IST 687 Homework 6 - Viz HW: Air Quality Analysis

Dan Burke

8/26/2021

Assignment Due: 8/23/2021

Submitted: 8/26/2021

Step 1: Load the data

We will use the airquality data set, which you should already have as part of your R installation.

```
airQual <- data.frame(airquality)
```

Step 2: Clean the data

After you load the data, there will be some NAs in the data. You need to figure out what to do about those nasty NAs.

```
#remove and store the clean data seperatly incase we'll need it
AirQualClean <- na.omit(airQual)
```

Step 3: Understand the data distribution

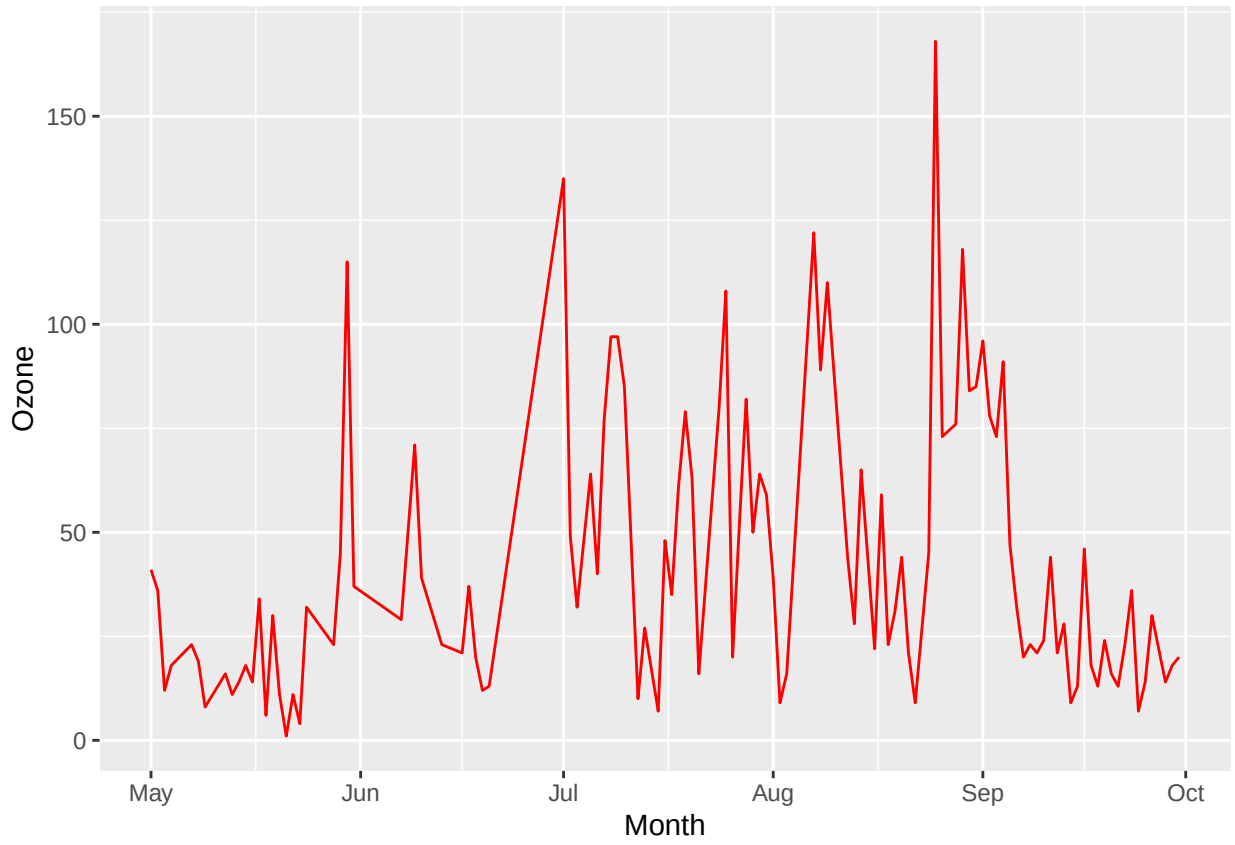
Create the following visualizations using ggplot: • Histograms for each of the variables
• Boxplot for Ozone • Boxplot for wind values (round the wind to get a good number of "buckets")

```
#Required to Knit to PDF
if (!require('ggplot2') || !require('tidyr'))
{
  install.packages('ggplot2');
  install.packages('tidyr')
  install.packages('formatR')
  library(formatR)
  library(tidyr)
  library(ggplot2);
  library(knitr)
}
```

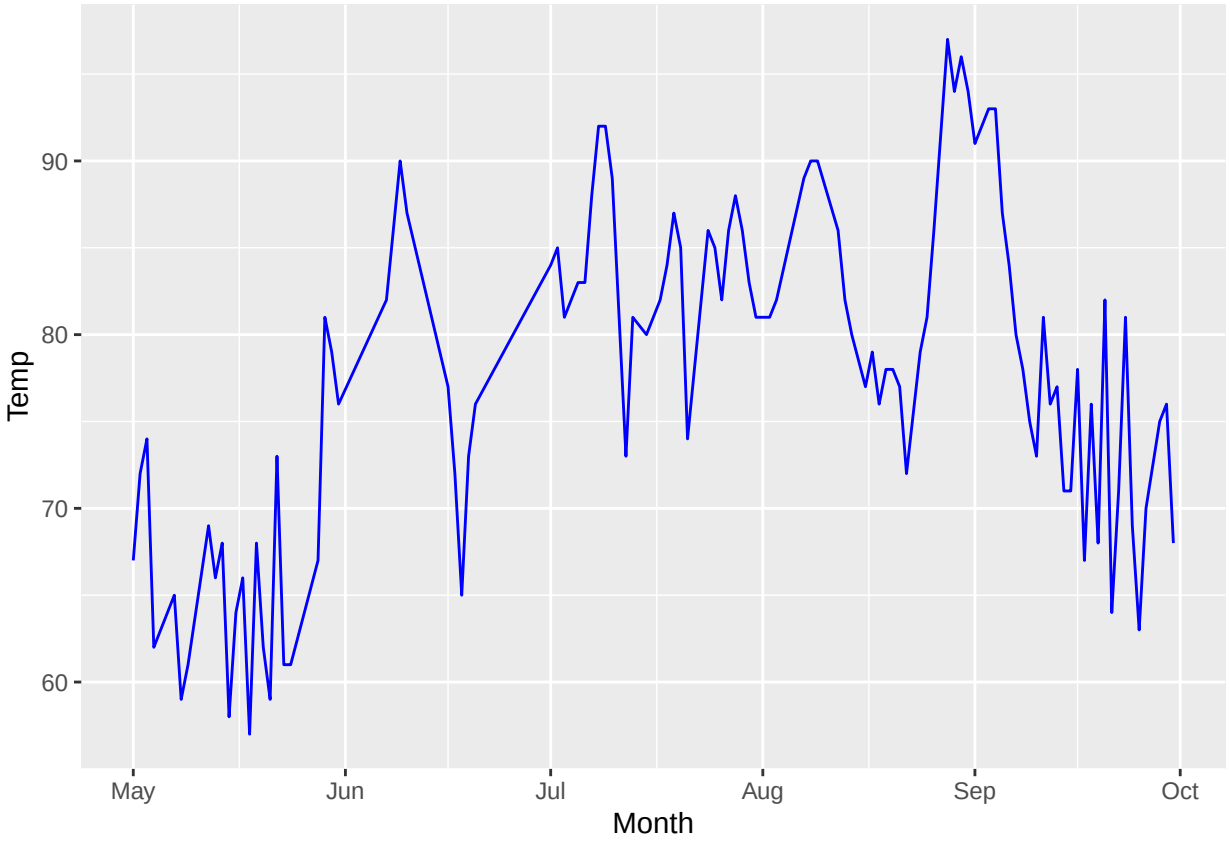
```
## Loading required package: ggplot2
```

```
## Loading required package: tidyr
```

