Ryan Timbrook

Applied Data Science

IST687 Intro to Data Science, Spring 2019
Due Date: 05/14/2019
Homework: 6
NetID: RTIMBROO
SUID: 386792749

#R Code - unexecuted

```
## Homework Week 6: Vizualization - Air Quality Analysis

#---Preprocess Steps:------------------------------------------------------------------

### Clear objects from Memory
rm(list=ls())
### Clear Console:
cat("\014")
### Set Working Directory
setwd("C:\\workspaces\\ms_datascience_su\\IST687-IntroDataScience\\R_workspace\\hw")

#---- Global Variable Assignments ---------------------------------------


#---- Load Required Packages ---------------------------------------------------
if(!require("ggplot2")){install.packages("ggplot2")}
if(!require("dplyr")) {install.packages("dplyr")}
if(!require("reshape2")) {install.packages("reshape2")}

#----Step 1: Load the data -------------------------------------------------
air <- airquality

#----Step 2: Clean the data -------------------------------------------------

### Replace NA with column means
na.2.mean <- function(x){
  replace(x, is.na(x), mean(x, na.rm = TRUE))
}

cleanDataSet <- function(ds){
  #Make all empty cells equal to NA
```

```
  ds[ds==""] <- NA

  #Clean NA Columns from Dataframe
  ds <- ds[ ,!apply(ds,2,function(x) all(is.na(x)))]

  #Clean empty Rows from Dataframe
  ds <- ds[!apply(ds,1,function(x) all(is.na(x))),]

  # replace NA's in Ozone col with mean of col (where NA is discarded when calculating the mean)
  ds$Ozone[is.na(ds$Ozone)] <- mean(ds$Ozone,na.rm=TRUE)
  ds$Ozone <- round(ds$Ozone)
  ds$Solar.R[is.na(ds$Solar.R)] <- mean(ds$Solar.R,na.rm=TRUE)
  ds$Solar.R <- round(ds$Solar.R)

  return(ds)
}


clean.air <- cleanDataSet(air)

#----Step 3: Understand the data --------------------------------------------
str(clean.air)
summary(clean.air)
head(clean.air)

#----Step 3.1: Visualizations -----------------------------------------------
## Step 3.1.1: Histograms for each of the variables

#colnames(clean.air)
## Ozone
summary(clean.air$Ozone)
ggplot(data=clean.air, aes(x=Ozone)) +
  geom_histogram(bins=10, color="black", fill="white", boundary=2) +
  ggtitle('Histogram of Ozone')
ggsave(filename='Histogram_of_Ozone.jpg', width = 6, height = 6)

## Solar.R
summary(clean.air$Solar.R)
ggplot(data=clean.air, aes(x=Solar.R)) +
  geom_histogram(bins=10, color="black", fill="white", boundary=2) +
  ggtitle('Histogram of Solar.R')
ggsave(filename='Histogram_of_Solar.R.jpg', width = 6, height = 6)
```

```
## Wind
summary(clean.air$Wind)
ggplot(data=clean.air, aes(x=Wind)) +
  geom_histogram(bins=10, color="black", fill="white", boundary=2) +
  ggtitle('Histogram of Wind')
ggsave(filename="Histogram_of_Wind.jpg", width = 6, height = 6)

## Temp
summary(clean.air$Temp)
ggplot(data=clean.air, aes(x=Temp)) +
  geom_histogram(bins=10, color="black", fill="white", boundary=2) +
  ggtitle('Histogram of Temp')
ggsave(filename="Histogram_of_Temp.jpg", width = 6, height = 6)

## Month
summary(clean.air$Month)
ggplot(data=clean.air, aes(x=Month)) +
  geom_histogram(bins=10, color="black", fill="white", boundary=2) +
  ggtitle('Histogram of Month')
ggsave(filename="Histogram_of_Month.jpg", width = 6, height = 6)

## Day
summary(clean.air$Day)
ggplot(data=clean.air, aes(x=Day)) +
  geom_histogram(bins=10, color="black", fill="white", boundary=2) +
  ggtitle('Histogram of Day')
ggsave(filename="Histogram_of_Day.jpg", width = 6, height = 6)



## Step 3.1.2: Boxplot for Ozone
summary(clean.air$Ozone)
ggplot(data=clean.air, aes(x=factor(0), y=Ozone)) +
  geom_boxplot() + ylab('Ozone') + xlab('Count') +
  ggtitle('Boxplot of Ozone')
ggsave(filename="Boxplot_of_Ozone.jpg", width = 6, height = 6)

## Step 3.1.2: Boxplot for wind values (rounded)
summary(round(clean.air$Wind))
ggplot(data=clean.air, aes(x=factor(0), y=round(Wind))) +
  geom_boxplot() +
  ylab("Wind") + xlab("Count") +
  ggtitle('Boxplot of Wind')
ggsave(filename="Boxplot_of_Wind.jpg", width = 6, height = 6)
```

```
#----Step 3.2: Explore how the data changes over time ------------------------
## Step 3.2.1: Create dates
clean.air$Date <- paste("1973",clean.air$Month,clean.air$Day,sep='-')
clean.air$Date <- as.Date(clean.air$Date,'%Y-%m-%d')
str(clean.air$Date)

## Step 3.2.2: Create Line Charts
## Ozone
ggplot(data=clean.air, aes(x=Date, y=Ozone)) +
  theme_classic(base_size = 8) +
  geom_line(color='Black') +
  ggtitle("Ozone Line Chart over Date Range")
ggsave("Ozone_Line_Chart_over_Date_Range.jpg", width = 6, height = 6)

## Wind
ggplot(data=clean.air, aes(x=Date, y=round(Wind))) +
  theme_classic(base_size = 8) +
  geom_line(color='Blue') +
  ggtitle("Wind Line Chart over Date Range")
ggsave("Wind_Line_Chart_over_Date_Range.jpg", width = 6, height = 6)

## Temp
ggplot(data=clean.air, aes(x=Date, y=round(Temp))) +
  theme_classic(base_size = 8) +
  geom_line(color='Red') +
  ggtitle("Temp Line Chart over Date Range")
ggsave("Temp_Line_Chart_over_Date_Range.jpg", width = 6, height = 6)

## Solar.R
ggplot(data=clean.air, aes(x=Date, y=round(Solar.R))) +
  theme_classic(base_size = 8) +
  geom_line(color='Green4') +
  ggtitle("Solar.R Line Chart over Date Range")
ggsave("Solar.R_Line_Chart_over_Date_Range.jpg", width = 6, height = 6)

## Grouped Line Chart of all four attributes on one chart
ggplot(data=clean.air, aes(x=Date)) +
  geom_line(aes(y=Ozone, color="Ozone")) +
  geom_line(aes(y=Temp, color="Temp")) +
  geom_line(aes(y=Wind, color="Wind")) +
  geom_line(aes(y=Solar.R, color="Solar.R")) +
```

