Temperature))
jan_vic_elec
```

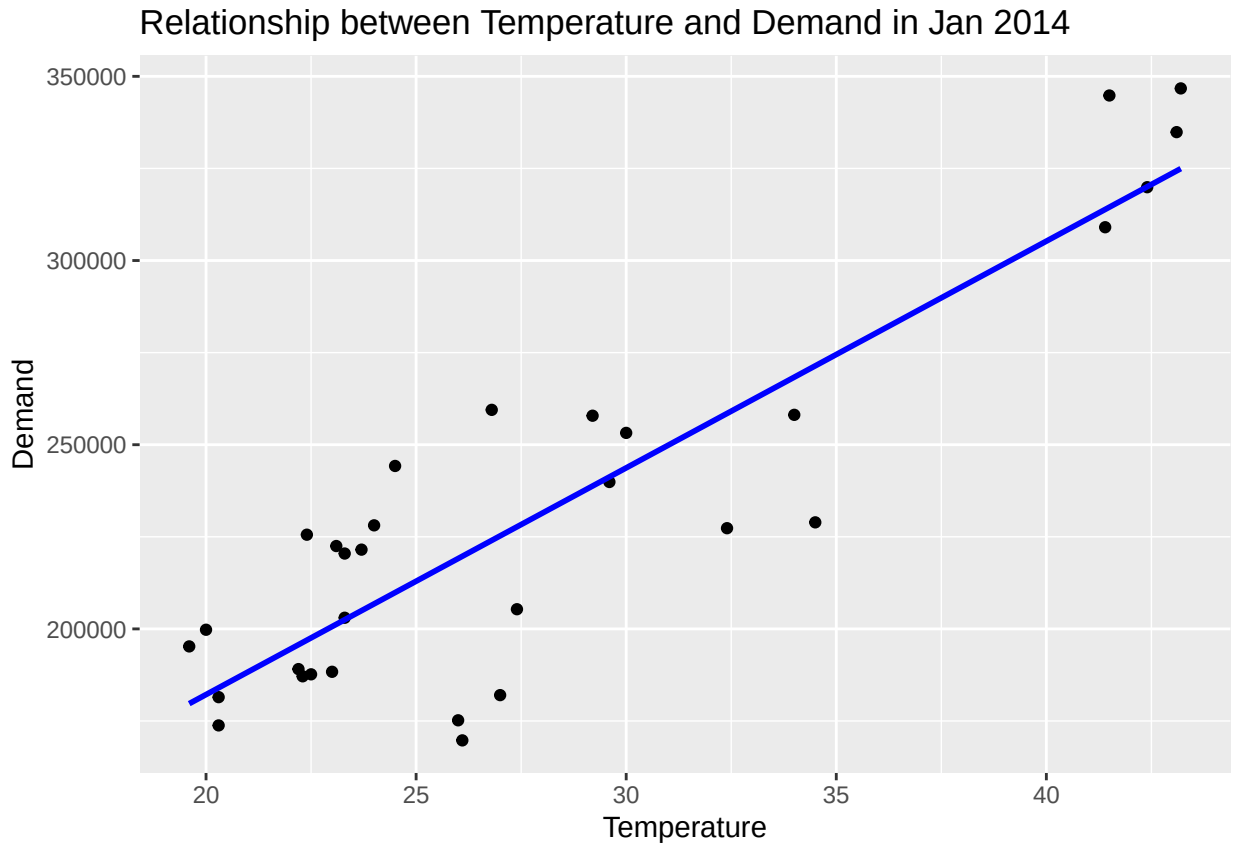
```
# A tsibble: 31 x 3 [1D]
   Date        Demand Temperature
   <date>       <dbl>       <dbl>
 1 2014-01-01 175185.        26
 2 2014-01-02 188351.        23
 3 2014-01-03 189086.        22.2
 4 2014-01-04 173798.        20.3
 5 2014-01-05 169733.        26.1
 6 2014-01-06 195241.        19.6
 7 2014-01-07 199770.        20
 8 2014-01-08 205339.        27.4
 9 2014-01-09 227334.        32.4
10 2014-01-10 258111.        34
# ... with 21 more rows
```

> Use ">" symbol to quote your writing when there are parts of the homework that require you to explain something (you can delete this quoted chunk)

a. Plot the data and find the regression model for Demand with temperature as an explanatory variable. Why is there a positive relationship?

```
# Plot data
library(ggplot2)

ggplot(jan_vic_elec, aes(x=Temperature, y=Demand)) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE, color="blue") +
```

```
    labs(title="Relationship between Temperature and Demand in Jan 2014",
         x="Temperature", y="Demand")
```



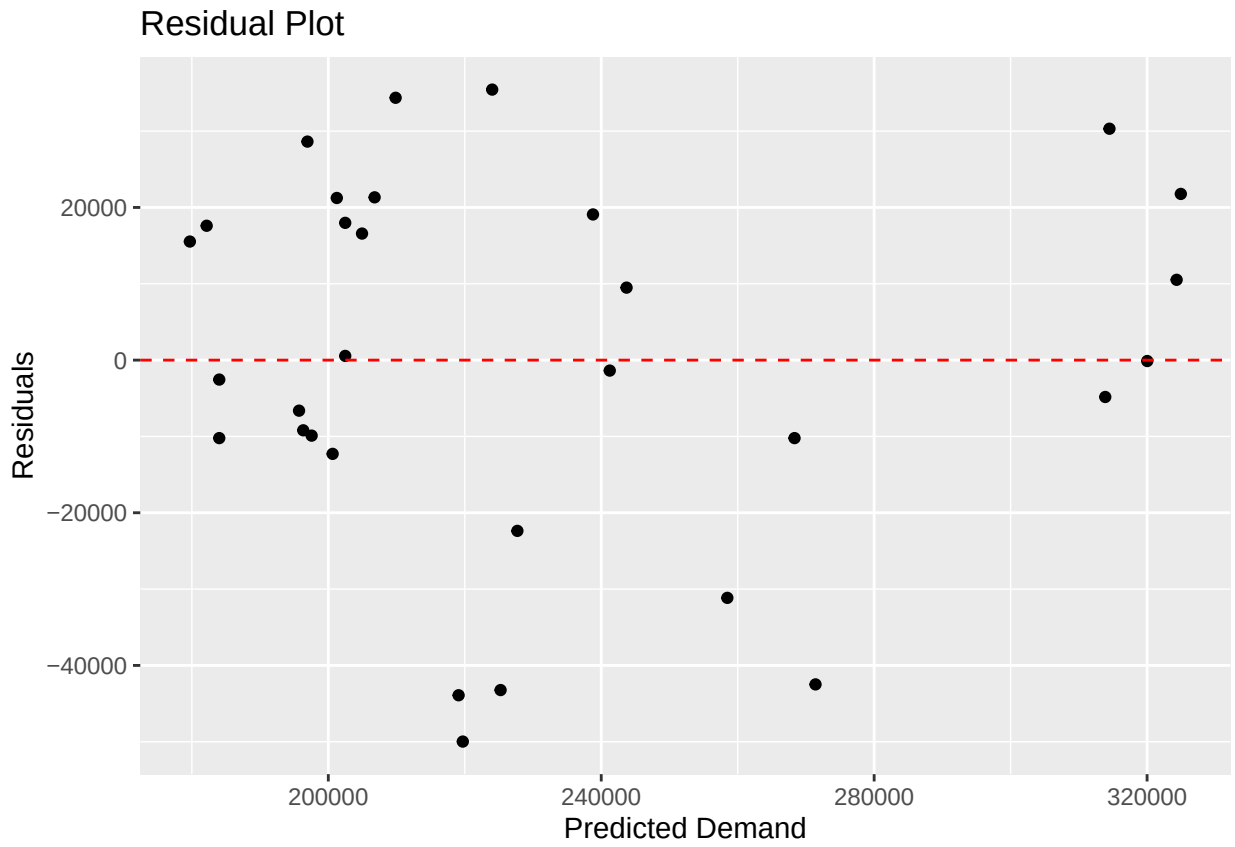

