# DATASCI 306, Fall 2024, Homework 1

Max Han, maxhan

Please `knit` your solution to a PDF document and then upload your PDF file to Gradescope.

## Taking Maizey's help (0.5 pt)

For all homework, you are allowed to take the help of Maizey.

However, invest at least two hours in thoroughly exploring all potential solutions to your problem before seeking assistance from Maizey (or any other GenAI or other friends). Only if you've exhausted all of your own ideas and remain stuck should you ask Maizey (or friends) for help. This approach ensures you're maximizing your learning and not relying on GenAI as a crutch.

**Understand the limitations of GenAI**

- GenAI is not a magic bullet: It can provide suggestions and assist with code generation, but it's not a replacement for understanding the fundamentals of coding.
- Potential for errors: GenAI can generate code with errors, bugs, or inefficient solutions.
- Always double-check its output. Context matters: Provide clear and specific instructions to ensure the GenAI understands your requirement.

**Enhance your coding skills**

- Focus on understanding the code generated: Study the generated code to learn how it works, identify potential issues, and improve your understanding of the concepts.
- Experiment with different inputs and parameters: Explore different ways of phrasing prompts or providing input to observe how the generated code changes.
- Combine GenAI with other tools: Use GenAI alongside documentation, and online resources to enhance your coding process.

**Be ethical and responsible**

- Acknowledge the assistance: Cite the GenAI tool and its contribution to your code.
- Avoid plagiarism: Ensure the code generated by GenAI is not submitted as your own original work. Adapt and improve it to make it your own.

**Remember: Using GenAI for coding assignments can be a valuable tool for learning and improving your skills. However, always prioritize understanding the fundamentals, ethical considerations, and responsible usage. Midterm exams require independent work with no access to any platform. Relying on Maizey exclusively to solve problems will hinder your understanding of the material.**

Please indicate that you will follow the above given advice by replacing NULL with your name in the code block below:

```
name = "Max Han"
print(paste('I acknowledge and will follow the advice given. -', name))
```

```
## [1] "I acknowledge and will follow the advice given. - Max Han"
```

## Question 1 (4.5 points)

### Keeping code DRY (1.5 points)

You are designing a rectangular garden bed. You need to calculate the perimeter and area of the bed, as well as the amount of topsoil needed to fill it to a certain depth. Assume all measurements are in ft

```
# A rectangular garden bed with length 10 and width 5 has a perimeter of
perimeter <- (10 + 5) * 2

# A rectangular garden bed with length 10 and width 5 has an area of
area <- 10 * 5

# A rectangular garden bed with length 10 and width 5, filled to a depth of 0.2, has a volume of
volume <- 10 * 5 * 0.2
```

Apply the DRY principle using meaningful variable names.

### Mean and Variance (2 points)

The sample mean is defined as:

$$\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$$

and the sample variance is defined as

$$\hat{\sigma}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X}_n)^2$$

Using *vectorized* computations, compute the sample mean and sample variance of the `hwy` column of the `mpg` data set. Do not use the functions `mean` or `var`. You may use `length` and `sum`.

Note: `mpg` dataset is part of `ggplot2` package, that is already loaded into the environment. So just typing `mpg` into the console can show you the dataset

```
sum <- sum(mpg$hwy)
length <- length(mpg$hwy)
mean <- sum / length
print(mean)
```

```
## [1] 23.44017
```

```
variance = sum((mpg$hwy - mean)^2) / (length - 1)
print(variance)
```

```
## [1] 35.45778
```

### Transform data using if_else (1 pt)

Here is a vector of numbers:

```
scores <- c(60, 50, 70, 90, 10, 80, 40, 30)

compute_score <- function(v){
  new_scores <- ifelse(scores < 70, "Fail", "Pass")
}
```

```
print(compute_score(scores))
```

```
## [1] "Fail" "Fail" "Pass" "Pass" "Fail" "Pass" "Fail" "Fail"
```

Complete the function to return another vector that has a value 'Fail' for values below 70, 'Pass' for values 70 and beyond

