# Homework #6 – Viz HW: air quality Analysis

## R Script (Code)

#

# Course: IST687

# Name: Joyce Woznica

# Homework 6 - Viz HW: air quality Analysts

# Due Date: 2/19/2019

# Date Submitted:

#

# Step 1: Load the data

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

myairQ<-airquality

# Step 2: Clean the data

# There will be NA's - figure out what you are going to do with that

# JLJW NOTE: I elected to use the mean for each column for the NAs as opposed to

# removing all the observations that had an NA for one of the values

# create a function to replace each column NA with mean for that column

replaceNAwMeans<-function(vec)

{

numcols<-length(colnames(vec))

index<-1

while(index<=numcols)

{

theColV <- vec[,index]

theColV[is.na(theColV)]<-mean(theColV,na.rm=TRUE)

vec[,index]<-theColV

index<-index+1

}

return(vec)

}

# Update the dataframe with the modified frame with means applied to NA values

myairQ<-replaceNAwMeans(myairQ)

# Step 3: Understand the data distribution

# Create the following visualizations using ggplot:

# 1) Histogrames for each of the variables

# Get the columns of myairQ

colnames(myairQ)

# function for correct packages

install.packages ("ggplot2")

library(ggplot2)

# histogram for Ozone with ggplot

g <- ggplot(myairQ, aes(x=Ozone))

g <- g + geom\_histogram(bins=20, color="blue", fill="green")

g <- g+ ggtitle("Air Quality:Ozone")

g

# histogram for Solar.R with ggplot

g <- ggplot(myairQ, aes(x=Solar.R))

g <- g + geom\_histogram(bins=20, color="red", fill="orange")

g <- g+ ggtitle("Air Quality:Solar.R")

g

# histogram for Wind with ggplot

g <- ggplot(myairQ, aes(x=Wind))

g <- g + geom\_histogram(bins=20, color="black", fill="lightblue")

g <- g+ ggtitle("Air Quality:Wind")

g

# histogram for Temperature with ggplot

g <- ggplot(myairQ, aes(x=Temp))

g <- g + geom\_histogram(bins=20, color="blue", fill="pink")

g <- g+ ggtitle("Air Quality:Temperature")

g

# - do we even need these two? They are not 'variables'

# histogram for Month with ggplot

g <- ggplot(myairQ, aes(x=Month))

g <- g + geom\_histogram(binwidth=1, bins=5, color="green", fill="navy")

g <- g+ ggtitle("Air Quality:Month")

g

# histogram for Day with ggplot

g <- ggplot(myairQ, aes(x=Day))

g <- g + geom\_histogram(bins=15, color="orange", fill="yellow")

g <- g+ ggtitle("Air Quality:Day")

g

# 2) Bloxplot for Ozone with ggplot

g<- ggplot(myairQ, aes(y=Ozone))

g<- g + geom\_boxplot(color="navy", fill="lightblue")

g<- g + ggtitle("Ozone Boxplot") + theme(plot.title=element\_text(hjust=0.5))

g

# also did against month - just for my own benefit

g<- ggplot(myairQ, aes(group=Month,x=Month,y=Ozone))

g<- g + geom\_boxplot(aes(fill=factor(Month)))

g<- g + ggtitle("Ozone by Month") + theme(plot.title=element\_text(hjust=0.5))

g

# 3) Boxplot for wind values (round the wind to get a good number of "buckets")

# Need to find "buckets" for the information

# maybe pick under 25%, 25% to 50%, 50% to 75%, 75% and up?

buildCutOffs<- function(mini, maxi, numcuts)

{

index<-numcuts

cutoffs<-c(0)

while(index>=1)

{

cutoffs<- c(cutoffs, round(maxi/index))

index<-index-1

}

return(cutoffs)

