

RWorksheet_Laguda#4c

Rey-Ann S. Laguda

2023-12-04

#1

```
install.packages("ggplot2")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'  
## (as 'lib' is unspecified)
```

```
library(ggplot2)  
install.packages("tidyverse")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'  
## (as 'lib' is unspecified)
```

#1a

```
# Show your solutions on how to import a csv file into the environment
```

```
data(mpg)  
mpg
```

```
## # A tibble: 234 x 11  
##   manufacturer model      displ  year   cyl trans drv     cty   hwy fl      class  
##   <chr>         <chr>    <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>  
## 1 audi         a4         1.8  1999     4 auto~ f      18    29 p      comp~  
## 2 audi         a4         1.8  1999     4 manu~ f      21    29 p      comp~  
## 3 audi         a4         2    2008     4 manu~ f      20    31 p      comp~  
## 4 audi         a4         2    2008     4 auto~ f      21    30 p      comp~  
## 5 audi         a4         2.8  1999     6 auto~ f      16    26 p      comp~  
## 6 audi         a4         2.8  1999     6 manu~ f      18    26 p      comp~  
## 7 audi         a4         3.1  2008     6 auto~ f      18    27 p      comp~  
## 8 audi         a4 quattro 1.8  1999     4 manu~ 4      18    26 p      comp~  
## 9 audi         a4 quattro 1.8  1999     4 auto~ 4      16    25 p      comp~  
## 10 audi        a4 quattro 2    2008     4 manu~ 4      20    28 p      comp~  
## # i 224 more rows
```

#1b

```
# Which variables from mpg dataset are categorical?
```

```
#The variables that are categorical in mpg dataset are manufacturer, model, trans, drv, and fl.
```

#1c

```
#Which are continuous variables?
```

```
#The continuous variables in the mpg dataset are displ, year, cyl, cty, and hwy.
```

#2

#Which manufacturer has the most models in this data set? Which model has the most variations? Show your code and result.

```
library(dplyr)
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
## filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
## intersect, setdiff, setequal, union
```

```
manufacturer_most_models <- mpg %>%
```

```
  group_by(manufacturer) %>%
```

```
  summarize(number_of_models = n_distinct(model)) %>%
```

```
  top_n(1, number_of_models)
```

```
model_most_variations <- mpg %>%
```

```
  group_by(model) %>%
```

```
  summarize(number_of_variations = n_distinct(trans)) %>%
```

```
  top_n(1, number_of_variations)
```

```
cat("Manufacturer with the most models:", manufacturer_most_models$manufacturer, "\n")
```

```
## Manufacturer with the most models: toyota
```

```
cat("Model with the most variations:", model_most_variations$model, "\n")
```

```
## Model with the most variations: a4 a4 quattro altima camry civic dakota pickup 4wd explorer 4wd gti
```

#2a

Group the manufacturers and find the unique models. Show your codes and result.

```
library(dplyr)
```

```
unique_models_by_manufacturer <- mpg %>%
```

```
  group_by(manufacturer) %>%
```

```
  distinct(model)
```

```
print(unique_models_by_manufacturer)
```

```
## # A tibble: 38 x 2
```

```
## # Groups:   manufacturer [15]
```

```
##   manufacturer model
```

```
##   <chr>          <chr>
```

```
## 1 audi          a4
```

```
## 2 audi          a4 quattro
```

```
## 3 audi          a6 quattro
```

```
## 4 chevrolet     c1500 suburban 2wd
```

```
## 5 chevrolet     corvette
```

```
## 6 chevrolet     k1500 tahoe 4wd
```

```
## 7 chevrolet     malibu
```

```
## 8 dodge         caravan 2wd
```

```
## 9 dodge         dakota pickup 4wd
```

```
## 10 dodge        durango 4wd
```

```
## # i 28 more rows
```

#2b

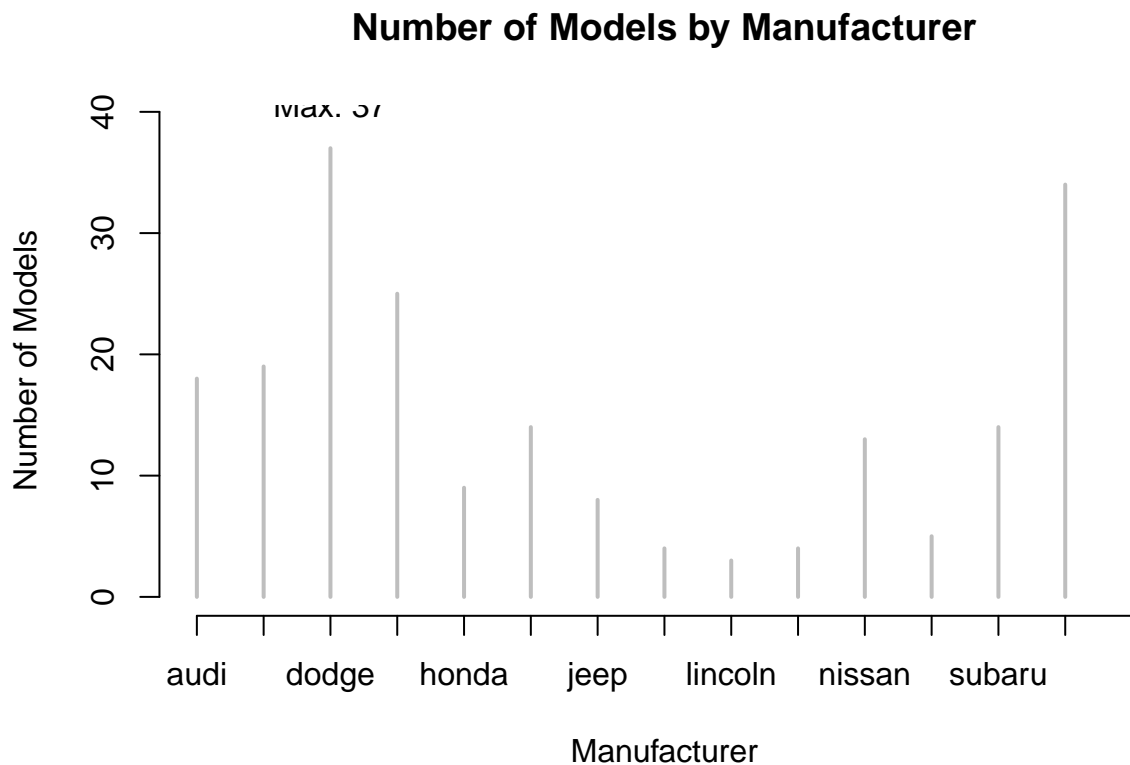
Graph the result by using plot() and ggplot(). Write the codes and its result.

```
models_per_manufacturer <- table(mpg$manufacturer)
```

#Using plot()

```
plot(models_per_manufacturer,  
      main = "Number of Models by Manufacturer",  
      xlab = "Manufacturer",  
      ylab = "Number of Models",  
      col = "grey",  
      ylim = c(0, max(models_per_manufacturer) + 2))
```