## Question 2 (2 points)

Consider using built-in functions for the next set of problems. Reference: http://adv-r.had.co.nz/Vocabulary.html

### Investigate Storms data

Investigate the data set `storms`. Run `glimpse(storms)` to get some idea on the dataset and then answer the following

- How many unique names are present in the `name` column? Write an expression to find this value. (0.5 pt)

```
unique(storms$name)
```

```
##   [1] "Amy"       "Blanche"   "Caroline"  "Doris"     "Eloise"    "Faye"
##   [7] "Gladys"    "Hallie"    "Belle"     "Dottie"    "Candice"   "Emmy"
##  [13] "Frances"   "Gloria"    "Holly"     "Anita"     "Babe"      "Clara"
##  [19] "Dorothy"   "Evelyn"    "Frieda"    "Amelia"    "Bess"      "Cora"
##  [25] "Debra"     "Ella"      "Flossie"   "Hope"      "Greta"     "Irma"
##  [31] "Juliet"    "Kendra"    "Ana"       "Bob"       "Claudette" "David"
##  [37] "Frederic"  "Elena"     "Henri"     "Allen"     "Bonnie"    "Charley"
##  [43] "Georges"   "Earl"      "Danielle"  "Hermine"   "Ivan"      "Jeanne"
##  [49] "Karl"      "Arlene"    "Bret"      "Cindy"     "Dennis"    "Emily"
##  [55] "Floyd"     "Gert"      "Harvey"    "Irene"     "Jose"      "Katrina"
##  [61] "Alberto"   "Beryl"     "Chris"     "Debby"     "Ernesto"   "Alicia"
##  [67] "Barry"     "Chantal"   "Dean"      "Arthur"    "Bertha"    "Cesar"
##  [73] "Diana"     "Edouard"   "Fran"      "Gustav"    "Hortense"  "Isidore"
##  [79] "Josephine" "Klaus"     "Lili"      "Danny"     "Fabian"    "Isabel"
##  [85] "Juan"      "Kate"      "Andrew"    "AL031987"  "AL061988"  "Florence"
##  [91] "Gilbert"   "Helene"    "Isaac"     "Joan"      "Keith"     "Allison"
##  [97] "Erin"      "Felix"     "Gabrielle" "Hugo"      "Iris"      "Jerry"
## [103] "Karen"     "Marco"     "Nana"      "AL041991"  "Erika"     "AL101991"
## [109] "Grace"     "AL121991"  "AL021992"  "AL031992"  "AL081992"  "AL011993"
## [115] "AL101993"  "AL021994"  "AL051994"  "AL081994"  "AL091994"  "AL101994"
## [121] "Gordon"    "AL061995"  "Humberto"  "Luis"      "AL141995"  "Marilyn"
## [127] "Noel"      "Opal"      "Pablo"     "Roxanne"   "Sebastien" "Tanya"
## [133] "Dolly"     "Kyle"      "Bill"      "AL061997"  "Alex"      "Lisa"
## [139] "Mitch"     "Nicole"    "AL021999"  "AL071999"  "AL111999"  "AL121999"
## [145] "Lenny"     "AL012000"  "AL022000"  "AL042000"  "AL092000"  "Joyce"
## [151] "Leslie"    "Michael"   "Nadine"    "AL022001"  "AL092001"  "Lorenzo"
## [157] "Michelle"  "Olga"      "Cristobal" "Fay"       "AL072002"  "Hanna"
## [163] "AL142002"  "AL022003"  "AL062003"  "AL072003"  "AL092003"  "AL142003"
## [169] "Larry"     "Mindy"     "Nicholas"  "Odette"    "Peter"     "Gaston"
## [175] "AL102004"  "Matthew"   "Otto"      "Franklin"  "Ten"       "Lee"
## [181] "Maria"     "Nate"      "Ophelia"   "Philippe"  "Rita"      "Nineteen"
## [187] "Stan"      "Tammy"     "Vince"     "Wilma"     "Alpha"     "Beta"
## [193] "Gamma"     "Delta"     "Epsilon"   "Zeta"      "AL022006"  "Ingrid"
## [199] "Melissa"   "Fifteen"   "Ike"       "Laura"     "Omar"      "Sixteen"
```

```
## [205] "Paloma"    "One"       "Fred"      "Eight"     "Ida"       "Two"
## [211] "Colin"     "Five"      "Fiona"     "Igor"      "Julia"     "Paula"
## [217] "Richard"   "Shary"     "Tomas"     "Don"       "Katia"     "Al202011"
## [223] "Rina"      "Sean"      "Kirk"      "Oscar"     "Patty"     "Rafael"
## [229] "Sandy"     "Tony"      "Andrea"    "Dorian"    "Fernand"   "Gonzalo"
## [235] "Nine"      "Joaquin"   "Ian"       "Four"      "Eleven"    "Three"
## [241] "Imelda"    "Nestor"    "Isaias"    "Paulette"  "Rene"      "Sally"
## [247] "Teddy"     "Vicky"     "Wilfred"   "Eta"       "Theta"     "Iota"
## [253] "Elsa"      "Julian"    "Rose"      "Sam"       "Victor"    "Wanda"
## [259] "Twelve"    "Martin"
```