}

# create bins/buckets for box plot

plotBuckets<-buildCutOffs(min(myairQ$Wind),max(myairQ$Wind),4)

myairQwb <- myairQ

myairQwb$Bucket<-cut(myairQwb$Wind,plotBuckets)

g<- ggplot(myairQwb)

g<- g + geom\_boxplot(aes(Bucket, Wind, fill=factor(Bucket)))

g<- g + ggtitle("Wind BoxPlot") + theme(plot.title=element\_text(hjust=0.5))

g

# Step 3 (again): 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, then one chart with

# four lines, each having a different color).

# create these visualizations with ggplot

# \*\*Note that for the chart with 4 lines, you need to think about how to

# effectively use the y-axis

# create a function to create dates from 1973

# read month, day and create 1973-<month>-<day>

createDate <- function (yyyy,mm,dd)

{

as.Date(paste(yyyy,mm,dd,sep='-'))

}

# function to create a new column from the created date

makeDateCol <- function(df)

{

for (row in 1:nrow(df))

{

dateVal<-createDate(1973,df$Month,df$Day)

}

return(dateVal)

}

# now add the new column which is the date in 1973 in the form

# yyyy-mm-dd

myairQ$Date<-makeDateCol(myairQ)

# now do a line plot against Date for Ozone

g <- ggplot(myairQ, aes(x=Date, y=Ozone))

g <- g + geom\_line(size=1, color="navy")

g <- g + ylab("Ozone")

g <- g + ggtitle("Ozone by Date")+theme(plot.title=element\_text(hjust=0.5))

g

# now do a line plot against Date for Temp

g <- ggplot(myairQ, aes(x=Date, y=Temp))

g <- g + geom\_line(size=1, color="orange")

g <- g + ylab("Temperature")

g <- g + ggtitle("Temperature by Date")+theme(plot.title=element\_text(hjust=0.5))

g

# now do a line plot against Date for Wind

g <- ggplot(myairQ, aes(x=Date, y=Wind))

g <- g + geom\_line(size=1, color="lightblue")

g <- g + ylab("Wind")

g <- g + ggtitle("Wind by Date")+theme(plot.title=element\_text(hjust=0.5))

g

# now do a line plot against Date for Solar.R

g <- ggplot(myairQ, aes(x=Date, y=Solar.R))

g <- g + geom\_line(size=1, color="red")

g <- g + ylab("Solar.R")

g <- g + ggtitle("Solar.R by Date")+theme(plot.title=element\_text(hjust=0.5))

g

# now plot all of these on the same plot

# might have to change the data in the frame using 'melt'

install.packages("reshape2")

library(reshape2)

# flip the dataframe

# need to remove, Day, Month variable before melting

redmyairQ <- myairQ[,1:4]

redmyairQ$Date <- myairQ$Date

# scale Wind by 10

redmyairQ$Wind <- redmyairQ$Wind \* 10

# now only have Ozone, Temp, Wind, Solar.R and then Melt with Date

conmyairQ <- melt(redmyairQ, id="Date")

# Now plot by all variables on one

g <- ggplot(data=conmyairQ, aes(y=value, x=Date, color=variable))

g <- g + geom\_line(size=1) + ggtitle("Ozone, Temp, Wind and Solar.R")

g <- g + theme(plot.title=element\_text(hjust=0.5))

g

# 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

# Create using geom\_tile (tiles instead of lines)

# You will need to figure out how to show the relative change

# equally across all the variables

# \*\* still missing something here \*\*

g <- ggplot(data=conmyairQ, aes(x=Date, y=variable))

g <- g + geom\_tile(aes(fill=value))

g <- g + scale\_fill\_gradient(low = "white", high="navy")

g <- g + ggtitle("Heatmap of Air Quality Data by Value")

g

# Step 5: Look at all the data via a scatter plot

# Create a scatter chart (geom\_point),

# x-axis is wind

# y-axis is temperature

# dot size represents Ozone

# color represents Solar.R

g <- ggplot(myairQ, aes(x=Wind, y=Temp))

g <- g + geom\_point(aes(color=Solar.R, size=Ozone))

g <- g + ggtitle("Wind versus Temperature, Ozone and Solar.R")

