Intro to Data Science HW 8

```
# Enter your name here: Hrishikesh Telang
```

Copyright Jeffrey Stanton, Jeffrey Saltz, and Jasmina Tacheva

Attribution statement: (choose only one and delete the rest)

```
# 1. I did this homework by myself, with help from the book and the professor.
```

The chapter on **linear models** ("Lining Up Our Models") introduces **linear predictive modeling** using the tool known as **multiple regression**. The term "multiple regression" has an odd history, dating back to an early scientific observation of a phenomenon called **"regression to the mean."** These days, multiple regression is just an interesting name for using **linear modeling** to assess the **connection between one or more predictor variables and an outcome variable**.

In this exercise, you will **predict Ozone air levels from three predictors**.

A. We will be using the **airquality** data set available in R. Copy it into a dataframe called **air** and use the appropriate functions to **summarize the data**.

```
air <- airquality
head(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    NA      NA 14.3   56     5   5
## 6    28      NA 14.9   66     5   6
```

B. In the analysis that follows, **Ozone** will be considered as the **outcome variable**, and **Solar.R**, **Wind**, and **Temp** as the **predictors**. Add a comment to briefly explain the outcome and predictor variables in the dataframe using **?airquality**.

```
?airquality #returns the R documentation for the New York Air
#Quality Measurements dataset
```

C. Inspect the outcome and predictor variables – are there any missing values? Show the code you used to check for that.

```
#We use is.na() function for all the four examples
air$Ozone[is.na(air$Ozone)] #Returns 37 NA values
```

```
##  [1] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [26] NA NA NA NA NA NA NA NA NA NA NA NA
```

```r
air$Solar.R[is.na(air$Solar.R)] #Returns 7 NA values
```

```
## [1] NA NA NA NA NA NA NA
```

```r
air$Wind[is.na(air$Wind)] #There are no NA values
```

```
## numeric(0)
```

```r
air$Temp[is.na(air$Temp)] #There are no NA values
```

```
## integer(0)
```

D. Use the **na_interpolation()** function from the **imputeTS package** (remember this was used in a previous HW) to fill in the missing values in each of the 4 columns. Make sure there are no more missing values using the commands from Step C.