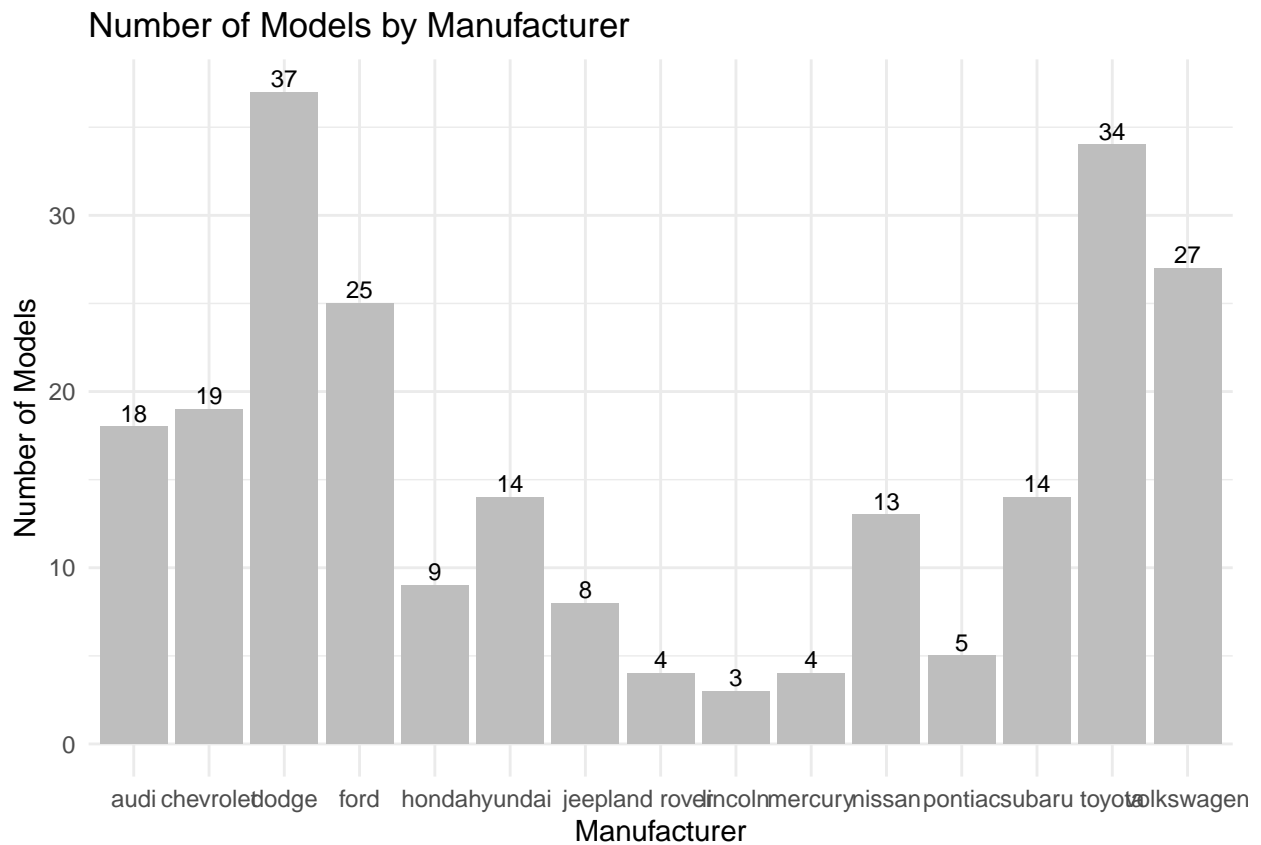
```
max_manufacturer <- which.max(models_per_manufacturer)  
text(max_manufacturer, models_per_manufacturer[max_manufacturer] + 1,  
      labels = paste("Max:", max(models_per_manufacturer)),  
      col = "black", pos = 3)
```



```
library(ggplot2)
```

#Using ggplot2

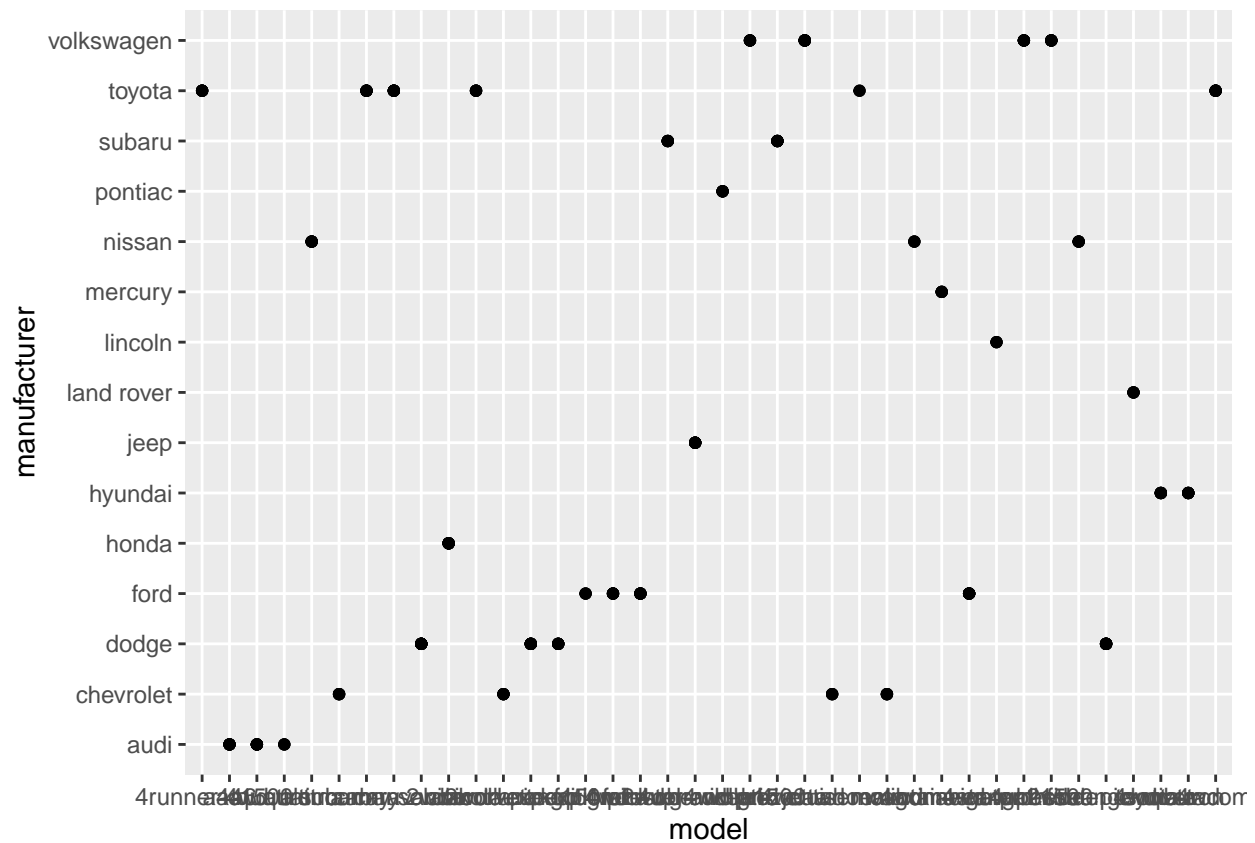
```
ggplot(data = as.data.frame(models_per_manufacturer), aes(x = Var1, y = Freq)) +  
  geom_bar(stat = "identity", fill = "grey") +  
  labs(title = "Number of Models by Manufacturer",  
        x = "Manufacturer",  
        y = "Number of Models") +  
  geom_text(aes(label = Freq), vjust = -0.3, col = "black", size = 3) +  
  theme_minimal()
```



#2aa

#. Same dataset will be used. You are going to show the relationship of the model and the manufacturer.

```
library(ggplot2)
ggplot(mpg, aes(model, manufacturer)) +
  geom_point()
```