g <- g + theme(plot.title=element\_text(hjust=0.5))

g

# Step 6: Final Analysis

# Do you see any patterns after exploring the data?

# ANSWER: a) Wind decreases in late summer, but is fairly high

# in the late Spring

# b) Lower the wind, lower the temperature, higher the Ozone

# c) Temperature, Ozone and Solar.R are related/correlated

#

# What was the most useful visualization?

# ANSWER: I found the scatterplot very useful. Without normalizing

# the data - it is hard to use the heat map. But you can

# see that there is definitely a relation between high

# Ozone and high Solar.R and Temperature in the heat map.

## Console Log (Executed Code)

> #

> # Course: IST687

> # Name: Joyce Woznica

> # Homework 6 - Viz HW: air quality Analysts

> # Due Date: 2/19/2019

> # Date Submitted:

> #

> # Step 1: Load the data

> # we will use the airquality data set, which you should already have as part of your R installation

> myairQ<-airquality

>

> # Step 2: Clean the data

> # There will be NA's - figure out what you are going to do with that

> # JLJW NOTE: I elected to use the mean for each column for the NAs as opposed to

> # removing all the observations that had an NA for one of the values

>

> # create a function to replace each column NA with mean for that column

> replaceNAwMeans<-function(vec)

+ {

+ numcols<-length(colnames(vec))

+ index<-1

+ while(index<=numcols)

+ {

+ theColV <- vec[,index]

+ theColV[is.na(theColV)]<-mean(theColV,na.rm=TRUE)

+ vec[,index]<-theColV

+ index<-index+1

+ }

+ return(vec)

+ }

>

> # Update the dataframe with the modified frame with means applied to NA values

> myairQ<-replaceNAwMeans(myairQ)

> # Step 3: Understand the data distribution

> # Create the following visualizations using ggplot:

> # 1) Histogrames for each of the variables

> # Get the columns of myairQ

> colnames(myairQ)

[1] "Ozone" "Solar.R" "Wind" "Temp" "Month" "Day"

> # function for correct packages

> install.packages ("ggplot2")

Error in install.packages : Updating loaded packages

> install.packages("ggplot2")

Installing package into ‘C:/Users/Joyce/Documents/R/win-library/3.5’

(as ‘lib’ is unspecified)

Warning in install.packages :

package ‘ggplot2’ is in use and will not be installed

> library(ggplot2)

> # histogram for Ozone with ggplot

> g <- ggplot(myairQ, aes(x=Ozone))

> g <- g + geom\_histogram(bins=20, color="blue", fill="green")

> g <- g+ ggtitle("Air Quality:Ozone")

> g

> # histogram for Solar.R with ggplot

> g <- ggplot(myairQ, aes(x=Solar.R))

> g <- g + geom\_histogram(bins=20, color="red", fill="orange")

> g <- g+ ggtitle("Air Quality:Solar.R")

> g

> # histogram for Wind with ggplot

> g <- ggplot(myairQ, aes(x=Wind))

> g <- g + geom\_histogram(bins=20, color="black", fill="lightblue")

> g <- g+ ggtitle("Air Quality:Wind")

> g

> # histogram for Temperature with ggplot

> g <- ggplot(myairQ, aes(x=Temp))

> g <- g + geom\_histogram(bins=20, color="blue", fill="pink")

> g <- g+ ggtitle("Air Quality:Temperature")

> g

> # - do we even need these two? They are not 'variables'

> # histogram for Month with ggplot

> g <- ggplot(myairQ, aes(x=Month))

> g <- g + geom\_histogram(binwidth=1, bins=5, color="green", fill="navy")

> g <- g+ ggtitle("Air Quality:Month")

> g

> # histogram for Day with ggplot

> g <- ggplot(myairQ, aes(x=Day))

> g <- g + geom\_histogram(bins=15, color="orange", fill="yellow")

> g <- g+ ggtitle("Air Quality:Day")

> g

> # 2) Bloxplot for Ozone with ggplot

> g<- ggplot(myairQ, aes(y=Ozone))

> g<- g + geom\_boxplot(color="navy", fill="lightblue")

> g<- g + ggtitle("Ozone Boxplot") + theme(plot.title=element\_text(hjust=0.5))

> g

> # 3) Boxplot for wind values (round the wind to get a good number of "buckets")

> # Need to find "buckets" for the information

> # maybe pick under 25%, 25% to 50%, 50% to 75%, 75% and up?