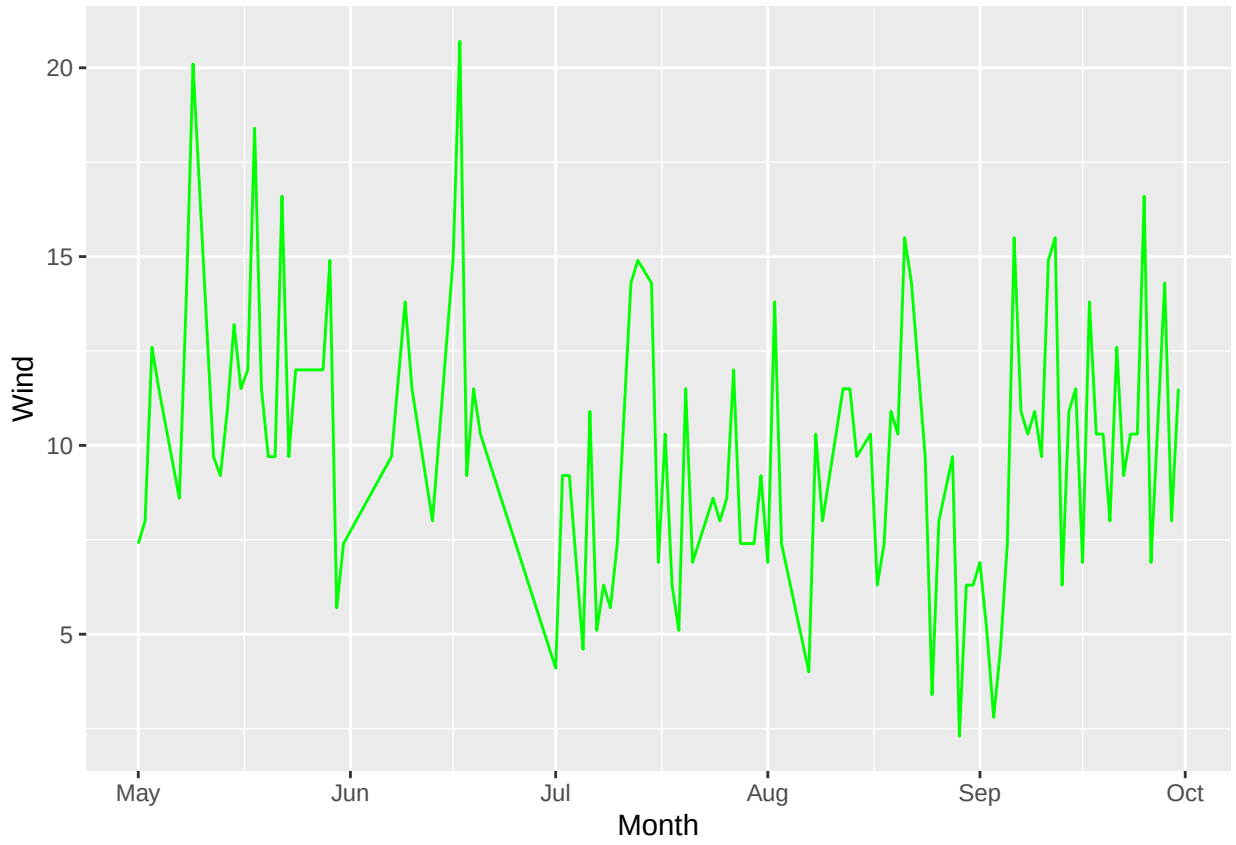
```
ggplot(AirQualClean, aes(x=Ozone)) + geom_histogram(binwidth=5)
```



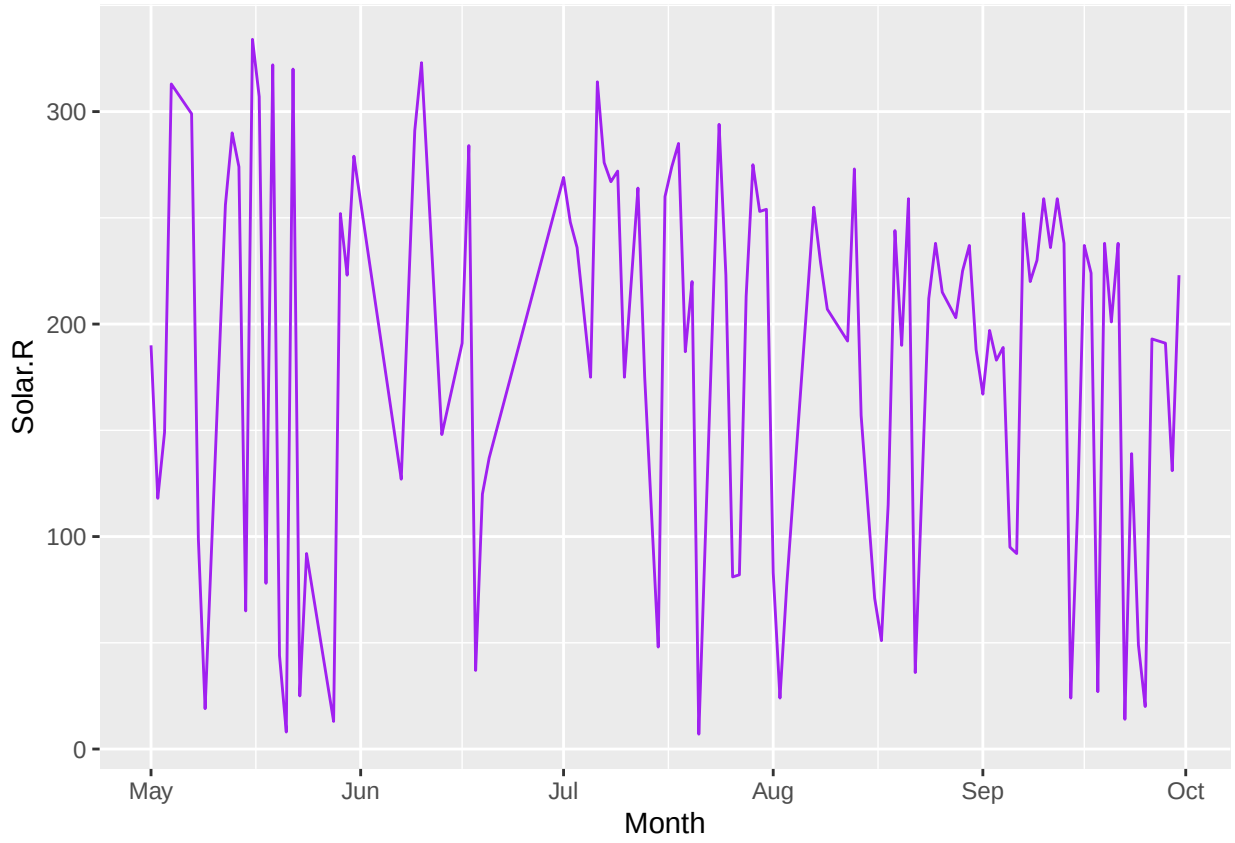

```
ggplot(AirQualClean, aes(x=Solar.R)) + geom_histogram(binwidth=8)
```



