Intro to Data Science HW 7


```
# Enter your name here: Benjamin Tisinger
```

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

```
# 1. I did this homework by myself, with help from the book and the professor and WEBSITE HERE h
ttps://www.rdocumentation.org/packages/datasets/versions/3.6.2/topics/airquality
```

```
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──
## ✓ ggplot2 3.3.6      ✓ purrr   0.3.5
## ✓ tibble  3.1.8      ✓ dplyr   1.0.10
## ✓ tidyr   1.2.1      ✓ stringr 1.4.1
## ✓ readr   2.1.3      ✓ forcats 0.5.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✕ dplyr::filter() masks stats::filter()
## ✕ dplyr::lag()    masks stats::lag()
```

```
library(dplyr)
library(imputeTS)
```

```
## Warning: package 'imputeTS' was built under R version 4.2.2
```

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

```
library(ggplot2)
```

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(airquality,5)
```

```
##   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
```

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**.

```
#https://www.rdocumentation.org/packages/datasets/versions/3.6.2/topics/airquality

#Ozone: Mean ozone in parts per billion from 1300 to 1500 hours at Roosevelt Island

#Solar.R: Solar radiation in Langleys in the frequency band 4000--7700 Angstroms from 0800 to 12
00 hours at Central Park

#Wind: Average wind speed in miles per hour at 0700 and 1000 hours at LaGuardia Airport

#Temp: Maximum daily temperature in degrees Fahrenheit at La Guardia Airport.
```

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

```
summary(is.na(air$Ozone))
```

```
##    Mode   FALSE    TRUE
## logical     116      37
```

```
#NA COUNT IS 37
```

```
summary(is.na(air$Solar.R))
```

```
##    Mode   FALSE    TRUE
## logical     146       7
```

```
#NA COUNT IS 7
```

```
summary(is.na(air$Wind))
```

```
##    Mode   FALSE
## logical     153
```

```
#NA COUNT IS 0
```

```
summary(is.na(air$Temp))
```

```
##    Mode   FALSE
## logical     153
```

```
#NA COUNT IS 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.

```
air$Ozone <- na_interpolation(air$Ozone)
air$Solar.R <- na_interpolation(air$Solar.R)
air$Wind <- na_interpolation(air$Wind)
air$Temp <- na_interpolation(air$Temp)

summary(is.na(air$Ozone))
```

```
##    Mode   FALSE
## logical     153
```

```
summary(is.na(air$Solar.R))
```

```
##    Mode   FALSE
## logical     153
```

```
summary(is.na(air$Wind))
```

