Tutorial 3 Memo

STSA2626

2025-05-08

First we load the necessary packages.

```
# Load necessary libraries
library(tidyverse)
library(ggplot2)
library(ggthemes)
library(reshape2)
library(corrplot)
library(GGally)
library(ggpubr)
library(dplyr)
```

Q1.

First we call the dataset. We transform the numeric Month variable (5–9) into a factor with descriptive labels as c("May", "June", "July", "August", "September").

```
# Load and clean the data
data("airquality")
airquality = airquality %>%
  mutate(Month = factor(Month, labels = c("May", "June", "July", "August",
"September")))
```

1. basic Statistical Analysis

Next we need to find if any missing value is there in every column of the dataset. Note that, each column represents a variable in the dataset.

```
# 1. Basic summary
summary(airquality)
```

```
##      Ozone           Solar.R           Wind             Temp
##  Min.   :  1.00   Min.   :  7.0   Min.   : 1.700   Min.   :56.00
##  1st Qu.: 18.00   1st Qu.:115.8   1st Qu.: 7.400   1st Qu.:72.00
##  Median : 31.50   Median :205.0   Median : 9.700   Median :79.00
##  Mean   : 42.13   Mean   :185.9   Mean   : 9.958   Mean   :77.88
##  3rd Qu.: 63.25   3rd Qu.:258.8   3rd Qu.:11.500   3rd Qu.:85.00
##  Max.   :168.00   Max.   :334.0   Max.   :20.700   Max.   :97.00
##  NA's   :37       NA's   :7
##        Month         Day
##  May      :31   Min.   : 1.0
##  June     :30   1st Qu.: 8.0
##  July     :31   Median :16.0
##  August   :31   Mean   :15.8
##  September:30   3rd Qu.:23.0
##                 Max.   :31.0
##

colSums(is.na(airquality))

##   Ozone Solar.R    Wind    Temp   Month     Day
##      37       7       0       0       0       0
```

Note that there are 37 missing values in the column ‘Ozone’ and 7 missing values in the column ‘Solar.R’. It is possible to remove those columns including missing values as follows:

```
# Load necessary packages
library(dplyr)

# Remove rows with missing values
airquality_clean = airquality %>%
  drop_na()  # or use: na.omit(airquality)

# 1. Basic summary after cleaning
summary(airquality_clean)

##      Ozone          Solar.R           Wind            Temp
Month
##  Min.   :  1.0   Min.   :  7.0   Min.   : 2.30   Min.   :57.00   May
:24
##  1st Qu.: 18.0   1st Qu.:113.5   1st Qu.: 7.40   1st Qu.:71.00   June
: 9
##  Median : 31.0   Median :207.0   Median : 9.70   Median :79.00   July
:26
##  Mean   : 42.1   Mean   :184.8   Mean   : 9.94   Mean   :77.79   August
:23
##  3rd Qu.: 62.0   3rd Qu.:255.5   3rd Qu.:11.50   3rd Qu.:84.50
September:29
##  Max.   :168.0   Max.   :334.0   Max.   :20.70   Max.   :97.00
##       Day
##  Min.   : 1.00
##  1st Qu.: 9.00
##  Median :16.00
##  Mean   :15.95
##  3rd Qu.:22.50
##  Max.   :31.00

colSums(is.na(airquality_clean))

##   Ozone Solar.R    Wind    Temp   Month     Day
##       0       0       0       0       0       0
```

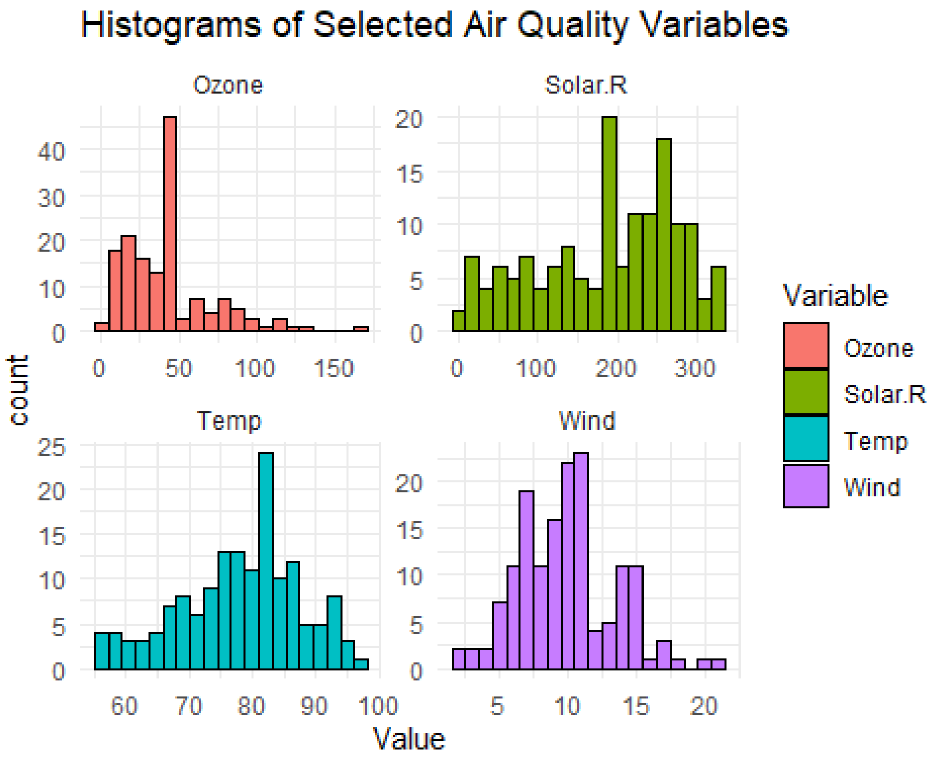
Now there are no missing values in the columns. But removing missing values would lead to loss of information we have in a dataset. We do not want to discard any information/data, rather use the available information to estimate the missing values. This can be done using **‘imputation’**.

```
# Impute missing values with column means
airquality_imputed = airquality %>%
  mutate(across(everything(), ~ ifelse(is.na(.), mean(., na.rm = TRUE), .)))

# We replace the actual data with the imputed data
airquality = airquality_imputed
```

