Capstone with Airquality Data

Min Gong

Table of Contents

# Problem 1: Importing and exploring data

## P1.1. Rename Data set

* Get a local copy of the dataset “airquality” and name it “df” so that you can use it.

#rename the dataset  
df <- airquality  
head(df)

## 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

* Identify data type, change it to tibble data type, and make a change to the df.

typeof(df)

## [1] "list"

class(df)

## [1] "data.frame"

#convert to tibble and save it as df  
df <- as\_tibble(df)

* Confirm the data type is tibble by printing df out.

print(df)

## # A tibble: 153 × 6  
## Ozone Solar.R Wind Temp Month Day  
## <int> <int> <dbl> <int> <int> <int>  
## 1 41 190 7.4 67 5 1  
## 2 36 118 8 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  
## 7 23 299 8.6 65 5 7  
## 8 19 99 13.8 59 5 8  
## 9 8 19 20.1 61 5 9  
## 10 NA 194 8.6 69 5 10  
## # ℹ 143 more rows

class(df)

## [1] "tbl\_df" "tbl" "data.frame"

## P1.2. Variable Definition and Background of the Topic

* Look up the help to understand the definition of the variables.

help("airquality")

* In addition, look up Ozone and related variables on the internet. A quick search on Ozone leads me to <https://www.epa.gov/ozone-pollution-and-your-patients-health/what-ozone>. Read a bit to gain domain knowledge, which is needed to analyze the data. It appears that Southern California has the highest concentration of Ozone.
* Given the definition of the data and the knowledge you gained from your research, what would you think are potential dependent variables and independent variables? Can you form a hypothesis regarding the relationships among the variables?

It seems reasonable to treat Ozone as a dependent variable and the Solar.R, Wind, and Temp as independent variables. Also, the Ozone amount may be dependent on the Month, such that Ozone amount is highest during hottest months.

Thus, I would form hypotheses as follows.

H1: Ozone amount will be associated positively with Solar radiation amount (Solar.R)  
H2: Ozone amount will be associated negatively with Wind speed (Wind)  
H3: Ozone amount will be associated positively with Maximum daily temperature (Temp)  
H4: Ozone amount will be highest during summer months.

* **Answer to this question**:  
  *Definition of data*: airquality is a daily air quality measurements in New York, from May to September in 1973.  
  It contains 153 observations on 6 variables:
* Ozone: Ozone concentration (parts per billion)
* Solar.R: Solar radiation (lang)
* Wind: Wind speed (mph)
* Temp: Temperature (degrees Fahrenheit)
* Month: Month of observation (1 = January, 2 = February, etc.)
* Day: Day of observation (1–31)  
  *Dependent Variable*: Ozone  
  *Independent variables*: Solar.R, Wind, Temp  
  Hypothesis 1: Higher solar radiation levels lead to higher ozone concentrations.  
  Hypothesis 2: Higher temperatures lead to higher ozone concentrations.  
  Hypothesis 3: Wind speed negatively affects ozone concentration.  
  Hypothesis 4: Ozone concentrations vary by month.

## P1.3. View data

* Next, show the first 7 rows of it. Pay attention to the names of the variables.

head(df, n = 7L)

## # A tibble: 7 × 6  
## Ozone Solar.R Wind Temp Month Day  
## <int> <int> <dbl> <int> <int> <int>  
## 1 41 190 7.4 67 5 1  
## 2 36 118 8 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  
## 7 23 299 8.6 65 5 7

#colnames(df)

* Look for unique values of categorical values (i.e., Month and Day variables). What did you find? Do you feel you should change the data type of the two variables? Why or why not?

unique(df$Month) #Month could be converted to character/factor for further analysis and visualization as categorical values.

## [1] 5 6 7 8 9

unique(df$Day) #Day could remain as numeric since most of the analyses is within each month

## [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25  
## [26] 26 27 28 29 30 31

* A change of calculation(sum/mean/min/max) for the variables Month and Day to get a monthly data, it would help with analysis the data better.

#calculation(sum/mean/min/max)   
df\_stats <- df %>%  
 group\_by(Month) %>%  
 summarise(across(c(Ozone, Solar.R, Wind, Temp),   
 list(sum = ~sum(.x, na.rm = TRUE),  
 mean = ~mean(.x, na.rm = TRUE),  
 min = ~min(.x, na.rm = TRUE),  
 max = ~max(.x, na.rm = TRUE)),  
 .names = "{col}\_{fn}"))  
print(df\_stats)

## # A tibble: 5 × 17  
## Month Ozone\_sum Ozone\_mean Ozone\_min Ozone\_max Solar.R\_sum Solar.R\_mean  
## <int> <int> <dbl> <int> <int> <int> <dbl>  
## 1 5 614 23.6 1 115 4895 181.  
## 2 6 265 29.4 12 71 5705 190.  
## 3 7 1537 59.1 7 135 6711 216.  
## 4 8 1559 60.0 9 168 4812 172.  
## 5 9 912 31.4 7 96 5023 167.  
## # ℹ 10 more variables: Solar.R\_min <int>, Solar.R\_max <int>, Wind\_sum <dbl>,  
## # Wind\_mean <dbl>, Wind\_min <dbl>, Wind\_max <dbl>, Temp\_sum <int>,  
## # Temp\_mean <dbl>, Temp\_min <int>, Temp\_max <int>

#tidy data  
df\_stats\_long <- df\_stats %>%   
 pivot\_longer(cols = -Month,  
 names\_to = c("Variable", "Statistics"),  
 names\_sep = "\_",  
 values\_to = "Value") %>%  
 pivot\_wider(names\_from = "Statistics",   
 values\_from = "Value")  
  
df\_stats\_long

## # A tibble: 20 × 6  
## Month Variable sum mean min max  
## <int> <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 5 Ozone 614 23.6 1 115   
## 2 5 Solar.R 4895 181. 8 334   
## 3 5 Wind 360. 11.6 5.7 20.1  
## 4 5 Temp 2032 65.5 56 81   
## 5 6 Ozone 265 29.4 12 71   
## 6 6 Solar.R 5705 190. 31 332   
## 7 6 Wind 308 10.3 1.7 20.7  
## 8 6 Temp 2373 79.1 65 93   
## 9 7 Ozone 1537 59.1 7 135   
## 10 7 Solar.R 6711 216. 7 314   
## 11 7 Wind 277. 8.94 4.1 14.9  
## 12 7 Temp 2601 83.9 73 92   
## 13 8 Ozone 1559 60.0 9 168   
## 14 8 Solar.R 4812 172. 24 273   
## 15 8 Wind 273. 8.79 2.3 15.5  
## 16 8 Temp 2603 84.0 72 97   
## 17 9 Ozone 912 31.4 7 96   
## 18 9 Solar.R 5023 167. 14 259   
## 19 9 Wind 305. 10.2 2.8 16.6  
## 20 9 Temp 2307 76.9 63 93

* There are only five months in the data while there are 31 days. For now, let’s change the month data type from a number to a factor.

df <- df %>%   
 mutate(Month = factor(Month, levels = 5:9,   
 labels = c("May", "Jun", "Jul", "Aug", "Sep")))   
head(df)

## # A tibble: 6 × 6  
## Ozone Solar.R Wind Temp Month Day  
## <int> <int> <dbl> <int> <fct> <int>  
## 1 41 190 7.4 67 May 1  
## 2 36 118 8 72 May 2  
## 3 12 149 12.6 74 May 3  
## 4 18 313 11.5 62 May 4  
## 5 NA NA 14.3 56 May 5  
## 6 28 NA 14.9 66 May 6

tail(df)

## # A tibble: 6 × 6  
## Ozone Solar.R Wind Temp Month Day  
## <int> <int> <dbl> <int> <fct> <int>  
## 1 14 20 16.6 63 Sep 25  
## 2 30 193 6.9 70 Sep 26  
## 3 NA 145 13.2 77 Sep 27  
## 4 14 191 14.3 75 Sep 28  
## 5 18 131 8 76 Sep 29  
## 6 20 223 11.5 68 Sep 30

is.factor(df$Month)

## [1] TRUE

* Write a code that reveals how many variables and observations are in the data set.