```
# Fit model
lm_model <- lm(Demand ~ Temperature, data=jan_vic_elec)
summary(lm_model)
```

```
Call:
lm(formula = Demand ~ Temperature, data = jan_vic_elec)

Residuals:
   Min     1Q Median     3Q    Max
-49978 -10219   -121  18533  35441

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  59083.9    17424.8   3.391  0.00203 **
Temperature   6154.3      601.3  10.235 3.89e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 24540 on 29 degrees of freedom
Multiple R-squared:  0.7832,    Adjusted R-squared:  0.7757
F-statistic: 104.7 on 1 and 29 DF,  p-value: 3.89e-11
```

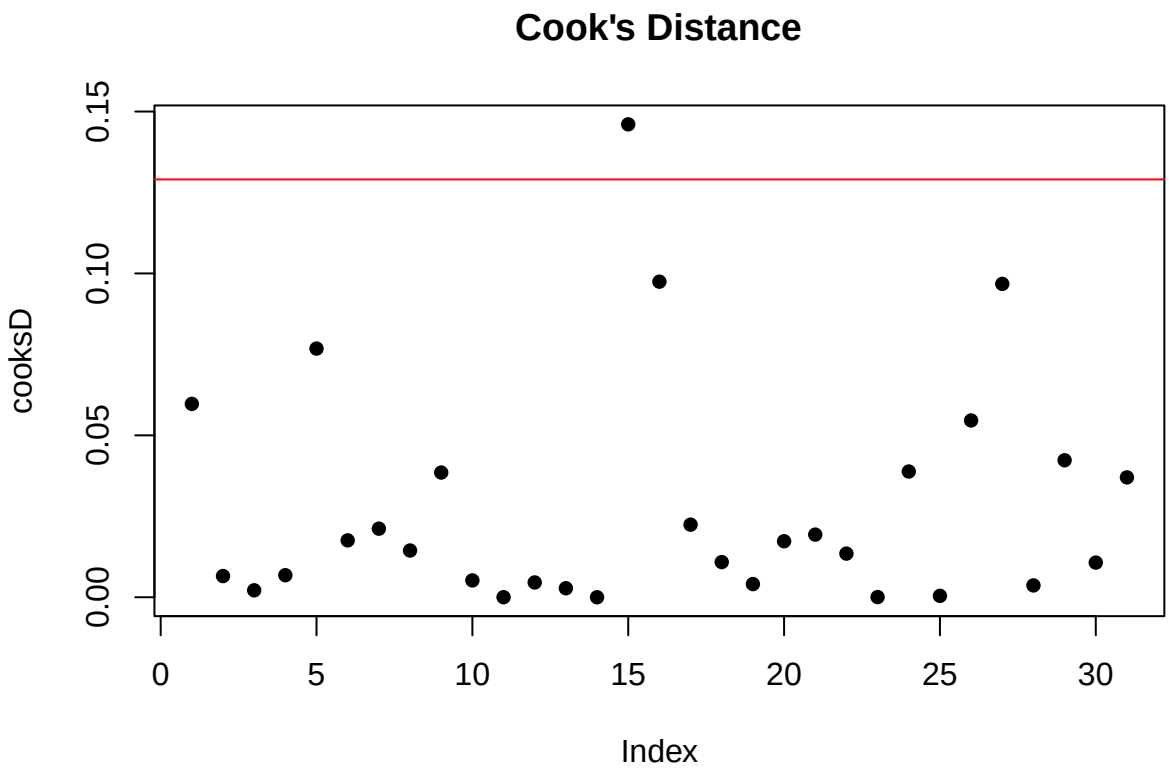
Answer "Why is there a positive relationship?" There is positive relationship with demand in electricity as temperature as people get hotter in Australia, people tend to turn on and use more AC cooling systems.

b. Produce a residual plot. Is the model adequate? Are there any outliers or influential observations?

```
# Plot residuals
residuals <- resid(lm_model)
predicted <- fitted(lm_model)

ggplot(jan_vic_elec, aes(x=predicted, y=residuals)) +
    geom_point() +
    geom_hline(yintercept = 0, linetype="dashed", color="red") +
    labs(title="Residual Plot", x="Predicted Demand", y="Residuals")
```




```
cooksD <- cooks.distance(lm_model)