2. Histogram

Next we create a set of histograms for each quantitative variables other than the ‘Month’ and ‘Day’ variables in the data.

```
# 2. Histogram plots
airquality %>%
  select(Ozone, Solar.R, Wind, Temp) %>%
  gather(key = "Variable", value = "Value") %>%
  ggplot(aes(x = Value, fill = Variable)) +
  geom_histogram(bins = 20, color = "black") +
  facet_wrap(~ Variable, scales = "free") +
  theme_minimal() +
  labs(title = "Histograms of Selected Air Quality Variables")
```



The histograms show the distribution of Ozone, Solar Radiation, Wind, and Temperature.

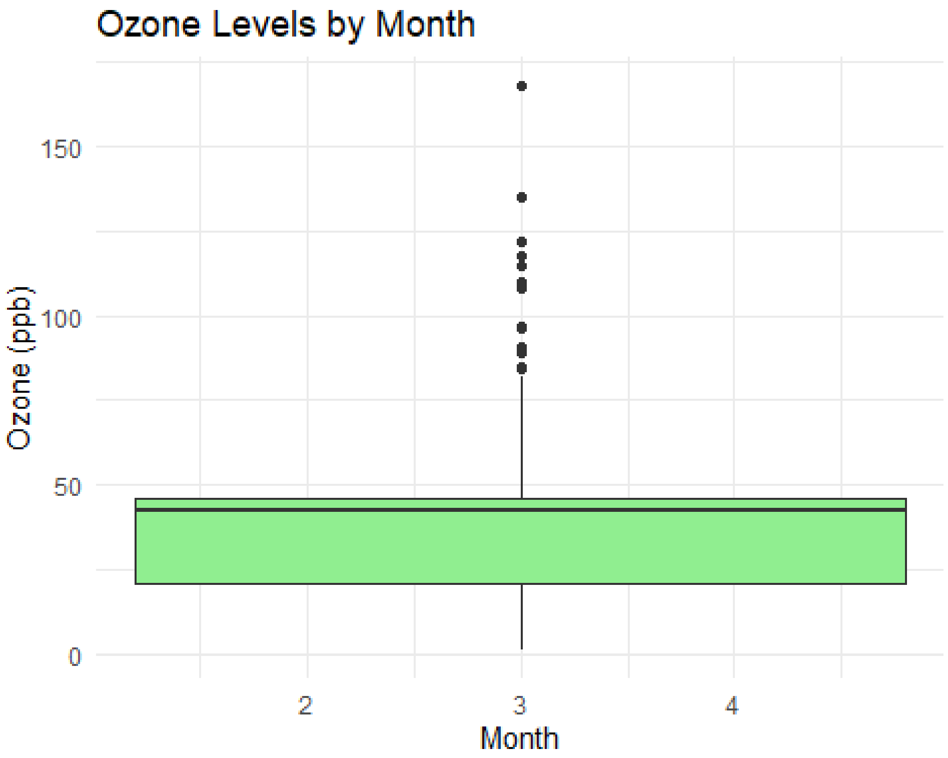
Ozone and Solar.R are right-skewed, suggesting most days have low to moderate levels, with a few very high outliers (exceptions).

Wind and Temp are more symmetrically distributed but still show slight skewness.

