# RWorksheet_Cautivar#4c.Rmd

James Clark Cautivar

2024-11-1

1. Use the dataset mpg

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

```r
library(ggplot2)

write.csv(mpg, "mpg.csv", row.names = FALSE)
mpg_data <- read.csv("mpg.csv")
str(mpg_data)
```

```
## 'data.frame':    234 obs. of  11 variables:
##  $ manufacturer: chr  "audi" "audi" "audi" "audi" ...
##  $ model       : chr  "a4" "a4" "a4" "a4" ...
##  $ displ       : num  1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
##  $ year        : int  1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
##  $ cyl         : int  4 4 4 4 6 6 6 4 4 4 ...
##  $ trans       : chr  "auto(l5)" "manual(m5)" "manual(m6)" "auto(av)" ...
##  $ drv         : chr  "f" "f" "f" "f" ...
##  $ cty         : int  18 21 20 21 16 18 18 18 16 20 ...
##  $ hwy         : int  29 29 31 30 26 26 27 26 25 28 ...
##  $ fl          : chr  "p" "p" "p" "p" ...
##  $ class       : chr  "compact" "compact" "compact" "compact" ...
```

b. Which variables from mpg dataset are categorical? The variables that are categorical in the mpg dataset are manufacturer, model, year, cyl, trans, drv, fl, and class.

c. Which are continuous variables? The continuous variables are displ, cty, and hwy.

2. Which manufacturer has the most models in this data set? Which model has the most variations? Show your answer.

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

```r
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
```

```r
manufacturerModelCount <- mpg %>%
  group_by(manufacturer) %>%
  summarize(num_models = n_distinct(model)) %>%
  arrange(desc(num_models))

manufacturerModelCount
```

```
## # A tibble: 15 x 2
##    manufacturer num_models
##    <chr>             <int>
##  1 toyota                6
##  2 chevrolet             4
##  3 dodge                 4
##  4 ford                  4
##  5 volkswagen            4
##  6 audi                  3
##  7 nissan                3
##  8 hyundai               2
##  9 subaru                2
## 10 honda                 1
## 11 jeep                  1
## 12 land rover            1
## 13 lincoln               1
## 14 mercury               1
## 15 pontiac               1
```

```
modelVariationCount <- table(mpg$model)
modelVariationCount [modelVariationCount  == max(modelVariationCount )]
```