- I'm trying to find the mean value of tropicalstorm_force_diameter column with this expression `mean(storms$tropicalstorm_force_diameter)` and getting a value of 'NA'. Why is this giving me 'NA' and how to fix it? (0.5 pt)

```
glimpse(storms)
```

```
## Rows: 19,537
## Columns: 13
## $ name                          <chr> "Amy", "Amy", "Amy", "Amy", "Amy", "Amy",~
## $ year                          <dbl> 1975, 1975, 1975, 1975, 1975, 1975, 1975,~
## $ month                         <dbl> 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6,~
## $ day                           <int> 27, 27, 27, 27, 28, 28, 28, 28, 29, 29, 2~
## $ hour                          <dbl> 0, 6, 12, 18, 0, 6, 12, 18, 0, 6, 12, 18,~
## $ lat                           <dbl> 27.5, 28.5, 29.5, 30.5, 31.5, 32.4, 33.3,~
## $ long                          <dbl> -79.0, -79.0, -79.0, -79.0, -78.8, -78.7,~
## $ status                        <fct> tropical depression, tropical depression,~
## $ category                      <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N~
## $ wind                          <int> 25, 25, 25, 25, 25, 25, 25, 30, 35, 40, 4~
## $ pressure                      <int> 1013, 1013, 1013, 1013, 1012, 1012, 1011,~
## $ tropicalstorm_force_diameter <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N~
## $ hurricane_force_diameter     <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N~
```

The reason we have NA for mean value of tropicalstorm_force_diameter is because all the values for column are defined with NA.

- What is the highest value in the tropicalstorm_force_diameter? (0.5 pt)

```
max(storms$tropicalstorm_force_diameter)
```

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

Since tropicalstorm_force_diameter has no value, max value is also NA, which means it doesn't exist.

- What is the mean value of wind speed when the `status` is 'tropical storm'? (0.5 pt)

```
mean(storms$status)
```

```
## Warning in mean.default(storms$status): argument is not numeric or logical:
## returning NA
```

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

Since status is not numeric data type, we can't calculate it.

## Question 3 (3 points)

Some problem sets may feature one or two questions or part of questions that go a bit beyond what we have covered in lab and lecture. The goal of these is for you to learn how to use online resources (R's help, Google,

Stack Overflow, GenAI, etc.) to solve programming challenges that you have not encountered before. This is an important skill that you will use constantly as a data scientist in the real world.