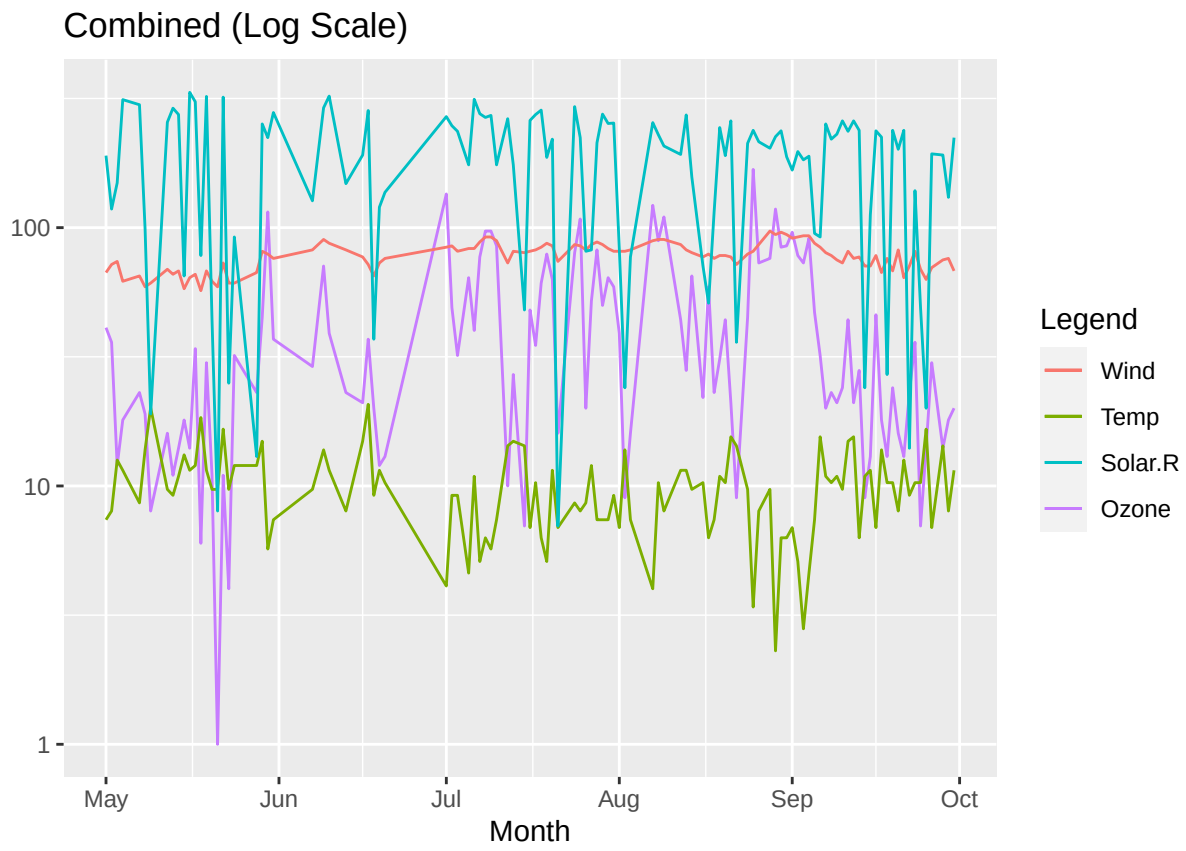

```
ggplot(AirQualClean, aes(x=Wind)) + geom_histogram(binwidth=1)
```



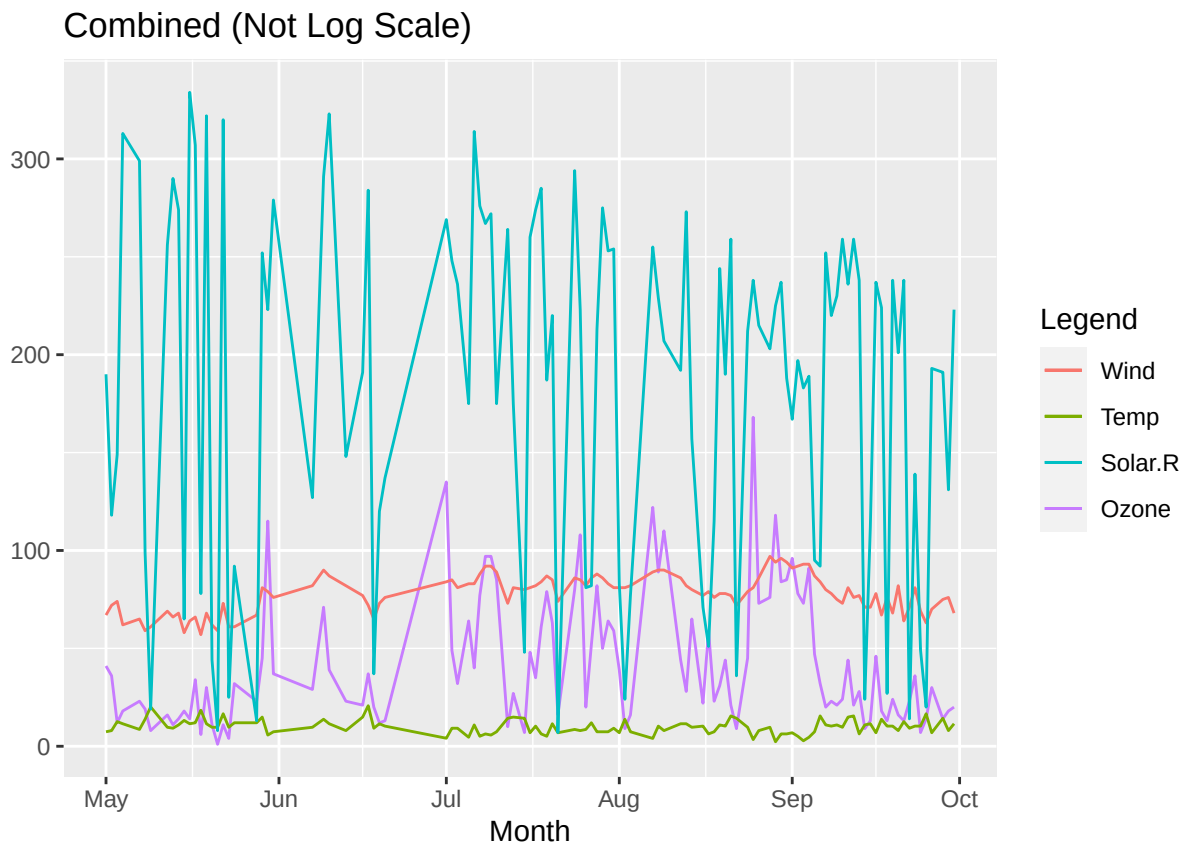

```
ggplot(AirQualClean, aes(x=Temp)) + geom_histogram(binwidth=1)
```