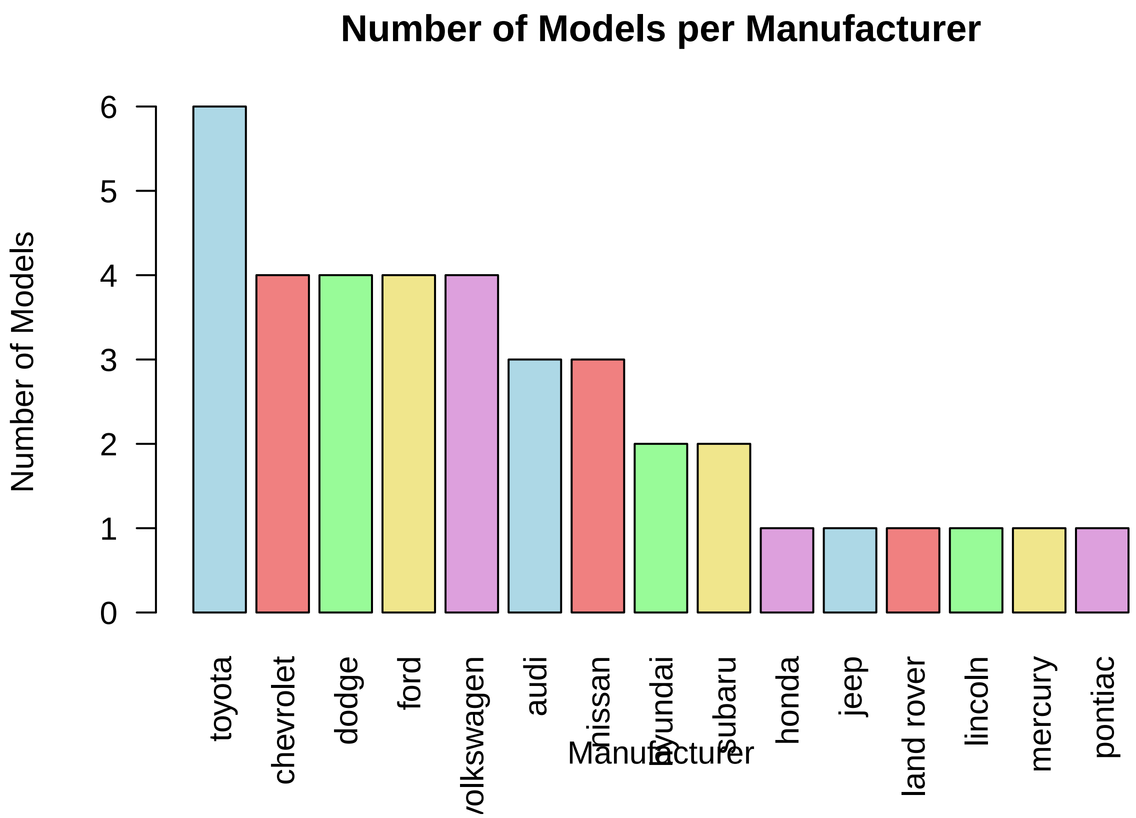
```
## caravan 2wd
##          11
```

The manufacturer that has the most models in this data set is toyota which has 6 models.

The model that has the most variations is the caravan 2wd which has 11C variarions.

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

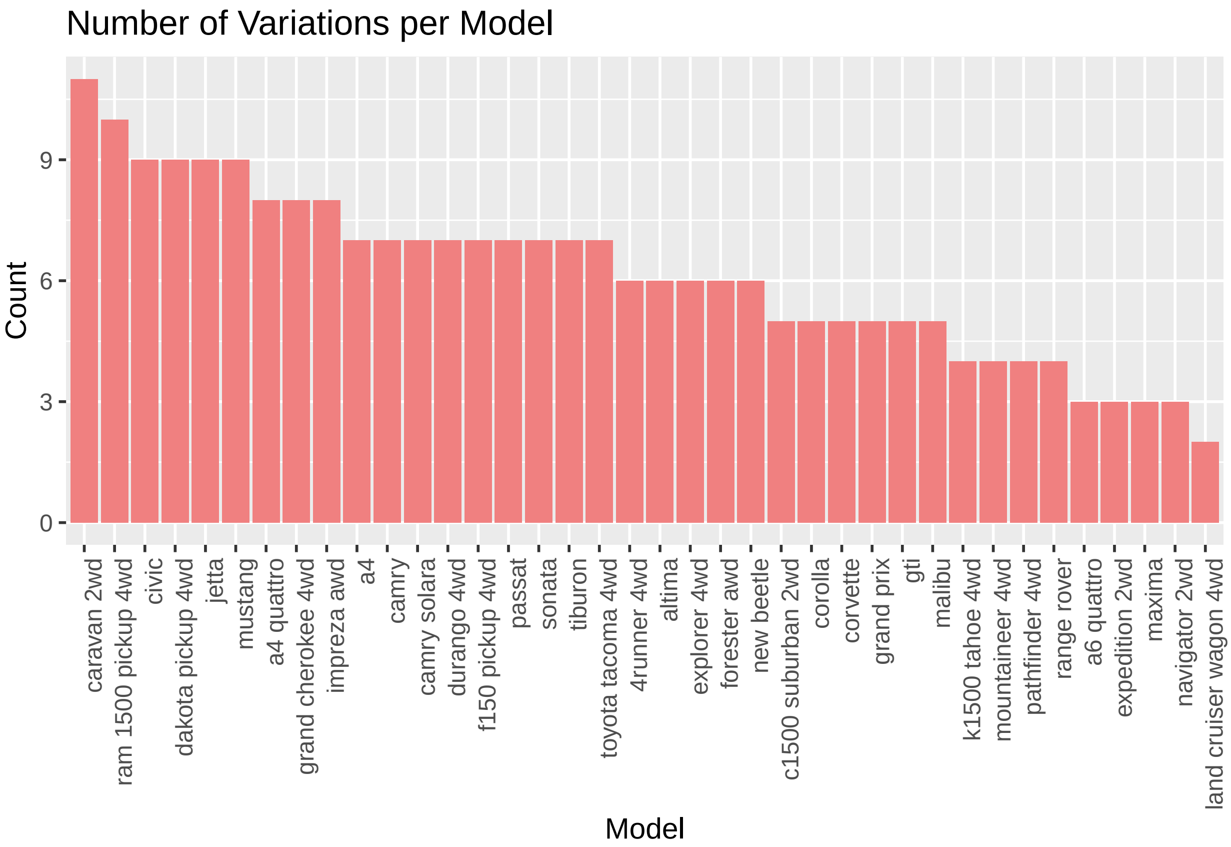
```
library(dplyr)
library(ggplot2)

manufacturer_counts <- setNames(manufacturerModelCount$num_models, manufacturerModelCount$manufacturer)

barplot(manufacturer_counts,
        main = "Number of Models per Manufacturer",
        xlab = "Manufacturer",
        ylab = "Number of Models",
        col = c("lightblue", "lightcoral", "palegreen", "khaki", "plum"),
        las = 2)
```



```r
modelVariationCount <- mpg %>%
  group_by(model) %>%
  summarize(count = n()) %>%
  arrange(desc(count))

print(modelVariationCount)
```

```
## # A tibble: 38 x 2
##    model                 count
##    <chr>                 <int>
##  1 caravan 2wd              11
##  2 ram 1500 pickup 4wd      10
##  3 civic                     9
##  4 dakota pickup 4wd         9
##  5 jetta                     9
##  6 mustang                   9
##  7 a4 quattro                8
##  8 grand cherokee 4wd        8
##  9 impreza awd               8
## 10 a4                        7
## # i 28 more rows
```