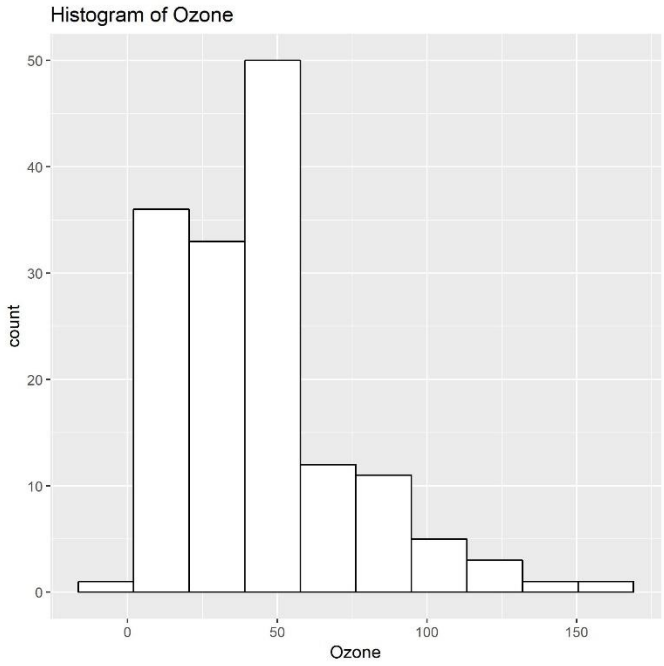
```
  scale_color_manual(values=c("Black","Blue","Red","Green4")) +
  theme(plot.title = element_text(hjust=.5)) +
  labs(title="Ozone - Temp - Wind - Solar.R -- over Date Range") +
  xlab("Date Range") + ylab("Values")
ggsave("Ozone_Temp_Wind_Solar.R_over_Date_Range.jpg", width = 6, height = 6)

## Using Melt
clean.air.reshape <- melt(clean.air[,-c(5,6)], id="Date")
#clean.air.reshape[order(clean.air.reshape$Date),]
ggplot(data=clean.air.reshape, aes(x=Date, y=value, color=variable)) +
  geom_line() +
  ggtitle("Ozone - Temp - Wind - Solar.R -- over Date Range")
ggsave("Melt_Ozone_Temp_Wind_Solar.R_over_Date_Range.jpg", width = 6, height = 6)

#----Step 4: Look at all the data via a Heatmap -----------------------------
## Each Day along the x-axis and Ozone, Temp, Wind, and Solar.R along y-axis and days as rows
along the y-axis
## Create the heatmap using geom_tile
## **Show the relative change equally acroos all the variables
ggplot(data=clean.air.reshape, aes(x=Date, y=variable)) +
  geom_tile(aes(fill=value)) +
  scale_fill_gradient(low = "white", high="red") +
  ggtitle("Heatmap of: Ozone - Temp - Wind - Solar.R")
ggsave("Heatmap_Ozone_Temp_Wind_Solar.R.jpg", width = 6, height = 6)

#----Step 5: Look at all the data via a Scatter Chart ------------------------
## Use geom_point, with the x-axis representing the Wind, the y-axis representing the Temp
# the size of each dot representing the Ozone and the color representing the Solar.R
ggplot(data=clean.air, aes(x=Wind, y=Temp, size=Ozone, color=Solar.R)) +
  geom_point() +
  ggtitle("Scatter Chart of: Wind - Temp - Ozone - Solar.R")
ggsave("Scatter_Chart_of_Wind_Temp_Ozone_Solar.R.jpg", width = 6, height = 6)

# Create a Scatter Chart with a smoother depicting standard error
ggplot(data=clean.air, aes(x=Wind, y=Temp, size=Ozone, color=Solar.R)) +
  geom_smooth() +
  geom_point(alpha=1/2) +
  ggtitle("Scatter Chart with Smooth Line Fitting of: Wind - Temp - Ozone - Solar.R")
ggsave("Scatter_Chart_with_Smooth_Line_Fitting_of_Wind_Temp_Ozone_Solar.R.jpg", width = 6,
height = 6)

#----Step 6: Final Alaysis ---------------------------------------------------
## What patterns immerged from the data?
```

What was the most useful visualization?

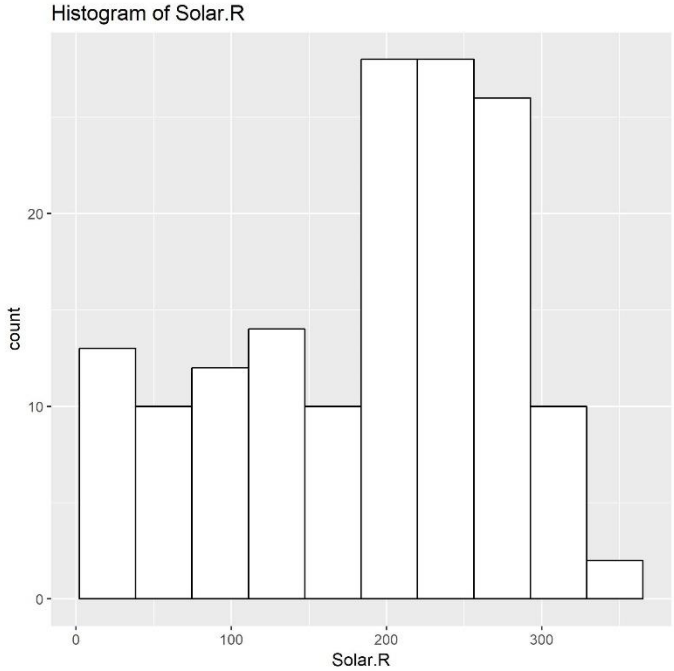
#R Code – executed

```
> ### Set Working Directory
> setwd("C:\\workspaces\\ms_datascience_su\\IST687-IntroDataScience\\R_worksp
ace\\hw")
>
> #---- Global Variable Assignments --------------------------------------
---
>
>
> #---- Load Required Packages -------------------------------------------
---
> if(!require("ggplot2")){install.packages("ggplot2")}
> if(!require("dplyr")) {install.packages("dplyr")}
> if(!require("reshape2")) {install.packages("reshape2")}
>
> #----Step 1: Load the data ----------------------------------------------
---
> air <- airquality
>
> #----Step 2: Clean the data ---------------------------------------------
---
>
> ### Replace NA with column means
> na.2.mean <- function(x){
+   replace(x, is.na(x), mean(x, na.rm = TRUE))
+ }
>
> cleanDataSet <- function(ds){
+   #Make all empty cells equal to NA
+   ds[ds==""] <- NA
+
+   #Clean NA Columns from Dataframe
+   ds <- ds[ ,!apply(ds,2,function(x) all(is.na(x)))]
+
+   #Clean empty Rows from Dataframe
+   ds <- ds[!apply(ds,1,function(x) all(is.na(x))),]
+
+   # replace NA's in Ozone col with mean of col (where NA is discarded when
calculating the mean)
+   ds$Ozone[is.na(ds$Ozone)] <- mean(ds$Ozone,na.rm=TRUE)
+   ds$Ozone <- round(ds$Ozone)
+   ds$Solar.R[is.na(ds$Solar.R)] <- mean(ds$Solar.R,na.rm=TRUE)
+   ds$Solar.R <- round(ds$Solar.R)
+
+   return(ds)
+ }
>
>
> clean.air <- cleanDataSet(air)
>
> #----Step 3: Understand the data ----------------------------------------
---
> str(clean.air)
```

```
'data.frame':  153 obs. of  6 variables:
 $ Ozone  : num  41 36 12 18 42 28 23 19 8 42 ...
 $ Solar.R: num  190 118 149 313 186 186 299 99 19 194 ...
 $ Wind   : num  7.4 8 12.6 11.5 14.3 14.9 8.6 13.8 20.1 8.6 ...
 $ Temp   : int  67 72 74 62 56 66 65 59 61 69 ...
 $ Month  : int  5 5 5 5 5 5 5 5 5 5 ...
 $ Day    : int  1 2 3 4 5 6 7 8 9 10 ...
> summary(clean.air)
     Ozone           Solar.R           Wind             Temp           Month
Day
 Min.   :  1.0   Min.   :  7.0   Min.   : 1.700   Min.   :56.00   Min.   :5.0
00   Min.   : 1.0
 1st Qu.: 21.0   1st Qu.:120.0   1st Qu.: 7.400   1st Qu.:72.00   1st Qu.:6.0
00   1st Qu.: 8.0
 Median : 42.0   Median :194.0   Median : 9.700   Median :79.00   Median :7.0
00   Median :16.0
 Mean   : 42.1   Mean   :185.9   Mean   : 9.958   Mean   :77.88   Mean   :6.9
93   Mean   :15.8
 3rd Qu.: 46.0   3rd Qu.:256.0   3rd Qu.:11.500   3rd Qu.:85.00   3rd Qu.:8.0
00   3rd Qu.:23.0
 Max.   :168.0   Max.   :334.0   Max.   :20.700   Max.   :97.00   Max.   :9.0
00   Max.   :31.0
> head(clean.air)
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    42     186 14.3   56     5   5
6    28     186 14.9   66     5   6
>
> #----Step 3.1: Visualizations ----------------------------------------------
-----
> ## Step 3.1.1: Histograms for each of the variables
>
> #colnames(clean.air)
> ## Ozone
> summary(clean.air$Ozone)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    1.0    21.0    42.0    42.1    46.0   168.0
> ggplot(data=clean.air, aes(x=Ozone)) +
+   geom_histogram(bins=10, color="black", fill="white", boundary=2) +
+   ggtitle('Histogram of Ozone')
> ggsave(filename='Histogram_of_Ozone.jpg', width = 6, height = 6)
```