plot(cooksD, pch=16, cex=1, main="Cook's Distance")
abline(h = 4/length(residuals), col="red")
```



Answer "Is the model adequate?" and "Are there any outliers?" See 5.3 and 5.4 for of FPP3 for extra guidance The model seems to be adequate as the data points are scattered in a random order with no distinguishable pattern. For outliers based on 5.3 and 5.4 on FPP3, detecting outliers with residuals represent the points in time where the model predictions were significantly off from the actual observed values. I tried the Cook's distance to visually identify any outliers. Cook's distance identifies influential observations that are defined by a data point that is significant enough that if it is removed, the results will change the regression equation. For the current dataset, Cook's Distance shows there is one data point over the red line that is an outlier that is also significant of changing the regression equation if removed.

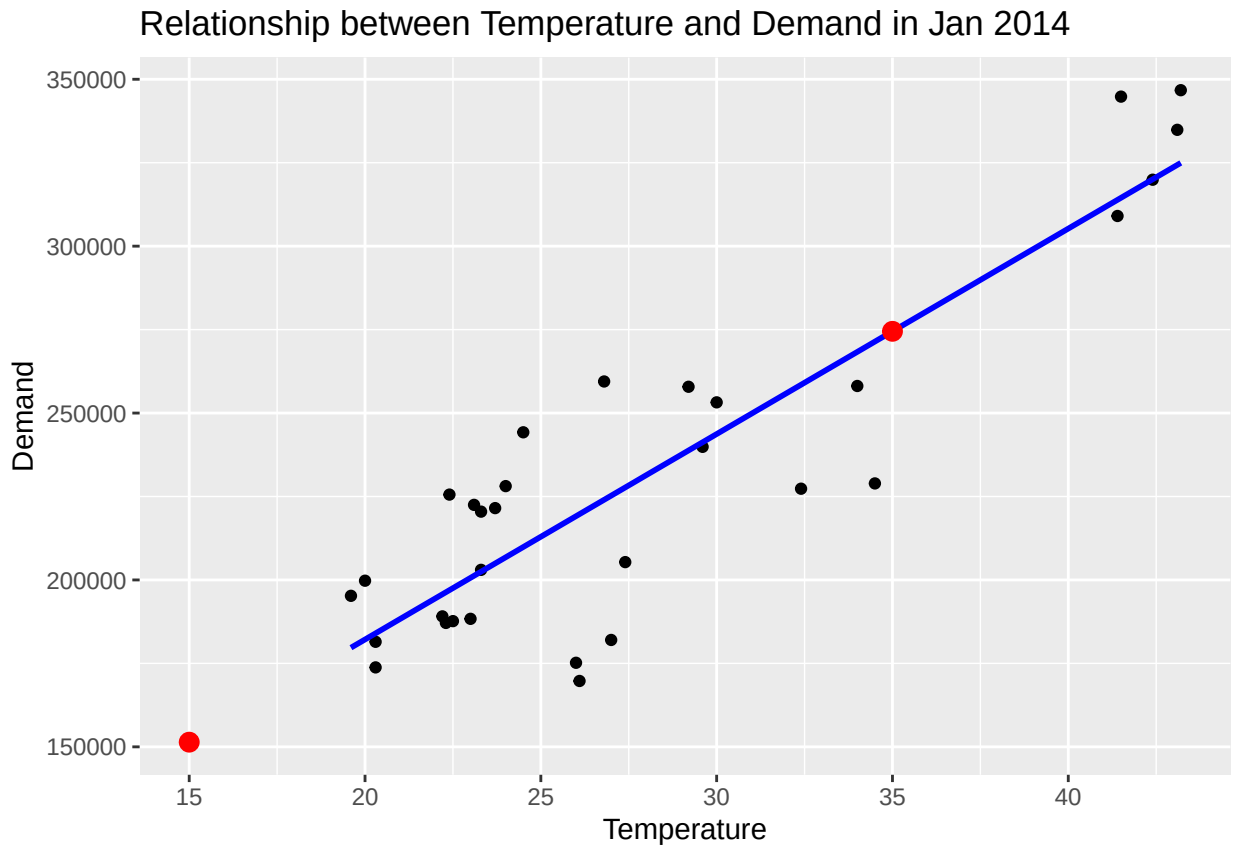
c. Use the model to forecast the electricity demand that you would expect for the next day if the maximum temperature was 15°C and compare it with the forecast if the with maximum temperature was 35°C. Do you believe these forecasts?

```
# Create new future scenarios
# Create a data frame for the new temperature scenarios
new_data <- data.frame(Temperature = c(15, 35))

# Use the model to forecast
forecasts <- predict(lm_model, new_data)

# Forecast new scenarios
forecasts
```

```
       1        2
151398.4 274484.2
```

```
library(ggplot2)

ggplot(jan_vic_elec, aes(x=Temperature, y=Demand)) +
    geom_point() +
    geom_smooth(method="lm", se=FALSE, color="blue") +
    labs(title="Relationship between Temperature and Demand in Jan 2014",
         x="Temperature", y="Demand") +
    annotate("point", x=c(15, 35), y=c(151398.4, 274484.2), color="red", size=3)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Answer "Do you believe these forecasts?" Short Answer: Yes, I believe in the forecast. First based on the results of the lm model report, we can break it down. The equation of the model line is Demand=59083.9+6154.3×Temperature. The intercept is 59083.9, which means that the demand is 59083.9 when the temperature is 0. The slope is 6154.3, which means for each unit increase in the tempearture is 6154.3. As the intercept and temperature coefficients are staistically significant as they are very close to 0.For the R^2 value of 0.7832, it means that 78.32% of the variability is explainable in demand using the temperature, which is relatively high with a good fit. For the residuals, the median value is close to 0 with a good sign. The range of residuals are large from minimum to maximum that indicate that some obeservations aren't that accurate due to possible outliers or non-linearity.Based on the context of the relationship as during January,

Australia has it's summer time it is reasonable with the increase of electricity for ac cooling with the increase in temperature. Also I plotted the two data points in red as it shows exactly on the best line of fit prediction for linear regression.

d. Give prediction intervals for your forecasts (hint: use `hilo %>% select(-.model)`).