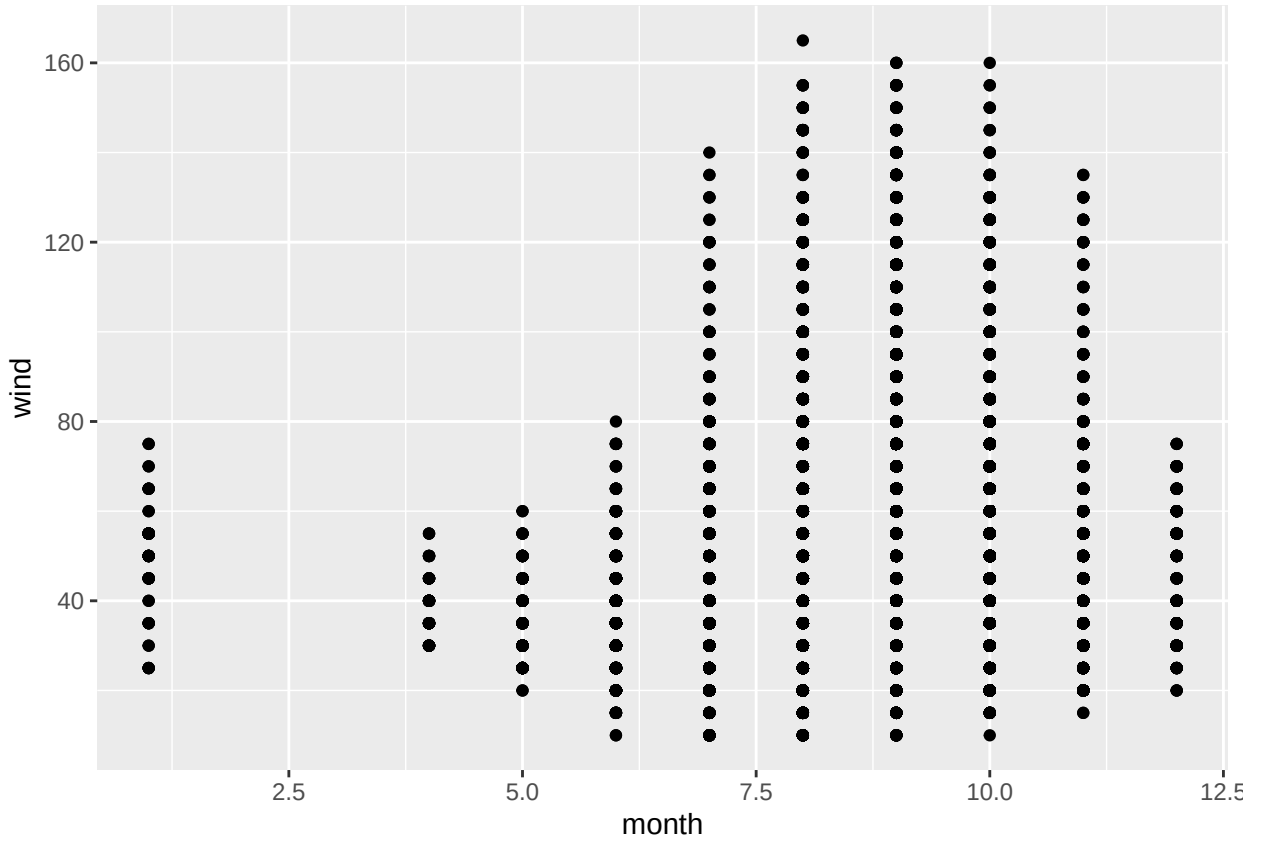
**Wind and Month**

I'm trying to create a scatter plot of 'month' and 'wind' columns with the below code

```
storms |> ggplot(x = month, y = wind) + geom_point()
```

But it is not working. I want the x-axis with only integer values representing months without any decimal values. Fix the code to get the desired plot.