#show variables and observations dimensions:  
dim(df)

## [1] 153 6

print(glue::glue("There are {ncol(df)} variables and {nrow(df)} observations in the airquality dataset."))

## There are 6 variables and 153 observations in the airquality dataset.

## P1.4. Simple Descriptive statistics

* Also, write code that gives you some basic descriptive statistics. You will notice that two variables have missing values.

summary(df) #Ozone and Solar.R have missing values.

## Ozone Solar.R Wind Temp Month   
## Min. : 1.00 Min. : 7.0 Min. : 1.700 Min. :56.00 May:31   
## 1st Qu.: 18.00 1st Qu.:115.8 1st Qu.: 7.400 1st Qu.:72.00 Jun:30   
## Median : 31.50 Median :205.0 Median : 9.700 Median :79.00 Jul:31   
## Mean : 42.13 Mean :185.9 Mean : 9.958 Mean :77.88 Aug:31   
## 3rd Qu.: 63.25 3rd Qu.:258.8 3rd Qu.:11.500 3rd Qu.:85.00 Sep:30   
## Max. :168.00 Max. :334.0 Max. :20.700 Max. :97.00   
## NA's :37 NA's :7   
## Day   
## Min. : 1.0   
## 1st Qu.: 8.0   
## Median :16.0   
## Mean :15.8   
## 3rd Qu.:23.0   
## Max. :31.0   
##

* Use the glimpse() function from dplyr package and skim() function from skimr package to understand the data. Skim function shows mean, sd, percentiles, and histogram.

glimpse(df)

## Rows: 153  
## Columns: 6  
## $ Ozone <int> 41, 36, 12, 18, NA, 28, 23, 19, 8, NA, 7, 16, 11, 14, 18, 14, …  
## $ Solar.R <int> 190, 118, 149, 313, NA, NA, 299, 99, 19, 194, NA, 256, 290, 27…  
## $ Wind <dbl> 7.4, 8.0, 12.6, 11.5, 14.3, 14.9, 8.6, 13.8, 20.1, 8.6, 6.9, 9…  
## $ Temp <int> 67, 72, 74, 62, 56, 66, 65, 59, 61, 69, 74, 69, 66, 68, 58, 64…  
## $ Month <fct> May, May, May, May, May, May, May, May, May, May, May, May, Ma…  
## $ Day <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,…

skimr::skim(df)

Data summary

|  |  |
| --- | --- |
| Name | df |
| Number of rows | 153 |
| Number of columns | 6 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| factor | 1 |
| numeric | 5 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

**Variable type: factor**

| skim\_variable | n\_missing | complete\_rate | ordered | n\_unique | top\_counts |
| --- | --- | --- | --- | --- | --- |
| Month | 0 | 1 | FALSE | 5 | May: 31, Jul: 31, Aug: 31, Jun: 30 |

**Variable type: numeric**

| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ozone | 37 | 0.76 | 42.13 | 32.99 | 1.0 | 18.00 | 31.5 | 63.25 | 168.0 | ▇▃▂▁▁ |
| Solar.R | 7 | 0.95 | 185.93 | 90.06 | 7.0 | 115.75 | 205.0 | 258.75 | 334.0 | ▅▃▅▇▅ |
| Wind | 0 | 1.00 | 9.96 | 3.52 | 1.7 | 7.40 | 9.7 | 11.50 | 20.7 | ▂▇▇▃▁ |
| Temp | 0 | 1.00 | 77.88 | 9.47 | 56.0 | 72.00 | 79.0 | 85.00 | 97.0 | ▂▃▇▇▃ |
| Day | 0 | 1.00 | 15.80 | 8.86 | 1.0 | 8.00 | 16.0 | 23.00 | 31.0 | ▇▇▇▇▆ |

* Looking at the histogram, which variable is most skewed?  
  The histograms shows that Ozone is the most skewed variable.  
  **Hint**. you may need to use skimr::skim() to make the skim function work.

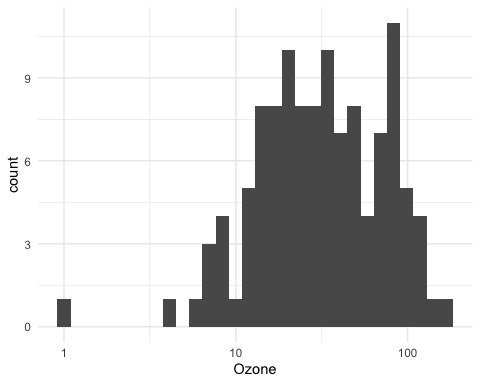
# Problem 2: Visualize numerical variables

## P2.1. Histograms

* Visualize numerical data with a histogram. Normality assumption is important when running a regression. If the data is severely skewed, change to a log-based scale to depict the variable on the chart.

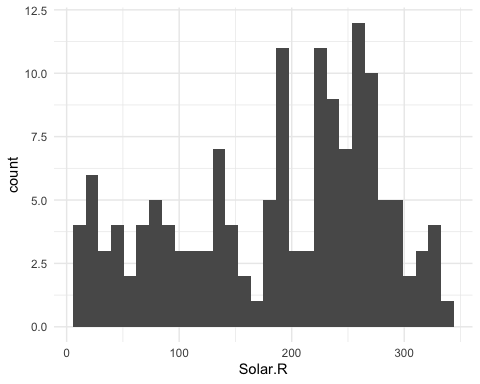
#Ozone  
df %>%   
 filter(!is.na(Ozone)) %>%   
 ggplot(aes(Ozone)) +  
 geom\_histogram() +  
 scale\_x\_log10()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



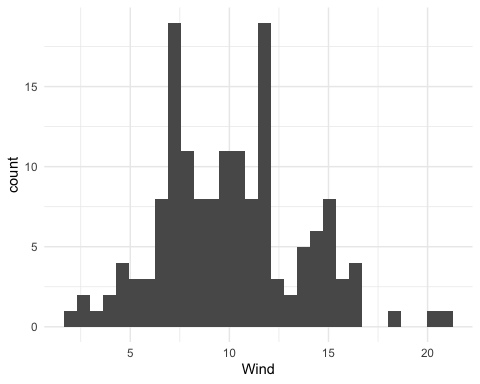
#Solar.R  
df %>%   
 filter(!is.na(Solar.R)) %>%   
 ggplot(aes(Solar.R)) +  
 geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