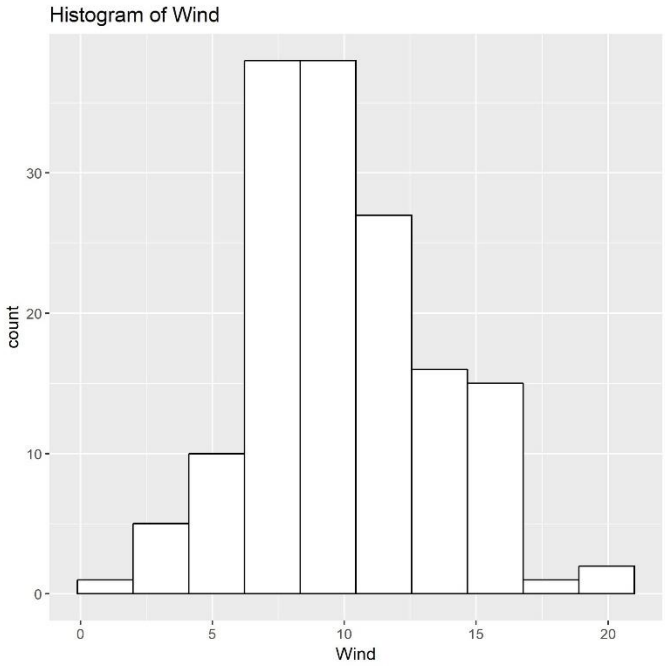



```
> ## Solar.R
> summary(clean.air$Solar.R)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    7.0   120.0   194.0   185.9   256.0   334.0
> ggplot(data=clean.air, aes(x=Solar.R)) +
+   geom_histogram(bins=10, color="black", fill="white", boundary=2) +
+   ggtitle('Histogram of Solar.R')
> ggsave(filename='Histogram_of_Solar.R.jpg', width = 6, height = 6)
```

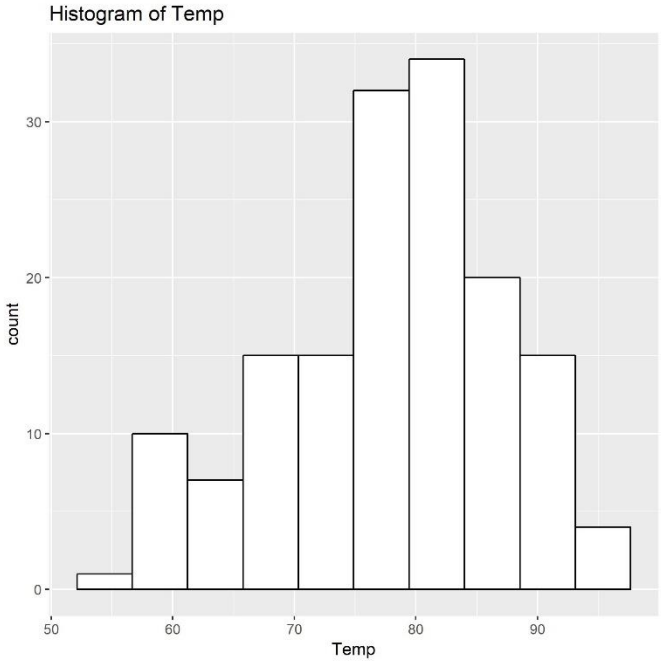



```
> ## Wind
> summary(clean.air$Wind)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.700   7.400   9.700   9.958  11.500  20.700
```