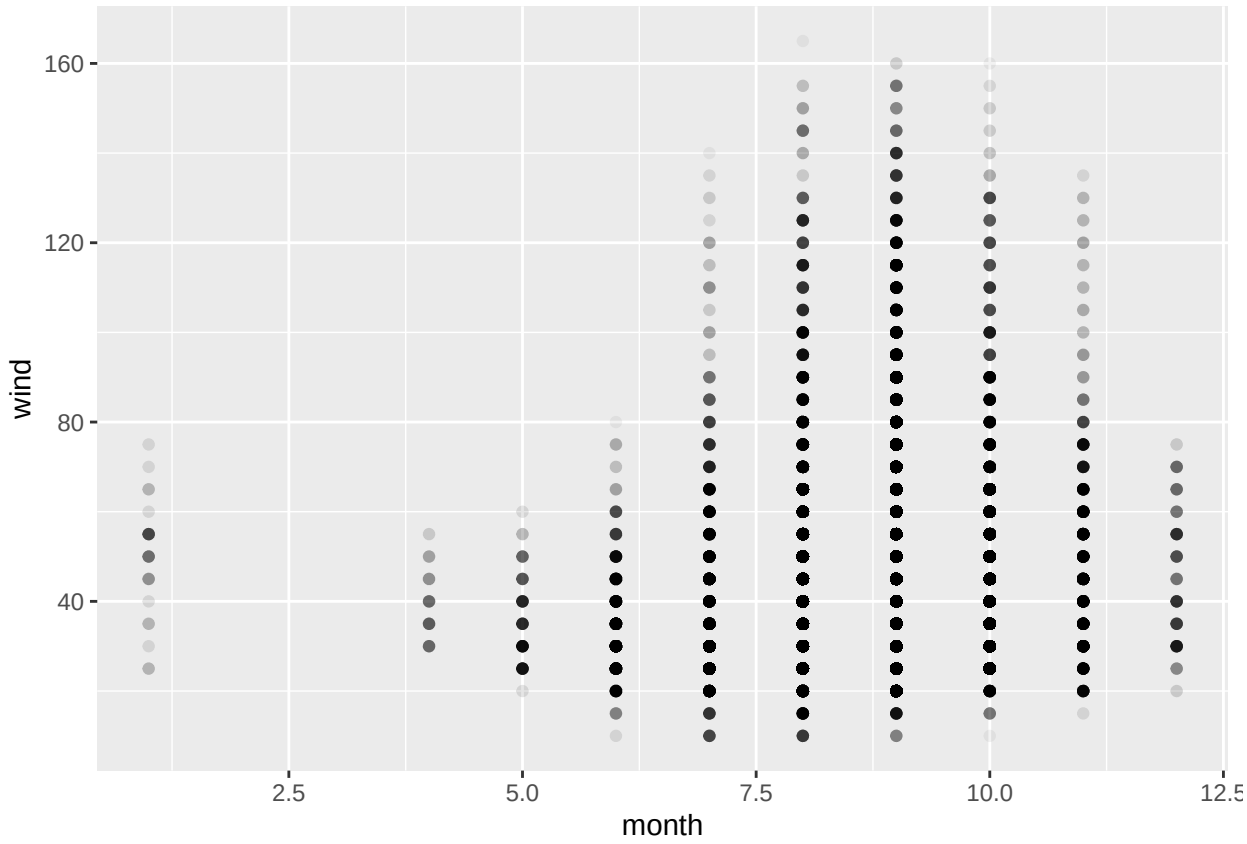
```
storms |> ggplot(aes(x = month, y = wind)) + geom_point()
```




Once plotted appropriately, answer the following? 1. Which two months are not windy as per this dataset? (0.5 point) Month 4 and 5 are not windy.

2. Is each dot representing a single data point, or are multiple data points overlayed on each dot? How could the chart be modified to clarify the number of observations represented by each dot? (0.5 point)

```
ggplot(data = storms, aes(x = month, y = wind)) +
  geom_point(alpha = 0.05)
```




We can modify alpha parameter to clarify overlapping points. The more darker a point is, the more overlapping points it represents.