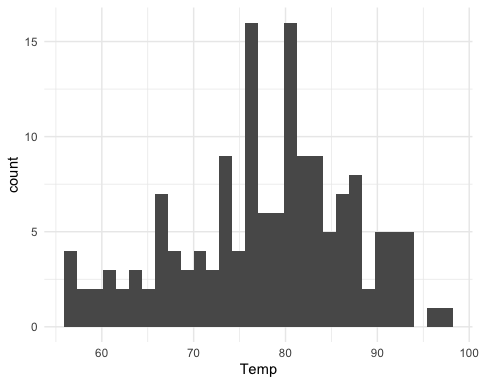
#Wind  
df %>%   
 filter(!is.na(Wind)) %>%   
 ggplot(aes(Wind)) +  
 geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#Temp  
df %>%   
 filter(!is.na(Temp)) %>%   
 ggplot(aes(Temp)) +  
 geom\_histogram()

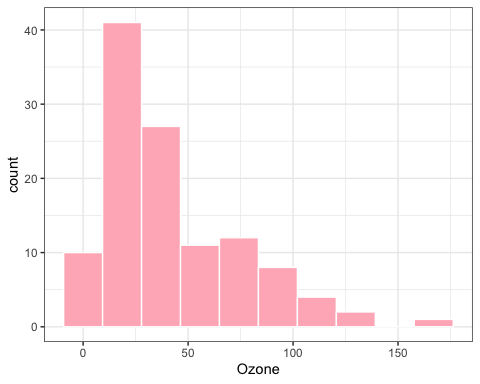
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#also can do it this way:  
#create a function to plot histogram  
histogram <- function(data, var) {  
 library(ggplot2); library(magrittr); library(rlang)  
 theme\_set(theme\_bw())  
   
 plot <- ggplot(data, aes(x = .data[[var]])) +  
 geom\_histogram(color = "white", fill = "lightpink",  
 binwidth = function(x) 2 \* IQR(x)/  
 length(x)^(1/3))  
 return(plot)  
}  
  
# extract only numerical valuables  
airquality\_num <- df %>% select(Ozone:Temp)  
  
# plot histograms  
histograms <-   
map2(.x = list(airquality\_num),   
 .y = names(airquality\_num),   
 .f = histogram)  
  
histograms

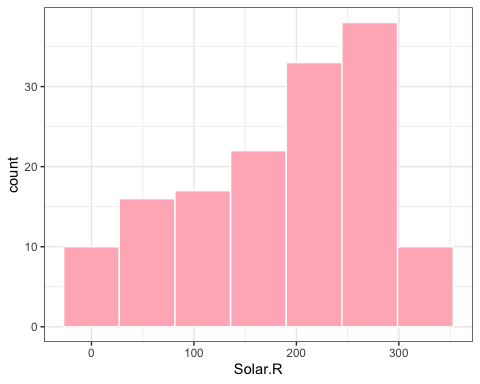
## [[1]]

## Warning: Removed 37 rows containing non-finite outside the scale range  
## (`stat\_bin()`).

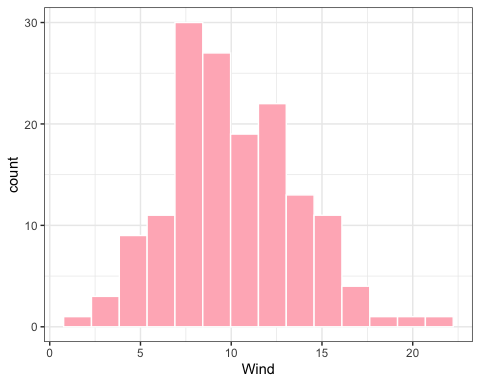


##   
## [[2]]

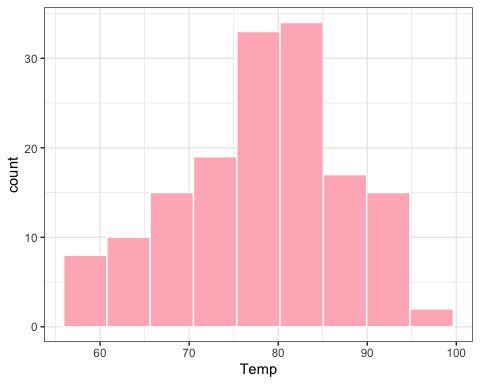
## Warning: Removed 7 rows containing non-finite outside the scale range  
## (`stat\_bin()`).



##   
## [[3]]



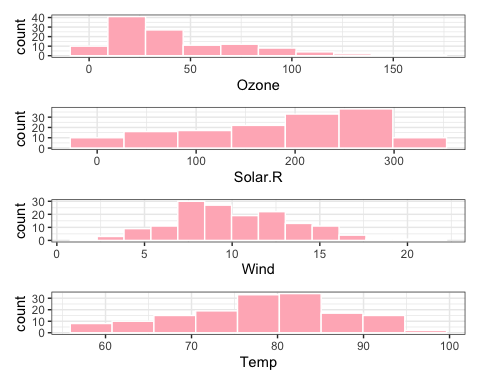
##   
## [[4]]



#combine all plots together  
wrap\_plots(histograms) +  
 plot\_layout(nrow = length(histograms))

## Warning: Removed 37 rows containing non-finite outside the scale range  
## (`stat\_bin()`).

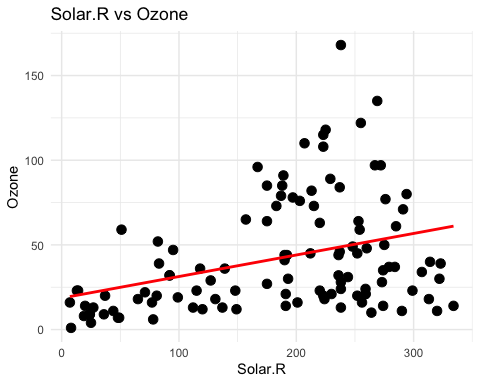
## Warning: Removed 7 rows containing non-finite outside the scale range  
## (`stat\_bin()`).



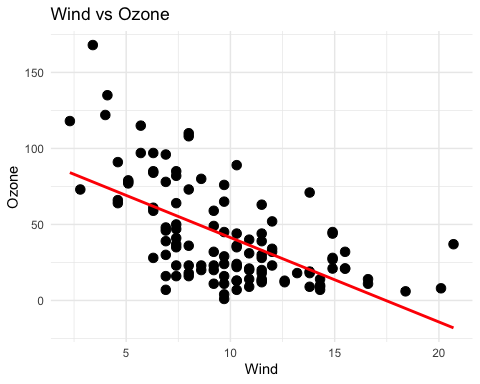
## P2.2. Ozone by Continuous variables

* Now, let’s examine the relationship between each continuous variable and Ozone at one pair at a time. Which plot should you use and why? Also, add a regression line on the plot.  
  Choice of Plot: Scatter plots are chosen because they visually represent the relationship between two continuous variables effectively. They allow you to see patterns, correlations, and potential outliers.