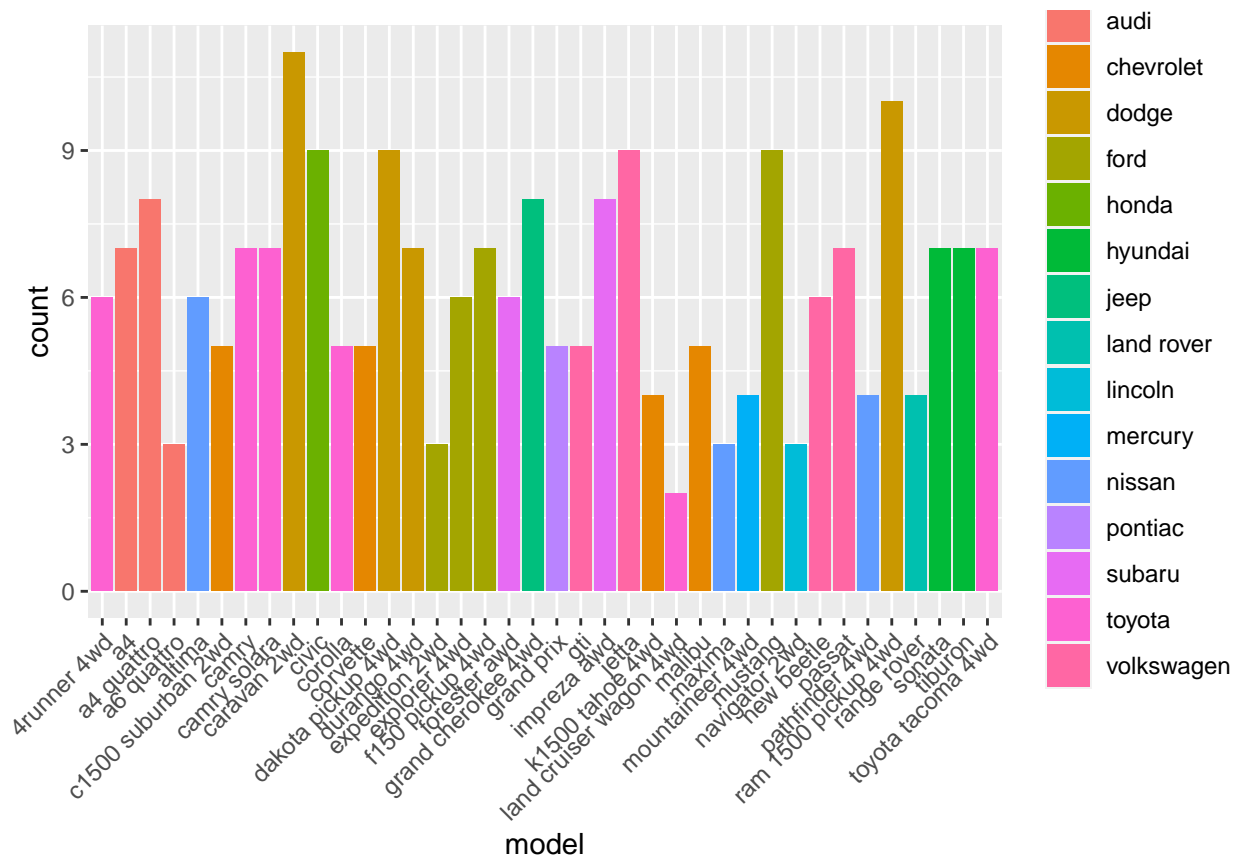


#2bb

The plot might not be very informative because it's attempting to create a scatter plot using categor

#Tallying Data

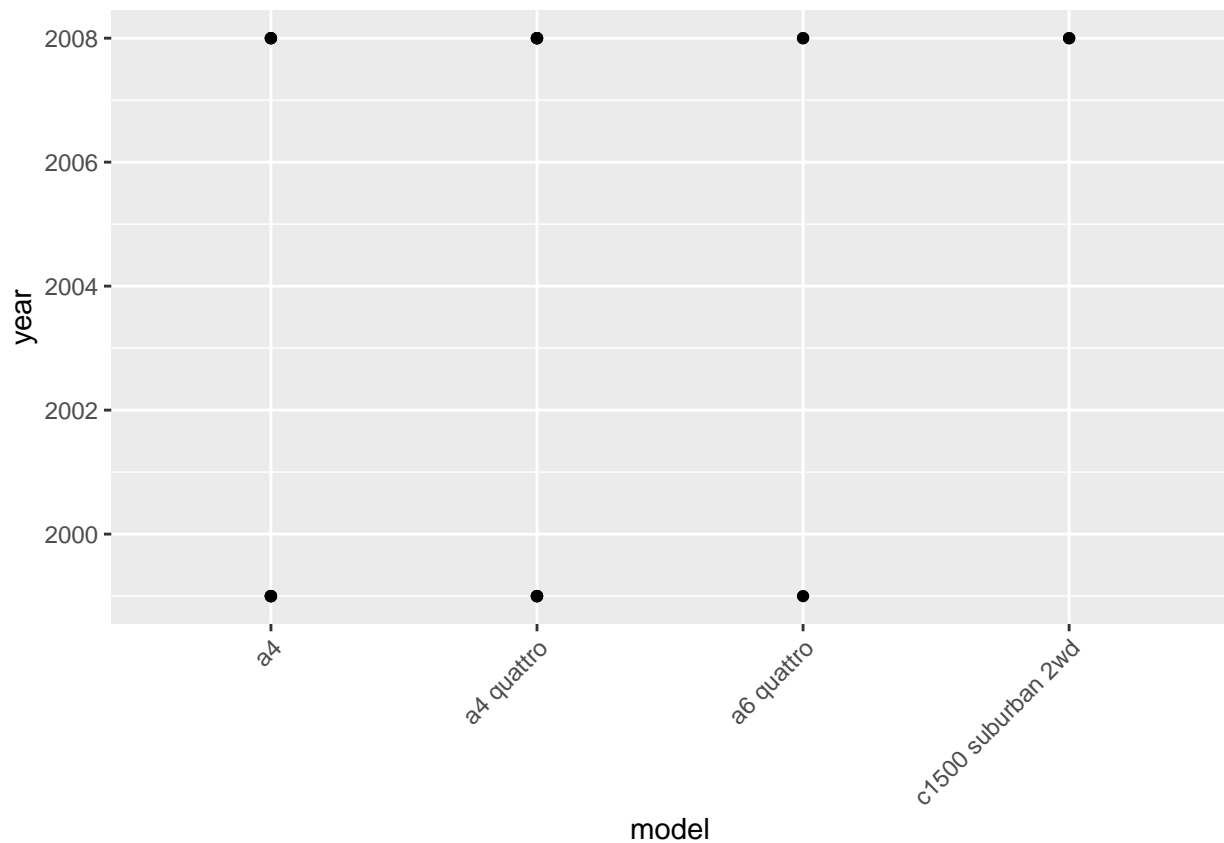
```
library(ggplot2)
ggplot(mpg, aes(model, fill = manufacturer)) +
  geom_bar(position = "dodge") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



#3

#Plot the model and the year using ggplot(). Use only the top 20 observations. Write the codes and its

```
library(ggplot2)
top_20 <- head(mpg, 20)
ggplot(top_20, aes(model, year)) +
  geom_point() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



#4

Using the pipe (%>), group the model and get the number of cars per model. Show codes and its result