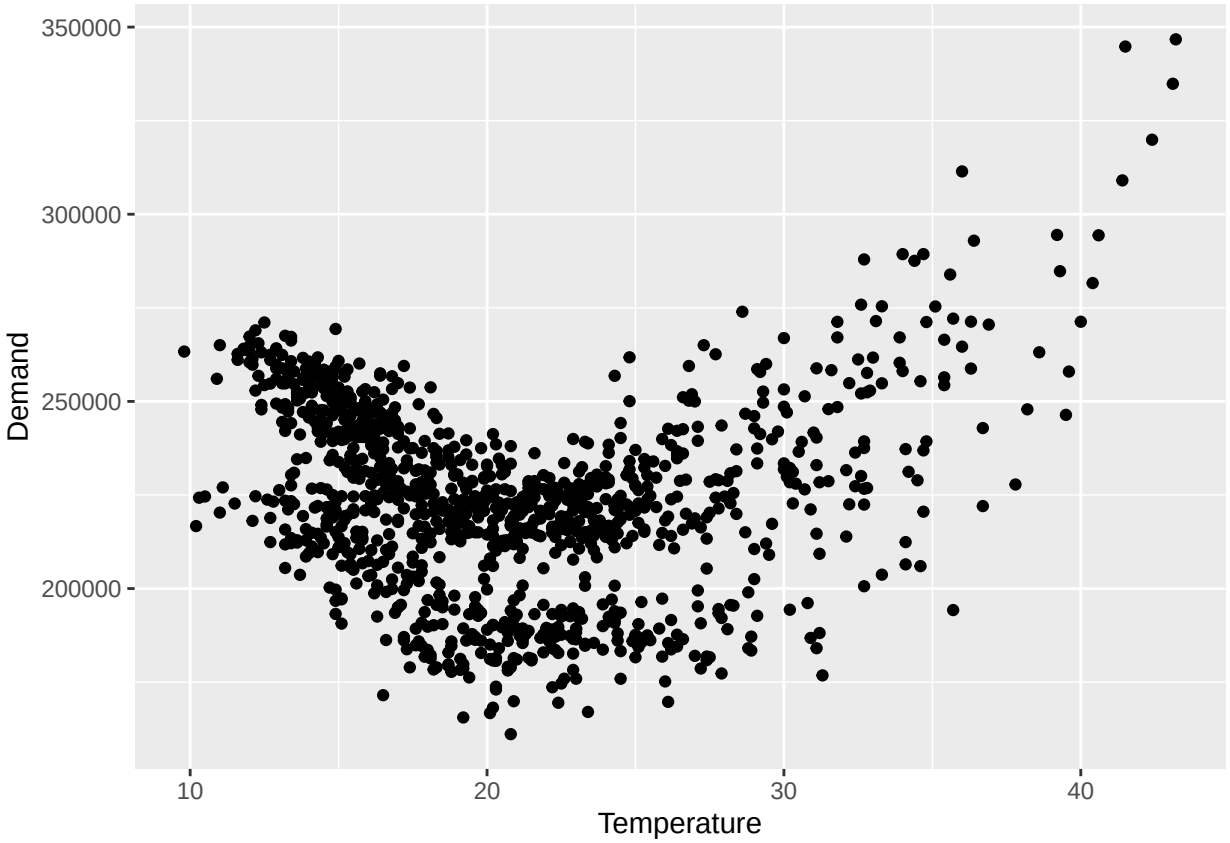
```
# Your provided new data scenarios
new_data <- data.frame(Temperature = c(15, 35))

# Use the model to forecast with prediction intervals
forecasts <- predict(lm_model, new_data, interval = "prediction")

# Print the forecasts with prediction intervals
print(forecasts)
```

```
       fit       lwr      upr
1 151398.4  97951.22 204845.5
2 274484.2 222783.69 326184.8
```

The prediction intervals for the forcasts of demand for 15 degrees is (97951.22,204845.5) and for 35 degree is (222783.69,326184.8)

e. Plot Demand vs Temperature for all of the available data in `vic_elec` aggregated to daily total demand and maximum temperature. What does this say about your model?

```
# This code is provided for you
vic_elec %>% # full dataset
  index_by(Date = as_date(Time)) %>% # index by time
  summarise( # summarize demand and temperature
    Demand = sum(Demand),
    Temperature = max(Temperature)
  ) %>%
  ggplot(aes(x = Temperature, y = Demand)) +
  geom_point() # scatterplot
```



Explain the pattern that you see and answer "What does this say about your model?" The pattern shows a bowl shaped curve U-shape pattern that indicates that there is high demand at low and high temperatures. This also indicates that there is a non-linear relationship between demand and temperature, leading to the requirement of a more complex model than a linear model.

2.

Data set `olympic_running` contains the winning times (in seconds) in each Olympic Games sprint, middle-distance and long-distance track events from 1896 to 2016.

```
#install.packages("broom")
#install.packages("dplyr")
#install.packages("glue")
library(broom)
library(dplyr)
library(glue)
```

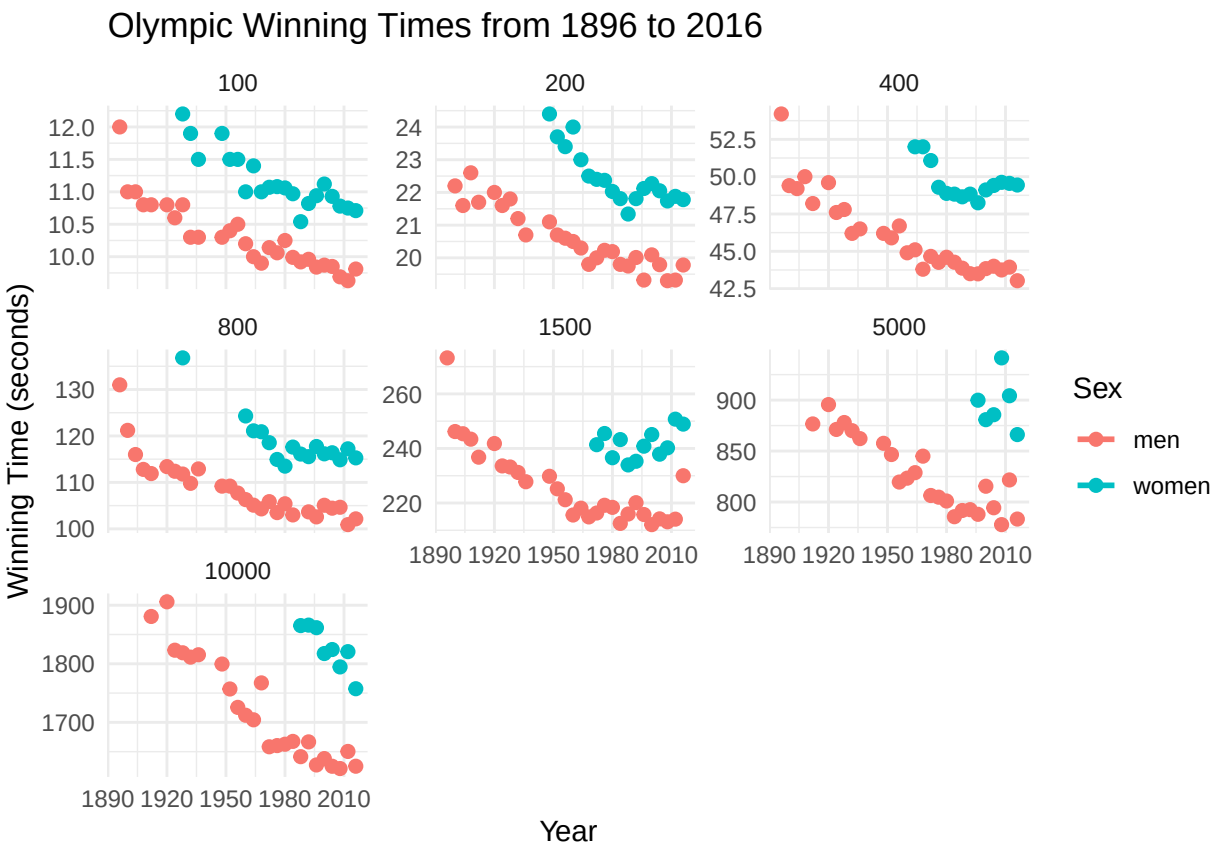
```
olympic_running
```

```
## # A tsibble: 312 x 4 [4Y]
## # Key:       Length, Sex [14]
##     Year Length Sex    Time
##    <int>  <int> <chr> <dbl>
```

```
##  1  1896    100 men    12
##  2  1900    100 men    11
##  3  1904    100 men    11
##  4  1908    100 men    10.8
##  5  1912    100 men    10.8
##  6  1916    100 men    NA
##  7  1920    100 men    10.8
##  8  1924    100 men    10.6
##  9  1928    100 men    10.8
## 10  1932    100 men    10.3
## # ... with 302 more rows
```