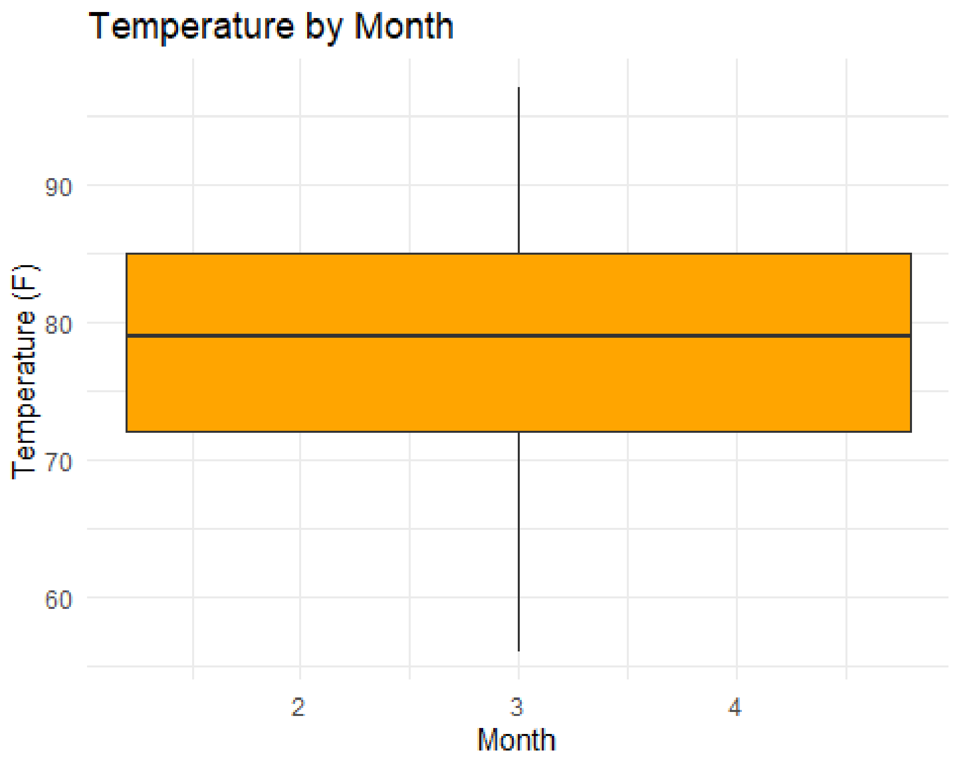
3. Boxplot

Next we create some boxplots.

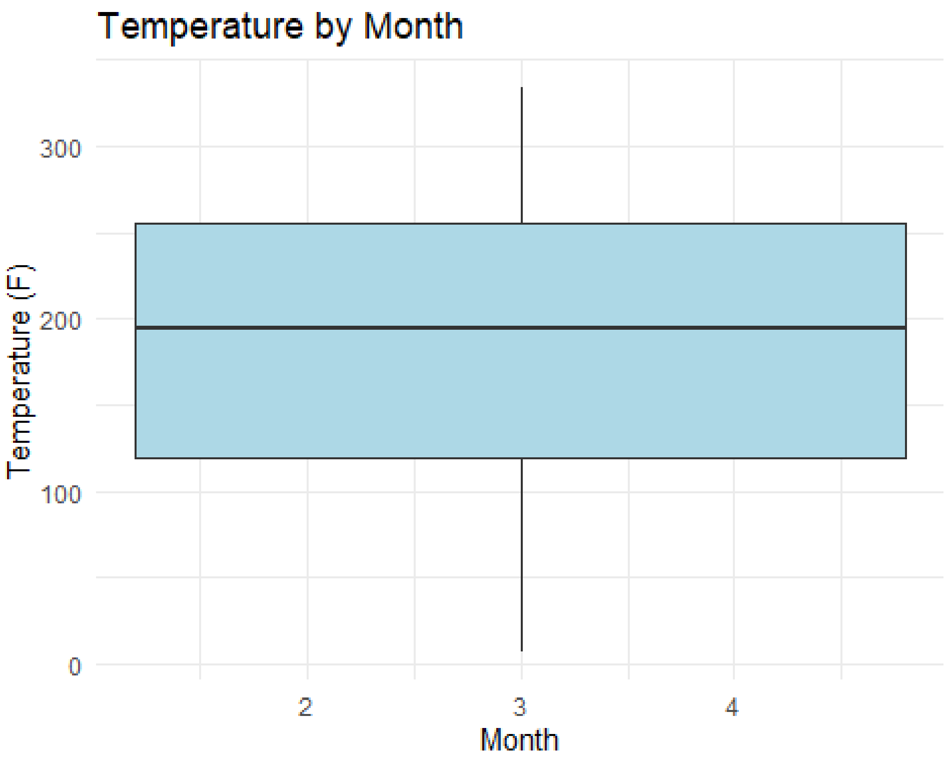
```
# 3. Boxplots by Month
ggplot(airquality, aes(x = Month, y = Ozone)) +
  geom_boxplot(fill = "lightgreen") +
  labs(title = "Ozone Levels by Month", y = "Ozone (ppb)") +
  theme_minimal()
```