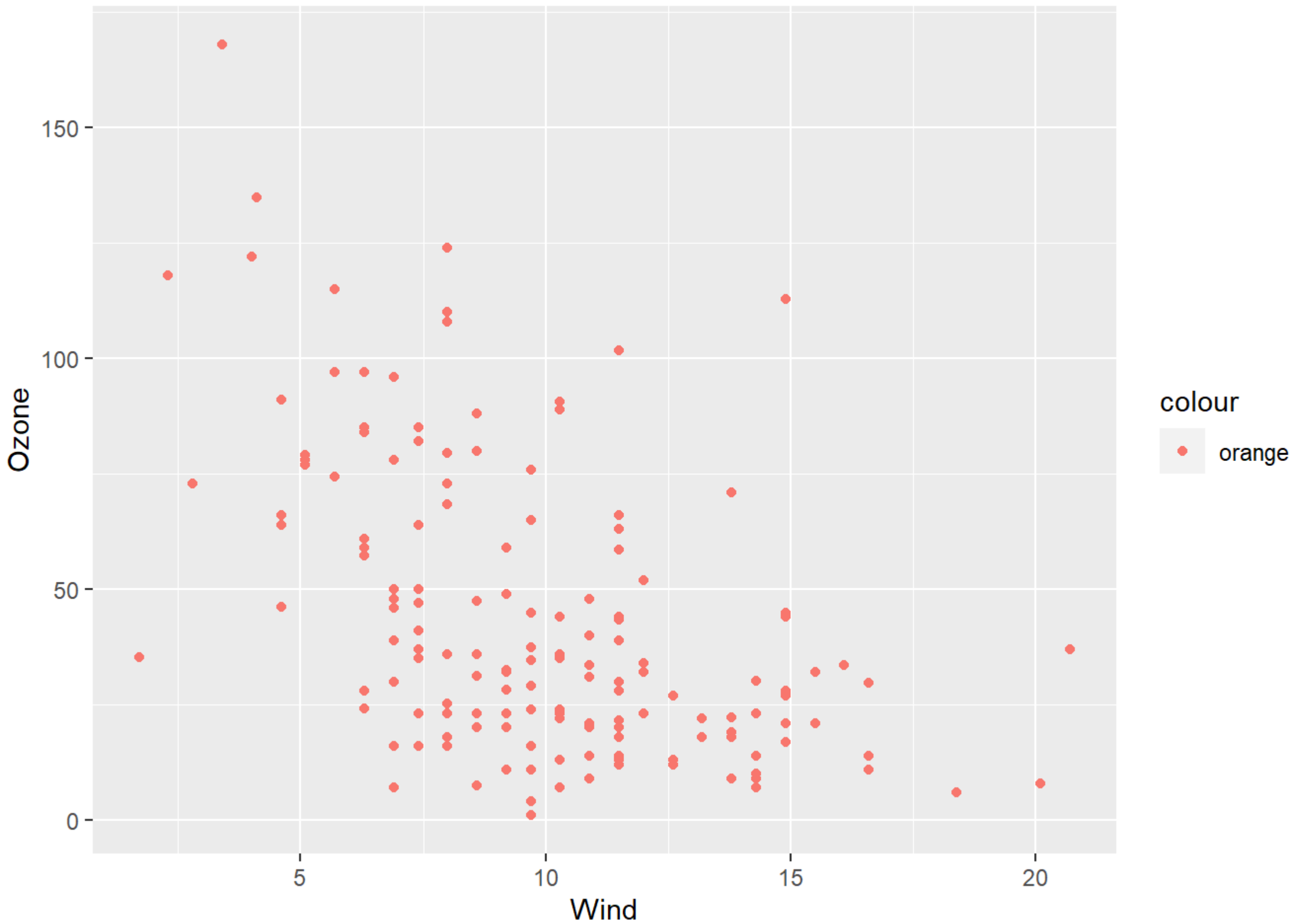
```
##    Mode   FALSE
## logical     153
```

```
summary(is.na(air$Temp))
```

```
##    Mode   FALSE
## logical     153
```