a. Plot the winning time against the year. Describe the main features of the plot. `facet_wrap` using `Length` and use `Sex` as `color`

```
library(ggplot2)

ggplot(olympic_running, aes(x = Year, y = Time, color = Sex)) +
  geom_line(aes(group = interaction(Sex, Length, Year)), size = 1) +
  geom_point(size = 2) +
  facet_wrap(~ Length, scales = "free_y") +
  labs(title = "Olympic Winning Times from 1896 to 2016",
       y = "Winning Time (seconds)",
       x = "Year",
       color = "Sex") +
  theme_minimal()
```



Describe the features (e.g., any patterns?) The features for the graph shows a negative relationship between winning time in seconds and years. In other words, men and women seem to become faster and faster with lower race times as the years past. Which means that there are more talented record breakers in recent years with athletes evolving.

b. Fit a regression (trend) line to the data. Obviously the winning times have been decreasing, but at what *average* rate per year?

```
fit <- lm(Time ~ ., data = olympic_running)
summary(fit)
```

```
Call:
lm(formula = Time ~ ., data = olympic_running)

Residuals:
    Min      1Q  Median      3Q     Max
-95.952 -13.610  -3.449  12.522 154.313

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.350e+02  1.205e+02   6.101 3.54e-09 ***
Year        -3.905e-01  6.152e-02  -6.347 8.92e-10 ***
```

```
Length       1.766e-01  6.132e-04 288.029  < 2e-16 ***
Sexwomen     3.297e+01  4.381e+00   7.526 7.36e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 31.96 on 277 degrees of freedom
  (31 observations deleted due to missingness)
Multiple R-squared:  0.9968,    Adjusted R-squared:  0.9967
F-statistic: 2.837e+04 on 3 and 277 DF,  p-value: < 2.2e-16
```

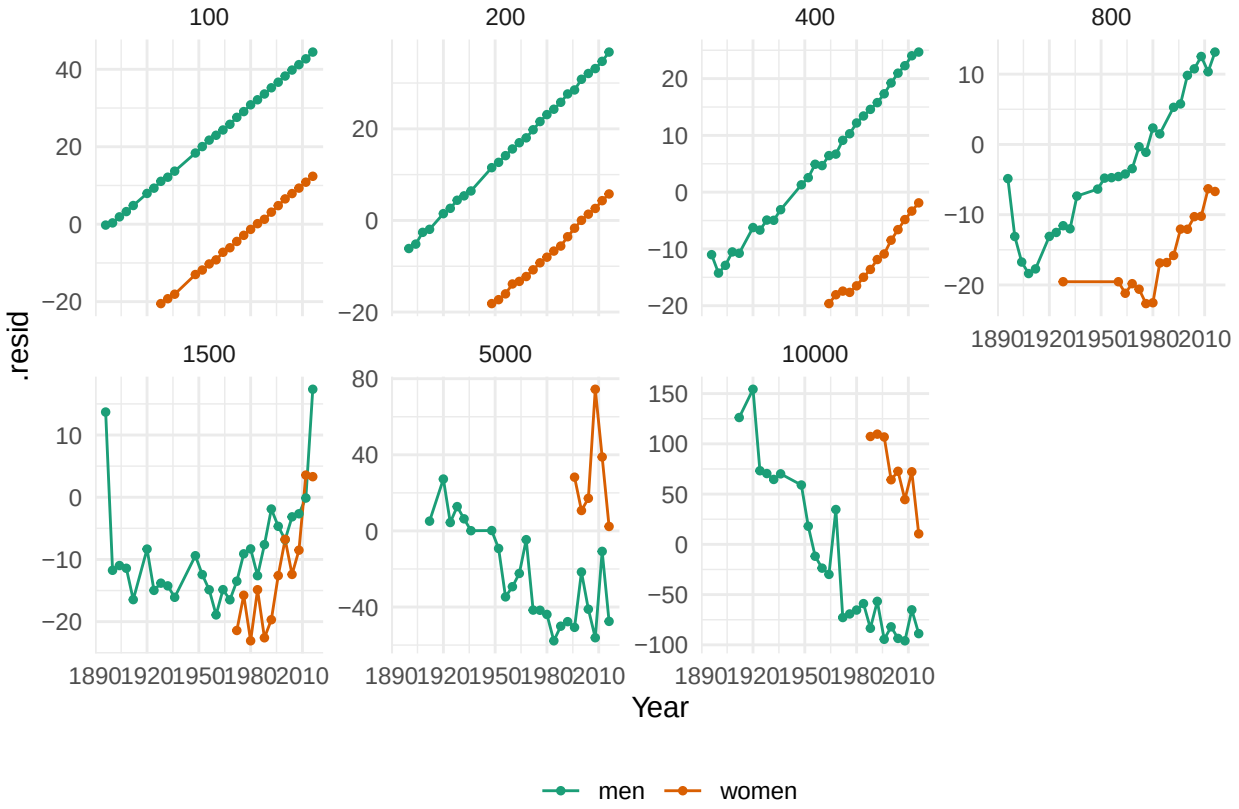
```
fit <- lm(Time ~ ., data = olympic_running)
tidy_output <- tidy(fit)

message <- tidy_output %>%
  filter(term == "Year") %>%
  glue::glue_data("The running time has been {ifelse(estimate<0, 'decreasing', 'increasing')} by an ave

print(message)
```

```
## The running time has been decreasing by an average of 0.391 seconds each year.<br>
```

The *average* rate per year of decreasing winning times are 0.391 seconds

c. Plot the residuals against the year. What does this indicate about the suitability of the fitted line?

```
# Code is provide for you
augment(fit) %>%
  ggplot(aes(x = Year, y = .resid, colour = Sex)) +
  geom_line() +
  geom_point(size = 1) +
  facet_wrap(~Length, scales = "free_y", nrow = 2) +
  theme_minimal() +
  scale_color_brewer(palette = "Dark2") +
  theme(legend.position = "bottom", legend.title = element_blank())
```