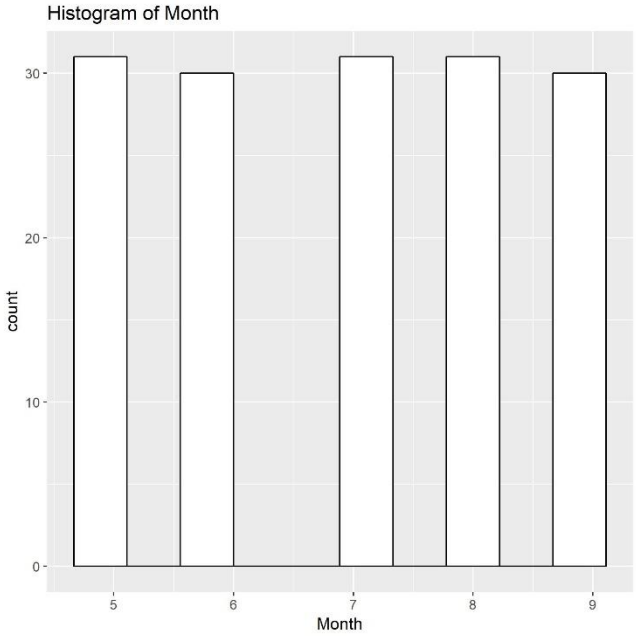
```
> ggplot(data=clean.air, aes(x=Wind)) +
+   geom_histogram(bins=10, color="black", fill="white", boundary=2) +
+   ggtitle('Histogram of Wind')
> ggsave(filename="Histogram_of_Wind.jpg", width = 6, height = 6)
```



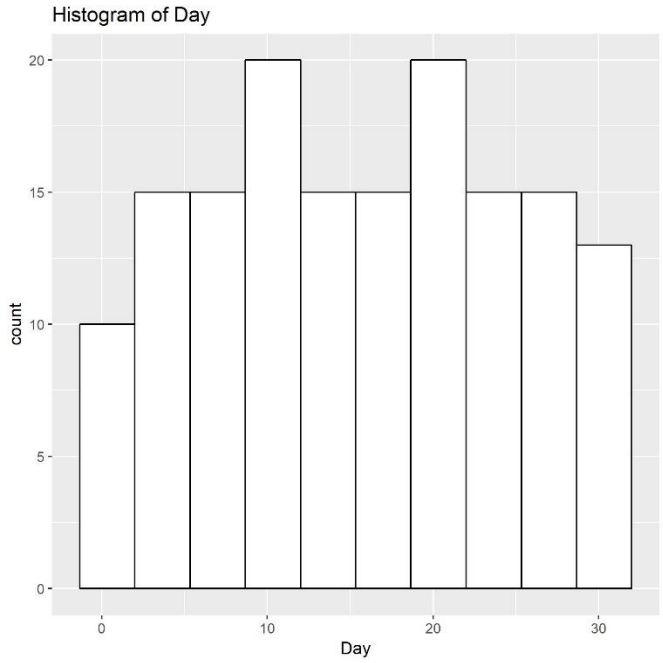

```
> ## Temp
> summary(clean.air$Temp)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  56.00   72.00   79.00   77.88   85.00   97.00
> ggplot(data=clean.air, aes(x=Temp)) +
+   geom_histogram(bins=10, color="black", fill="white", boundary=2) +
+   ggtitle('Histogram of Temp')
> ggsave(filename="Histogram_of_Temp.jpg", width = 6, height = 6)
```



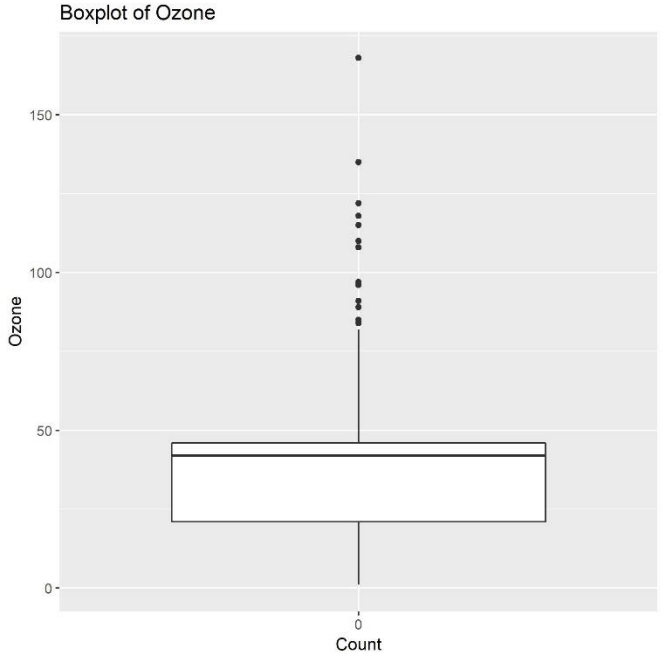
```
> ## Month
> summary(clean.air$Month)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  5.000   6.000   7.000   6.993   8.000   9.000
> ggplot(data=clean.air, aes(x=Month)) +
+   geom_histogram(bins=10, color="black", fill="white", boundary=2) +
+   ggtitle('Histogram of Month')
> ggsave(filename="Histogram_of_Month.jpg", width = 6, height = 6)
```




```
> ## Day
> summary(clean.air$Day)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    1.0     8.0    16.0    15.8    23.0    31.0
> ggplot(data=clean.air, aes(x=Day)) +
+   geom_histogram(bins=10, color="black", fill="white", boundary=2) +
+   ggtitle('Histogram of Day')
> ggsave(filename="Histogram_of_Day.jpg", width = 6, height = 6)
```

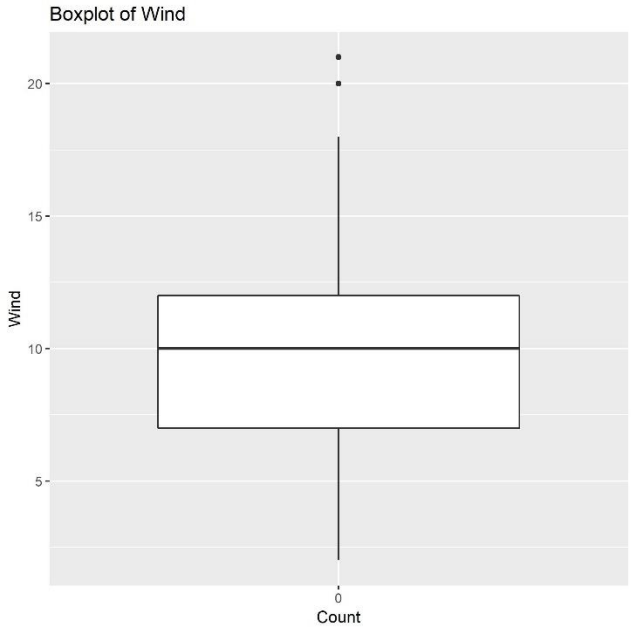



```
>
> ## Step 3.1.2: Boxplot for Ozone
> summary(clean.air$Ozone)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    1.0    21.0    42.0    42.1    46.0   168.0
> ggplot(data=clean.air, aes(x=factor(0), y=Ozone)) +
+   geom_boxplot() + ylab('Ozone') + xlab('Count') +
+   ggtitle('Boxplot of Ozone')
> ggsave(filename="Boxplot_of_Ozone.jpg", width = 6, height = 6)
```