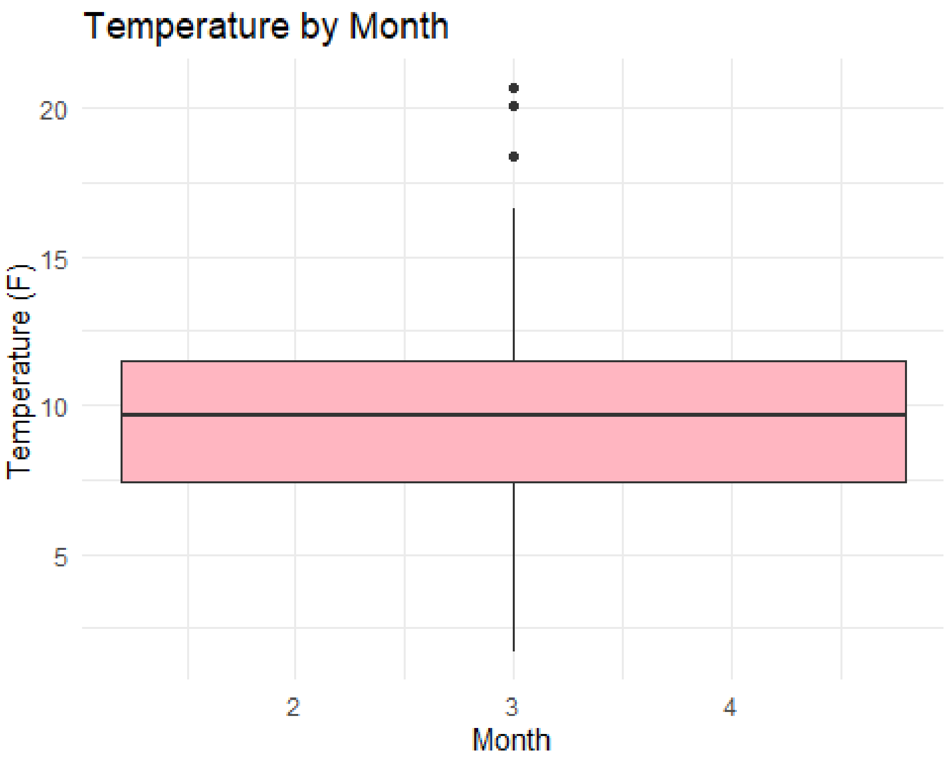
```
ggplot(airquality, aes(x = Month, y = Temp)) +
  geom_boxplot(fill = "orange") +
  labs(title = "Temperature by Month", y = "Temperature (F)") +
  theme_minimal()
```



```
ggplot(airquality, aes(x = Month, y = Solar.R)) +
  geom_boxplot(fill = "lightblue") +
  labs(title = "Temperature by Month", y = "Temperature (F)") +
  theme_minimal()
```



```r
ggplot(airquality, aes(x = Month, y = Wind)) +
  geom_boxplot(fill = "lightpink") +
  labs(title = "Temperature by Month", y = "Temperature (F)") +
  theme_minimal()
```



Boxplots by Month indicate seasonal variation:

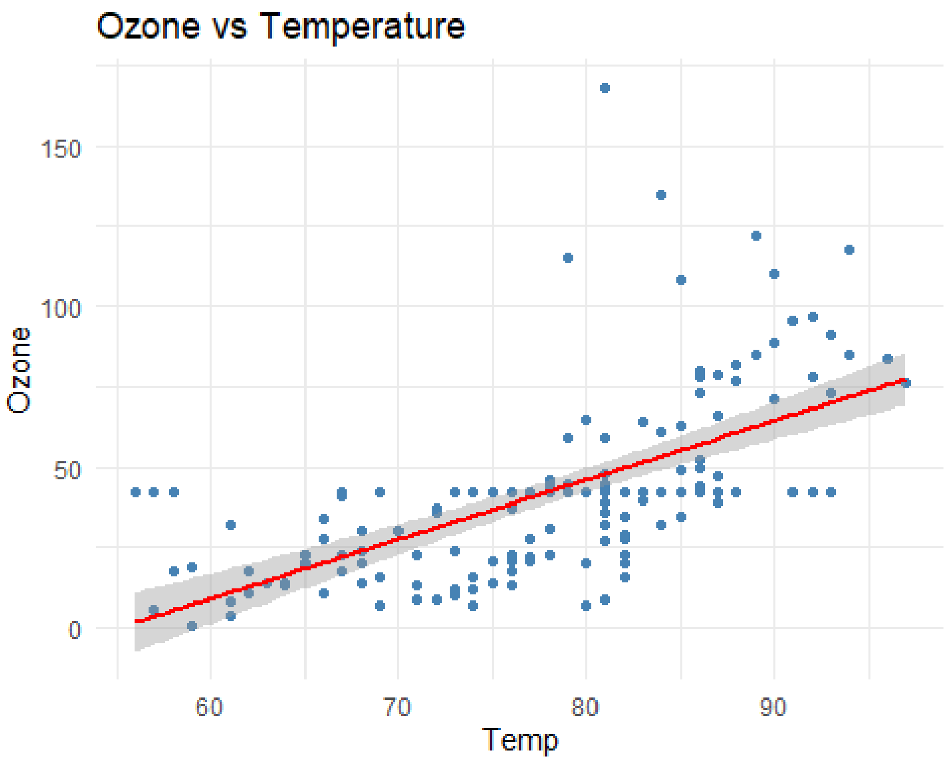
Ozone increases from May to July and slightly decreases afterward.

Temperature peaks in July and August.