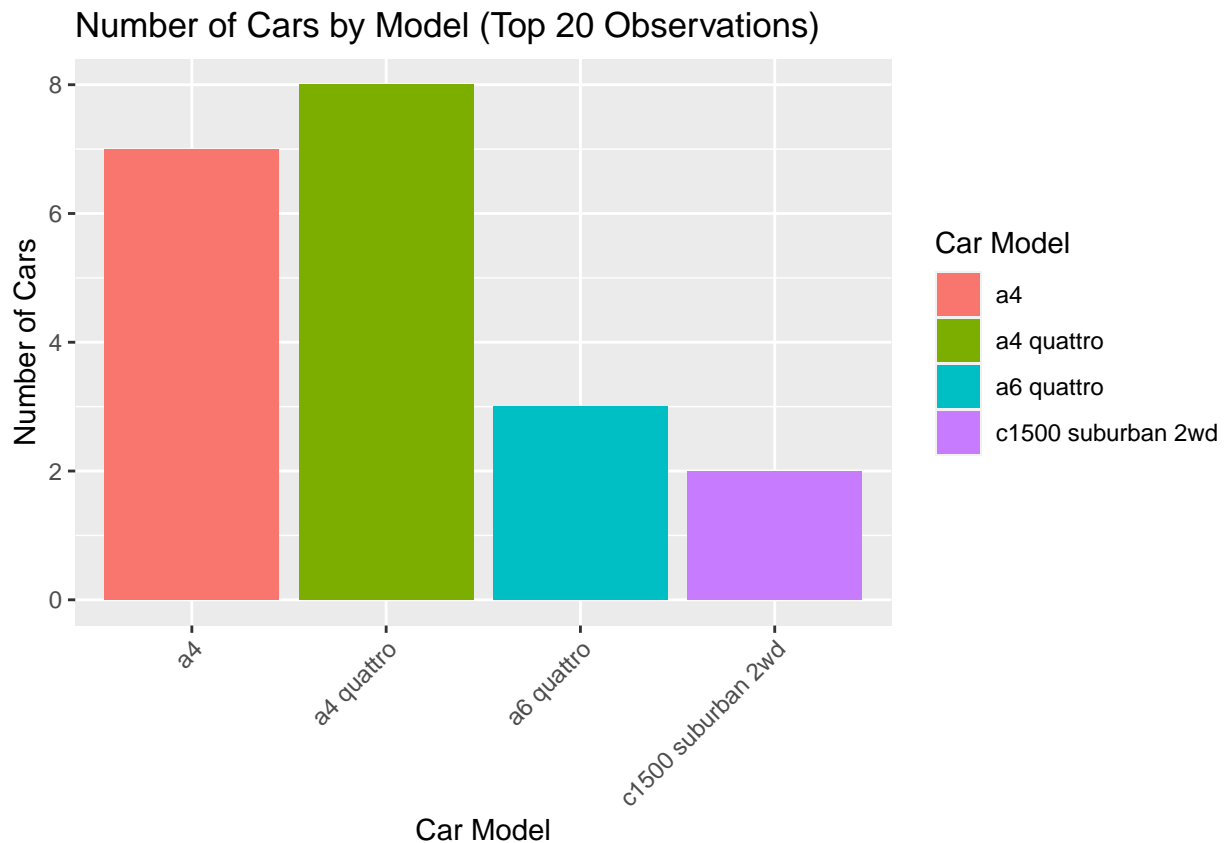
```
library(dplyr)
car_counts <- mpg %>%
  group_by(model) %>%
  summarize(number_of_cars = n())
print(car_counts)
```

```
## # A tibble: 38 x 2
##   model          number_of_cars
##   <chr>          <int>
## 1 4runner 4wd           6
## 2 a4                  7
## 3 a4 quattro           8
## 4 a6 quattro           3
## 5 altima              6
## 6 c1500 suburban 2wd    5
## 7 camry               7
## 8 camry solara         7
## 9 caravan 2wd          11
## 10 civic              9
## # i 28 more rows
```

#4a

a. Plot using geom_bar() using the top 20 observations only. The graphs should have a title, labels a