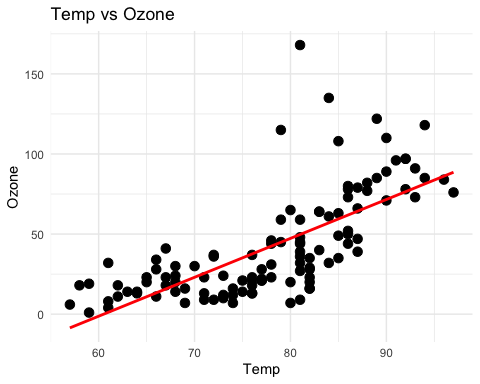
#create a function to draw Ozone with different continuous variables  
plot\_Ozone\_continuous <- function(data, var) {  
 library(ggplot2)  
 library(stringr)  
   
 plot <- df %>%   
 filter(!is.na(Ozone), !is.na(.data[[var]])) %>%  
 ggplot(aes(x = .data[[var]], y = Ozone)) +  
 geom\_point(size = 3) +  
 geom\_smooth(formula = 'y ~ x', method = "lm",   
 se = FALSE, color = "red") +  
 labs(title = paste(var, "vs Ozone"),  
 x = var,  
 y = "Ozone") +  
 theme\_minimal()  
 return(plot)  
}  
  
#Solar.R vs. Ozone  
plot\_Ozone\_continuous(df, "Solar.R")



#Wind vs. Ozone  
plot\_Ozone\_continuous(df, "Wind")



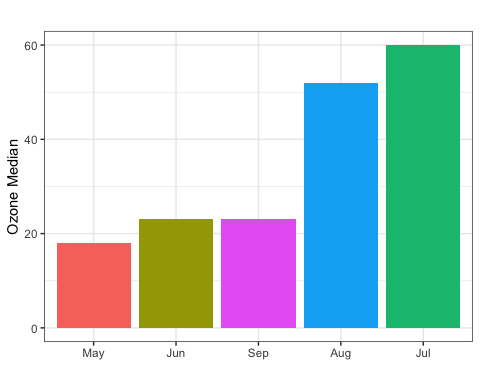
#Temp vs. Ozone  
plot\_Ozone\_continuous(df, "Temp")



## P2.3. Ozone by Month (Monthly ozone amount)

* This time, draw a chart showing the impact of the categorical independent variables on the ozone amount.

df %>%   
 group\_by(Month) %>%   
 summarise(Ozone\_Median = median(Ozone, na.rm = TRUE)) %>%   
 ggplot(aes(x=fct\_reorder(Month, Ozone\_Median), y = Ozone\_Median, fill = Month))+  
 geom\_col(show.legend = FALSE)+  
 labs(title = "",  
 x = "",  
 y = "Ozone Median")



# Problem 3: The moderating role of the Month?

## P3.1. Using group\_by() and summarise(), find out how many cases exist for each month.

df %>%   
 group\_by(Month) %>%   
 summarise(case = n())

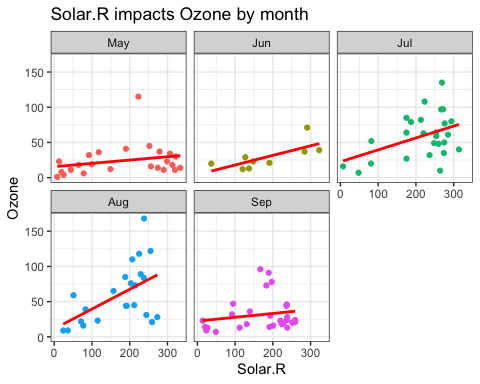
## # A tibble: 5 × 2  
## Month case  
## <fct> <int>  
## 1 May 31  
## 2 Jun 30  
## 3 Jul 31  
## 4 Aug 31  
## 5 Sep 30

## P3.2. Draw a series of charts showing the impact of Solar.R on Ozone cut by Month.

#create a function to draw different variables' impact on Ozone cut by month   
plot\_Ozone\_continuous\_monthly <- function(data, var) {  
 library(ggplot2)  
 library(stringr)  
 theme\_set(theme\_bw())  
   
 plot <- ggplot(df, aes(x = .data[[var]], y = Ozone, color = Month)) +  
 geom\_point(show.legend = FALSE) +  
 geom\_smooth(color = "red", formula = 'y ~ x',   
 method = "lm", se = FALSE) +  
 facet\_wrap(~Month) +  
 labs(title = paste(var, "impacts Ozone by month"),  
 x = var,  
 y = "Ozone")  
 return(plot)  
}  
  
#apply function with Solar.R  
plot\_Ozone\_continuous\_monthly(df, "Solar.R")

## Warning: Removed 42 rows containing non-finite outside the scale range  
## (`stat\_smooth()`).

## Warning: Removed 42 rows containing missing values or values outside the scale range  
## (`geom\_point()`).

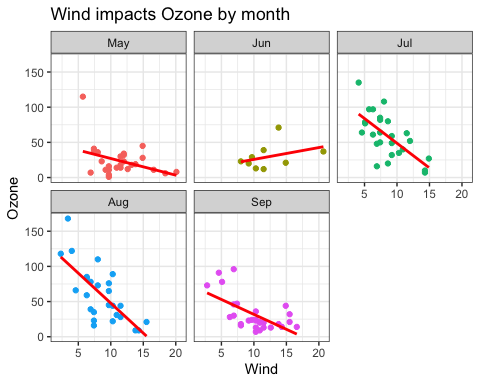


## P3.3. Draw a series of charts showing the impact of Wind on Ozone cut by Month.

plot\_Ozone\_continuous\_monthly(df, "Wind")

## Warning: Removed 37 rows containing non-finite outside the scale range  
## (`stat\_smooth()`).

## Warning: Removed 37 rows containing missing values or values outside the scale range  
## (`geom\_point()`).

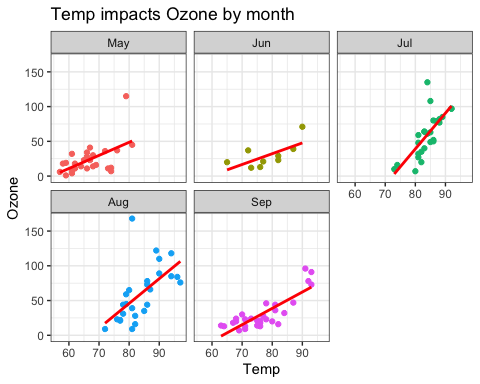


## P3.4. Draw a series of charts showing the impact of Temp on Ozone cut by Month.

plot\_Ozone\_continuous\_monthly(df, "Temp")

## Warning: Removed 37 rows containing non-finite outside the scale range  
## (`stat\_smooth()`).

## Warning: Removed 37 rows containing missing values or values outside the scale range  
## (`geom\_point()`).



## P3.5 Based on the descriptive statistics above, can you conclude that the impact of Solar.R and Wind on Ozone changes by Month?

* Solar.R impact: Each month’s plot shows a trend where Ozone increases with Solar.R, indicating that more solar radiation contributes to higher ozone levels.
* Wind impact: Wind appears to have a more complex and less direct impact on ozone levels. High wind speeds might reduce Ozone concentrations by dispersing pollutants, but this effect can vary depending on other atmospheric conditions.