> buildCutOffs<- function(mini, maxi, numcuts)

+ {

+ index<-numcuts

+ cutoffs<-c(0)

+ while(index>=1)

+ {

+ cutoffs<- c(cutoffs, round(maxi/index))

+ index<-index-1

+ }

+ return(cutoffs)

+ }

> # create bins/buckets for box plot

> plotBuckets<-buildCutOffs(min(myairQ$Wind),max(myairQ$Wind),4)

> myairQwb <- myairQ

> myairQwb$Bucket<-cut(myairQwb$Wind,plotBuckets)

> g<- ggplot(myairQwb)

> g<- g + geom\_boxplot(aes(Bucket, Wind, fill=factor(Bucket)))

> g<- g + ggtitle("Wind BoxPlot") + theme(plot.title=element\_text(hjust=0.5))

> g

> # Step 3 (again): 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, then one chart with

> # four lines, each having a different color).

> # create these visualizations with ggplot

> # \*\*Note that for the chart with 4 lines, you need to think about how to

> # effectively use the y-axis

> # create a function to create dates from 1973

> # read month, day and create 1973-<month>-<day>

> createDate <- function (yyyy,mm,dd)

+ {

+ as.Date(paste(yyyy,mm,dd,sep='-'))

+ }

>

> # function to create a new column from the created date

> makeDateCol <- function(df)

+ {

+ for (row in 1:nrow(df))

+ {

+ dateVal<-createDate(1973,df$Month,df$Day)

+ }

+ return(dateVal)

+ }

> # now add the new column which is the date in 1973 in the form

> # yyyy-mm-dd

> myairQ$Date<-makeDateCol(myairQ)

> # now do a line plot against Date for Ozone

> g <- ggplot(myairQ, aes(x=Date, y=Ozone))

> g <- g + geom\_line(size=1, color="navy")

> g <- g + ylab("Ozone")

> g <- g + ggtitle("Ozone by Date")+theme(plot.title=element\_text(hjust=0.5))

> g

> # now do a line plot against Date for Temp

> g <- ggplot(myairQ, aes(x=Date, y=Temp))

> g <- g + geom\_line(size=1, color="orange")

> g <- g + ylab("Temperature")

> g <- g + ggtitle("Temperature by Date")+theme(plot.title=element\_text(hjust=0.5))

> g

> # now do a line plot against Date for Wind

> g <- ggplot(myairQ, aes(x=Date, y=Wind))

> g <- g + geom\_line(size=1, color="lightblue")

> g <- g + ylab("Wind")

> g <- g + ggtitle("Wind by Date")+theme(plot.title=element\_text(hjust=0.5))

> g

> # now do a line plot against Date for Solar.R

> g <- ggplot(myairQ, aes(x=Date, y=Solar.R))

> g <- g + geom\_line(size=1, color="red")

> g <- g + ylab("Solar.R")

> g <- g + ggtitle("Solar.R by Date")+theme(plot.title=element\_text(hjust=0.5))

> g

> # now plot all of these on the same plot

> # might have to change the data in the frame using 'melt'

> install.packages("reshape2")

Error in install.packages : Updating loaded packages

> install.packages("reshape2")

Installing package into ‘C:/Users/Joyce/Documents/R/win-library/3.5’

(as ‘lib’ is unspecified)

Warning in install.packages :

package ‘reshape2’ is in use and will not be installed

> library(reshape2)

> # flip the dataframe

> # need to remove, Day, Month variable before melting

> redmyairQ <- myairQ[,1:4]