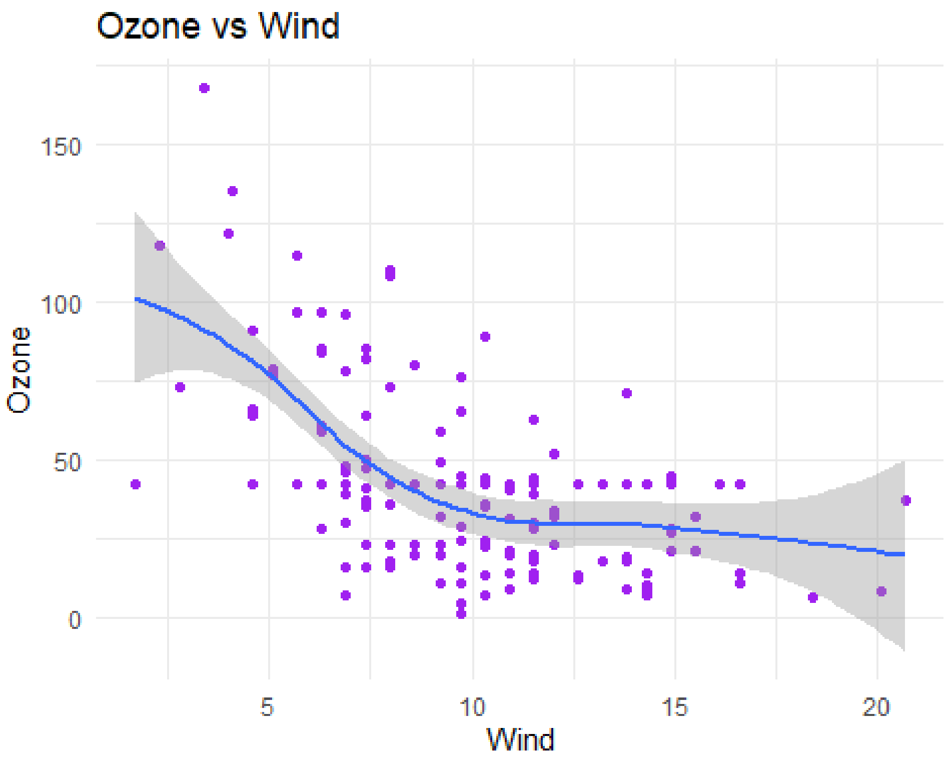
Solar.R and Wind also vary across months but with more fluctuations and outliers.

4. Scatter plot with smoothing

```
# 4. Scatter plots with smoothing
ggplot(airquality, aes(x = Temp, y = Ozone)) +
  geom_point(color = "steelblue") +
  geom_smooth(method = "lm", se = TRUE, color = "red") +
  labs(title = "Ozone vs Temperature") +
  theme_minimal()
```



```r
ggplot(airquality, aes(x = Wind, y = Ozone)) +
  geom_point(color = "purple") +
  geom_smooth(method = "loess", se = TRUE) +
  labs(title = "Ozone vs Wind") +
  theme_minimal()
```



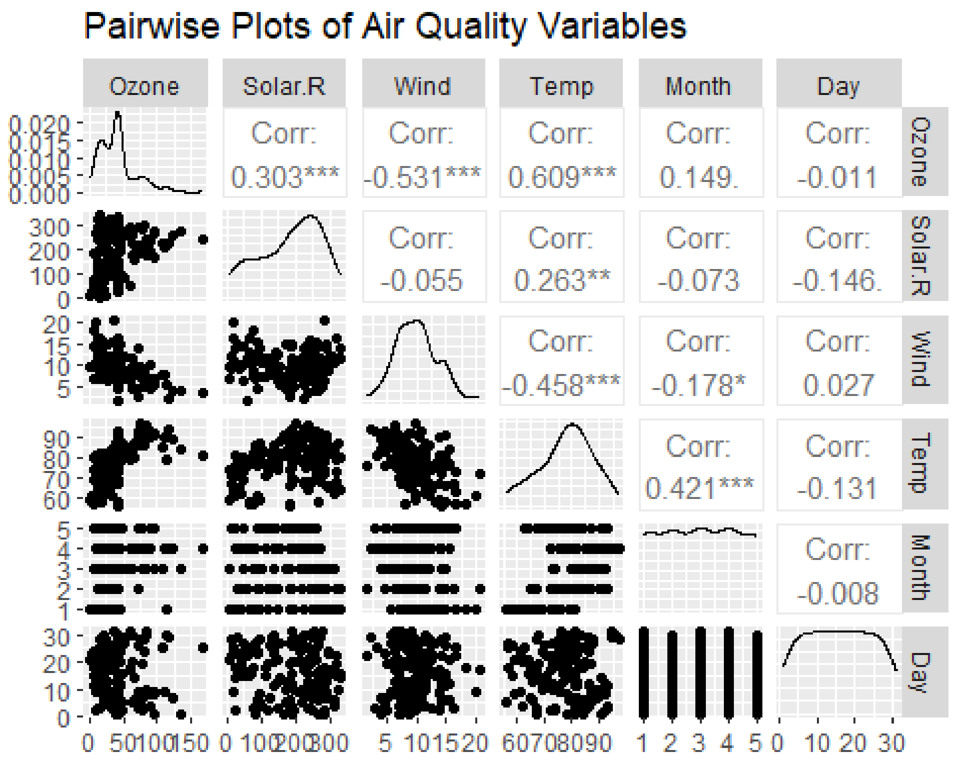
The Ozone vs Temp plot shows a positive linear relationship: higher temperatures are associated with higher ozone levels.