# Problem 4: Correlations

## P4.1. The data visualization so far should have helped you form associations among the variables. Now, let’s try to quantify the associations by running correlations among all numeric variables.

df %>% select(Ozone:Temp) %>% cor(use = "pairwise.complete.obs")

## Ozone Solar.R Wind Temp  
## Ozone 1.0000000 0.34834169 -0.60154653 0.6983603  
## Solar.R 0.3483417 1.00000000 -0.05679167 0.2758403  
## Wind -0.6015465 -0.05679167 1.00000000 -0.4579879  
## Temp 0.6983603 0.27584027 -0.45798788 1.0000000

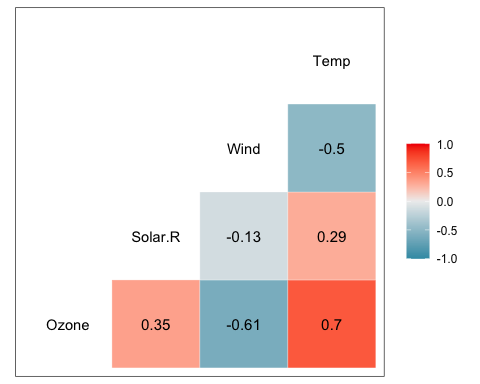
#revision  
df %>% select(Ozone:Temp) %>% cor(use = "complete.obs")

## Ozone Solar.R Wind Temp  
## Ozone 1.0000000 0.3483417 -0.6124966 0.6985414  
## Solar.R 0.3483417 1.0000000 -0.1271835 0.2940876  
## Wind -0.6124966 -0.1271835 1.0000000 -0.4971897  
## Temp 0.6985414 0.2940876 -0.4971897 1.0000000

#alternative ggcor:  
#install.packages("GGally")  
library(GGally)

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

library(tidyverse)  
df %>%   
 filter(Ozone!=is.na(Ozone), Solar.R != is.na(Solar.R)) %>%   
 select(Ozone:Temp) %>%   
 ggcorr(label = TRUE, label\_round = 2)



## P4.2. Which variables are correlated highly with Ozone? Describe the nature of the association – whether the association is positive or negative, strongly or weakly correlated.

* Strongly Positive Correlation: Ozone has strongly positive correlation with Temperature (Temp), as temperature increases, Ozone tends to increase as well.
* Strongly Negative Correlation: Ozone has strongly negative correlation with Wind speed, indicating that higher wind speeds are associated with lower Ozone levels.
* Moderate Positive Correlation: Ozone seems to have moderate positive correlation with Solar Radiation (Solar.R), suggesting that higher solar radiation levels are associated with higher Ozone levels.
* Weak Correlation: There is a very weak correlation between the Day of the month and Ozone, indicating that the day of the month does not significantly influence Ozone in this dataset.

# Problem 5: Examine Missing values

When you ran simple descriptive statistics previously, you would have noticed that two variables had missing values, which might have given you some trouble while you visualized the data.

Write the codes that tell you (1)where the missing values are located,

df %>% is.na()

## Ozone Solar.R Wind Temp Month Day  
## [1,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [2,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [3,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [4,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [5,] TRUE TRUE FALSE FALSE FALSE FALSE  
## [6,] FALSE TRUE FALSE FALSE FALSE FALSE  
## [7,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [8,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [9,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [10,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [11,] FALSE TRUE FALSE FALSE FALSE FALSE  
## [12,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [13,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [14,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [15,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [16,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [17,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [18,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [19,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [20,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [21,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [22,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [23,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [24,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [25,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [26,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [27,] TRUE TRUE FALSE FALSE FALSE FALSE  
## [28,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [29,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [30,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [31,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [32,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [33,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [34,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [35,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [36,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [37,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [38,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [39,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [40,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [41,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [42,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [43,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [44,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [45,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [46,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [47,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [48,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [49,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [50,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [51,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [52,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [53,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [54,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [55,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [56,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [57,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [58,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [59,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [60,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [61,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [62,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [63,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [64,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [65,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [66,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [67,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [68,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [69,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [70,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [71,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [72,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [73,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [74,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [75,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [76,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [77,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [78,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [79,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [80,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [81,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [82,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [83,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [84,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [85,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [86,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [87,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [88,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [89,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [90,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [91,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [92,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [93,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [94,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [95,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [96,] FALSE TRUE FALSE FALSE FALSE FALSE  
## [97,] FALSE TRUE FALSE FALSE FALSE FALSE  
## [98,] FALSE TRUE FALSE FALSE FALSE FALSE  
## [99,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [100,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [101,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [102,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [103,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [104,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [105,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [106,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [107,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [108,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [109,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [110,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [111,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [112,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [113,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [114,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [115,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [116,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [117,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [118,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [119,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [120,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [121,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [122,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [123,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [124,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [125,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [126,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [127,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [128,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [129,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [130,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [131,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [132,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [133,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [134,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [135,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [136,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [137,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [138,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [139,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [140,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [141,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [142,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [143,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [144,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [145,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [146,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [147,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [148,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [149,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [150,] TRUE FALSE FALSE FALSE FALSE FALSE  
## [151,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [152,] FALSE FALSE FALSE FALSE FALSE FALSE  
## [153,] FALSE FALSE FALSE FALSE FALSE FALSE

# which()

1. the number of missing values in the dataset (df),

df %>% summarise\_all(~sum(is.na(.)))

## # A tibble: 1 × 6  
## Ozone Solar.R Wind Temp Month Day  
## <int> <int> <int> <int> <int> <int>  
## 1 37 7 0 0 0 0

# 2  
df %>% is.na() %>% sum()

## [1] 44

1. the number of missing values in the Solar.R column, and

df %>% select(Solar.R) %>%   
 filter(is.na(Solar.R)) %>% count()

## # A tibble: 1 × 1  
## n  
## <int>  
## 1 7

#or   
df %>% summarise\_all(~sum(is.na(.))) %>% select(Solar.R)

## # A tibble: 1 × 1  
## Solar.R  
## <int>  
## 1 7

#or  
sum(is.na(df['Solar.R']))

## [1] 7

1. all the rows that include at least one missing value.

df\_na <- df %>% filter(if\_any(everything(), is.na))  
  
df\_na

## # A tibble: 42 × 6  
## Ozone Solar.R Wind Temp Month Day  
## <int> <int> <dbl> <int> <fct> <int>  
## 1 NA NA 14.3 56 May 5  
## 2 28 NA 14.9 66 May 6  
## 3 NA 194 8.6 69 May 10  
## 4 7 NA 6.9 74 May 11  
## 5 NA 66 16.6 57 May 25  
## 6 NA 266 14.9 58 May 26  
## 7 NA NA 8 57 May 27  
## 8 NA 286 8.6 78 Jun 1  
## 9 NA 287 9.7 74 Jun 2  
## 10 NA 242 16.1 67 Jun 3  
## # ℹ 32 more rows

1. Lastly, write the code that returns the number of rows with at least one missing value. Hint: some rows have more than one missing value.

nrow(df\_na)

## [1] 42

sum(is.na(df))

## [1] 44

which(is.na(df))

## [1] 5 10 25 26 27 32 33 34 35 36 37 39 42 43 45 46 52 53 54  
## [20] 55 56 57 58 59 60 61 65 72 75 83 84 102 103 107 115 119 150 158  
## [39] 159 164 180 249 250 251

# Problem 6: Missing value imputation

1. Replace all the missing values in the Solar.R column with the median of the values in the column.

Solar\_median<- df %>%   
 filter(!is.na(Solar.R)) %>%   
 summarise(Solar\_median = median(Solar.R))  
print(Solar\_median)