3. Create a boxplot as shown below for wind data:

```
ggplot(data = storms, aes(x = factor(month), y = wind)) +
  geom_boxplot(outlier.color = "red") +
  labs(x = "Month", y = "Wind")
```

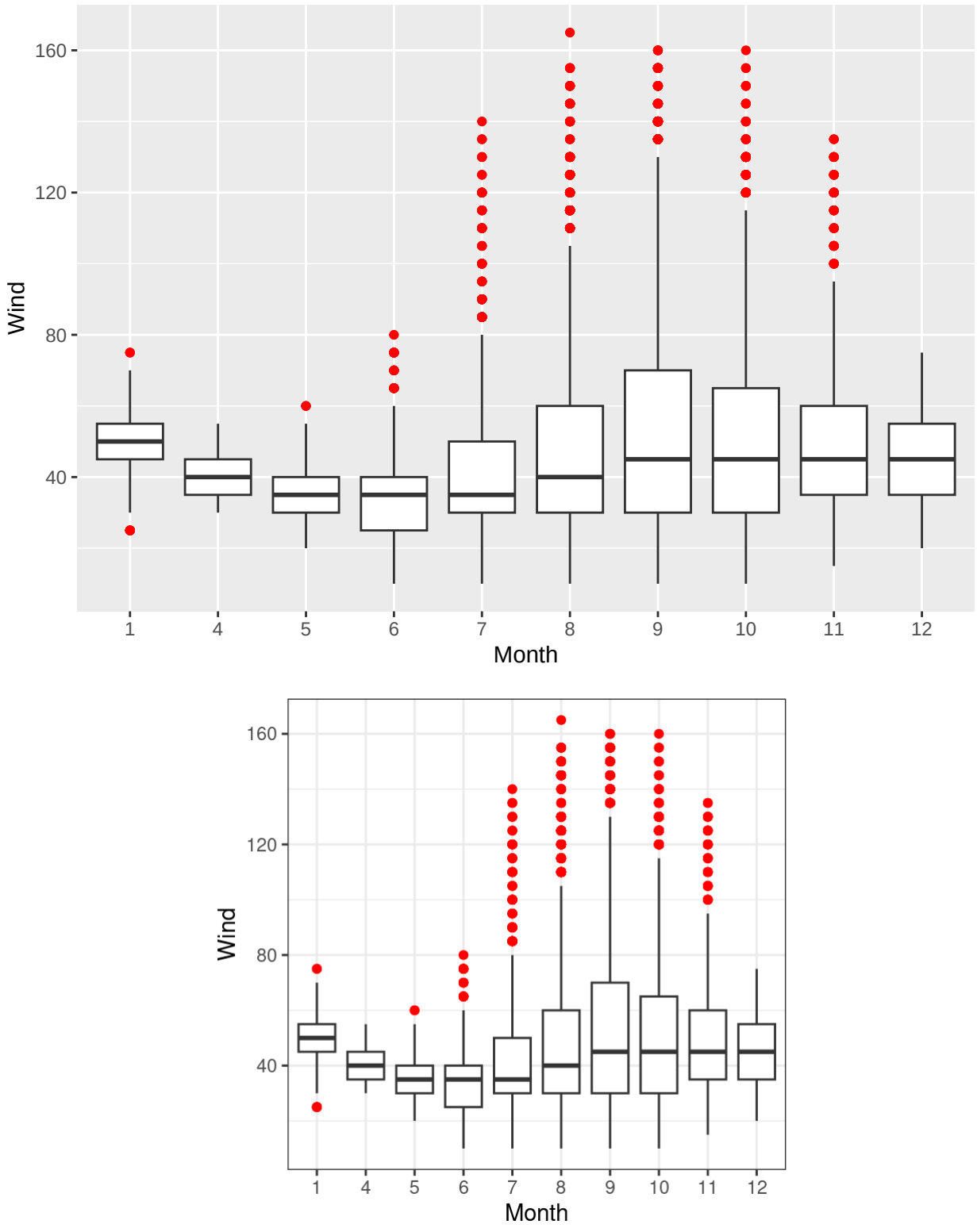



Figure 1: Wind vs month

Show the outliers in red color. Explain what is considered as an outlier in this chart (2 points)

**Hint:** The provided plot requires categorical data on the x-axis. As the 'month' column in our dataset is

currently numeric, you'll need to convert it to a categorical type, such as a factor, before plotting.