The Ozone vs Wind plot suggests a negative nonlinear relationship: ozone tends to decrease as wind increases, especially at lower wind speeds.

Smoothing lines (linear and LOESS) shows general downward nonlinear pattern for ozone values against the wind speed values.

5. Pairwise scatterplot matrix

```r
# 5. Pairwise scatterplot matrix
cor_data = airquality %>% select_if(is.numeric) %>% na.omit()
ggpairs(cor_data, title = "Pairwise Plots of Air Quality Variables")
```



This matrix highlights pairwise relationships among numeric variables.

We can observe:

Strongest positive correlation: Ozone vs Temp.

Weak negative correlation: Ozone vs Wind.

6. Linear Regression Model

```
# 6. Linear regression model
model = lm(Ozone ~ Temp + Wind + Solar.R, data = airquality)
summary(model)

##
## Call:
## lm(formula = Ozone ~ Temp + Wind + Solar.R, data = airquality)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -38.618 -14.491  -5.054  12.270 101.176
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -38.22315   18.88338  -2.024  0.04474 *
## Temp          1.24126    0.20906   5.937 1.96e-08 ***
## Wind         -2.71725    0.54280  -5.006 1.55e-06 ***
## Solar.R       0.05775    0.02003   2.883  0.00452 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.9 on 149 degrees of freedom
## Multiple R-squared:   0.48,  Adjusted R-squared:  0.4696
## F-statistic: 45.85 on 3 and 149 DF,  p-value: < 2.2e-16
```

All three predictors are statistically significant.

Temperature has the strongest positive effect on ozone levels.

Wind has a strong negative effect.

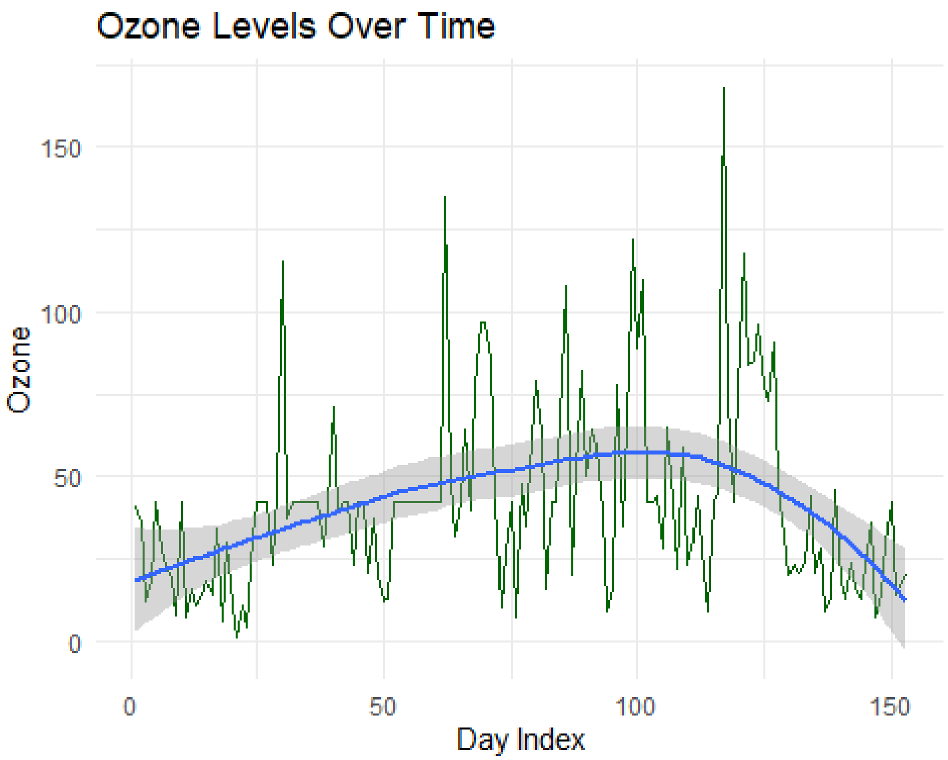
Solar.R has a weak positive effect.

Model's R^2 = 0.48 indicates that the model explains 48% of the variation in Ozone — a reasonably good fit.

Residual diagnostics suggest model assumptions are largely met.

7. Time Series Plot

```r
# 7. Ozone over time (index)
airquality$DayIndex = 1:nrow(airquality)
ggplot(airquality, aes(x = DayIndex, y = Ozone)) +
  geom_line(color = "darkgreen") +
  geom_smooth(method = "loess") +
  labs(title = "Ozone Levels Over Time", x = "Day Index") +
  theme_minimal()
```