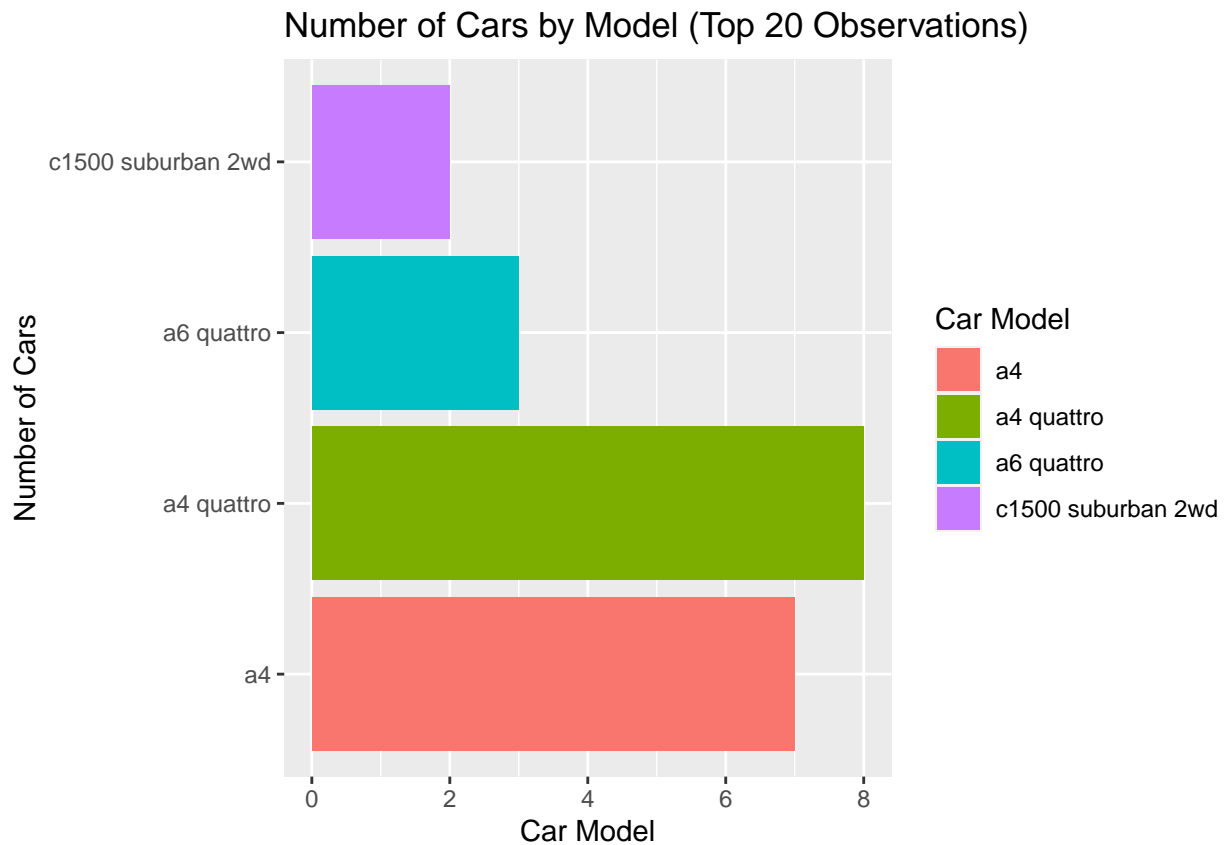
```
library(ggplot2)
top_20 <- head(mpg, 20)
ggplot(top_20, aes(x = model, fill = factor(model))) +
  geom_bar() +
  labs(title = "Number of Cars by Model (Top 20 Observations)",
       x = "Car Model",
       y = "Number of Cars") +
  scale_fill_discrete(name = "Car Model") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



#4b

Plot using the geom_bar() + coord_flip() just like what is shown below. Show codes and its result.

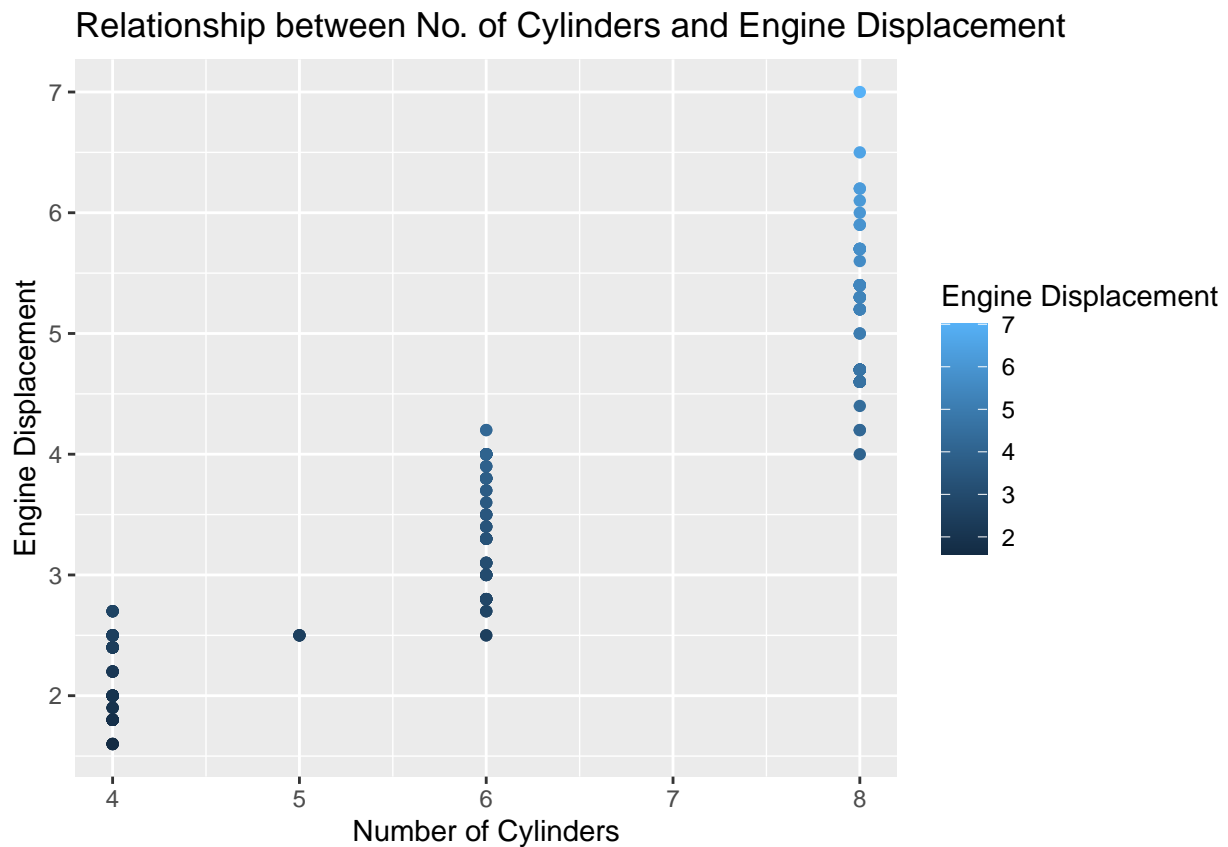
```
library(ggplot2)
top_20 <- head(mpg, 20)
ggplot(top_20, aes(x = model, fill = factor(model))) +
  geom_bar() +
  labs(title = "Number of Cars by Model (Top 20 Observations)",
       x = "Number of Cars",
       y = "Car Model") +
  scale_fill_discrete(name = "Car Model") +
  coord_flip()
```

#5

Plot the relationship between cyl - number of cylinders and displ - engine displacement using geom_point

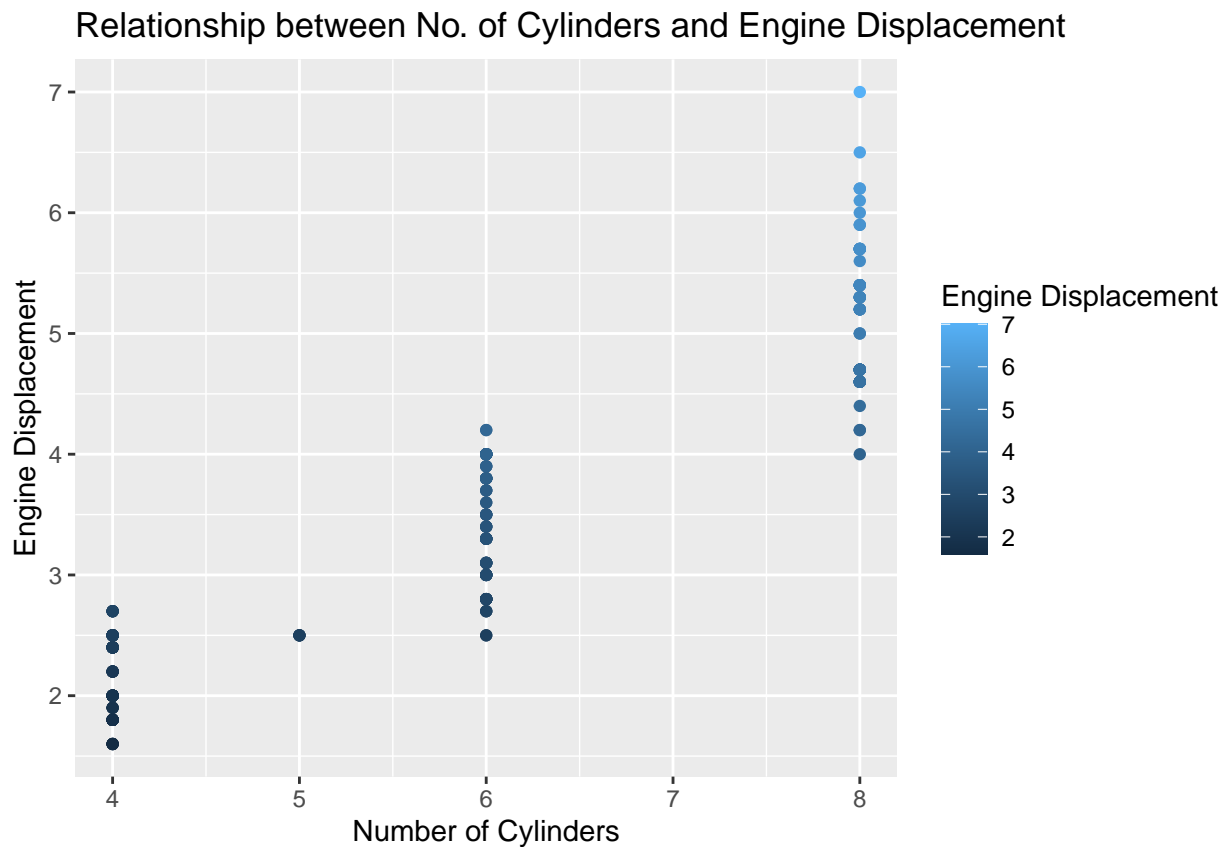
```
library(ggplot2)
ggplot(mpg, aes(x = cyl, y = displ, color = displ)) +
  geom_point() +
  labs(title = "Relationship between No. of Cylinders and Engine Displacement",
       x = "Number of Cylinders",
       y = "Engine Displacement") +
  scale_color_continuous(name = "Engine Displacement")
```



#5a

How would you describe its relationship? Show the codes and its result.