> redmyairQ$Date <- myairQ$Date

> # scale Wind by 10

> redmyairQ$Wind <- redmyairQ$Wind \* 10

> # now only have Ozone, Temp, Wind, Solar.R and then Melt with Date

> conmyairQ <- melt(redmyairQ, id="Date")

> # Now plot by all variables on one

> g <- ggplot(data=conmyairQ, aes(y=value, x=Date, color=variable))

> g <- g + geom\_line(size=1) + ggtitle("Ozone, Temp, Wind and Solar.R")

> g <- g + theme(plot.title=element\_text(hjust=0.5))

> g

> # 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

> # Create using geom\_tile (tiles instead of lines)

> # You will need to figure out how to show the relative change

> # equally across all the variables

> # \*\* still missing something here \*\*

> g <- ggplot(data=conmyairQ, aes(x=Date, y=variable))

> g <- g + geom\_tile(aes(fill=value))

> g <- g + scale\_fill\_gradient(low = "white", high="navy")

> g <- g + ggtitle("Heatmap of Air Quality Data by Value")

> g

> # Step 5: Look at all the data via a scatter plot

> # Create a scatter chart (geom\_point),

> # x-axis is wind

> # y-axis is temperature

> # dot size represents Ozone

> # color represents Solar.R

> g <- ggplot(myairQ, aes(x=Wind, y=Temp))

> g <- g + geom\_point(aes(color=Solar.R, size=Ozone))

> g <- g + ggtitle("Wind versus Temperature, Ozone and Solar.R")

> g <- g + theme(plot.title=element\_text(hjust=0.5))

> g

> # Step 6: Final Analysis

> # Do you see any patterns after exploring the data?

> # ANSWER: a) Wind decreases in late summer, but is fairly high

> # in the late Spring

> # b) Lower the wind, lower the temperature, higher the Ozone

> # c) Temperature, Ozone and Solar.R are related/correlated

> #

> # What was the most useful visualization?

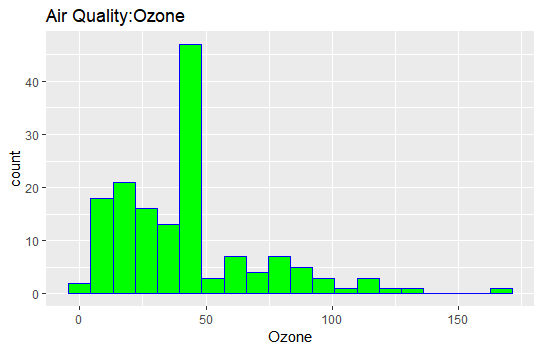
> # ANSWER: I found the scatterplot very useful. Without normalizing

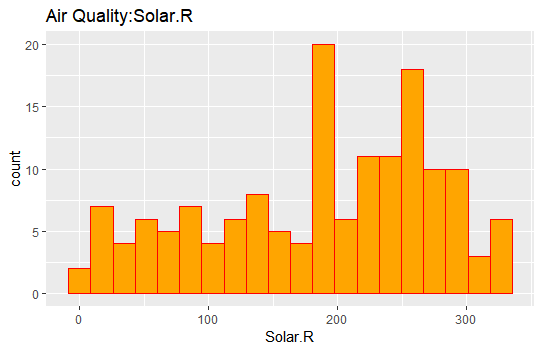
> # the data - it is hard to use the heat map. But you can

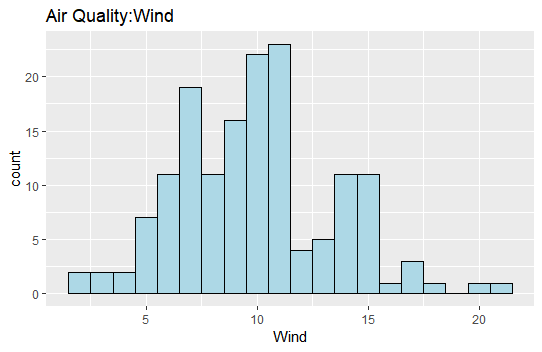
> # see that there is definitely a relation between high