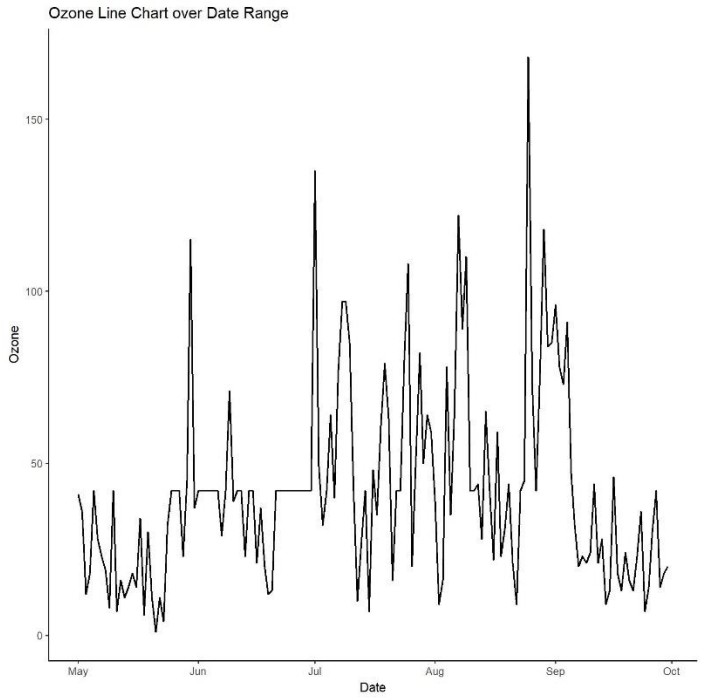



```
> ## Step 3.1.2: Boxplot for wind values (rounded)
> summary(round(clean.air$Wind))
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   2.00    7.00   10.00   10.02   12.00   21.00
```

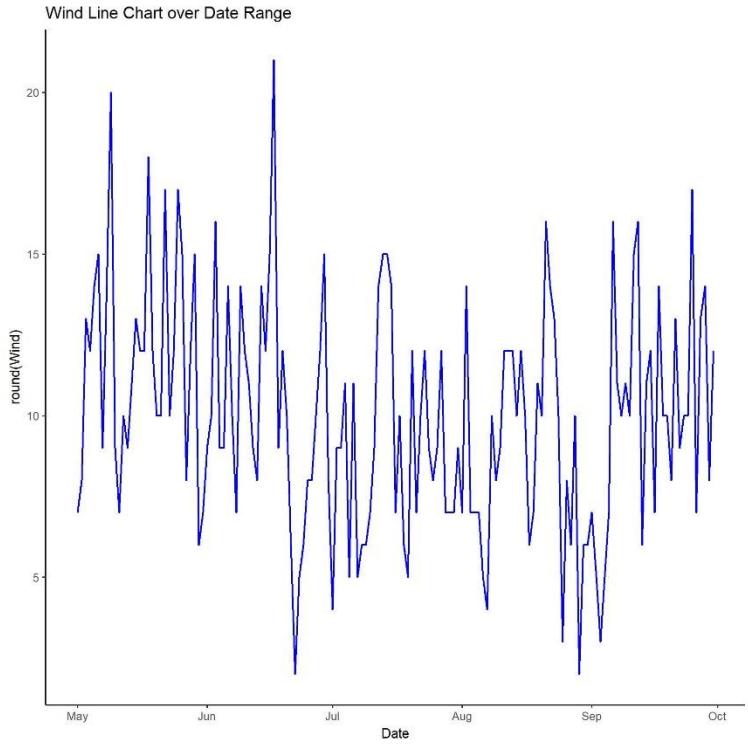
```
> ggplot(data=clean.air, aes(x=factor(0), y=round(Wind))) +
+   geom_boxplot() +
+   ylab("Wind") + xlab("Count") +
+   ggtitle('Boxplot of Wind')
> ggsave(filename="Boxplot_of_Wind.jpg", width = 6, height = 6)
```




```
>
> #----Step 3.2: Explore how the data changes over time ----------------------
---
> ## Step 3.2.1: Create dates
> clean.air$Date <- paste("1973",clean.air$Month,clean.air$Day,sep='-')
> clean.air$Date <- as.Date(clean.air$Date,'%Y-%m-%d')
> str(clean.air$Date)
 Date[1:153], format: "1973-05-01" "1973-05-02" "1973-05-03" "1973-05-04" "19
73-05-05" "1973-05-06" "1973-05-07" "1973-05-08" "1973-05-09" ...
>
> ## Step 3.2.2: Create Line Charts
> ## Ozone
> ggplot(data=clean.air, aes(x=Date, y=Ozone)) +
+   theme_classic(base_size = 8) +
+   geom_line(color='Black') +
+   ggtitle("Ozone Line Chart over Date Range")
> ggsave("Ozone_Line_Chart_over_Date_Range.jpg", width = 6, height = 6)
```

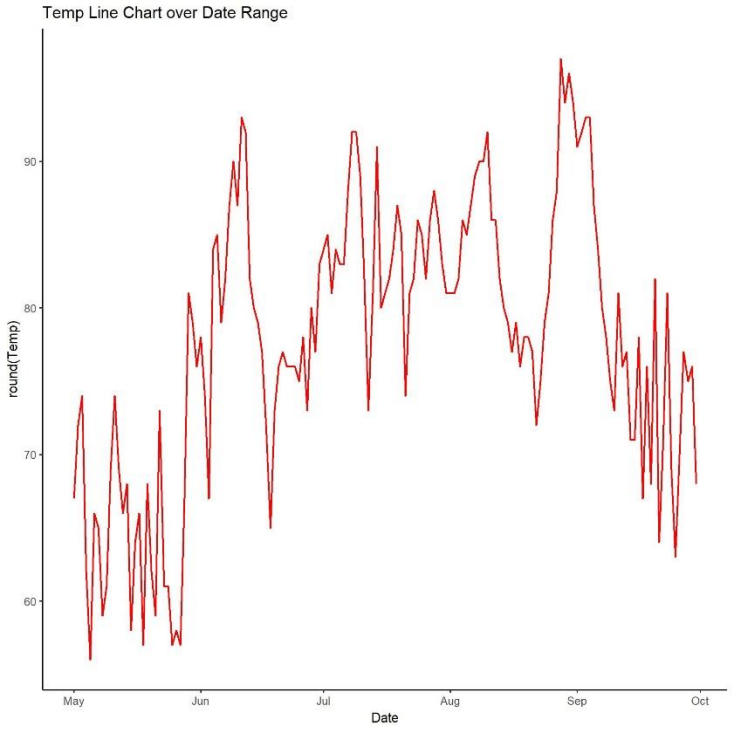



```
> ## Wind
> ggplot(data=clean.air, aes(x=Date, y=round(Wind))) +
+   theme_classic(base_size = 8) +
+   geom_line(color='Blue') +
+   ggtitle("Wind Line Chart over Date Range")
> ggsave("Wind_Line_Chart_over_Date_Range.jpg", width = 6, height = 6)
```