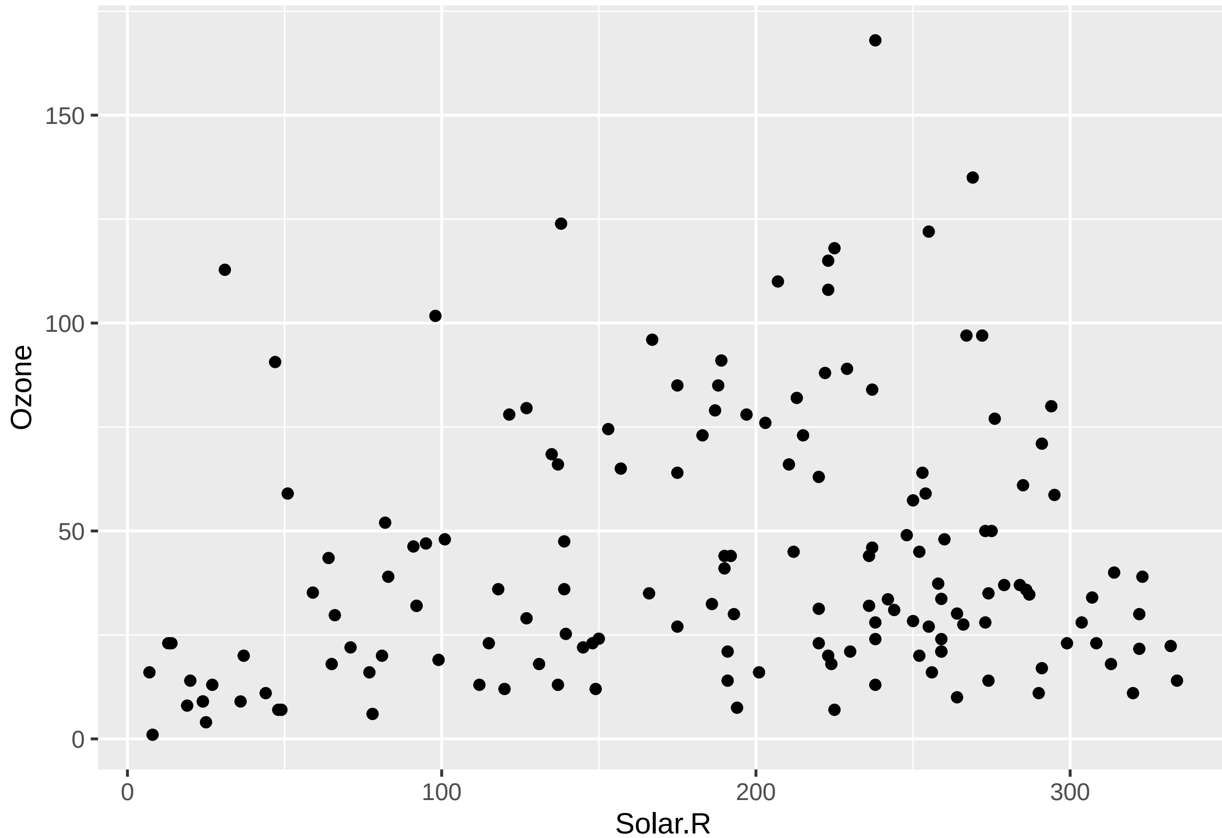
```r
library(imputeTS)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

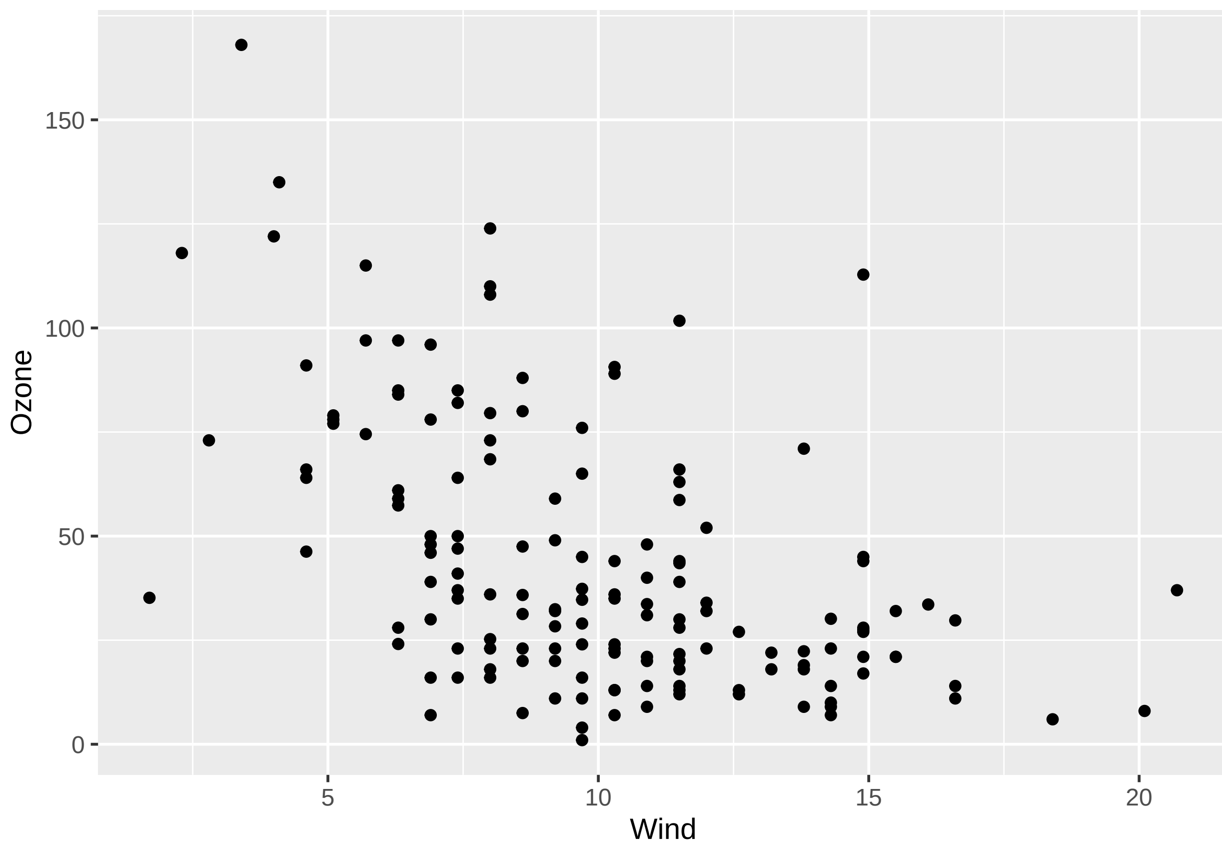
```r
air$Ozone <- na_interpolation(air$Ozone) #Filling in the 37 missing values in air$Ozone
air$Solar.R <- na_interpolation(air$Solar.R) #Filling in the 37 missing values in air$Solar.R
air$Wind <- na_interpolation(air$Wind) #The prompt requested me to use the function for air$Wind
air$Temp <- na_interpolation(air$Temp) #The prompt requested me to use the function for air$Temp
```