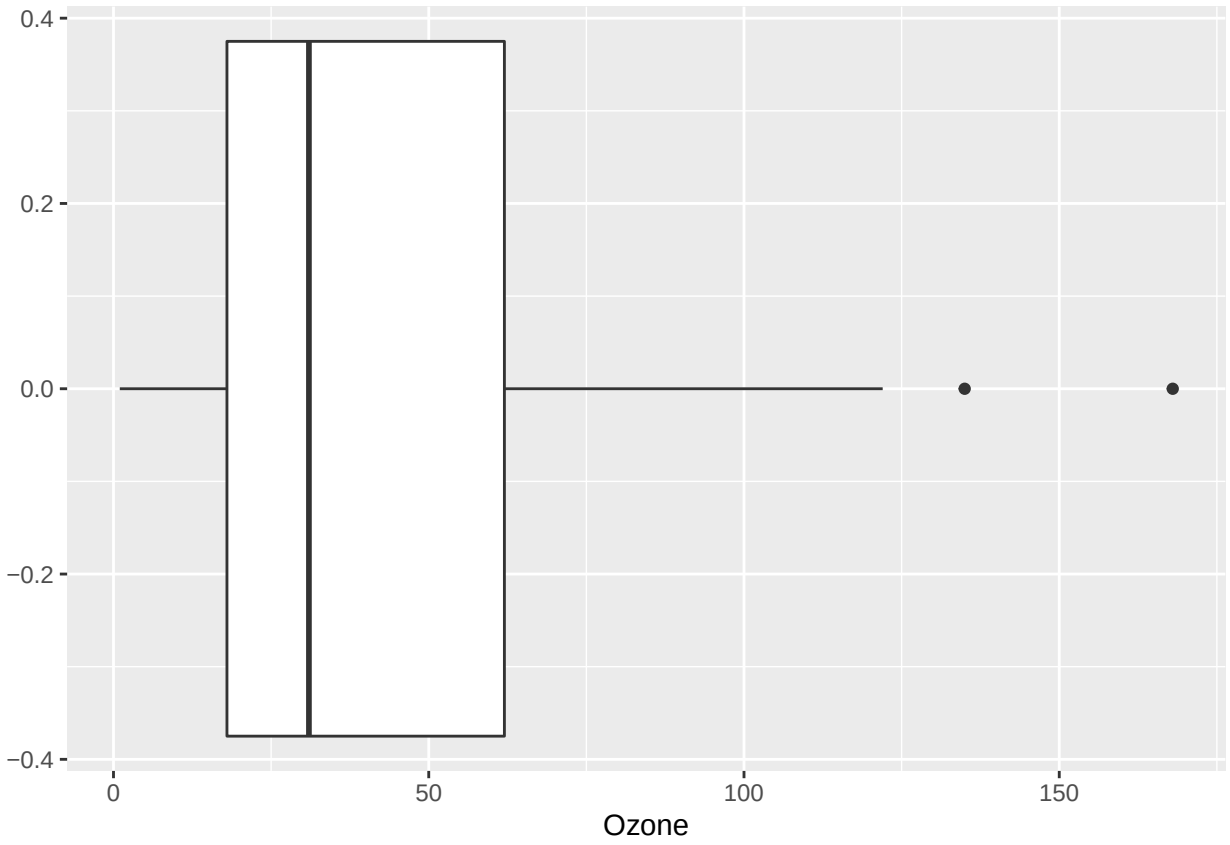

```
ggplot(AirQualClean, aes(x=Month)) + geom_histogram(binwidth=0.9)
```



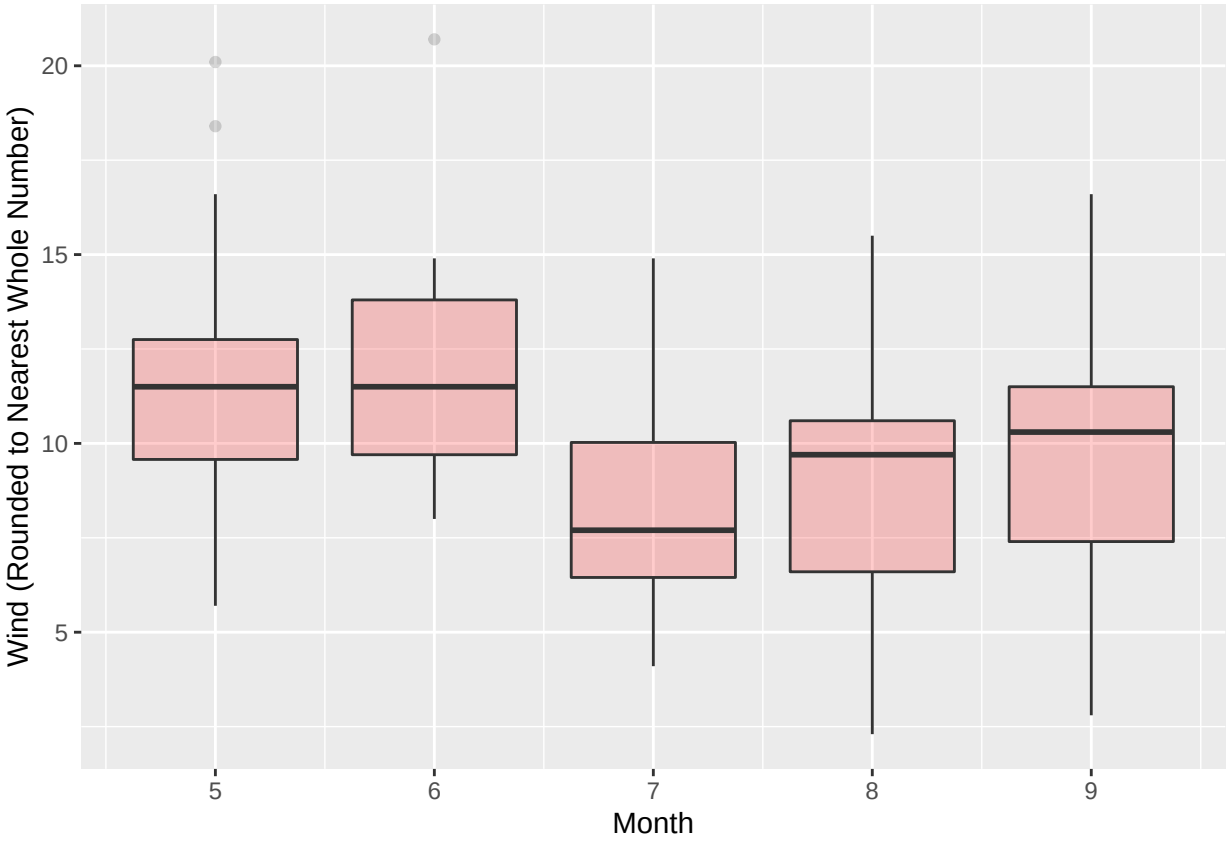

```
ggplot(AirQualClean, aes(x=Day)) + geom_histogram(binwidth=1)
```




```
#BoxPlot for Ozone
ggplot(AirQualClean, aes(x=Ozone)) + geom_boxplot()
```




```
#BoxPlot for wind Values (round the wind    to  get a   good    number  of  "buckets")

#Instead of rounding the values of Wind, I've chosen to create buckets by month
roundedWindBoxPlot <- ggplot(AirQualClean, aes(x=Month, y=round(Wind,1),group=Month))
roundedWindBoxPlot <- roundedWindBoxPlot + geom_boxplot( fill="red", alpha=0.2)
roundedWindBoxPlot <- roundedWindBoxPlot + ylab("Wind (Rounded to Nearest Whole Number)")
roundedWindBoxPlot
```