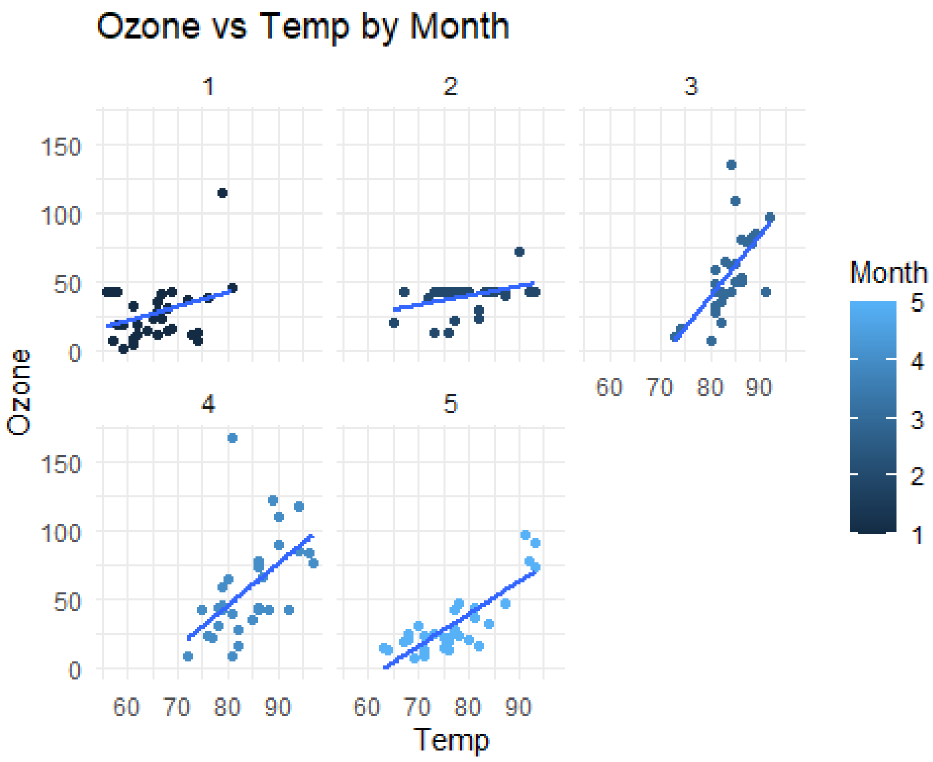


Shows temporal variation in ozone levels from May to September.

Loess smoothing shows an increase in ozone mid-year (summer peak), then tapering off.

8. Faceted scatter plot

```r
# 8. Faceted scatter plot
ggplot(airquality, aes(x = Temp, y = Ozone)) +
  geom_point(aes(color = Month)) +
  geom_smooth(method = "lm", se = FALSE) +
  facet_wrap(~ Month) +
  theme_minimal() +
  labs(title = "Ozone vs Temp by Month")
```



Ozone vs Temp by month shows:

Positive relationships across months.

But steeper trends in warmer months like July and August.

Q2.

The dataset contains the following variables for 100 countries:

- Country_ID: Unique country identifier
- GDP_per_capita: Income per person
- Unemployment_rate: Unemployment rate (%)
- Education_index: Composite measure of education

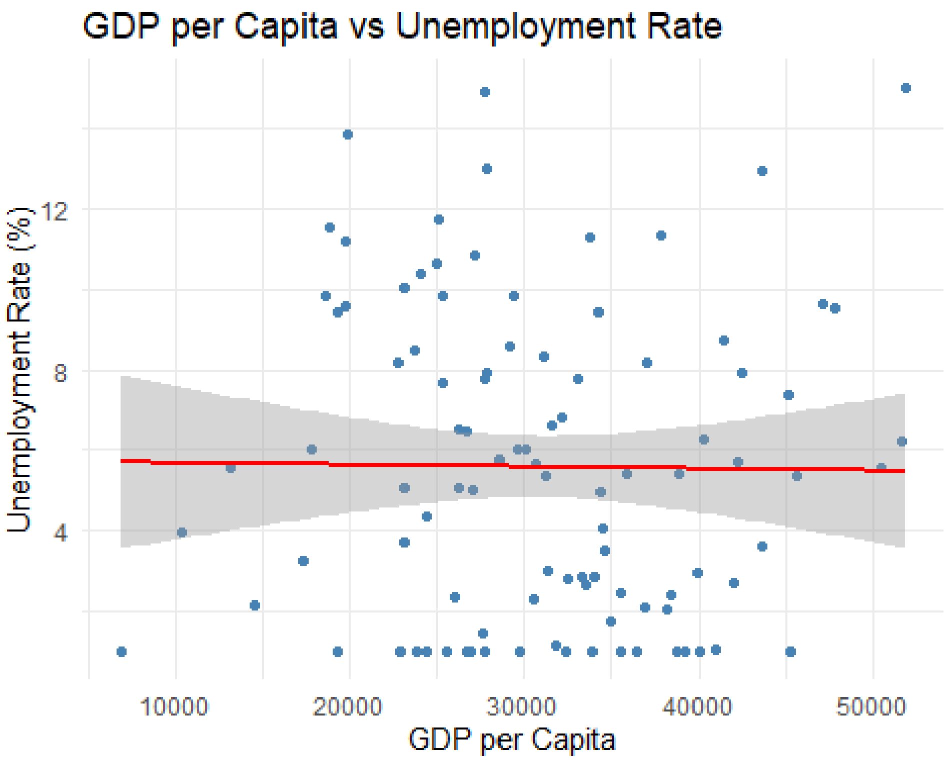
To verify the claim by the data analyst, we perform the following analysis in R:

- Visualization
- Correlation
- Regression

Scatter Plot

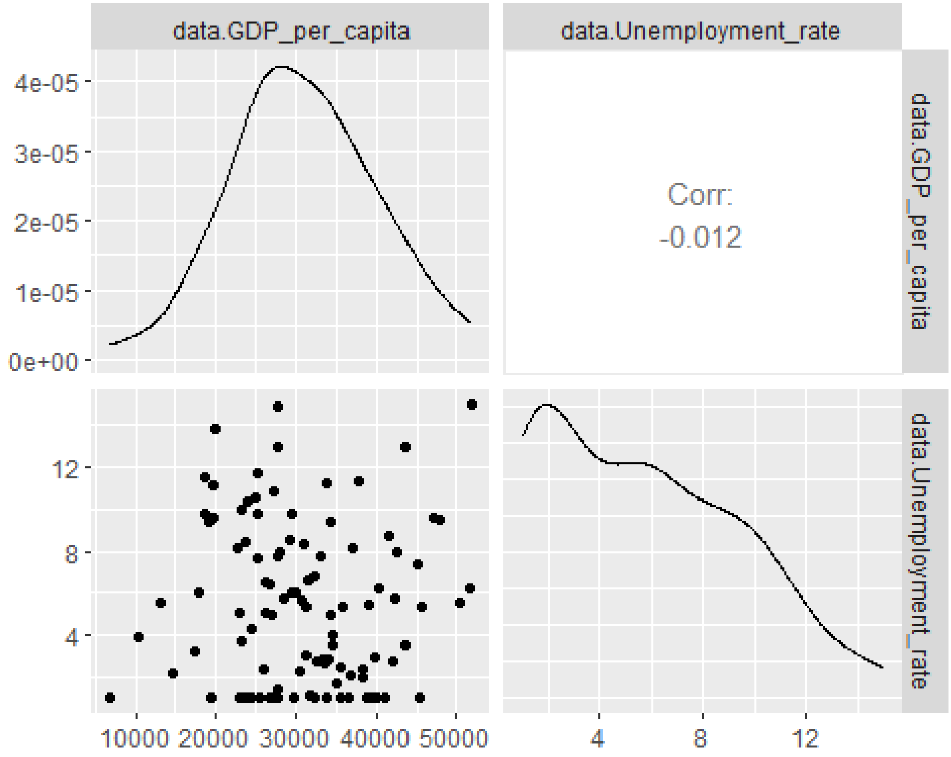
First we create a scatter plot.

```
# Load data
data = read.csv("C:\\Users\\ChakrabortyN\\OneDrive - University of the Free
State\\Desktop\\economics2.csv", header = T)

str(data)

## 'data.frame':    100 obs. of  5 variables:
##  $ X                : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Country_ID       : chr  "C1" "C2" "C3" "C4" ...
##  $ GDP_per_capita   : int  24395 27698 45587 30705 31293 47151 34609 17349
23131 25543 ...
##  $ Unemployment_rate: num  4.35 14.92 5.35 5.66 3.03 ...
##  $ Education_index  : num  0.81 0.31 0.81 0.77 0.71 0.61 0.4 0.31 0.59
0.62 ...

# Scatter plot of GDP vs Unemployment
ggplot(data, aes(x = GDP_per_capita, y = Unemployment_rate)) +
  geom_point(color = "steelblue") +
  geom_smooth(method = "lm", se = TRUE, color = "red") +
  labs(title = "GDP per Capita vs Unemployment Rate",
       x = "GDP per Capita",
       y = "Unemployment Rate (%)") +
  theme_minimal()
```



From the scatter plot, any linear relationship between the GDP/capita and the unemployment rate is not obvious.

```
ggpairs(data.frame( data$GDP_per_capita,data$Unemployment_rate ))
```



The correlation plot and the correlation value shows that there is no significant correlation between these two variables. Hence, the statement by the analyst is unlikely to be true.

It is not reasonable to fit a linear regression model to a dataset when the variables are uncorrelated. This is due to the absence of linear relationship among the variables in the data. However, to justify the conclusion that a liner regression is invalid for this data, we fit a regression model as follows.

```
# Fit a linear regression model
model = lm(Unemployment_rate ~ GDP_per_capita, data = data)
summary(model)

##
## Call:
## lm(formula = Unemployment_rate ~ GDP_per_capita, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7023 -3.3111 -0.1623  2.7727  9.5182
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     5.736e+00  1.361e+00   4.213  5.6e-05 ***
## GDP_per_capita -4.903e-06  4.227e-05  -0.116    0.908
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 3.839 on 98 degrees of freedom
## Multiple R-squared:  0.0001373,  Adjusted R-squared:  -0.01007
## F-statistic: 0.01346 on 1 and 98 DF,  p-value: 0.9079
```

The output for the linear regression model fit for this dataset shows that GDP/capita is indeed **statistically insignificant** at 5% level of significance. Hence, there is no significant influence of GDP/capita on the unemployment. Also the p-value for the F-test for the regression model is 0.91, approximately. This implies the model is not a good fit for the data. This aligns with our previous finding that there is no significant correlation between these two variables.