```
library(ggplot2)
ggplot(mpg, aes(x = cyl, y = displ, color = displ)) +
  geom_point() +
  labs(title = "Relationship between No. of Cylinders and Engine Displacement",
       x = "Number of Cylinders",
       y = "Engine Displacement") +
  scale_color_continuous(name = "Engine Displacement")
```



#6

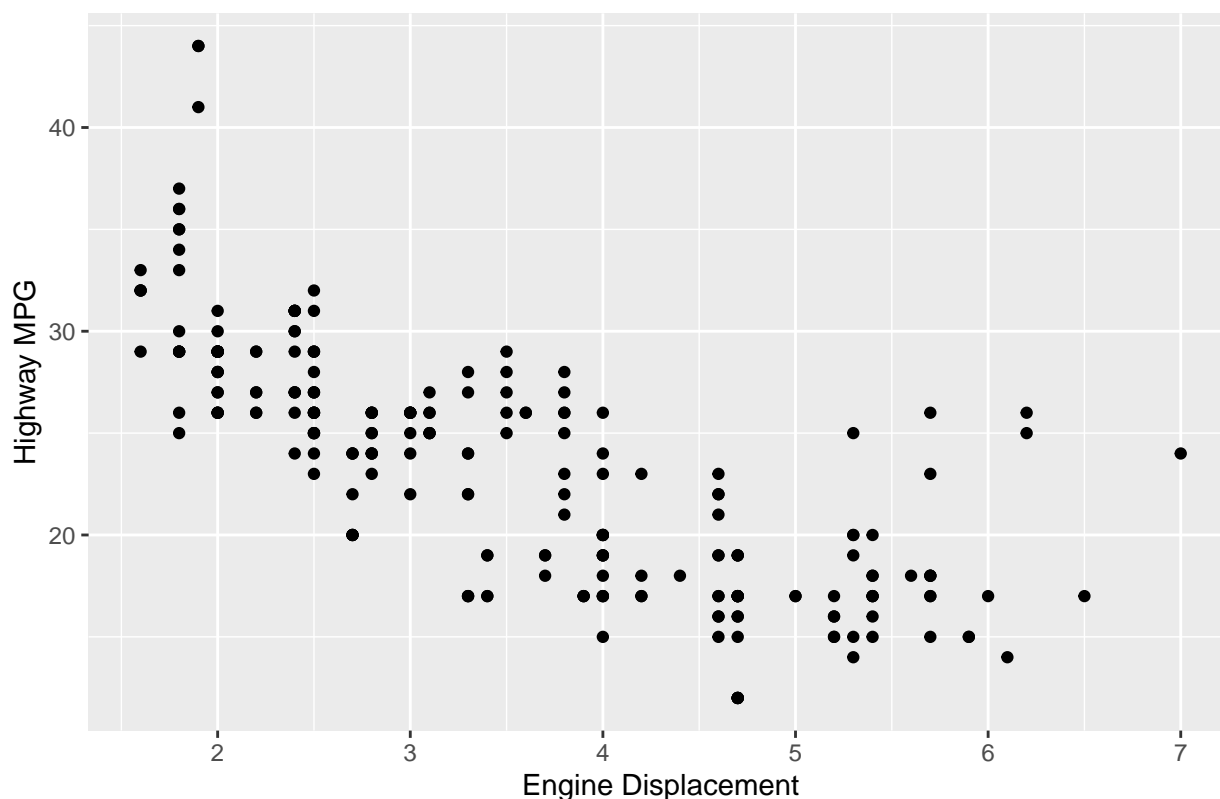
Plot the relationship between displ (engine displacement) and hwy(highway miles per gallon). Mapped i

```
library(ggplot2)
continuous_variable <- mpg$your_continuous_variable

## Warning: Unknown or uninitialised column: `your_continuous_variable`.

ggplot(mpg, aes(x = displ, y = hwy, color = continuous_variable)) +
  geom_point() +
  labs(title = "Relationship between Engine Displacement and Highway MPG",
       x = "Engine Displacement",
       y = "Highway MPG") +
  scale_color_continuous(name = "Your Continuous Variable")
```

Relationship between Engine Displacement and Highway MPG



#6.1

```
library(readr)
traffic <- read_csv("/cloud/project/RWorksheet4/traffic.csv")

## Rows: 48120 Columns: 4
## -- Column specification -----
## Delimiter: ","
## dbl (3): Junction, Vehicles, ID
## dtm (1): DateTime
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
traffic
```

```
## # A tibble: 48,120 x 4
##   DateTime          Junction Vehicles      ID
##   <dtm>            <dbl>    <dbl>    <dbl>
## 1 2015-11-01 00:00:00      1      15 20151101001
## 2 2015-11-01 01:00:00      1      13 20151101011
## 3 2015-11-01 02:00:00      1      10 20151101021
## 4 2015-11-01 03:00:00      1       7 20151101031
## 5 2015-11-01 04:00:00      1       9 20151101041
## 6 2015-11-01 05:00:00      1       6 20151101051
## 7 2015-11-01 06:00:00      1       9 20151101061
## 8 2015-11-01 07:00:00      1       8 20151101071
## 9 2015-11-01 08:00:00      1      11 20151101081
## 10 2015-11-01 09:00:00      1      12 20151101091
```

```
## # i 48,110 more rows
```

```
#6a
```

```
observations <- nrow(traffic)
observations
```

```
## [1] 48120
```

```
columns <- ncol(traffic)
columns
```

```
## [1] 4
```

```
#6b
```

```
junction1 <- subset(traffic, Junction ==1)
junction1
```

```
## # A tibble: 14,592 x 4
```

```
##   DateTime          Junction Vehicles      ID
##   <dtm>              <dbl>    <dbl>    <dbl>
## 1 2015-11-01 00:00:00         1        15 20151101001
## 2 2015-11-01 01:00:00         1        13 20151101011
## 3 2015-11-01 02:00:00         1        10 20151101021
## 4 2015-11-01 03:00:00         1         7 20151101031
## 5 2015-11-01 04:00:00         1         9 20151101041
## 6 2015-11-01 05:00:00         1         6 20151101051
## 7 2015-11-01 06:00:00         1         9 20151101061
## 8 2015-11-01 07:00:00         1         8 20151101071
## 9 2015-11-01 08:00:00         1        11 20151101081
## 10 2015-11-01 09:00:00        1        12 20151101091
## # i 14,582 more rows
```

```
junction2 <- subset(traffic, Junction ==2)
junction2
```

```
## # A tibble: 14,592 x 4
```

```
##   DateTime          Junction Vehicles      ID
##   <dtm>              <dbl>    <dbl>    <dbl>
## 1 2015-11-01 00:00:00         2         6 20151101002
## 2 2015-11-01 01:00:00         2         6 20151101012
## 3 2015-11-01 02:00:00         2         5 20151101022
## 4 2015-11-01 03:00:00         2         6 20151101032
## 5 2015-11-01 04:00:00         2         7 20151101042
## 6 2015-11-01 05:00:00         2         2 20151101052
## 7 2015-11-01 06:00:00         2         4 20151101062
## 8 2015-11-01 07:00:00         2         4 20151101072
## 9 2015-11-01 08:00:00         2         3 20151101082
## 10 2015-11-01 09:00:00        2         3 20151101092
## # i 14,582 more rows
```

```
junction3 <- subset(traffic, Junction ==3)
junction3
```

```
## # A tibble: 14,592 x 4
```

```
##   DateTime          Junction Vehicles      ID
##   <dtm>              <dbl>    <dbl>    <dbl>
## 1 2015-11-01 00:00:00         3         9 20151101003
```