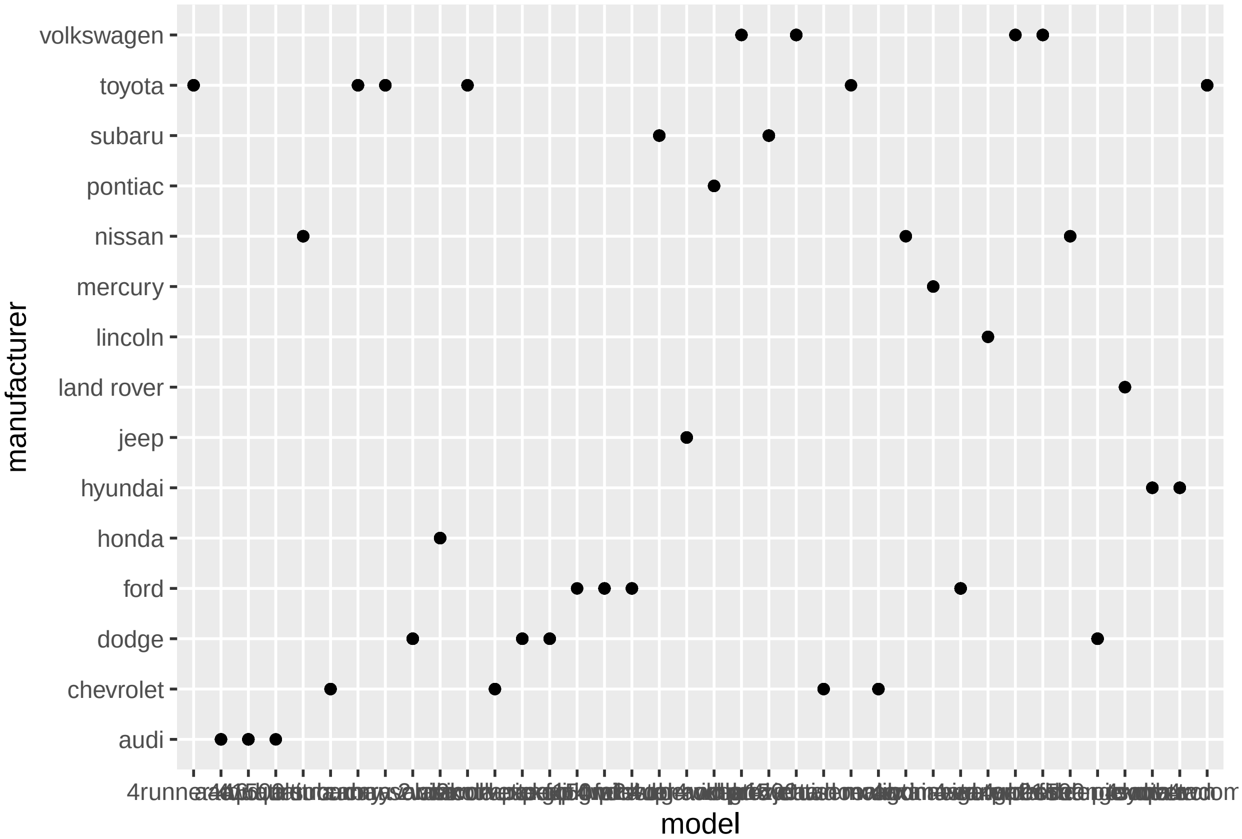
```r
ggplot(modelVariationCount, aes(x = reorder(model, -count), y = count)) +
  geom_bar(stat = "identity", fill = "lightcoral") +
  labs(title = "Number of Variations per Model", x = "Model", y = "Count") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



2. Same dataset will be used. You are going to show the relationship of the modeland the manufacturer.

a. What does ggplot(mpg, aes(model, manufacturer)) + geom_point() show?

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

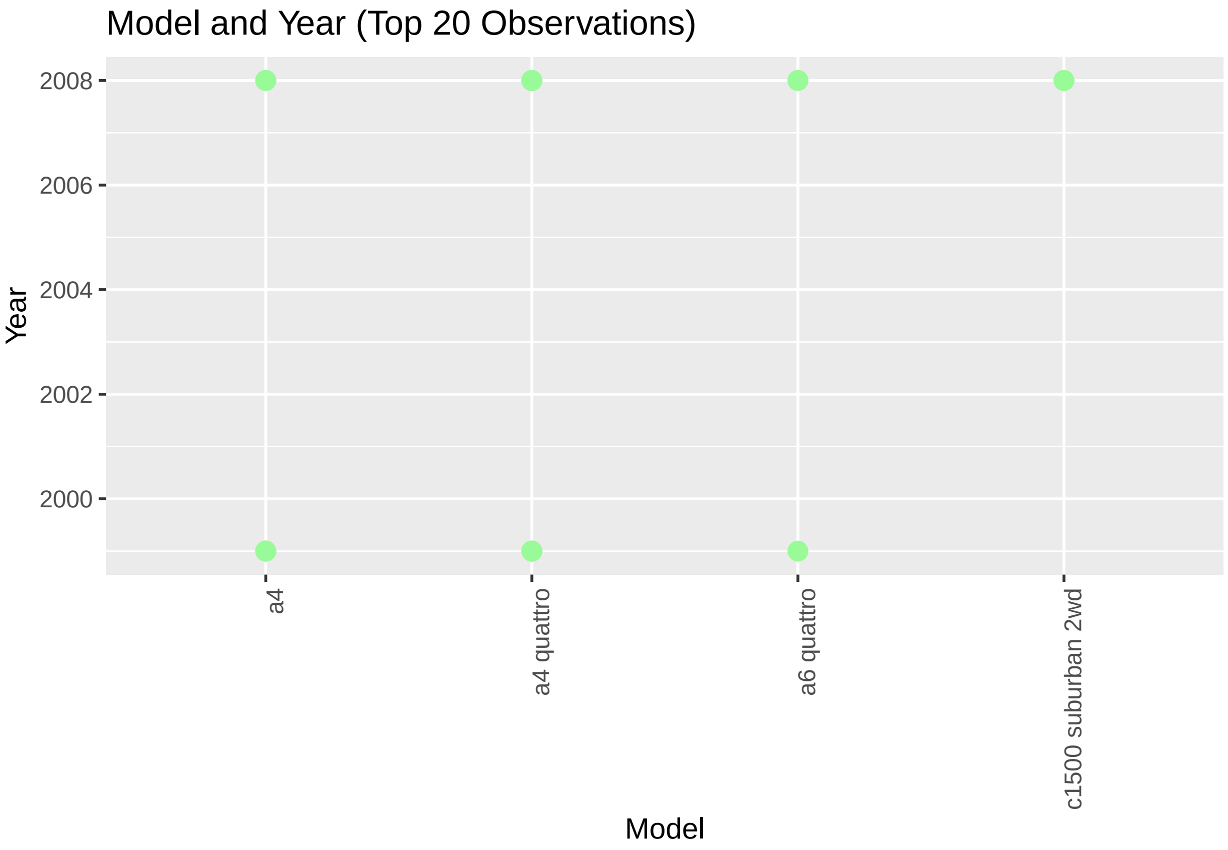


It shows a scatter plot of the mpg models and manufacturers.

b. For you, is it useful? If not, how could you modify the data to make it more informative? For me, it's not that useful due to the visualization of it. It's not clear and the labels are covering each other which is confusing. To make it more informative, i'll change it into a bar graph and fix the labels to make it organized.

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