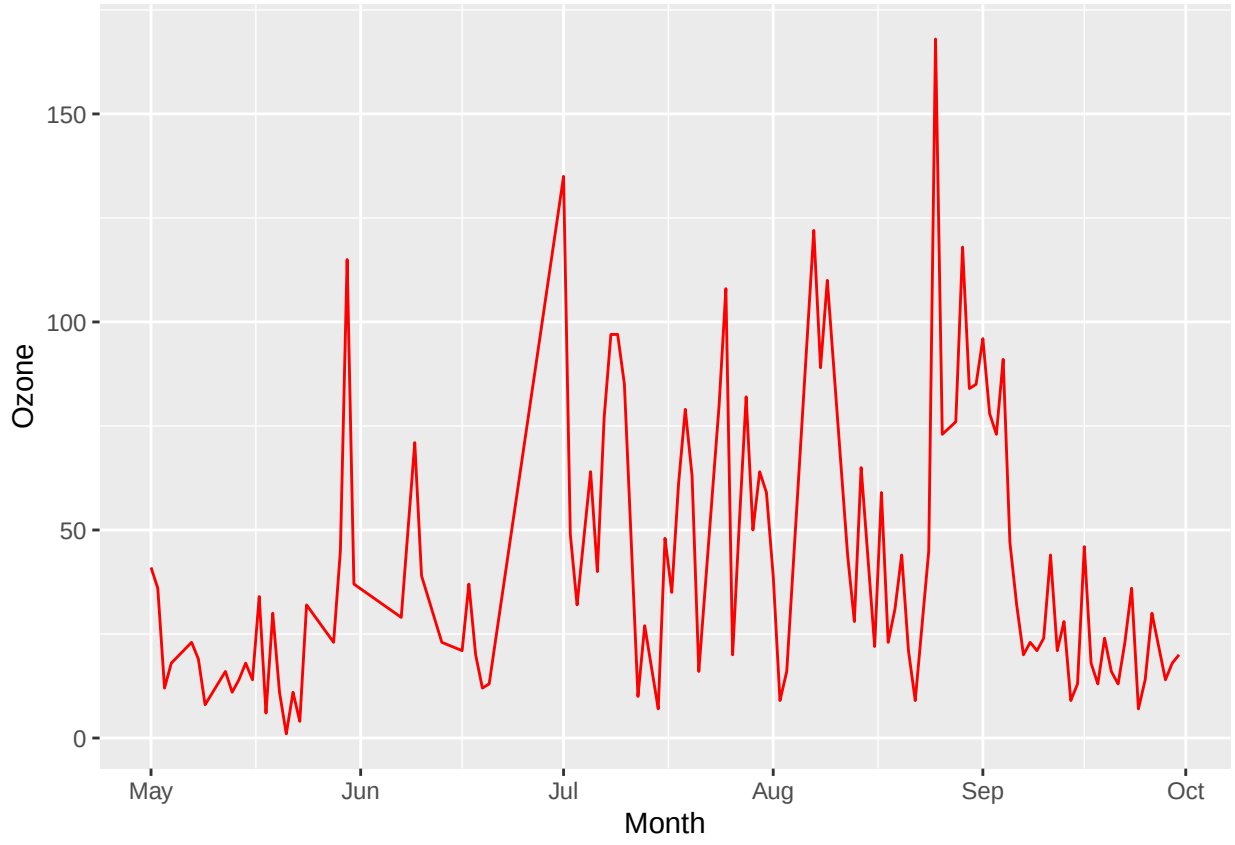


Step 3: Explore how the data changes over time

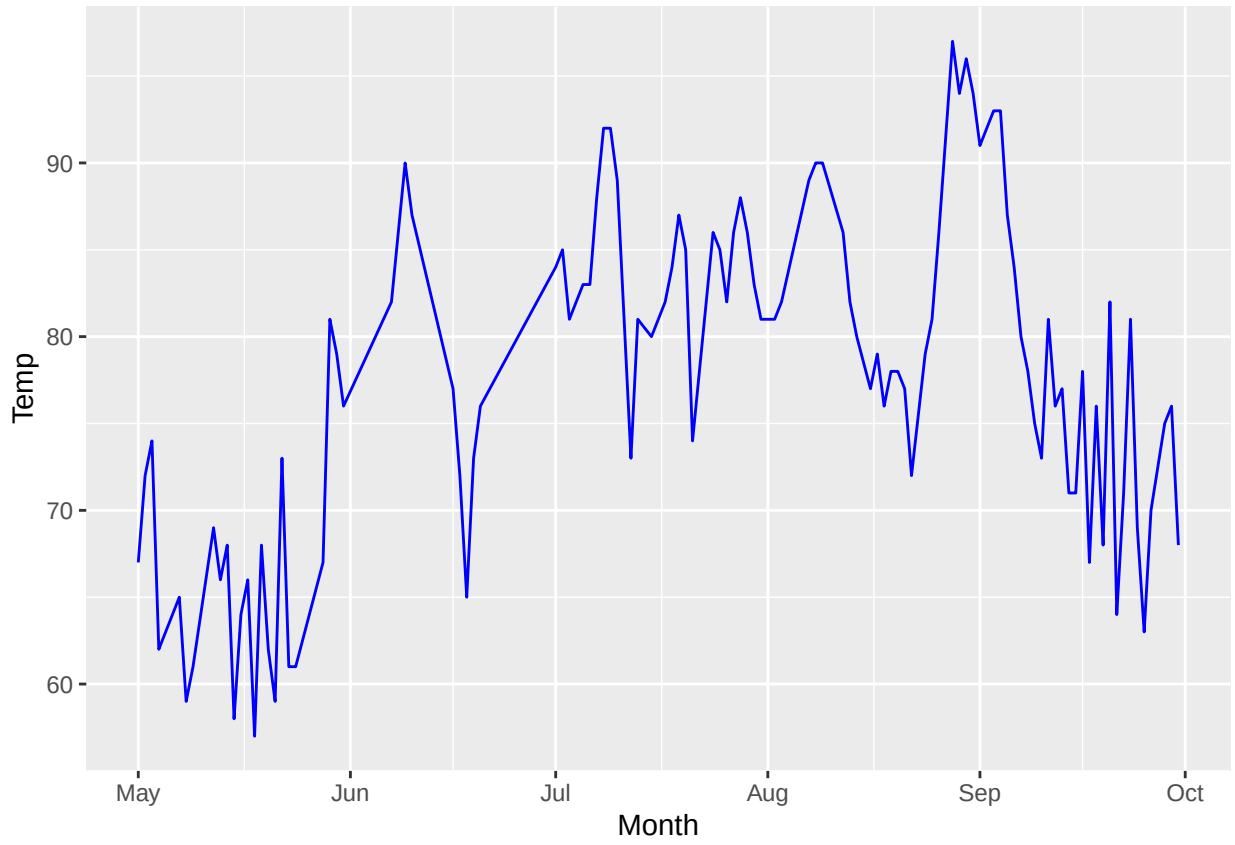
First, make sure to create appropriate dates (this data was from 1973). Then create line charts for ozone, temp, wind and solar.R (one line chart for each, and then one chart with 4 lines, each having a different color). Create these visualizations using ggplot. Note that for the chart with 4 lines, you need to think about how to effectively use the yaxis.

```
#Create dates from the Given Month and Day Values
AirQualClean$CalcDate <- as.Date(with(AirQualClean, paste(1973, Month, Day,sep="-")), "%Y-%m-%d")

#LineChart Ozone
ozoneLineChart <- ggplot(AirQualClean, aes(y=Ozone, x= CalcDate))
ozoneLineChart <- ozoneLineChart + geom_line(color="red") + xlab("Month")


#LineChart Temp
tempLineChart <- ggplot(AirQualClean, aes(y=Temp, x= CalcDate))
tempLineChart <- tempLineChart + geom_line(color="blue") + xlab("Month")


#LineChart Wind
windLineChart <- ggplot(AirQualClean, aes(y=Wind, x= CalcDate))
windLineChart <- windLineChart + geom_line(color="green") + xlab("Month")
```