## # A tibble: 1 × 1  
## Solar\_median  
## <dbl>  
## 1 205

df['Solar.R'][is.na(df['Solar.R'])] <- as.integer(Solar\_median)  
df

## # A tibble: 153 × 6  
## Ozone Solar.R Wind Temp Month Day  
## <int> <int> <dbl> <int> <fct> <int>  
## 1 41 190 7.4 67 May 1  
## 2 36 118 8 72 May 2  
## 3 12 149 12.6 74 May 3  
## 4 18 313 11.5 62 May 4  
## 5 NA 205 14.3 56 May 5  
## 6 28 205 14.9 66 May 6  
## 7 23 299 8.6 65 May 7  
## 8 19 99 13.8 59 May 8  
## 9 8 19 20.1 61 May 9  
## 10 NA 194 8.6 69 May 10  
## # ℹ 143 more rows

1. Replace all the missing values in the Ozone column with the median of the values in the column.

Ozone\_median<- df %>%   
 filter(!is.na(Ozone)) %>%   
 summarise(Ozone\_median = median(Ozone))  
print(Ozone\_median)

## # A tibble: 1 × 1  
## Ozone\_median  
## <dbl>  
## 1 31.5

df['Ozone'][is.na(df['Ozone'])] <- as.integer(Ozone\_median)  
df

## # A tibble: 153 × 6  
## Ozone Solar.R Wind Temp Month Day  
## <int> <int> <dbl> <int> <fct> <int>  
## 1 41 190 7.4 67 May 1  
## 2 36 118 8 72 May 2  
## 3 12 149 12.6 74 May 3  
## 4 18 313 11.5 62 May 4  
## 5 31 205 14.3 56 May 5  
## 6 28 205 14.9 66 May 6  
## 7 23 299 8.6 65 May 7  
## 8 19 99 13.8 59 May 8  
## 9 8 19 20.1 61 May 9  
## 10 31 194 8.6 69 May 10  
## # ℹ 143 more rows

1. Take a look at the descriptive statistics again.

summary(df)

## Ozone Solar.R Wind Temp Month   
## Min. : 1.00 Min. : 7.0 Min. : 1.700 Min. :56.00 May:31   
## 1st Qu.: 21.00 1st Qu.:120.0 1st Qu.: 7.400 1st Qu.:72.00 Jun:30   
## Median : 31.00 Median :205.0 Median : 9.700 Median :79.00 Jul:31   
## Mean : 39.44 Mean :186.8 Mean : 9.958 Mean :77.88 Aug:31   
## 3rd Qu.: 46.00 3rd Qu.:256.0 3rd Qu.:11.500 3rd Qu.:85.00 Sep:30   
## Max. :168.00 Max. :334.0 Max. :20.700 Max. :97.00   
## Day   
## Min. : 1.0   
## 1st Qu.: 8.0   
## Median :16.0   
## Mean :15.8   
## 3rd Qu.:23.0   
## Max. :31.0

1. Also, get the mean and standard deviation of all continuous variables.

df %>%   
 summarize(across(c(Ozone:Temp),   
 list(Mean = mean, SD = sd),  
 .names = "{.col}\_{.fn}")  
 )

## # A tibble: 1 × 8  
## Ozone\_Mean Ozone\_SD Solar.R\_Mean Solar.R\_SD Wind\_Mean Wind\_SD Temp\_Mean  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 39.4 29.1 187. 88.1 9.96 3.52 77.9  
## # ℹ 1 more variable: Temp\_SD <dbl>

# Problem 7: Correlations after missing value imputation

## P7.1. Correlation with raw Ozone

* Run the correlation analysis you did in Problem 4 again.

df %>% select(Ozone:Temp) %>% cor(use = "pairwise.complete.obs")

## Ozone Solar.R Wind Temp  
## Ozone 1.0000000 0.29495314 -0.53162566 0.6001034  
## Solar.R 0.2949531 1.00000000 -0.05789854 0.2571538  
## Wind -0.5316257 -0.05789854 1.00000000 -0.4579879  
## Temp 0.6001034 0.25715377 -0.45798788 1.0000000

* Compare the results of the correlations before and after missing value imputations. What can you tell about the strength of the association between Ozone and the other three variables?  
  Filling in the missing values in the dataset slightly affected the relationships involving Ozone. The connection between Ozone and Solar.R, and between Ozone and Temp, became a bit weaker. However, the strong negative relationship between Ozone and Wind stayed the same.

## P7.2. Correlations with Logged Ozone

* This time, create a new variable by taking the log of Ozone – log(Ozone) – as Ozone is severely skewed. Write the code to repeat (1) with the log-transformed form of Ozone.

df %>%   
 mutate(Ozone\_logged = log(Ozone)) %>%   
 select(-Month, -Day) %>%   
 cor(use = "pairwise.complete.obs")

## Ozone Solar.R Wind Temp Ozone\_logged  
## Ozone 1.0000000 0.29495314 -0.53162566 0.6001034 0.8794206  
## Solar.R 0.2949531 1.00000000 -0.05789854 0.2571538 0.3937871  
## Wind -0.5316257 -0.05789854 1.00000000 -0.4579879 -0.4747741  
## Temp 0.6001034 0.25715377 -0.45798788 1.0000000 0.6448765  
## Ozone\_logged 0.8794206 0.39378709 -0.47477406 0.6448765 1.0000000

* What can you tell about the strength of association between the transformed Ozone and the other three variables? Do you see a pattern of relationships that differ in the two sets of correlations? Why do you think there were discrepancies?

The log transformation of Ozone made its correlation with Solar.R and Temp stronger and clearer by normalizing the data and reducing the effect of outliers. It also slightly weakened the negative correlation with Wind. Raw Ozone correlations were less reliable due to data skewness, which hid stronger positive relationships with Solar.R and Temp. Imputing missing values had a minor effect, slightly lowering the correlations with Solar.R and Temp but keeping the strong negative correlation with Wind stable. This highlights the importance of handling missing data and transforming skewed variables for better correlation results.

# Problem 8: Adding a new variable to the data sets and adjusting data types

1. Since the logged Ozone variable seems to be helpful, let’s add the variable.
2. Look for the unique value of the Month variable. Since Month is categorical data, change it to a factor data type in preparation for visualization
3. Also, change the data type to tibble permanently.
4. Confirm that the changes you made are successful by printing out the data sets.