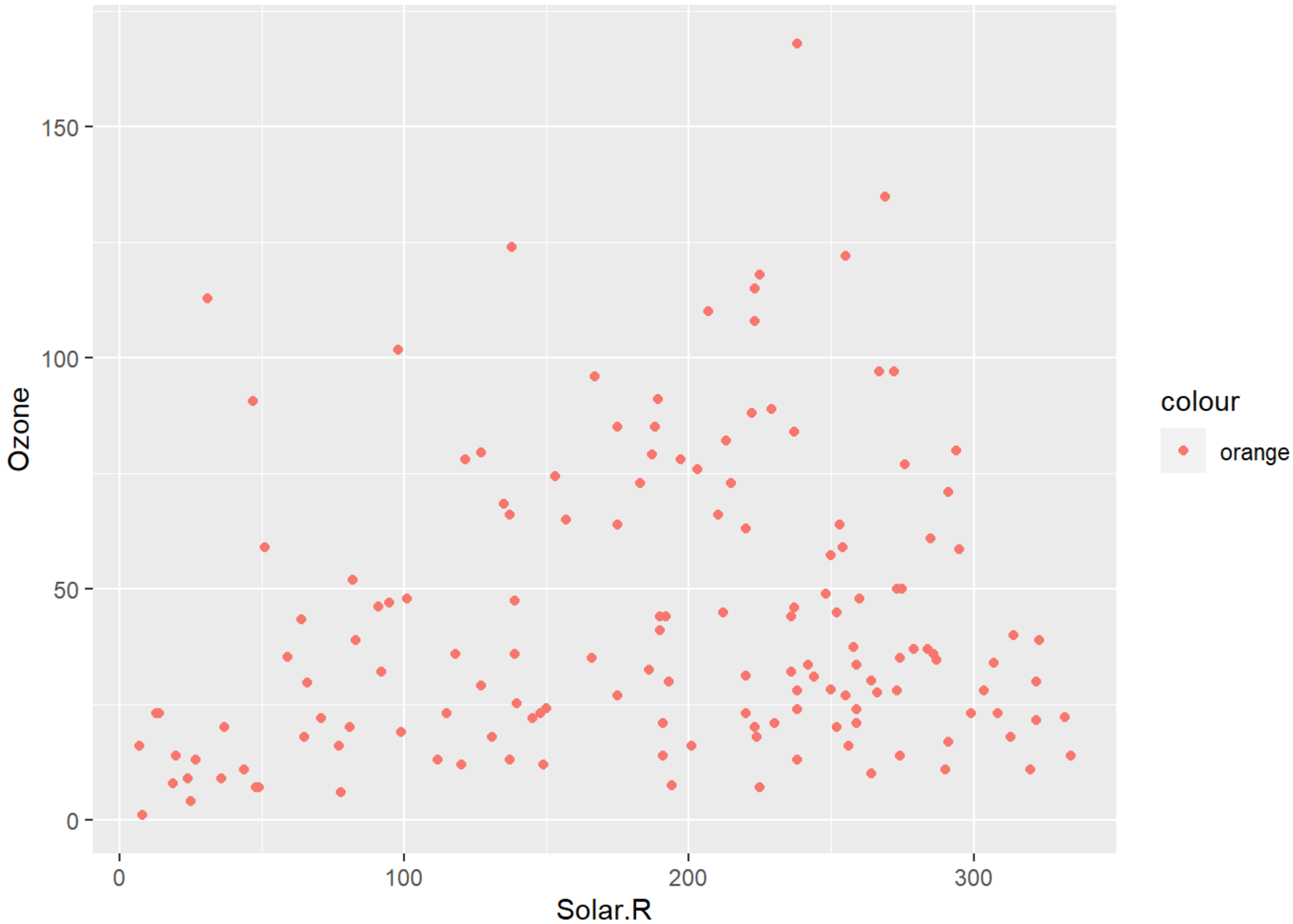
E. Create **3 bivariate scatterplots (X-Y) plots** (using ggplot), for each of the predictors with the outcome. **Hint:** In each case, put **Ozone on the Y-axis**, and a **predictor on the X-axis**. Add a comment to each, describing the plot and explaining whether there appears to be a **linear relationship**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

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



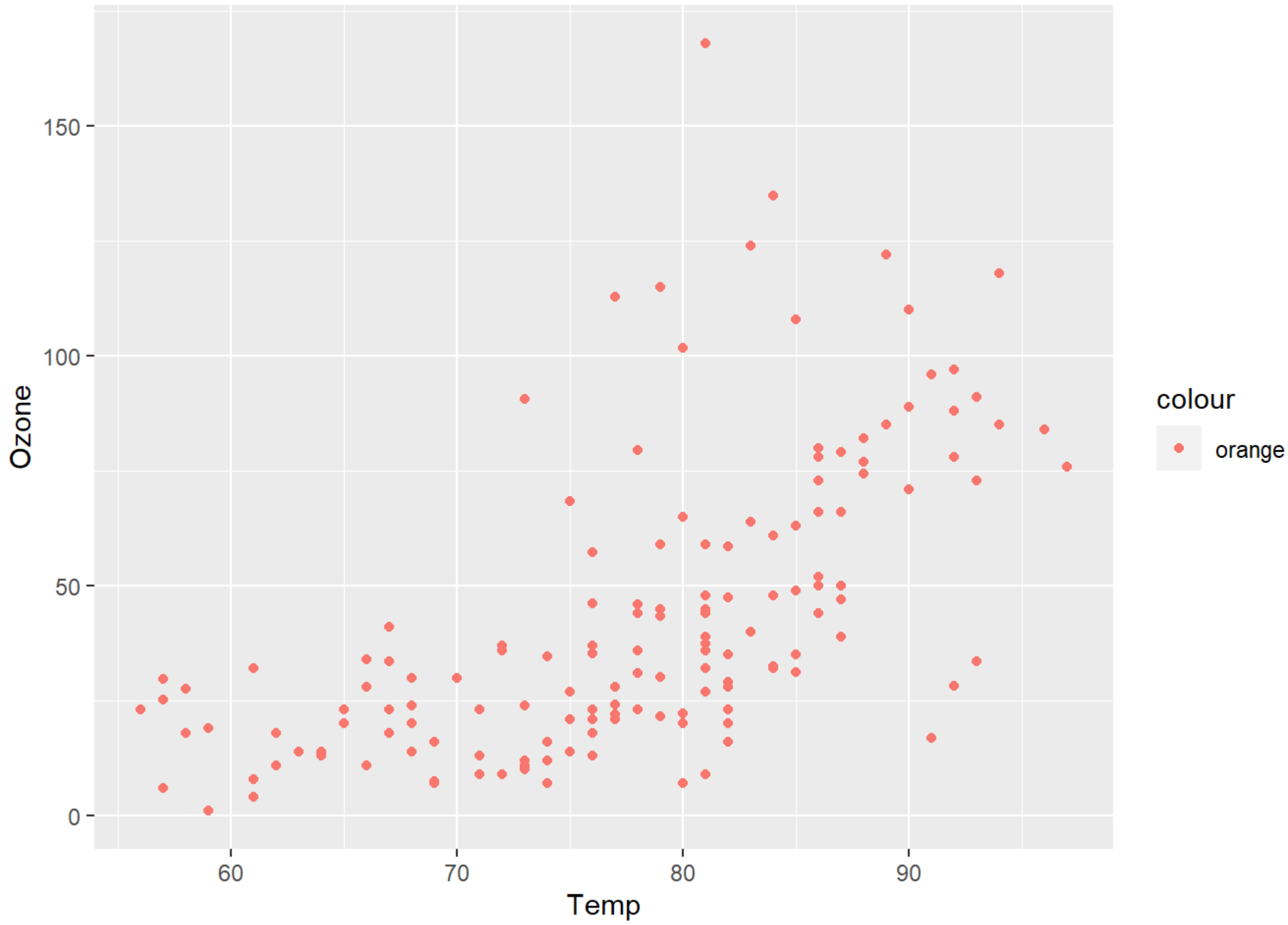

```
#Negative Linear or non linear. Dots are higher on the left than the Right.
```

```
ggplot(air) +
  geom_point(aes(x=Solar.R, y=Ozone,color="orange"))
```




```
#Appears to be Somewhat linear. Some points move in a Linear Direction, Others do not.
```