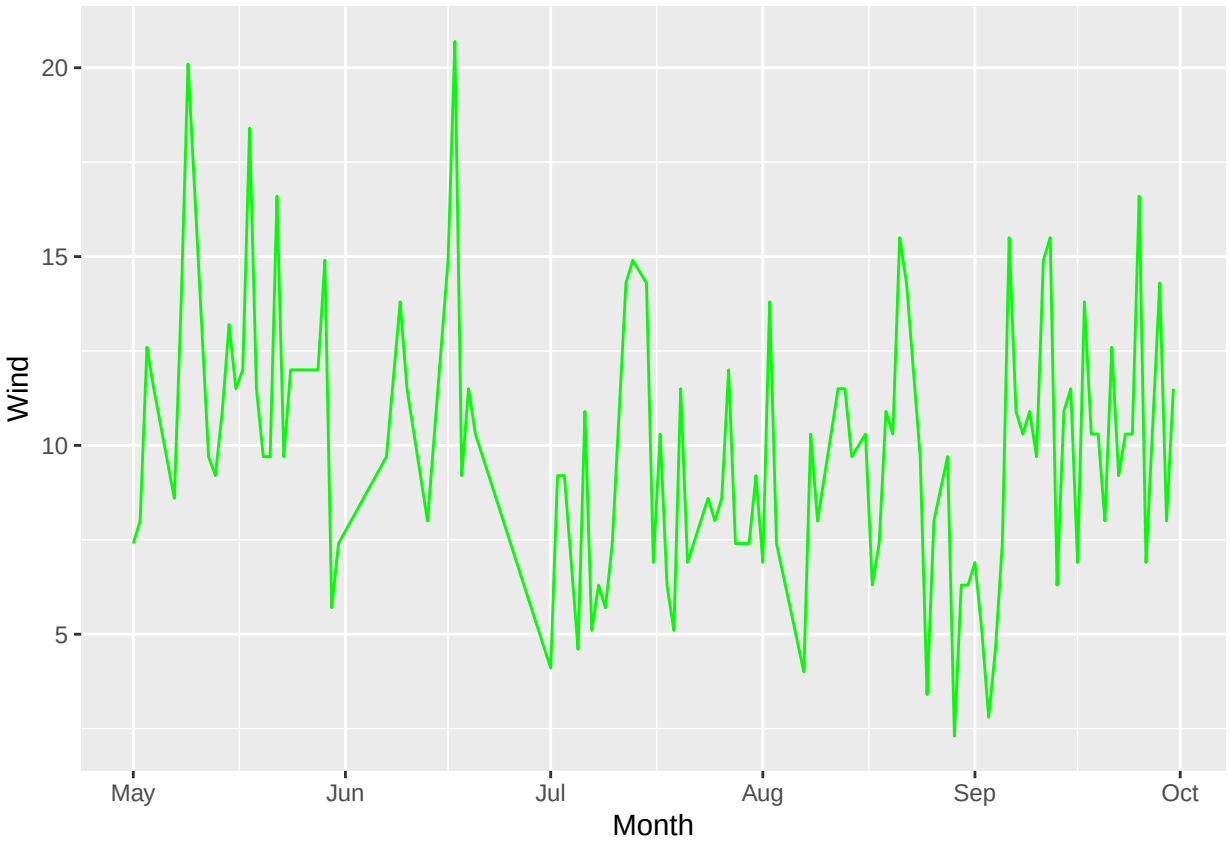
```
#LineChart Solar.R
solar.RLineChart <- ggplot(AirQualClean, aes(y=Solar.R, x= CalcDate))
solar.RLineChart <- solar.RLineChart + geom_line(color="purple") + xlab("Month")

#show the Charts
ozoneLineChart
```