```r
library(ggplot2)

top20Obs <- mpg[1:20, ]

ggplot(top20Obs, aes(x = model, y = year)) +
  geom_point(color = "palegreen", size = 3) +
  labs(title = "Model and Year (Top 20 Observations)",
       x = "Model",
       y = "Year") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



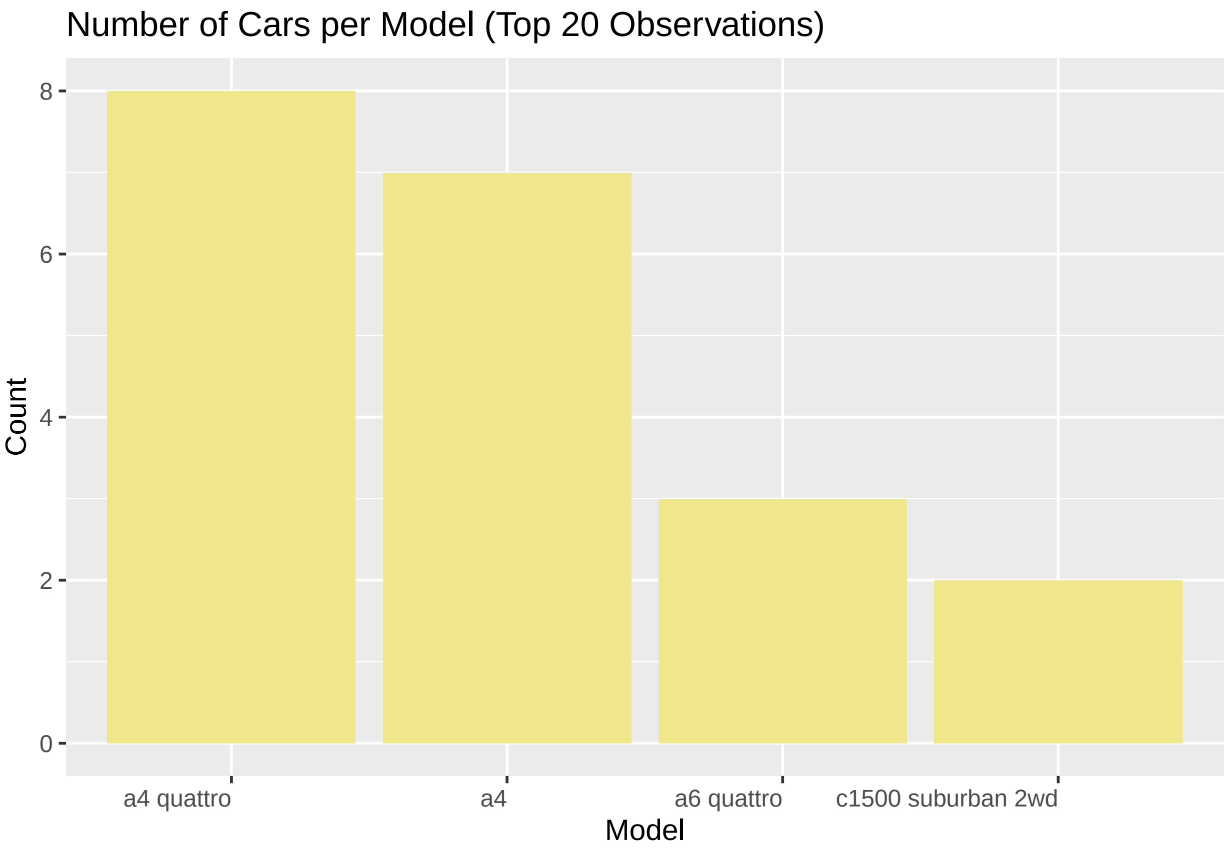
4. Using the pipe (%>%), group the model and get the number of cars per model. Show codes and its result

```
library(dplyr)

carCounts <- mpg %>%
  group_by(model) %>%
  summarise(count = n())

carCounts
```

```
## # A tibble: 38 x 2
##    model              count
##    <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
```