Answer "What does this indicate about the suitability of the fitted line?" Hint: Do the residuals follow any pattern? A normal distribution? The suitability of the fitted line shows a positive relationship(shorter lengths) and a negative relationship(longer lengths) between the residuals and the year with a normal distribution that indicates a systemmatic pattern in the errors that the model isn't capturing. The residuals should show no patterns with randomness that would incidate a good model, but this model clearly shows a pattern.

d. Predict the winning time for each race in the 2020 Olympics. Give a prediction interval for your forecasts. What assumptions have you made in these calculations? Hint: Do the times seem reasonable?

```
# Code is provide for you
#fit %>%
#  forecast(h = 1) %>%
#  mutate(PI = hilo(Time, 95)) %>%
#  select(-.model)
```

```
# 1. Create a dataframe for 2020 races
lengths <- c(100, 200, 400, 800, 1500, 5000, 10000)
sexes <- c("men", "women")

# Create all combinations of Year, Sex, and Length for 2020
data_2020 <- expand.grid(Year = 2020, Sex = sexes, Length = lengths)
```

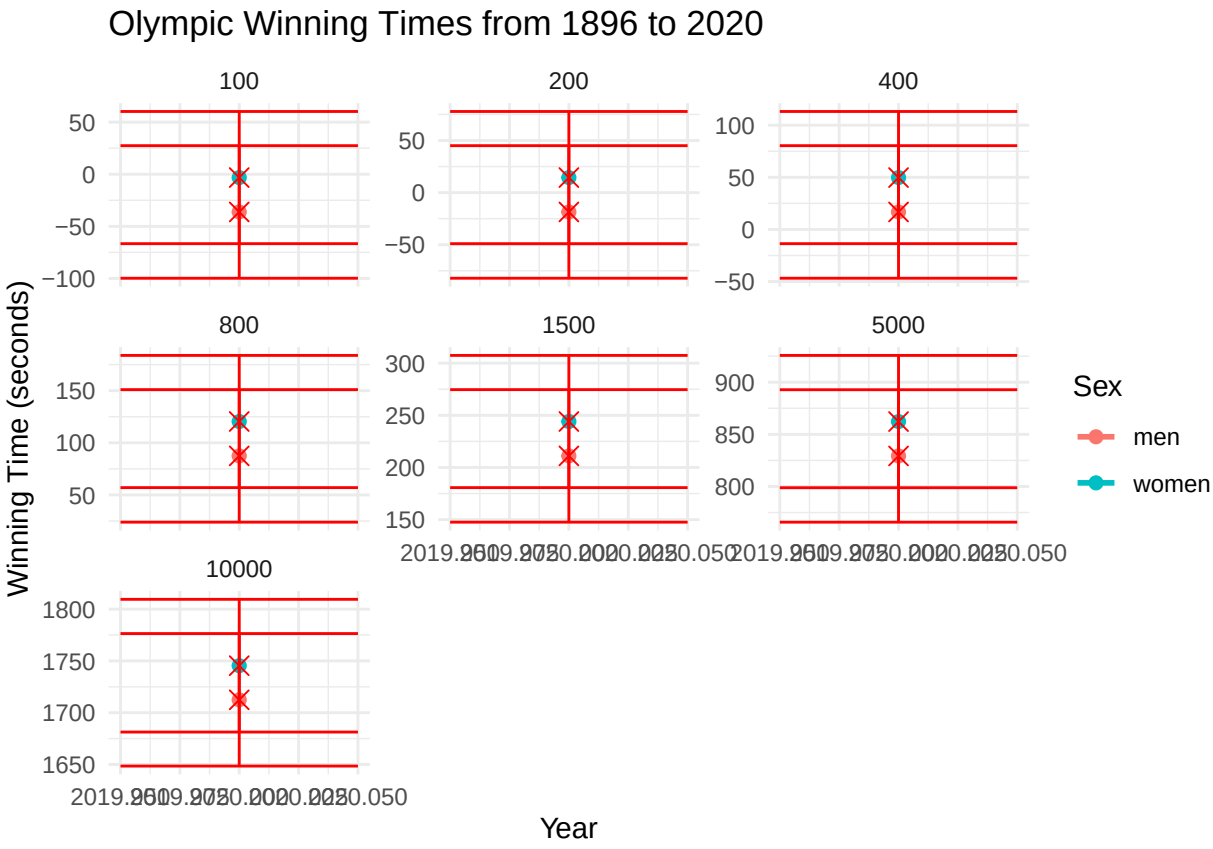
```r
# 2. Predict the winning times and the prediction intervals
predictions <- predict(fit, newdata = data_2020, interval = "prediction", level = 0.95)

# Assign predicted values and intervals to data_2020
data_2020$Time <- predictions[, "fit"]
data_2020$Lower <- predictions[, "lwr"]
data_2020$Upper <- predictions[, "upr"]

# Display the predictions and intervals
data_2020
```

```
##    Year   Sex Length        Time      Lower      Upper
## 1  2020   men    100  -36.198643  -99.80030   27.40301
## 2  2020 women    100   -3.225449  -66.65147   60.20057
## 3  2020   men    200  -18.535672  -82.13021   45.05886
## 4  2020 women    200   14.437522  -48.98403   77.85907
## 5  2020   men    400   16.790269  -46.79071   80.37125
## 6  2020 women    400   49.763463  -13.64983  113.17676
## 7  2020   men    800   87.442152   23.88555  150.99875
## 8  2020 women    800  120.415347   57.01581  183.81489
## 9  2020   men   1500  211.082948  147.56019  274.60570
## 10 2020 women   1500  244.056142  180.67183  307.44046
## 11 2020   men   5000  829.286925  765.76482  892.80903
## 12 2020 women   5000  862.260119  798.78301  925.73723
## 13 2020   men  10000 1712.435463 1648.42862 1776.44230
## 14 2020 women  10000 1745.408657 1681.31423 1809.50309
```

```r
# Assuming you've created the 'data_2020' dataframe with predicted values and intervals

ggplot(data_2020, aes(x = Year, y = Time, color = Sex)) +
  geom_line(aes(group = interaction(Sex, Length, Year)), size = 1) +
  geom_point(size = 2) +
  geom_point(data = data_2020, aes(x = Year, y = Time), color = "red", size = 3, shape = 4) +
  geom_errorbar(data = data_2020, aes(x = Year, ymin = Lower, ymax = Upper), color = "red", width = 0.1
  facet_wrap(~ Length, scales = "free_y") +
  labs(title = "Olympic Winning Times from 1896 to 2020",
       y = "Winning Time (seconds)",
       x = "Year",
       color = "Sex") +
  theme_minimal()
```

```
## geom_path: Each group consists of only one observation. Do you need to adjust
## the group aesthetic?
## geom_path: Each group consists of only one observation. Do you need to adjust
## the group aesthetic?
## geom_path: Each group consists of only one observation. Do you need to adjust
## the group aesthetic?
## geom_path: Each group consists of only one observation. Do you need to adjust
## the group aesthetic?
## geom_path: Each group consists of only one observation. Do you need to adjust
## the group aesthetic?
## geom_path: Each group consists of only one observation. Do you need to adjust
## the group aesthetic?
```

```
## geom_path: Each group consists of only one observation. Do you need to adjust
## the group aesthetic?
```