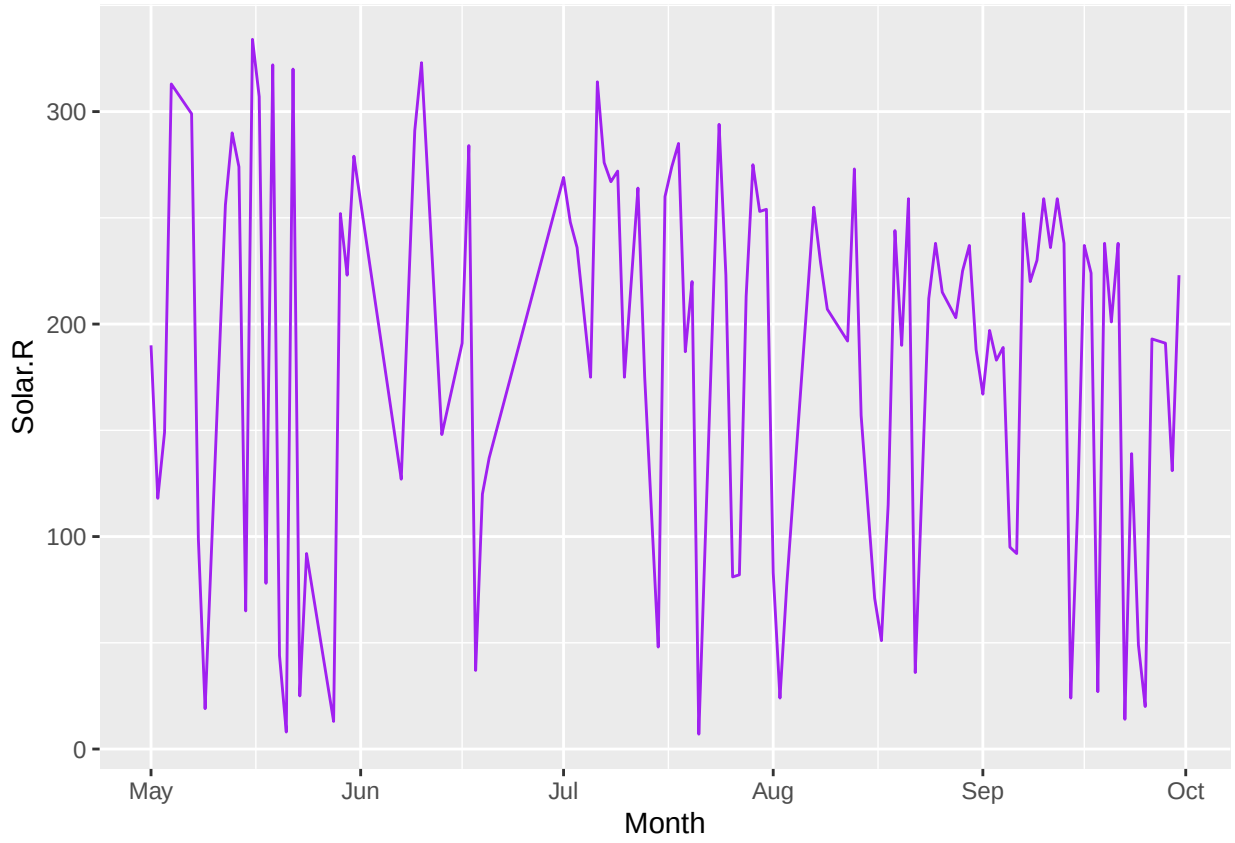

```
tempLineChart
```



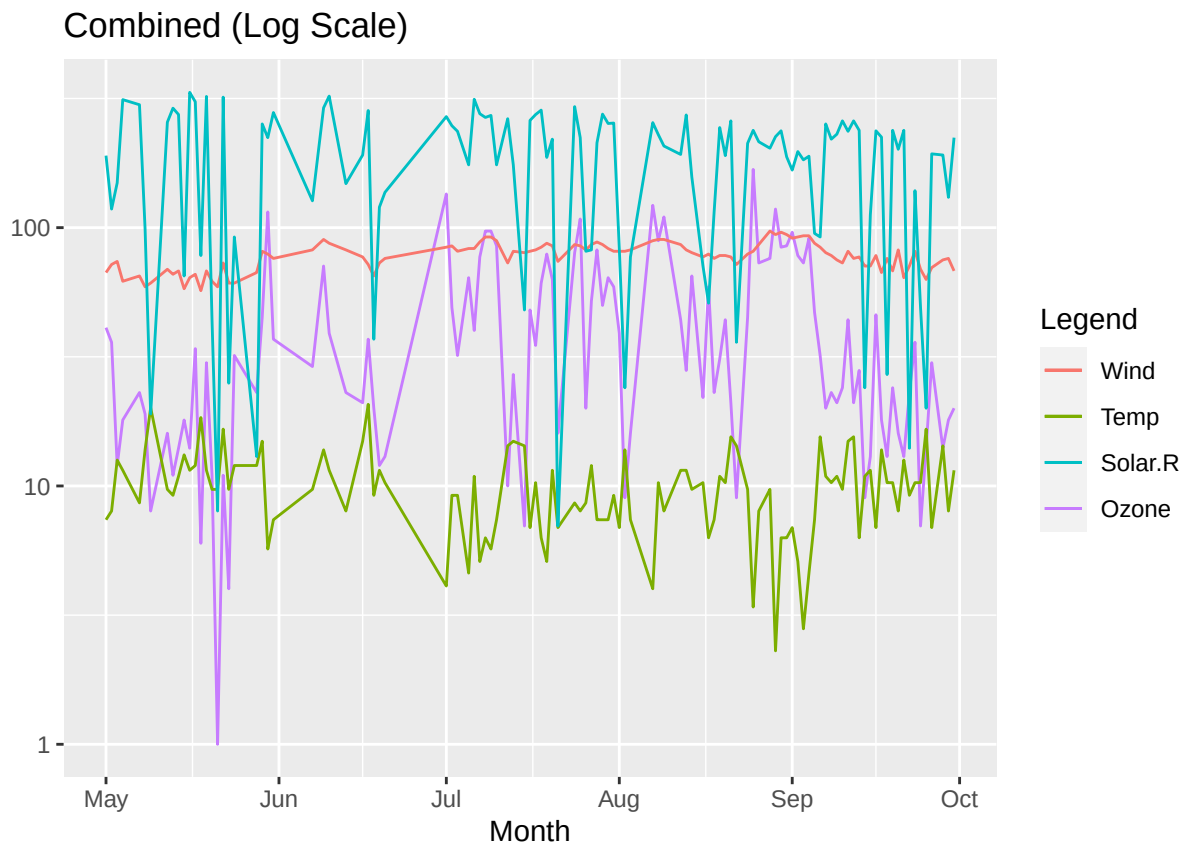

```
windLineChart
```



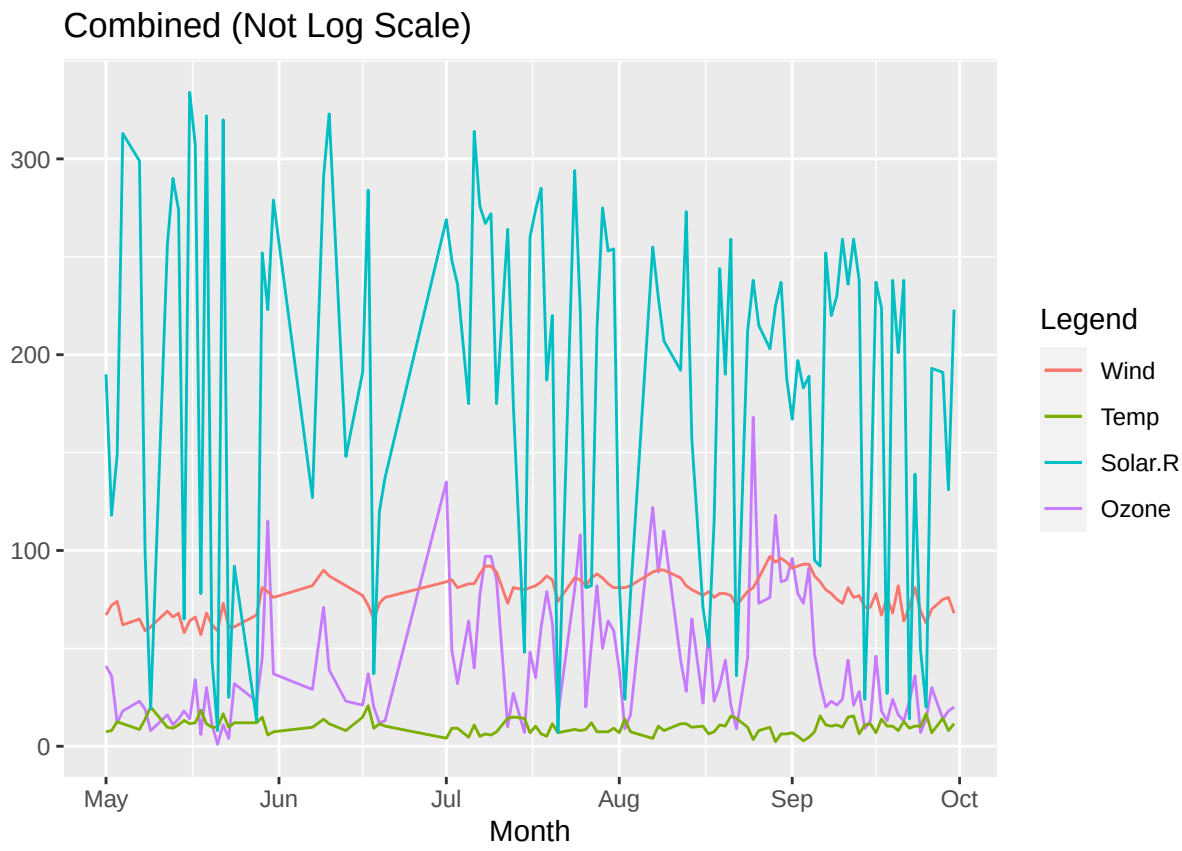

```
solar.RLineChart
```




```
#Combined Line Chart

varLables <- c("Wind", "Temp", "Solar.R", "Ozone" )

combinedChart <- ggplot(AirQualClean, aes(x= CalcDate, y=Ozone, color="red"))+ geom_line()
combinedChart <- combinedChart + geom_line(aes(y=Temp,color="blue"))
combinedChart <- combinedChart + geom_line(aes(y=Wind, color="green"))
combinedChart <- combinedChart + geom_line(aes(y=Solar.R, color="purple"))
combinedChart <- combinedChart + labs(title="Combined", size=10) + xlab("Month") + ylab(" ")  +
  scale_color_discrete(labels = paste(varLables))+
  guides(color = guide_legend(title = "Legend"))

#change to Log scale along the Y axis to make more readable across all data columns
logscalecombinedChart <- combinedChart + labs(title="Combined (Log Scale)", size=10)

logscalecombinedChart <- logscalecombinedChart +scale_y_continuous(trans='log10')

logscalecombinedChart
```