E. Create **3 bivariate scatterplots (X-Y) plots** (using ggplot), for each of the predictors with the outcome. **Hint:** In each case, put **Ozone on the Y-axis**, and a **predictor on the X-axis**. Add a comment to each, describing the plot and explaining whether there appears to be a **linear relationship** between the outcome variable and the respective predictor.

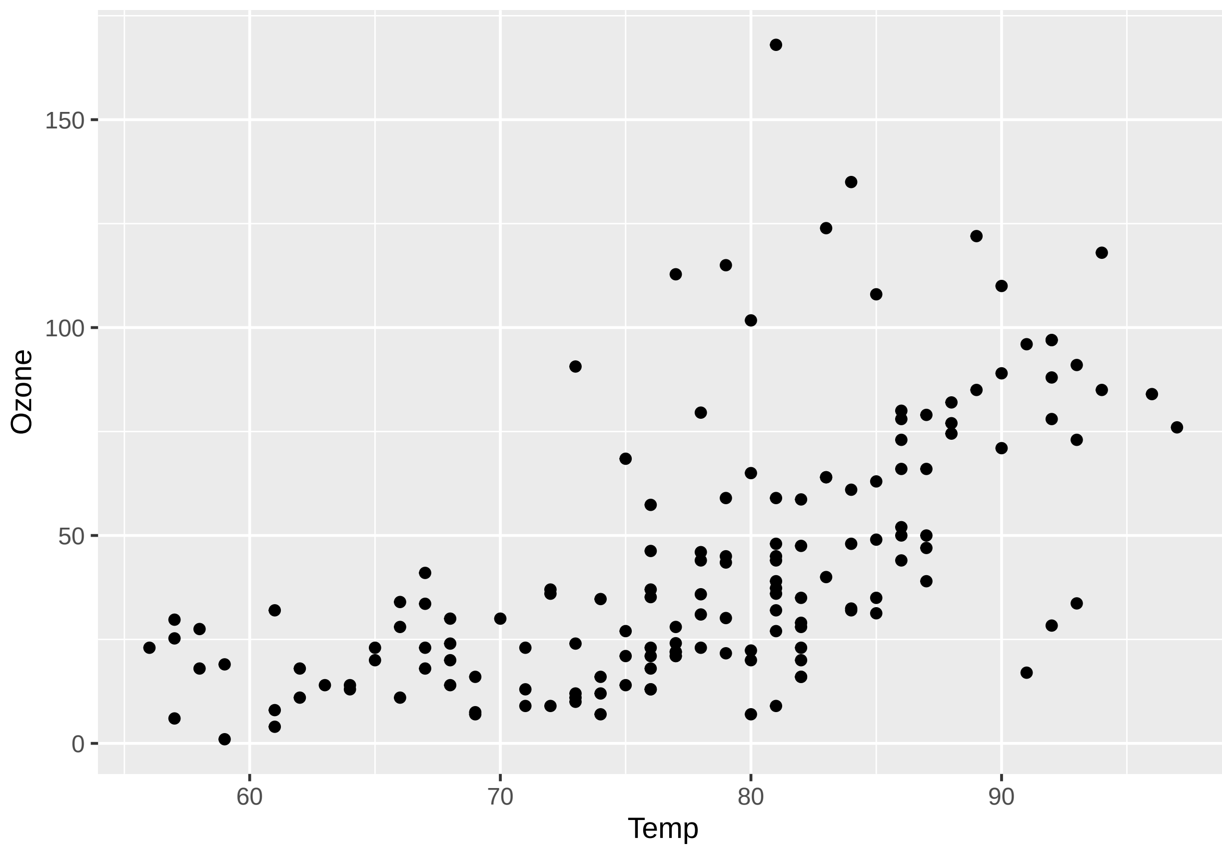
```r
library(ggplot2)
#We are performing scatter plots with changing x axis but constant y axis
ggplot(air) + geom_point(aes(x=Solar.R, y=Ozone))
```