```
## 2 2015-11-01 01:00:00      3      7 20151101013
## 3 2015-11-01 02:00:00      3      5 20151101023
## 4 2015-11-01 03:00:00      3      1 20151101033
## 5 2015-11-01 04:00:00      3      2 20151101043
## 6 2015-11-01 05:00:00      3      2 20151101053
## 7 2015-11-01 06:00:00      3      3 20151101063
## 8 2015-11-01 07:00:00      3      4 20151101073
## 9 2015-11-01 08:00:00      3      3 20151101083
## 10 2015-11-01 09:00:00     3      6 20151101093
## # i 14,582 more rows
```

```
junction4 <- subset(traffic, Junction ==4)
junction4
```

```
## # A tibble: 4,344 x 4
##   DateTime      Junction Vehicles      ID
##   <dtm>          <dbl>    <dbl>    <dbl>
## 1 2017-01-01 00:00:00      4      3 20170101004
## 2 2017-01-01 01:00:00      4      1 20170101014
## 3 2017-01-01 02:00:00      4      4 20170101024
## 4 2017-01-01 03:00:00      4      4 20170101034
## 5 2017-01-01 04:00:00      4      2 20170101044
## 6 2017-01-01 05:00:00      4      1 20170101054
## 7 2017-01-01 06:00:00      4      1 20170101064
## 8 2017-01-01 07:00:00      4      4 20170101074
## 9 2017-01-01 08:00:00      4      4 20170101084
## 10 2017-01-01 09:00:00      4      2 20170101094
## # i 4,334 more rows
```

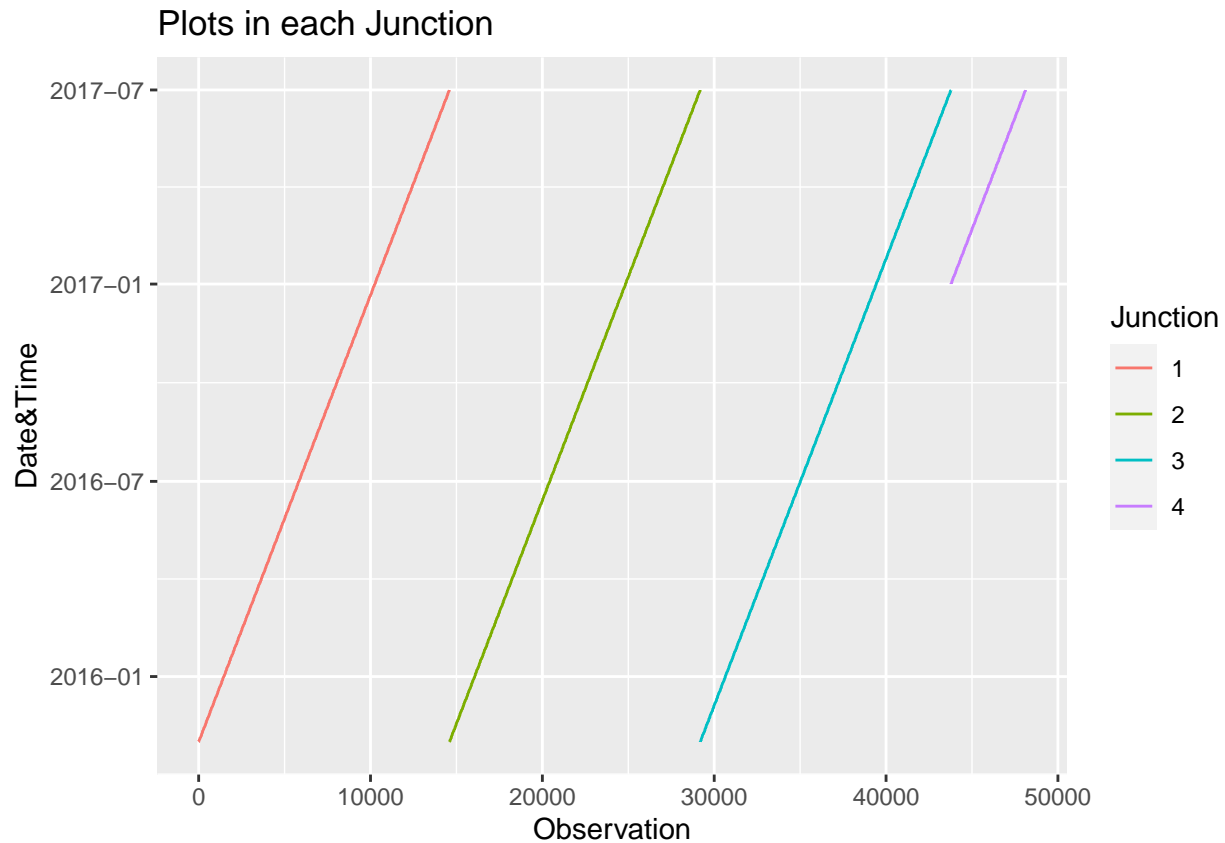
#6c

```
# Assuming 'traffic' is your dataset
```

```
library(ggplot2)
```

```
# Plot each junction using geom_line()
```

```
ggplot(traffic, aes(x = seq_along(Junction), y = DateTime, group = Junction, color = factor(Junction)))
  geom_line() +
  labs(title = "Plots in each Junction",
       x = "Observation",
       y = "Date&Time") +
  scale_color_discrete(name = "Junction")
```



#7

```
library(readxl)
alexa_file <- read_xlsx("/cloud/project/RWorksheet4/alexa_file.xlsx")
alexa_file
```

```
## # A tibble: 3,150 x 5
##   rating date          variation      verified_reviews      feedback
##   <dbl> <dtm>          <chr>          <chr>          <dbl>
## 1     5 2018-07-31 00:00:00 Charcoal Fabric Love my Echo!          1
## 2     5 2018-07-31 00:00:00 Charcoal Fabric Loved it!              1
## 3     4 2018-07-31 00:00:00 Walnut Finish  Sometimes while play~  1
## 4     5 2018-07-31 00:00:00 Charcoal Fabric I have had a lot of ~  1
## 5     5 2018-07-31 00:00:00 Charcoal Fabric Music              1
## 6     5 2018-07-31 00:00:00 Heather Gray Fabric I received the echo ~  1
## 7     3 2018-07-31 00:00:00 Sandstone Fabric Without having a cel~  1
## 8     5 2018-07-31 00:00:00 Charcoal Fabric I think this is the ~  1
## 9     5 2018-07-30 00:00:00 Heather Gray Fabric looks great      1
## 10    5 2018-07-30 00:00:00 Heather Gray Fabric Love it! I've listen~ 1
## # i 3,140 more rows
```

#7a

```
observ <- nrow(alexa_file)
observ
```

```
## [1] 3150
```

```
colm <- ncol(alexa_file)
colm
```

```
## [1] 5
```

```
#7b
```

```
# Assuming 'your_dataset' is your actual dataset and 'variations' is the variable of interest  
library(dplyr)
```

```
# Group by variations and count the occurrences of each variation
```

```
result <- alexa_file %>%  
  group_by(variation) %>%  
  summarize(total = n())
```