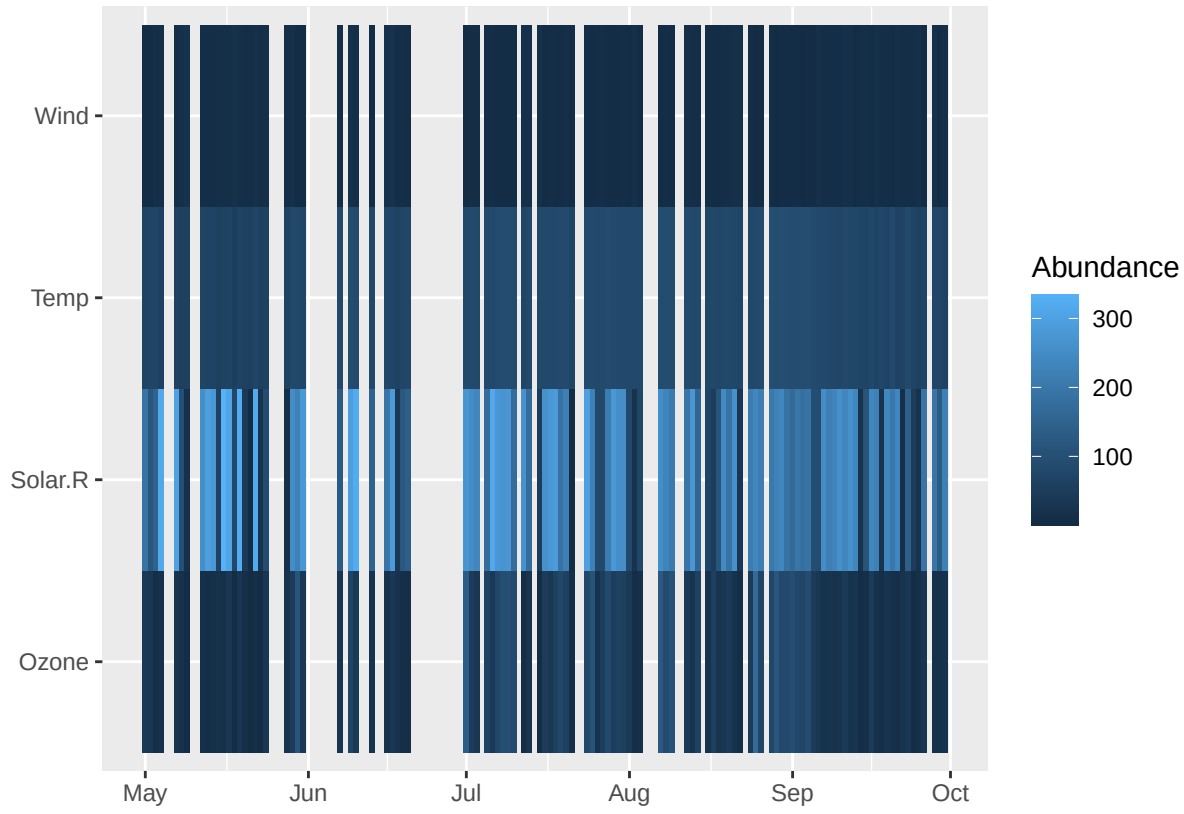

```
#For comparison same data without the Y axis in Log scale
combinedChart <- combinedChart + labs(title="Combined (Not Log Scale)", size=10)
combinedChart
```



Step 4: Look at all the data via a Heatmap

Create a heatmap, with each day along the x-axis and ozone, temp, wind and solar.r along the y-axis, and days as rows along the y-axis. Great the heatmap using geom_tile (this defines the ggplot geometry to be 'tiles' as opposed to 'lines' and the other geometry we have previously used).Note that you need to figure out how to show the relative change equally across all the variables.

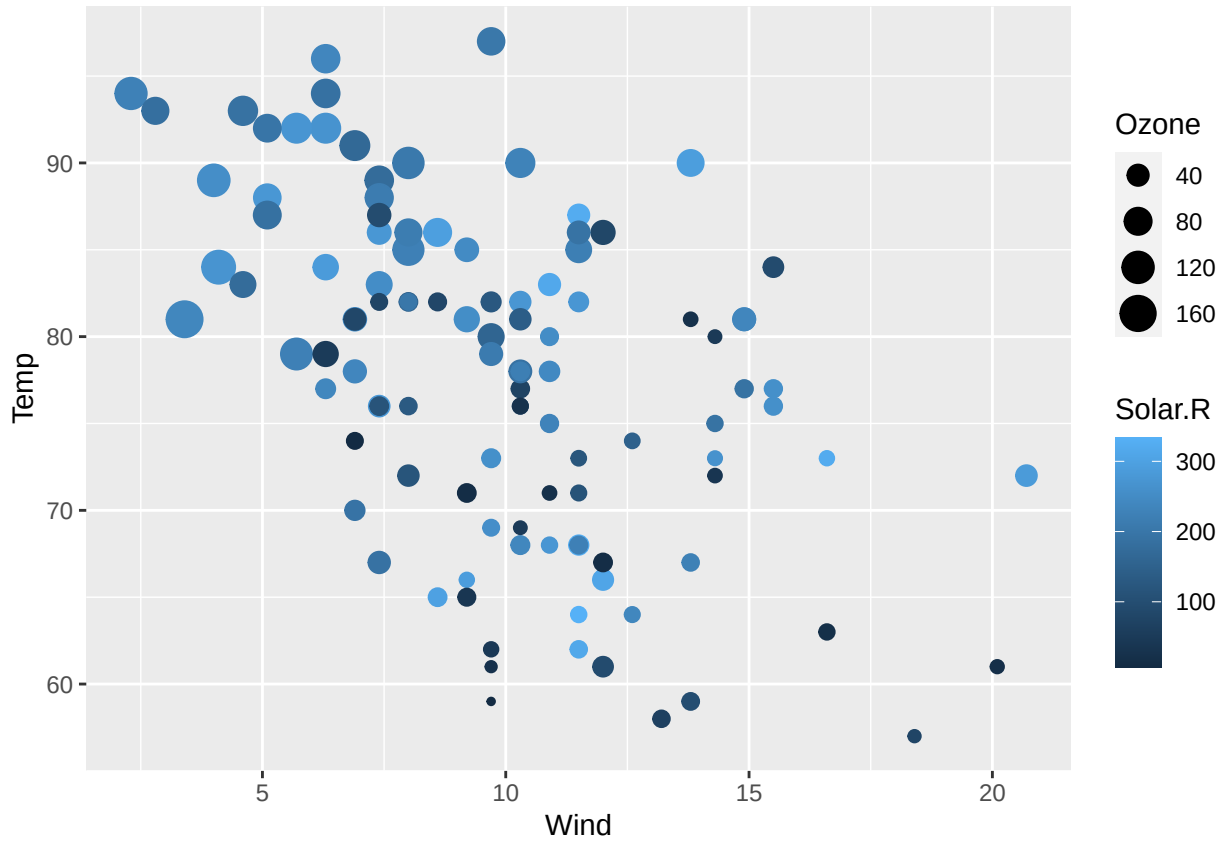
```
#format the data for use within a heatmap
heatmapdata <- pivot_longer(data = AirQualClean, cols = c(1:4),names_to = "Class", values_to = "Abundan

#Create Via ggplot2
airheatmap <- ggplot(data = heatmapdata, mapping = aes(x=CalcDate, y=Class, fill = Abundance))
airheatmap <- airheatmap + geom_tile() + ylab("") + xlab("")

#Show the heatmap
airheatmap
```



Step 5: Look at all the data via a scatter chart

Create a scatter chart (using ggplot geom_point), with the x-axis representing the wind, the y-axis representing the temperature, the size of each dot representing the ozone and the color representing the solar.R

```
scatterData <- AirQualClean[,1:4]

scatterPlot <- ggplot(scatterData, aes(x=Wind, y=Temp)) + geom_point(aes(size=Ozone, color=Solar.R))

scatterPlot
```



Step 6: Final Analysis

Do you see any patterns after exploring the data?
What was the most useful visualization?

The most intuitive visualization is the Scatter Plot. With this the audience is immediately able to identify a trend where at high temperatures there is also higher levels of ozone and vice versa. The audience is also able to quickly understand within the data present a general trend whereas wind increases temperatures decrease along with ozone. The relationship between decrease in temperature and wind speed more as a cluster that is not conclusive and should only cause a data scientist to seek more data or explore the data further.

The second most useful visualization would be the combined line chart where the Y Axis is scaled logarithmically. When not scaled logarithmically, it is difficult to identify trends as variability of some of the dataset's columns (most notably temperature) are flattened due to the range in values of adjacent columns (Solar.R).