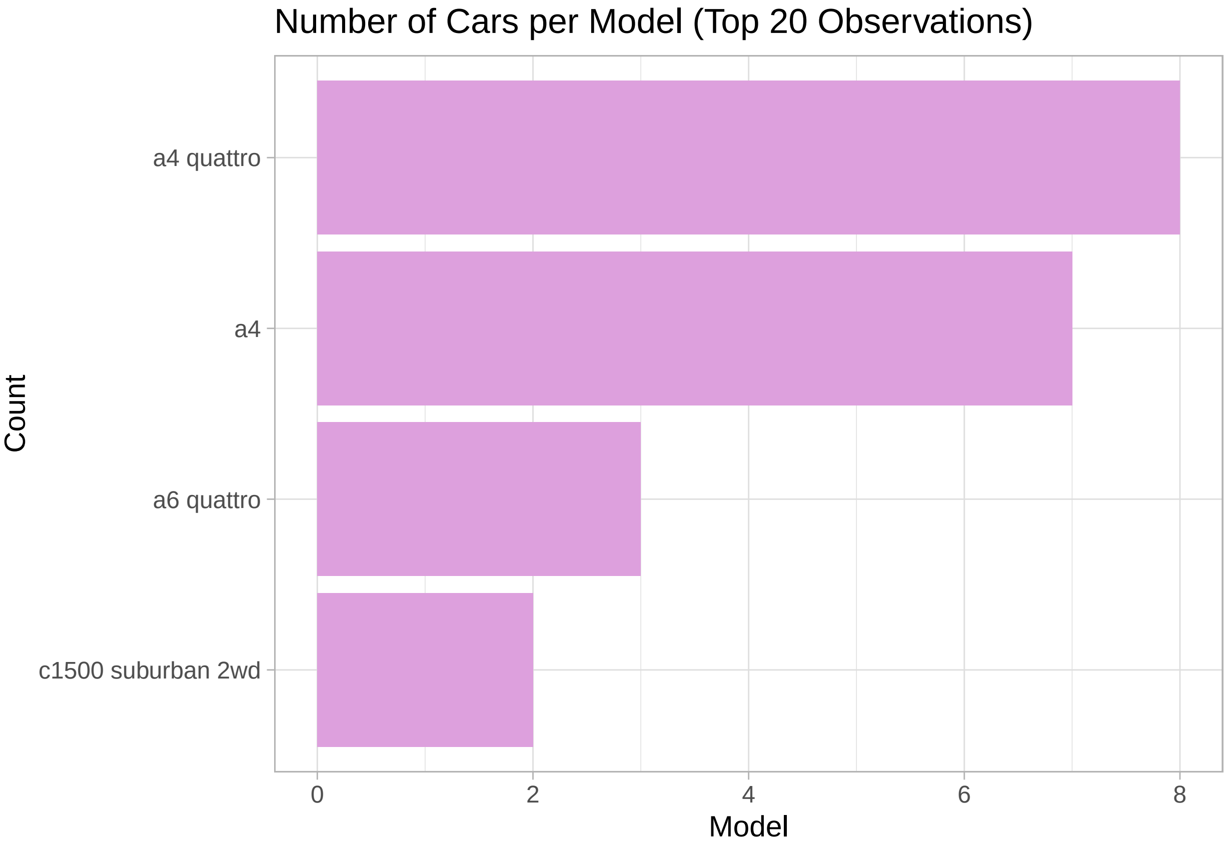
a. Plot using geom_bar() using the top 20 observations only. The graphs shoudl have a title, labels and colors. Show code and results.

```
library(ggplot2)
library(dplyr)

top20Obs <- mpg[1:20, ]

carCounts20 <- top20Obs %>%
  group_by(model) %>%
  summarise(count = n())

ggplot(carCounts20, aes(x = reorder(model, -count), y = count)) +
  geom_bar(stat = "identity", fill = "khaki") +
  labs(title = "Number of Cars per Model (Top 20 Observations)",
       x = "Model",
       y = "Count") +
  theme(axis.text.x = element_text(hjust = 1))
```