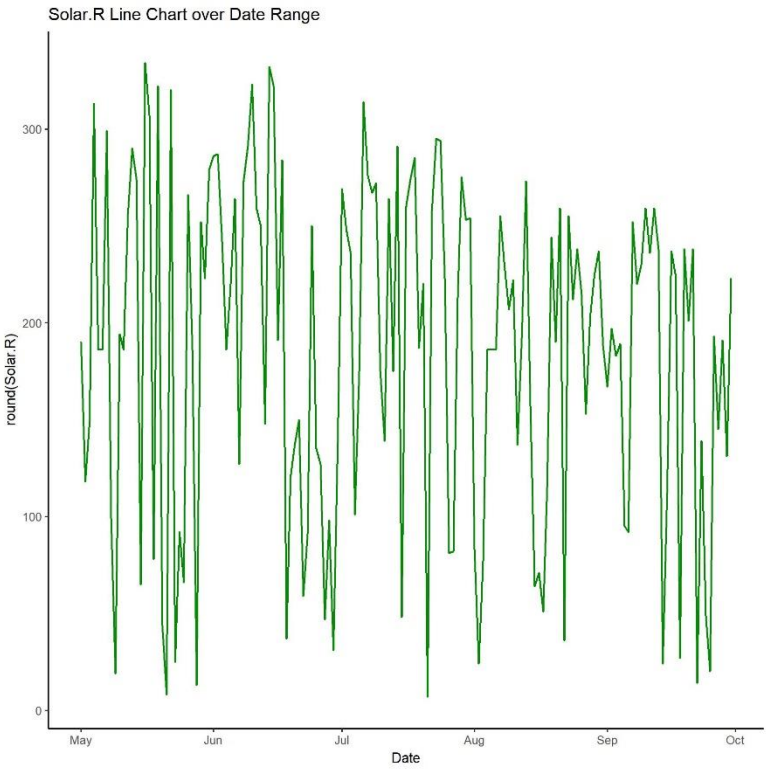



```
> ## Temp
> ggplot(data=clean.air, aes(x=Date, y=round(Temp))) +
+   theme_classic(base_size = 8) +
```

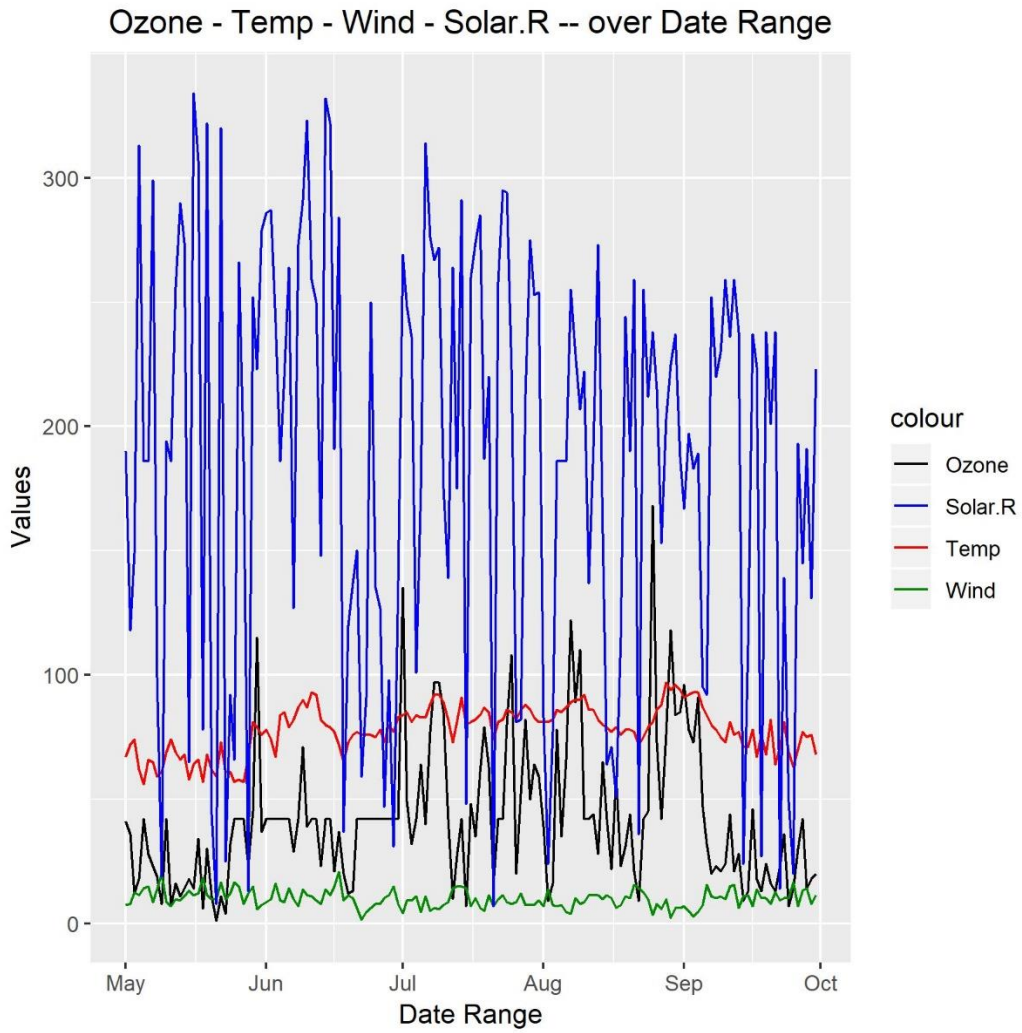
```
+   geom_line(color='Red') +
+   ggtitle("Temp Line Chart over Date Range")
> ggsave("Temp_Line_Chart_over_Date_Range.jpg", width = 6, height = 6)
```