Answer "What assumptions have you made in these calculations?" Comment on whether the times are reasonable. For the assumptions I made it based on principles of GLMs - general linear models 1. Linearity: I assumed that there was a linear relationship between the predictor variables(year) and response variables(winning times, which means that the predictor variables are associate with constant changes respective to the response variable. 2. Independence: I assumed that the obervations were indpendent of each other that means that the peformance of one athlete does not affect another athlete. 3. Homoscedasticity: I assumed that the variance of residuals(errors) is constant across all of the levels of the predictor variable changes 4. Normality of Errors: I assumed that the residuals()errors: are normally distributed as it is important for valid hypothesis testing and confidence interval construction 5. No Multicollinearity: I assumed that the predictor variables are not highly correlated with each other. Higher multi collinearity can mmake it hard to interpret the individuals of predictors 6. Correctness of the model: I assumed that model is correctly configured with no important predcitors missing 7. Constant Predictors for Forecasting: I assumed that the model forecasting for 2020 remain constant with no new abnormal events that would affect the winning times Conclusion: (Not sufficient model)For the winning times, I think that the winning times are unreasonable as there are results with negative time stamps and time stamps that are abnormally high. This is due to the simplicity of the model as shown with the residuals showing a pattern in the previous steps, which indicates that we need a more complex model to predict the winning times.

4.

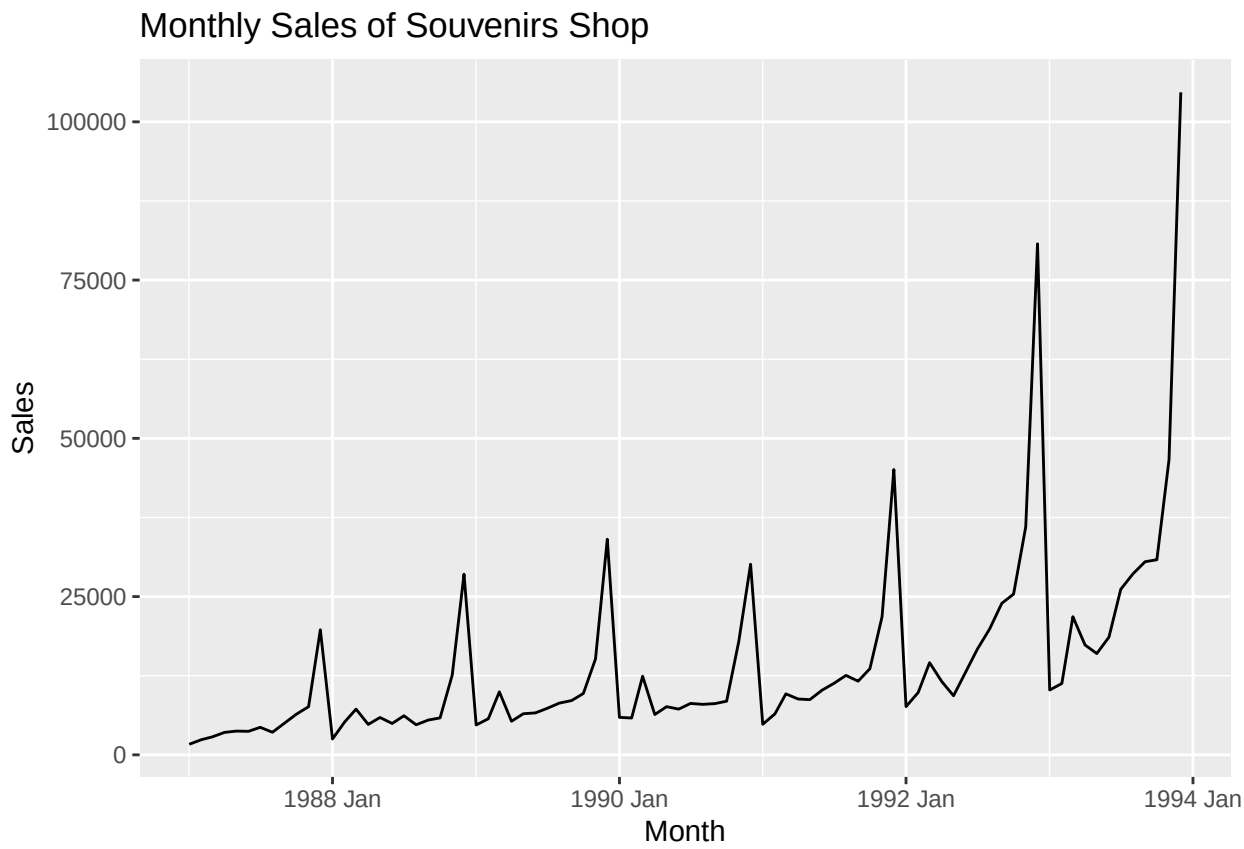
The data set `souvenirs` concerns the monthly sales figures of a shop which opened in January 1987 and sells gifts, souvenirs, and novelties. The shop is situated on the wharf at a beach resort town in Queensland, Australia. The sales volume varies with the seasonal population of tourists. There is a large influx of visitors to the town at Christmas and for the local surfing festival, held every March since 1988. Over time, the shop has expanded its premises, range of products, and staff.

a. Produce a time plot of the data and describe the patterns in the graph. Identify any unusual or unexpected fluctuations in the time series.

```
# Plot `souvenirs` data
souvenirs
```

```
# A tsibble: 84 x 2 [1M]
      Month Sales
      <mth> <dbl>
 1 1987 Jan 1665.
 2 1987 Feb 2398.
 3 1987 Mar 2841.
 4 1987 Apr 3547.
 5 1987 May 3753.
 6 1987 Jun 3715.
 7 1987 Jul 4350.
 8 1987 Aug 3566.
 9 1987 Sep 5022.
10 1987 Oct 6423.
# ... with 74 more rows
```