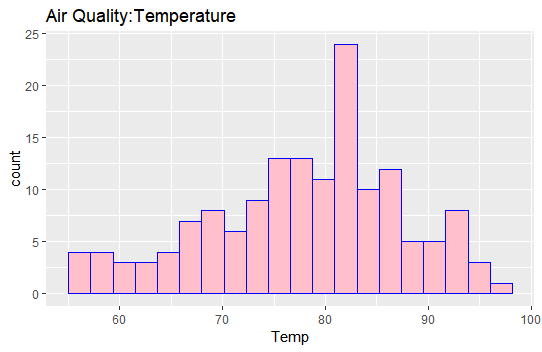
> # Ozone and high Solar.R and Temperature in the heat map.

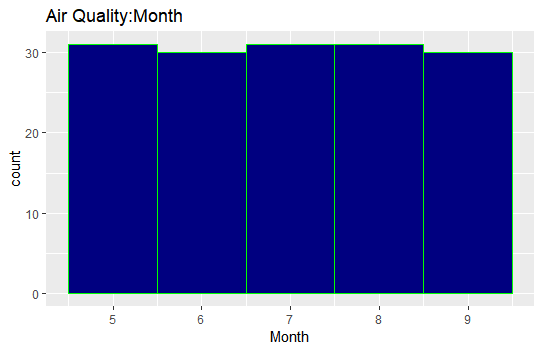
## Visualizations

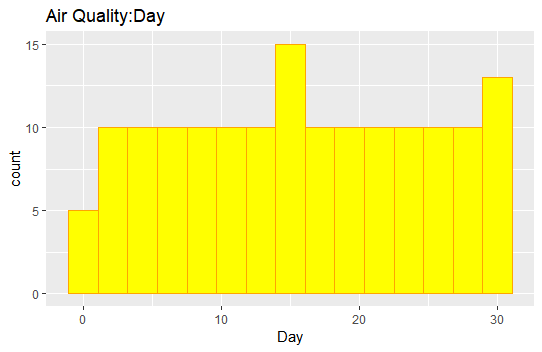


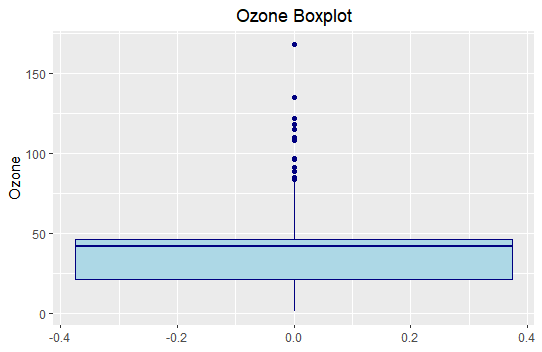


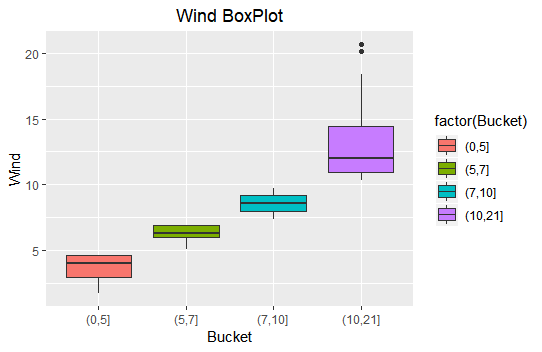