#add log(Ozone) to df  
df<- df %>%   
 mutate(Ozone\_logged = log(Ozone))  
df

## # A tibble: 153 × 7  
## Ozone Solar.R Wind Temp Month Day Ozone\_logged  
## <int> <int> <dbl> <int> <fct> <int> <dbl>  
## 1 41 190 7.4 67 May 1 3.71  
## 2 36 118 8 72 May 2 3.58  
## 3 12 149 12.6 74 May 3 2.48  
## 4 18 313 11.5 62 May 4 2.89  
## 5 31 205 14.3 56 May 5 3.43  
## 6 28 205 14.9 66 May 6 3.33  
## 7 23 299 8.6 65 May 7 3.14  
## 8 19 99 13.8 59 May 8 2.94  
## 9 8 19 20.1 61 May 9 2.08  
## 10 31 194 8.6 69 May 10 3.43  
## # ℹ 143 more rows

#convert Month to factor  
df$Month <- factor(df$Month)  
class(df$Month)

## [1] "factor"

#convert df to tibble   
df <- as\_tibble(df)  
class(df)

## [1] "tbl\_df" "tbl" "data.frame"

df

## # A tibble: 153 × 7  
## Ozone Solar.R Wind Temp Month Day Ozone\_logged  
## <int> <int> <dbl> <int> <fct> <int> <dbl>  
## 1 41 190 7.4 67 May 1 3.71  
## 2 36 118 8 72 May 2 3.58  
## 3 12 149 12.6 74 May 3 2.48  
## 4 18 313 11.5 62 May 4 2.89  
## 5 31 205 14.3 56 May 5 3.43  
## 6 28 205 14.9 66 May 6 3.33  
## 7 23 299 8.6 65 May 7 3.14  
## 8 19 99 13.8 59 May 8 2.94  
## 9 8 19 20.1 61 May 9 2.08  
## 10 31 194 8.6 69 May 10 3.43  
## # ℹ 143 more rows

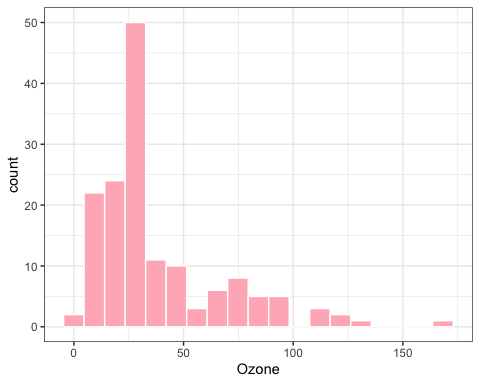
# Problem 9: Data Visualization using imputed data

Let’s repeat the visualization you did in Problem 2 and Problem 3, using the imputed data and Log-transformed Ozone variable. Specifically, do the following data visualizations.

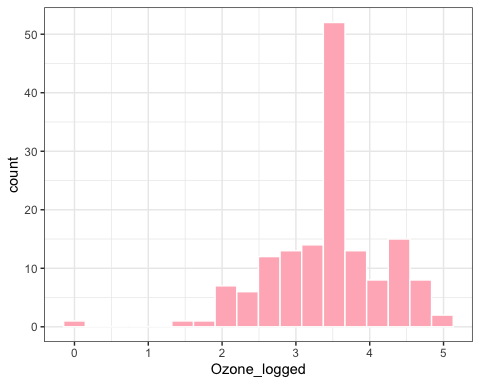
## P9.1. Histogram of Ozone, Ozone\_logged, and Solar.R

# plot histograms using function created in P2  
df\_p9 <- df %>% select(Ozone, Ozone\_logged, Solar.R)  
  
histograms\_p9 <-   
map2(.x = list(df\_p9),   
 .y = names(df\_p9),   
 .f = histogram)  
  
histograms\_p9

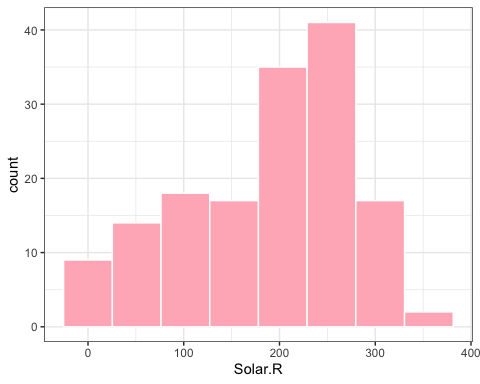
## [[1]]



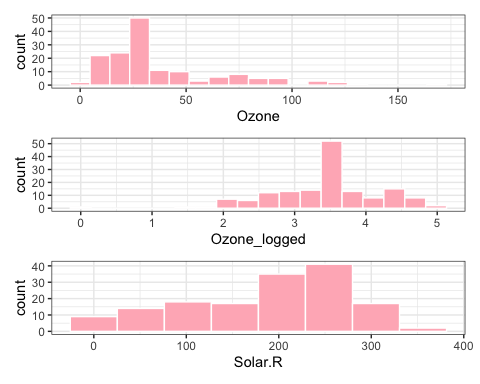
##   
## [[2]]



##   
## [[3]]

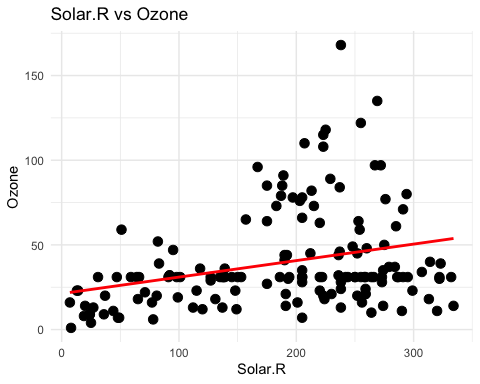


#combine all plots together  
wrap\_plots(histograms\_p9) +  
 plot\_layout(nrow = length(histograms\_p9))

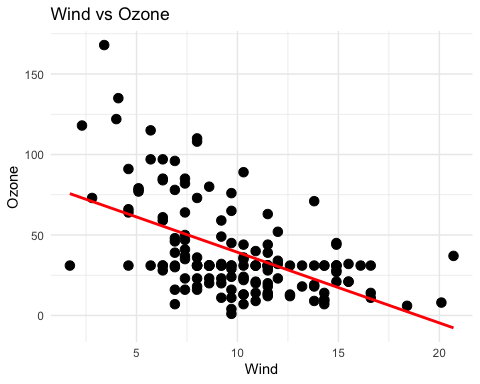


## P9.2. Ozone by Continuous variables

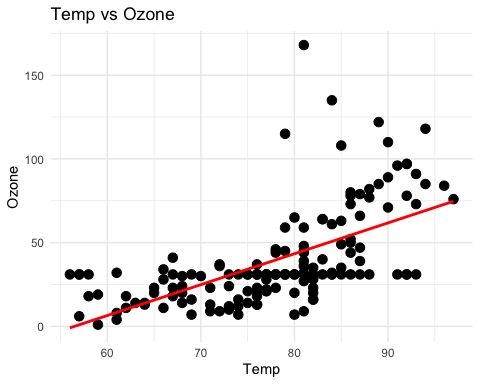
#using function created in p2.2  
#Solar.R vs. Ozone  
plot\_Ozone\_continuous(df, "Solar.R")



#Wind vs. Ozone  
plot\_Ozone\_continuous(df, "Wind")

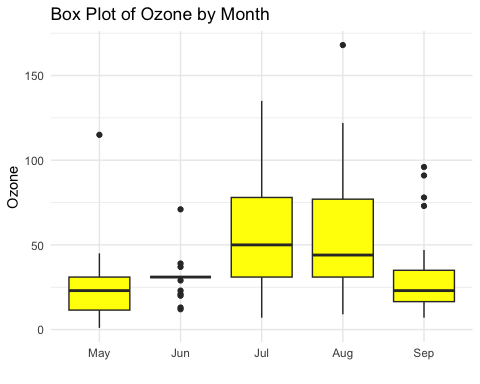


#Temp vs. Ozone  
plot\_Ozone\_continuous(df, "Temp")