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




```
#Appears to be pretty linear. Points move in a Linear motion to the right
```

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.

```
ozone_wind <- lm(Ozone ~ Wind, data = air)
show(ozone_wind)
```

```
##
## Call:
## lm(formula = Ozone ~ Wind, data = air)
##
## Coefficients:
## (Intercept)         Wind
##      89.021       -4.592
```

```
summary(ozone_wind)
```

```
##
## 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
```

```
#CoEff -4.592
#Adjusted RSquare - 0.2576
#Pvalue  2.148e-11
#Statistically Important
```

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.

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

```
##
## 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(ozone_all)
```

```
##
## 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.

```
#Adjusted RSquare is 0.4207
#The coefficient of Wind was -2.69669 (Only Two that Showed Improvement)
# The coefficient of Temp was 0.153072 (Only Two that Showed Improvement)
#Looking at the Two Different Outcomes from Above and the one just ran, the correlation proves t
hat an improvement was shown by including the other variables (Wind and Temp) to the statistic.
```

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

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

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

```
predict(ozone_all,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**.

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

```
##
## Call:
## lm(formula = Temp ~ Ozone + Wind + Solar.R, data = air)
##
## Coefficients:
## (Intercept)        Ozone         Wind      Solar.R
##    74.69322      0.13905     -0.58018      0.01575
```

```
summary(temp_main)
```

```
##
## Call:
## lm(formula = Temp ~ Ozone + Wind + Solar.R, data = air)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.831  -4.802   1.174   4.880  18.004
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 74.693222   2.796787  26.707  < 2e-16 ***
## Ozone        0.139055   0.021907   6.348 2.49e-09 ***
## Wind        -0.580176   0.195774  -2.963  0.00354 **
## Solar.R      0.015751   0.006737   2.338  0.02072 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.313 on 149 degrees of freedom
## Multiple R-squared:  0.4148, Adjusted R-squared:  0.403
## F-statistic: 35.21 on 3 and 149 DF,  p-value: < 2.2e-16
```

```
# Adjusted R-squared:  0.403 - 40.3% of Data can be Explained by Correlation of all Variables
# All other Variables are important for correlation reporting to Temp. (Wind, Solar and Ozone)
```