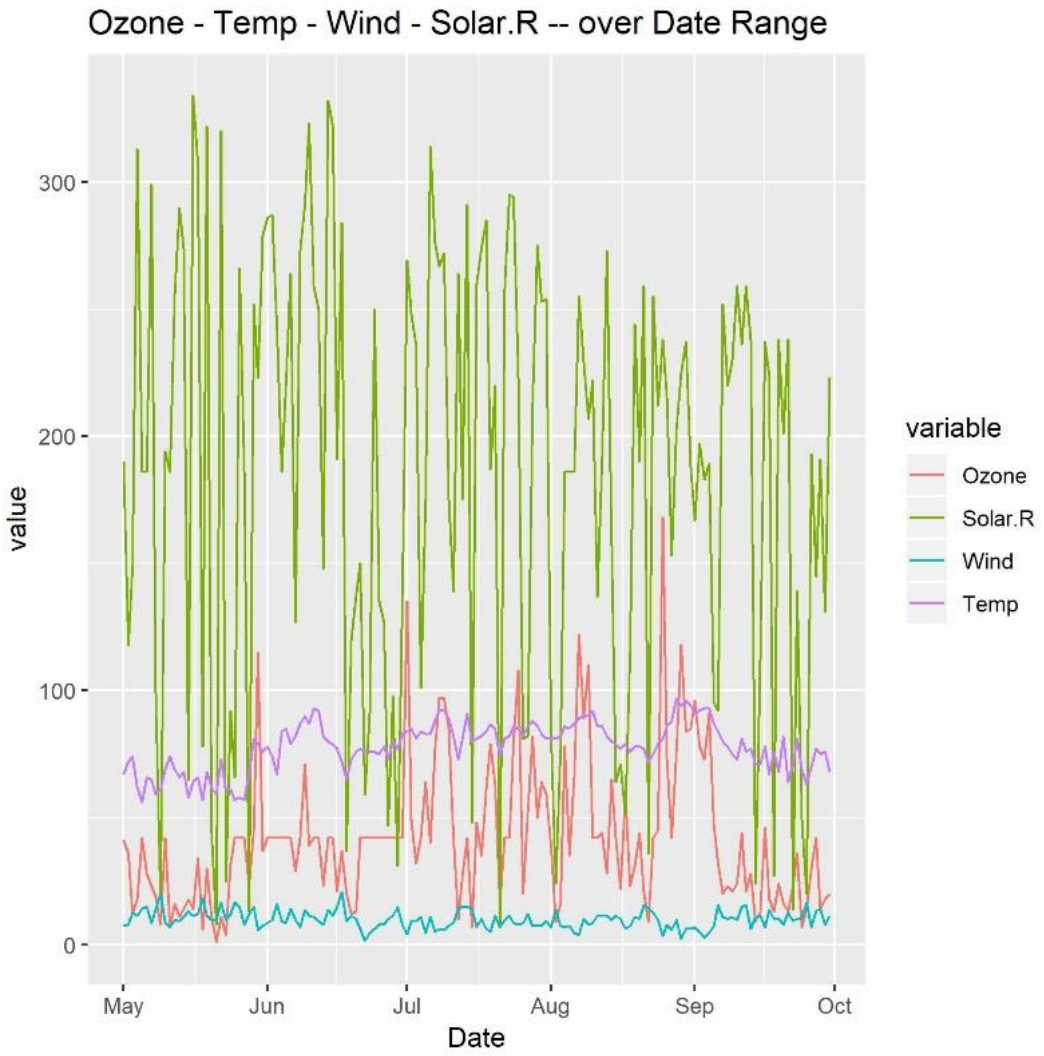

```
> ## Solar.R
> ggplot(data=clean.air, aes(x=Date, y=round(Solar.R))) +
+   theme_classic(base_size = 8) +
+   geom_line(color='Green4') +
+   ggtitle("Solar.R Line Chart over Date Range")
> ggsave("Solar.R_Line_Chart_over_Date_Range.jpg", width = 6, height = 6)
```



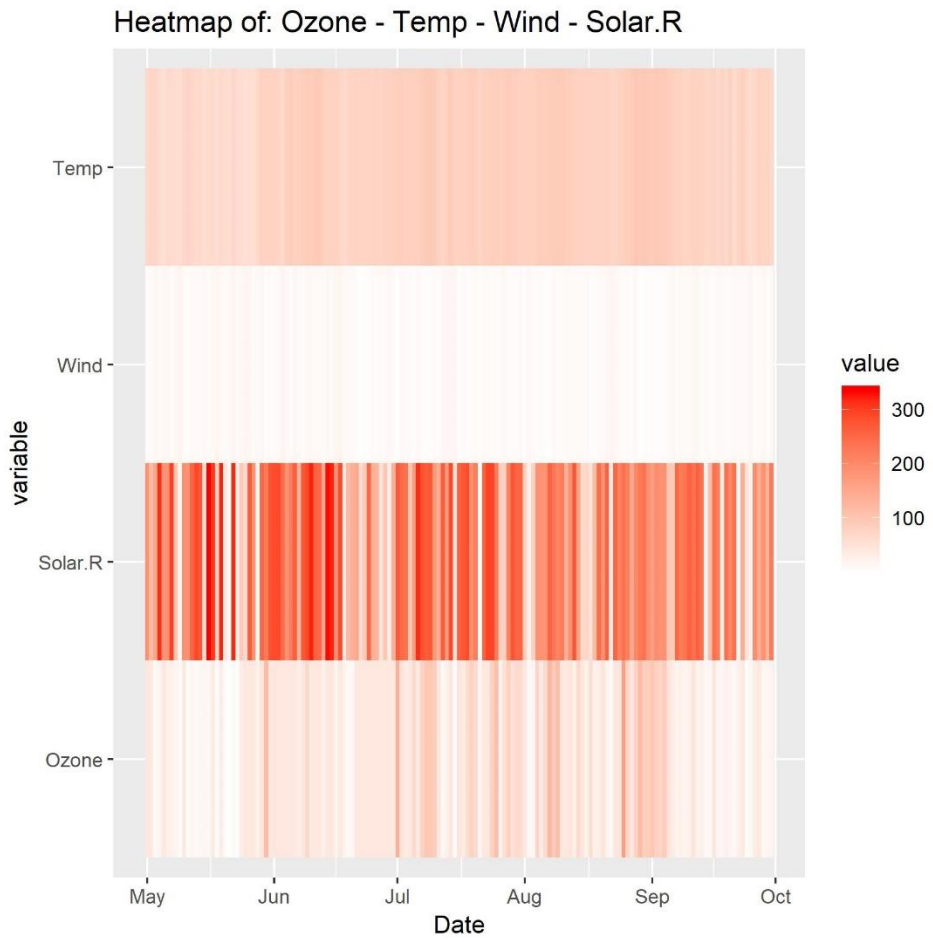

```
> ## Grouped Line Chart of all four attributes on one chart
> ggplot(data=clean.air, aes(x=Date)) +
+   geom_line(aes(y=Ozone, color="Ozone")) +
+   geom_line(aes(y=Temp, color="Temp")) +
+   geom_line(aes(y=Wind, color="Wind")) +
+   geom_line(aes(y=Solar.R, color="Solar.R")) +
+   scale_color_manual(values=c("Black","Blue","Red","Green4")) +
+   theme(plot.title = element_text(hjust=.5)) +
+   labs(title="Ozone - Temp - Wind - Solar.R -- over Date Range") +
+   xlab("Date Range") + ylab("Values")
> ggsave("Ozone_Temp_Wind_Solar.R_over_Date_Range.jpg", width = 6, height = 6
)
```