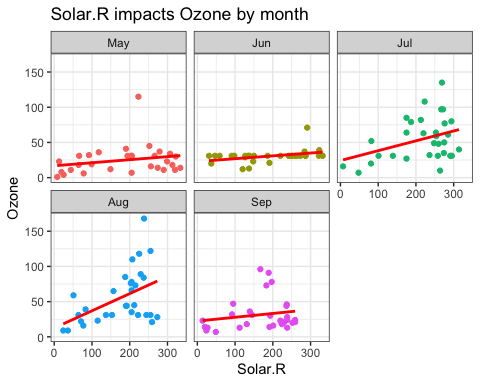
## P9.3. Ozone by Month

df %>%  
 ggplot(aes(x = Month, y = Ozone)) +  
 geom\_boxplot(fill = "yellow") +  
 labs(title = "Box Plot of Ozone by Month",  
 x = "",  
 y = "Ozone") +  
 theme\_minimal()

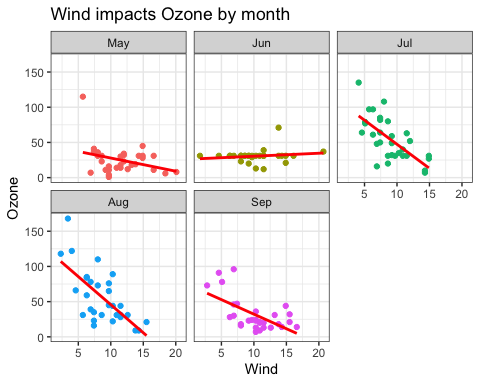


## P9.4. Moderating Role of Month in the impact of Continuous variables on Ozone

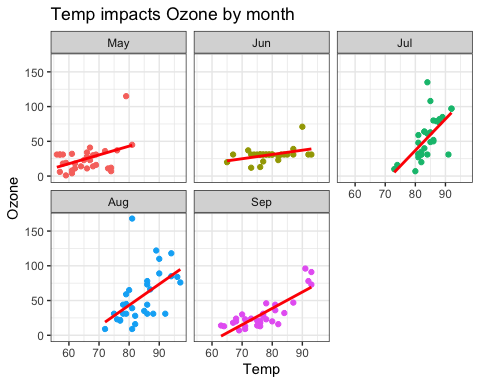
#using function created in 3.2  
#Solar.R impacts on Ozone  
plot\_Ozone\_continuous\_monthly(df, "Solar.R")



#Wind impacts on Ozone  
plot\_Ozone\_continuous\_monthly(df, "Wind")



#Temp impacts on Ozone  
plot\_Ozone\_continuous\_monthly(df, "Temp")



## P9.5: Do you find the same relationships as before?

* Yes, most of the impacts show on the plots for each month remain the same as before, except for June. The denser alignment of points and the smoother fit line for June indicate that imputation had a significant effect on the data distribution for that month. This suggests that the missing values in June were numerous and replacing them with summary statistics of median altered the overall distribution of the data.

# Problem 10: Using categorical Ozone amount

## P10.1: categorical Ozone

Create a new column called “Ozone\_cat.” If the Ozone of the imputed dataset is less than or equal to the 25th quantile of the Ozone amount in the data, put “Low” in the new column, if it is greater than 25th quantile and less than the 75th quantile, put “Middle,” and if it is greater than 75th quantile, put “high” in the new column (use the pipe operator).

**Hint**: You may use quantile() to find 25th and 75 quantile. You may also use case\_when() from dplr.

quantile(df$Ozone)

## 0% 25% 50% 75% 100%   
## 1 21 31 46 168

df <- df %>%   
 mutate(Ozone\_cat = case\_when(Ozone <= 21 ~ "Low",  
 Ozone > 21 & Ozone <75 ~ "Middle",  
 Ozone > 75 ~ "High",  
 .default = as.character(Ozone)))  
df

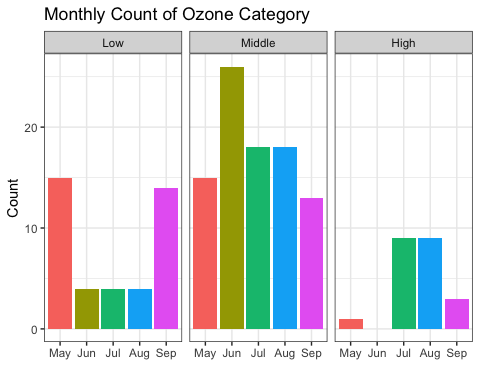
## # A tibble: 153 × 8  
## Ozone Solar.R Wind Temp Month Day Ozone\_logged Ozone\_cat  
## <int> <int> <dbl> <int> <fct> <int> <dbl> <chr>   
## 1 41 190 7.4 67 May 1 3.71 Middle   
## 2 36 118 8 72 May 2 3.58 Middle   
## 3 12 149 12.6 74 May 3 2.48 Low   
## 4 18 313 11.5 62 May 4 2.89 Low   
## 5 31 205 14.3 56 May 5 3.43 Middle   
## 6 28 205 14.9 66 May 6 3.33 Middle   
## 7 23 299 8.6 65 May 7 3.14 Middle   
## 8 19 99 13.8 59 May 8 2.94 Low   
## 9 8 19 20.1 61 May 9 2.08 Low   
## 10 31 194 8.6 69 May 10 3.43 Middle   
## # ℹ 143 more rows

## P10.2: Monthly Ozone Severity

Now that you have created Ozone\_cat, which is a factor, let’s draw a chart that shows monthly counts of each of the three levels of Ozone\_cat – Low, Middle, and High in that order. Make the chart as professional as it can be.

Hints: When you created the Ozone\_cat variable previously, you might have created the level differently than the low-middle-high order. If so, you can change the order of the level using a combination of mutate and fct\_relevel() and manually type the order you like: “c(”Low”, “Middle”, “High”)“. To generate the count of Ozone\_cat, you would like to use”group\_by()” and “count().”

df %>%  
 mutate(Ozone\_cat = fct\_relevel(Ozone\_cat,   
 c("Low", "Middle", "High")))%>%   
 group\_by(Month, Ozone\_cat) %>%   
 count() %>%   
 ggplot(aes(x = Month, y = n, fill = Month)) +  
 geom\_col(position = "dodge", show.legend = FALSE) +  
 facet\_wrap(~Ozone\_cat) +  
 labs(title = "Monthly Count of Ozone Category",  
 x = "",  
 y = "Count")



## P10.3: Insights from the chart

* What can you tell about the monthly Ozone severity?
  + The severity of ozone levels varies from month to month. May and September consistently have the lowest ozone levels, indicating cleaner air during these months. On the other hand, July and August consistently have the highest ozone levels, suggesting more pollution during the peak summer months. This matches Hypothesis 4, which suggests that ozone levels change with the seasons. Solar radiation and temperature have a notable impact on ozone levels, contributing to higher concentrations in July and August. In contrast, wind speed shows a negative relationship with ozone levels, meaning lower wind speeds may contribute to higher ozone concentrations, especially noticeable in July and August. These findings support Hypotheses 1, 2, and 3, which focus on the effects of sunlight, temperature, and wind on ozone concentrations throughout the year.