```
ggplot(air) + geom_point(aes(x=Wind, y=Ozone))
```




```
ggplot(air) + geom_point(aes(x=Temp, y=Ozone))
```




```r
#There is barely any linear relationship in between Ozone and Solar.R, there is
#a gradual inverse correlation between Ozone and Wind, and a linear relationship
#between Ozone and Temp
```

F. Next, create a **simple regression model** predicting **Ozone based on Wind**, using the **lm( )** command. In a comment, report the **coefficient** (aka **slope** or **beta weight**) of **Wind** in the regression output and, **if it is statistically significant**, **interpret it** with respect to **Ozone**. Report the **adjusted R-squared** of the model and try to explain what it means.

```r
a <- lm(formula = Ozone ~ Wind, data=air)
summary(a)
```

```
##
## Call:
## lm(formula = Ozone ~ Wind, data = air)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.332 -18.332  -4.155  14.163  94.594
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  89.0205     6.6991  13.288  < 2e-16 ***
## Wind         -4.5925     0.6345  -7.238 2.15e-11 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 27.56 on 151 degrees of freedom
## Multiple R-squared:  0.2576, Adjusted R-squared:  0.2527
## F-statistic: 52.39 on 1 and 151 DF,  p-value: 2.148e-11
```

```
#The intercept is about 89 and the Wind is -4.592. Since the p value is <= 0.05,
#it is statistically significant
```

G. Create a **multiple regression model** predicting **Ozone** based on **Solar.R**, **Wind**, and **Temp**. **Make sure to include all three predictors in one model – NOT three different models each with one predictor.**

```
lmOut <- lm(formula = Ozone ~ Solar.R + Wind + Temp, data=air)
lmOut
```

```
##
## Call:
## lm(formula = Ozone ~ Solar.R + Wind + Temp, data = air)
##
## Coefficients:
## (Intercept)      Solar.R         Wind         Temp
##   -52.16596      0.01654     -2.69669      1.53072
```

```
summary(lmOut)
```

```
##
## Call:
## lm(formula = Ozone ~ Solar.R + Wind + Temp, data = air)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -39.651 -15.622  -4.981  12.422 101.411
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -52.16596   21.90933  -2.381   0.0185 *
## Solar.R       0.01654    0.02272   0.728   0.4678
## Wind         -2.69669    0.63085  -4.275 3.40e-05 ***
## Temp          1.53072    0.24115   6.348 2.49e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 24.26 on 149 degrees of freedom
## Multiple R-squared:  0.4321, Adjusted R-squared:  0.4207
## F-statistic: 37.79 on 3 and 149 DF,  p-value: < 2.2e-16
```

H. Report the **adjusted R-Squared** in a comment – how does it compare to the adjusted R-squared from Step F? Is this better or worse? Which of the predictors are **statistically significant** in the model? In a comment, report the coefficient of each predictor that is statistically significant. Do not report the coefficients for predictors that are not significant.

```r
#The adjusted R squared in G is 0.4207. It is way better than that in F, with
#adjusted R squared as 0.2527, making G a relatively higher fitting variable.
```

I. Create a one-row data frame like this:

```r
predDF <- data.frame(Solar.R=290, Wind=13, Temp=61)
```

and use it with the **predict( )** function to predict the **expected value of Ozone**:

```r
predict(lmOut, predDF)
```

```
##       1
## 10.9464
```

J. Create an additional **multiple regression model**, with **Temp** as the **outcome variable**, and the other **3 variables** as the **predictors**.

Review the quality of the model by commenting on its **adjusted R-Squared**.

```r
lmOut2 <- lm(formula = Temp ~ Solar.R + Wind + Temp, data=air)
```

```
## Warning in model.matrix.default(mt, mf, contrasts): the response appeared on the
## right-hand side and was dropped

## Warning in model.matrix.default(mt, mf, contrasts): problem with term 3 in
## model.matrix: no columns are assigned
```

```r
summary(lmOut2)
```

```
##
## Call:
## lm(formula = Temp ~ Solar.R + Wind + Temp, data = air)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -22.167  -5.301   1.178   5.183  18.440
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 85.675648   2.468453  34.708  < 2e-16 ***
## Solar.R      0.022932   0.007461   3.074  0.00251 **
## Wind        -1.213450   0.189226  -6.413 1.76e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.215 on 150 degrees of freedom
## Multiple R-squared:  0.2566, Adjusted R-squared:  0.2467
## F-statistic: 25.88 on 2 and 150 DF,  p-value: 2.202e-10
```

```
#The model has a poor fitting model as the adjusted R square is a meager 0.2467
```