```
# Print the result
```

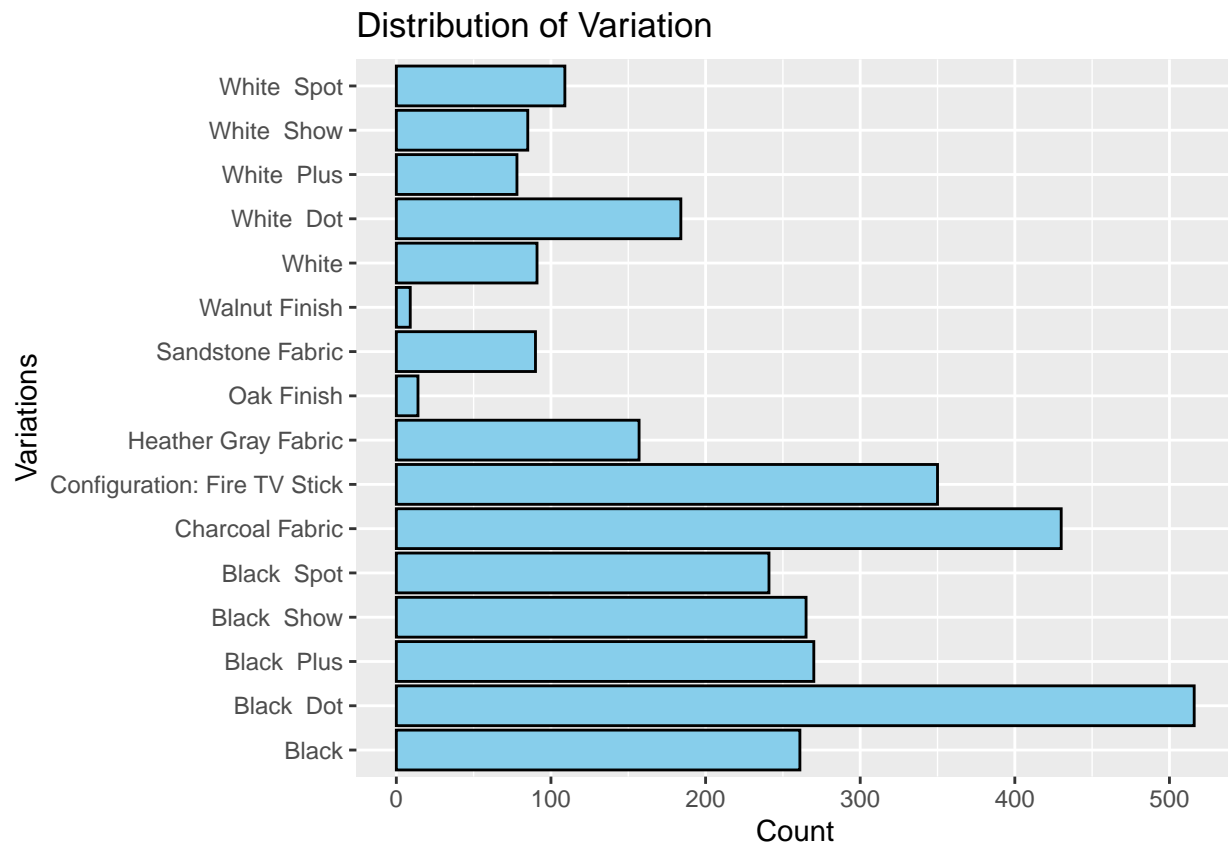
```
print(result)
```

```
## # A tibble: 16 x 2
```

##	variation	total
##	<chr>	<int>
##	1 Black	261
##	2 Black Dot	516
##	3 Black Plus	270
##	4 Black Show	265
##	5 Black Spot	241
##	6 Charcoal Fabric	430
##	7 Configuration: Fire TV Stick	350
##	8 Heather Gray Fabric	157
##	9 Oak Finish	14
##	10 Sandstone Fabric	90
##	11 Walnut Finish	9
##	12 White	91
##	13 White Dot	184
##	14 White Plus	78
##	15 White Show	85
##	16 White Spot	109

```
#7c
```

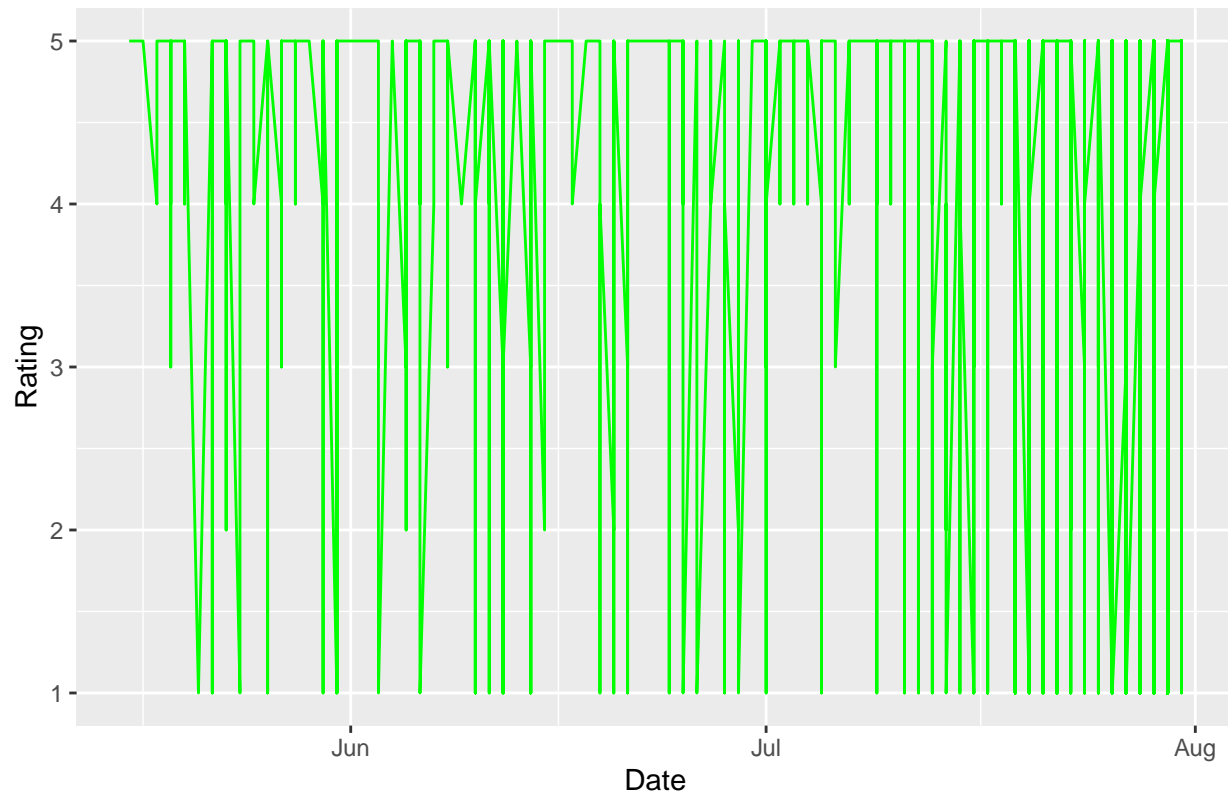
```
ggplot(alexa_file, aes(x = variation)) +  
  geom_bar(fill = "skyblue", color = "black") +  
  labs(title = "Distribution of Variation",  
        y = "Count",  
        x = "Variations") +  
  coord_flip()
```

#7d

```
ggplot(alexa_file, aes(x = date, y = rating)) +  
  geom_line(color= "green") +  
  labs (title = "Date and the number of verified reviews",  
        x = "Date",  
        y = "Rating")
```

Date and the number of verified reviews



#7e

```
ggplot(alexa_file, aes(x = variation, y = rating)) +
  geom_boxplot(fill = "skyblue", color = "black") +
  labs(title = "Relationship Between Variations and Ratings",
        x = "Variations",
        y = "Ratings") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Relationship Between Variations and Ratings