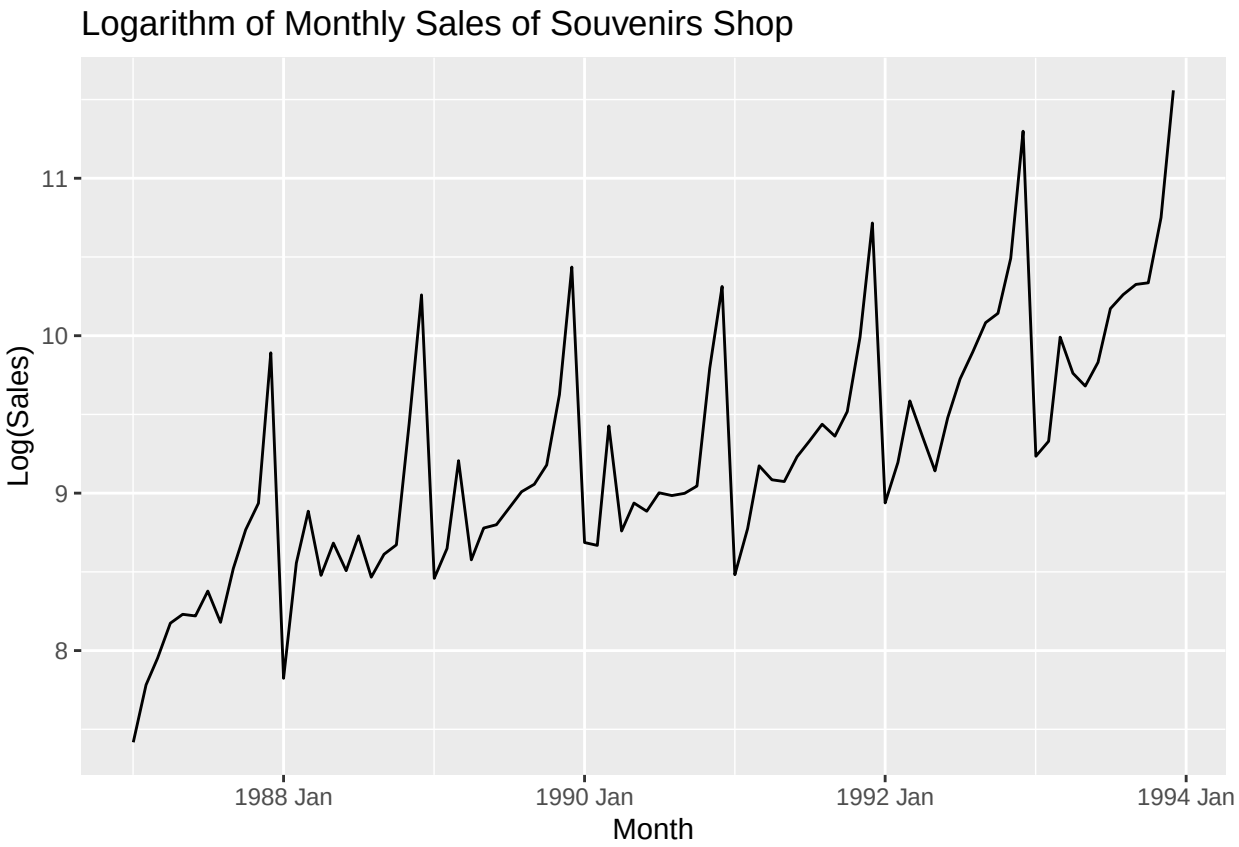
```
# Plot `Sales`
# Create a time plot using ggplot2
ggplot(souvenirs, aes(x = Month, y = Sales)) +
  geom_line() +
  labs(x = "Month", y = "Sales", title = "Monthly Sales of Souvenirs Shop")
```



Discuss any patterns you see in the data The data has patterns of a constant sudden fluctuation right before Janurary with a rapid spike of sales and then a rapid decrease.

b. Explain why it is necessary to take logarithms of these data before fitting a model.

```
# Taking logarithm of the data (and plot)

# Taking logarithm of the `Sales` data
souvenirs$log_sales <- log(souvenirs$Sales)

# Create a plot of the logarithm of sales using ggplot2
ggplot(souvenirs, aes(x = Month, y = log_sales)) +
  geom_line() +
  labs(x = "Month", y = "Log(Sales)", title = "Logarithm of Monthly Sales of Souvenirs Shop")
```



Explain what differences you see from the first plot I see the differences with a better representation of the fluctions and plotting of the data. On a side note for logarithmic plots, it is done to address issues related to the distribution and the variance of the data when the data shows exponential or multiplicative behavior. 1. Stabilitizing Variance: The variance of the data increases as the values get larger leading to heteroscedasticity, where the spread of residuals varies across the range of the dependent variable. Takeing the lgarithm of the data can stabilizze the variance, making it more constant across the range of the values and meeting one of the assumptions of linear regression models 2. Lineariziaing Relationships: The logarithm can make it easier to transform the relationship into a more linear form. 3. Normlalizing Distributions: If the original data is skewed or not a normal distribution, the logarithm can transform the data into a more symmetric and more normal distribution like representation. 4. Interpretable Coefficient: When you fit a linear regression model to logarithmically transformed data, the coefficients have an interpretations in terms of percentage change rather than absolute change that could be more meaningful in different contexts 5. Homoscedasticity: By transforming the data to stabilize the variance, taking logarithms can help the assumption of homoscedasticity, which assumes that the residuals have constant variance 6. Outlier Handling> Logarithmic transformation can reduce the impact of extreme values (outliers) by compressing their influence on the model 7. Residual patterns: Taking logarithms can help mitigate specific patterns in the residuals, such as funnel shaped patterns, which can occur when the spread of residuals changes with the level of the dependent varible.

c. Fit a regression model to the logarithms of these sales data with a linear trend, seasonal dummies and a "surfing festival" dummy variable.

```
# Data with festival is created for you
souvenirs_festival <- souvenirs %>%
  mutate(festival = month(Month) == 3 & year(Month) != 1987)

# Fit model with trend (log), season, and festival
model <- lm(log_sales ~ Month + festival, data = souvenirs_festival)

# Predict the log of sales using the fitted model
souvenirs_festival$predicted_log_sales <- predict(model, newdata = souvenirs_festival)

# Transform the predicted log sales back to the original scale
souvenirs_festival$predicted_sales <- exp(souvenirs_festival$predicted_log_sales)

# Plot fitted model with `souvenirs` data
# Plot `Sales`
ggplot() +
  geom_line(data = souvenirs, aes(x = Month, y = Sales, color = "Original Sales"), alpha = 0.5, size = 1
  geom_line(data = souvenirs_festival, aes(x = Month, y = predicted_sales, color = "Fitted Model"), size
  labs(x = "Month", y = "Sales", title = "Fitted Model and Original Sales of Souvenirs Shop") +
  theme_minimal() +
  scale_color_manual(values = c("blue", "red"), labels = c("Fitted Model", "Original Sales")) +
  guides(color = guide_legend(title = NULL))
```