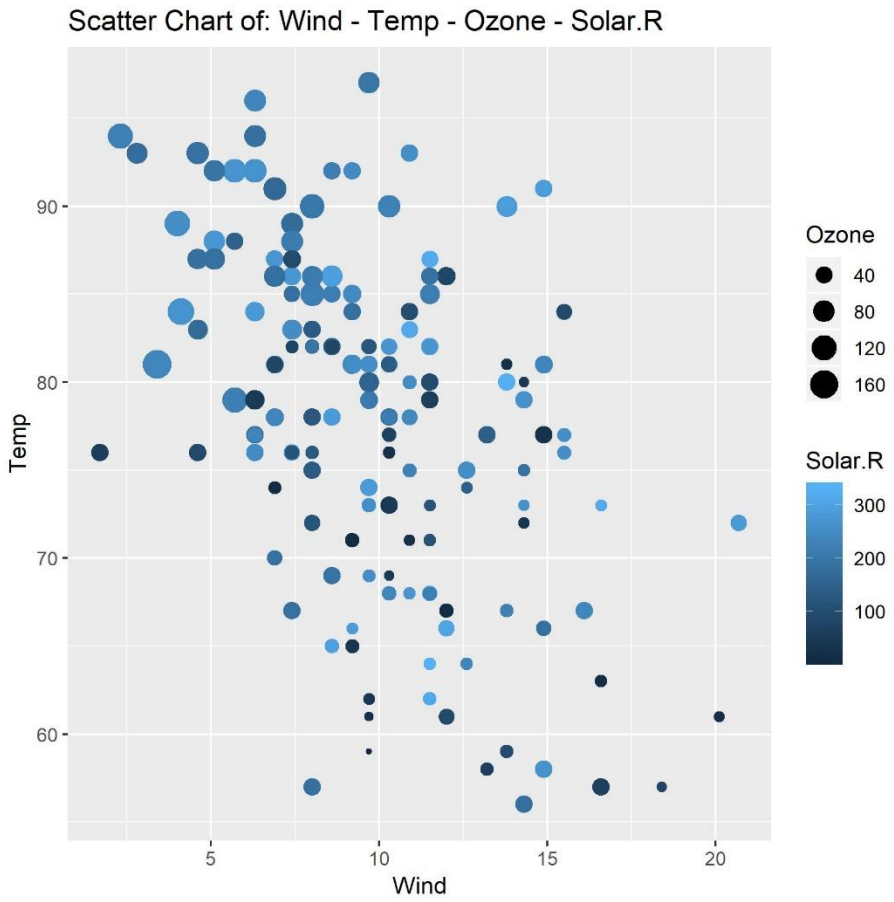

```
> ## Using Melt
> clean.air.reshape <- melt(clean.air[,-c(5,6)], id="Date")
> #clean.air.reshape[order(clean.air.reshape$Date),]
> ggplot(data=clean.air.reshape, aes(x=Date, y=value, color=variable)) +
+   geom_line() +
+   ggtitle("Ozone - Temp - Wind - Solar.R -- over Date Range")
> ggsave("Melt_Ozone_Temp_Wind_Solar.R_over_Date_Range.jpg", width = 6, heigh
t = 6)
```



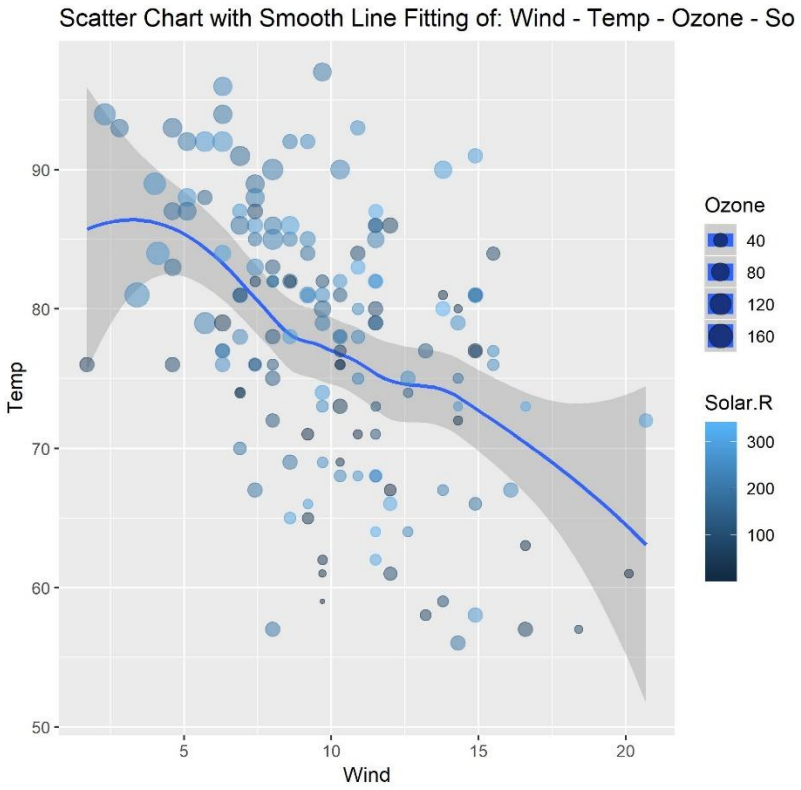

```
> #----Step 4: Look at all the data via a Heatmap ---------------------------
---
> ## Each Day along the x-axis and Ozone, Temp, Wind, and Solar.R along y-axi
s and days as rows along the y-axis
> ## Create the heatmap using geom_tile
> ## **Show the relative change equally acroos all the variables
> ggplot(data=clean.air.reshape, aes(x=Date, y=variable)) +
+   geom_tile(aes(fill=value)) +
+   scale_fill_gradient(low = "white", high="red") +
+   ggtitle("Heatmap of: Ozone - Temp - Wind - Solar.R")
> ggsave("Heatmap_Ozone_Temp_Wind_Solar.R.jpg", width = 6, height = 6)
```




```
> #----Step 5: Look at all the data via a Scatter Chart ----------------------
---
> ## Use geom_point, with the x-axis representing the Wind, the y-axis repres
enting the Temp
> # the size of each dot representing the Ozone and the color representing th
e Solar.R
> ggplot(data=clean.air, aes(x=Wind, y=Temp, size=Ozone, color=Solar.R)) +
+   geom_point() +
+   ggtitle("Scatter Chart of: Wind - Temp - Ozone - Solar.R")
> ggsave("Scatter_Chart_of_Wind_Temp_Ozone_Solar.R.jpg", width = 6, height =
6)
```




```
> ggplot(data=clean.air, aes(x=Wind, y=Temp, size=Ozone, color=Solar.R)) +
+   geom_smooth() +
+   geom_point(alpha=1/2) +
+   ggtitle("Scatter Chart with Smooth Line Fitting of: Wind - Temp - Ozone -
Solar.R")
`geom_smooth()` using method = 'loess' and formula 'y ~ x'
> ggsave("Scatter_Chart_with_Smooth_Line_Fitting_of_Wind_Temp_Ozone_Solar.R.j
pg", width = 6, height = 6)
```




```
> #----Step 6: Final Alaysis ------------------------------------------------
---
> ## What patterns immerged from the data?
As Wind increases, it's observed that Temp decreases.
As Temp rises, it's observed that Ozone levels rise.
> ## What was the most useful visualization?
```

I found the Scatter Chart to be the most useful in uncovering data patterns among the four core attributes, Wind, Temp, Ozone and Solar.R