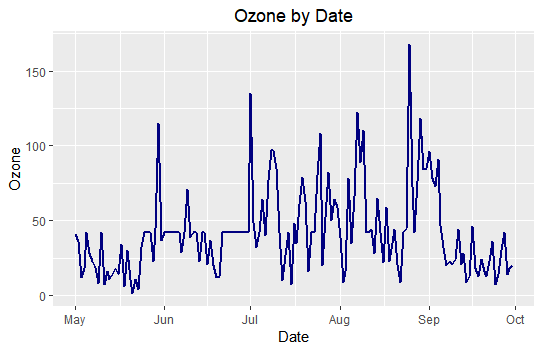


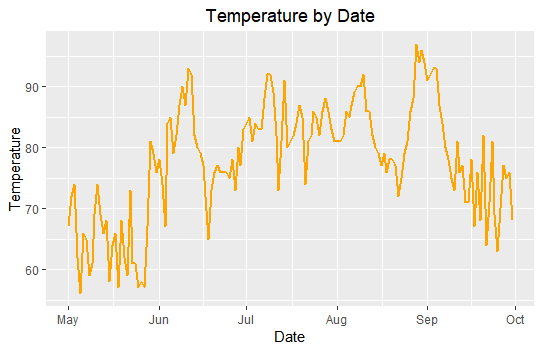


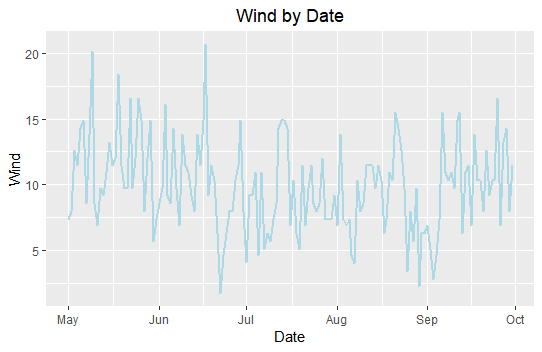


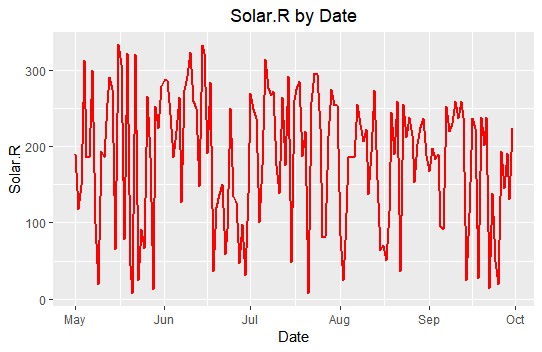


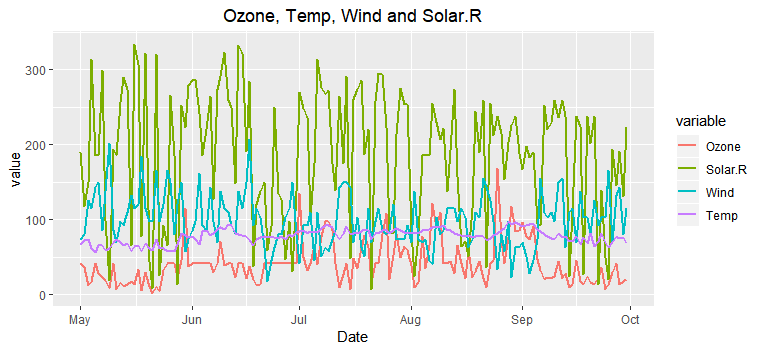


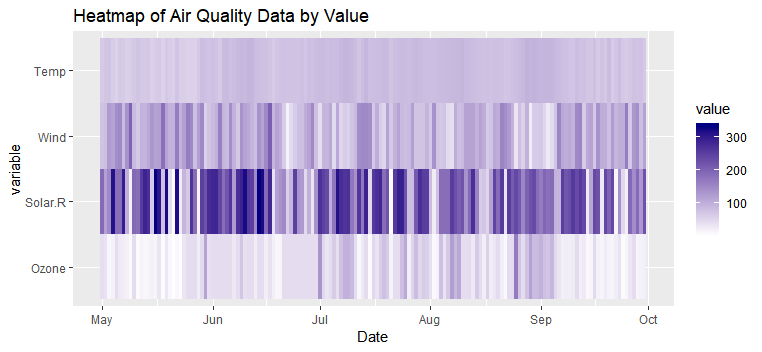


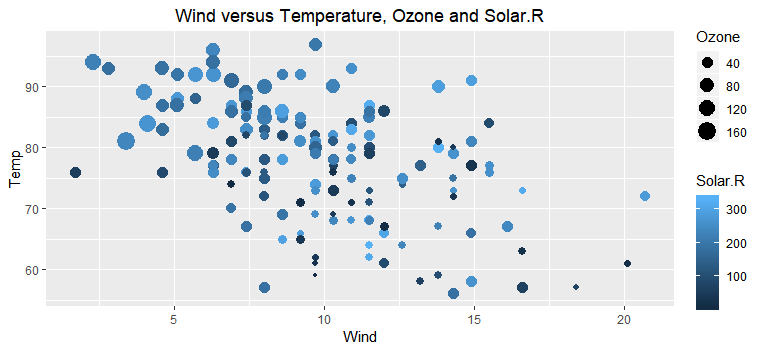






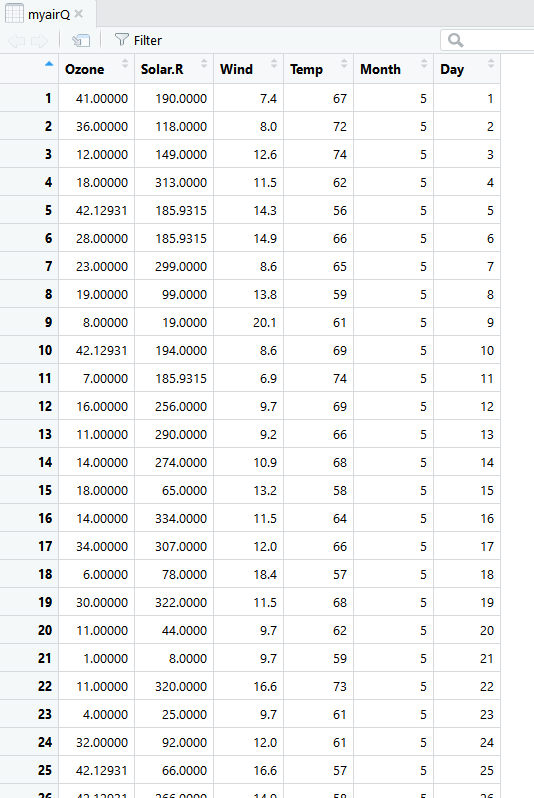




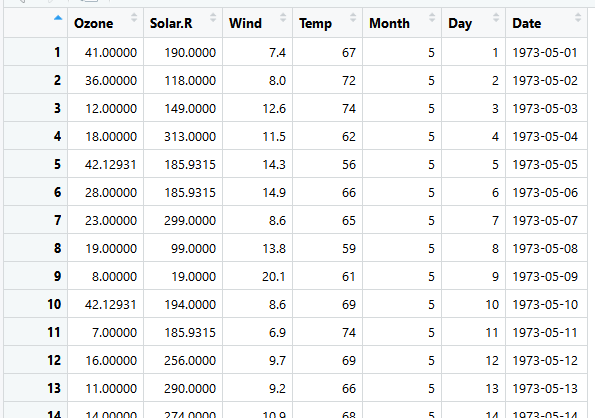


## Other Artifacts

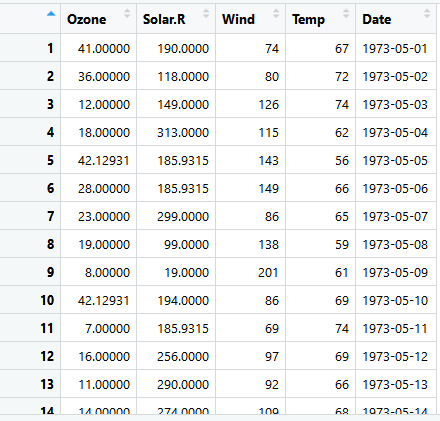
### myairQ



### myairQ with Date Column



### myairQ with Month and Day removed and Wind scaled by 10



### myairQ converted with melt function