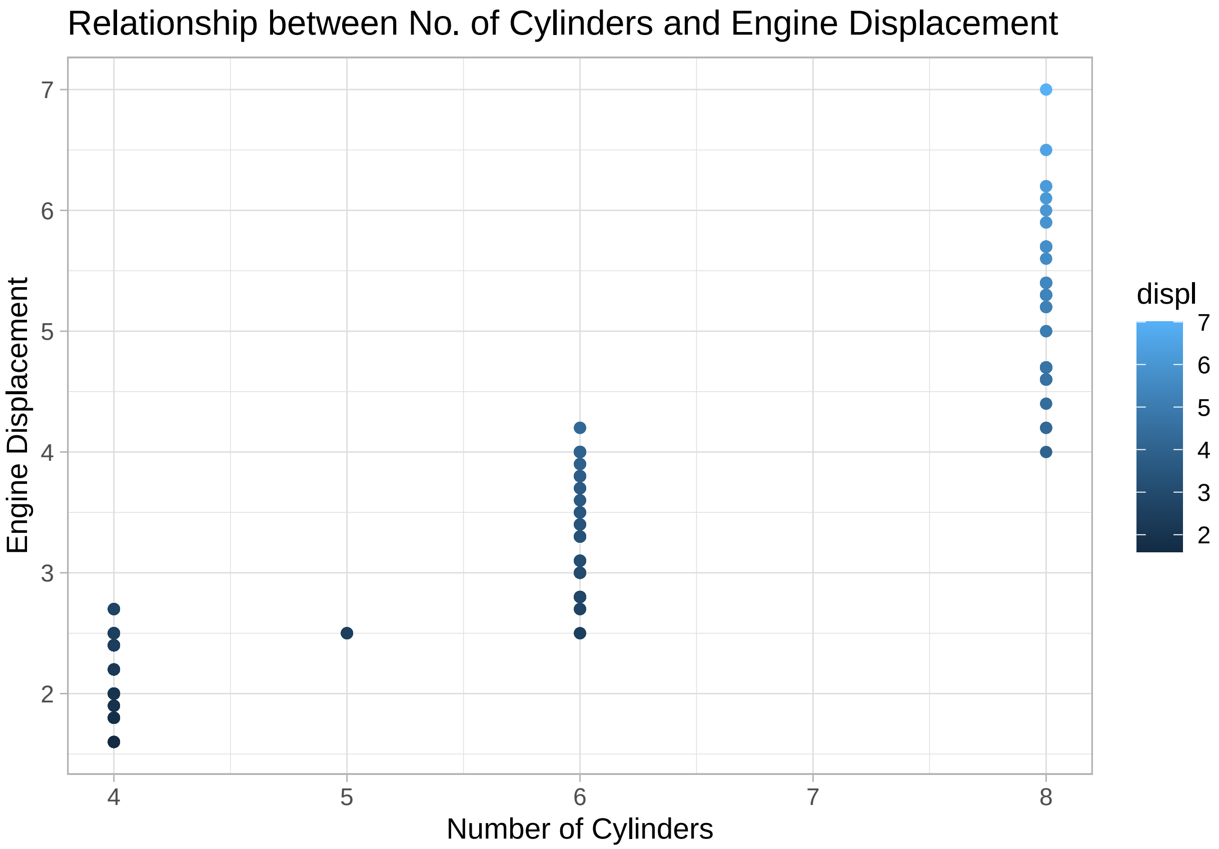
b. Plot using the geom_bar() + coord_flip() just like what is shown below. Show codes and its result.

```
library(ggplot2)
library(dplyr)

top20Obs <- mpg[1:20, ]

carCounts20 <- top20Obs %>%
  group_by(model) %>%
  summarise(count = n())

ggplot(carCounts20, aes(x = reorder(model, count), y = count)) +
  geom_bar(stat = "identity", fill = "plum") +
  coord_flip() +
  labs(title = "Number of Cars per Model (Top 20 Observations)",
       x = "Count",
       y = "Model") +
  theme_light()
```



5. Plot the relationship between cyl - number of cylinders and displ - engine displacement using geom_point with aesthetic color = engine displacement. Title should be "Relationship between No. of Cylinders and Engine Displacement". a. How would you describe its relationship? Show the codes and its result.

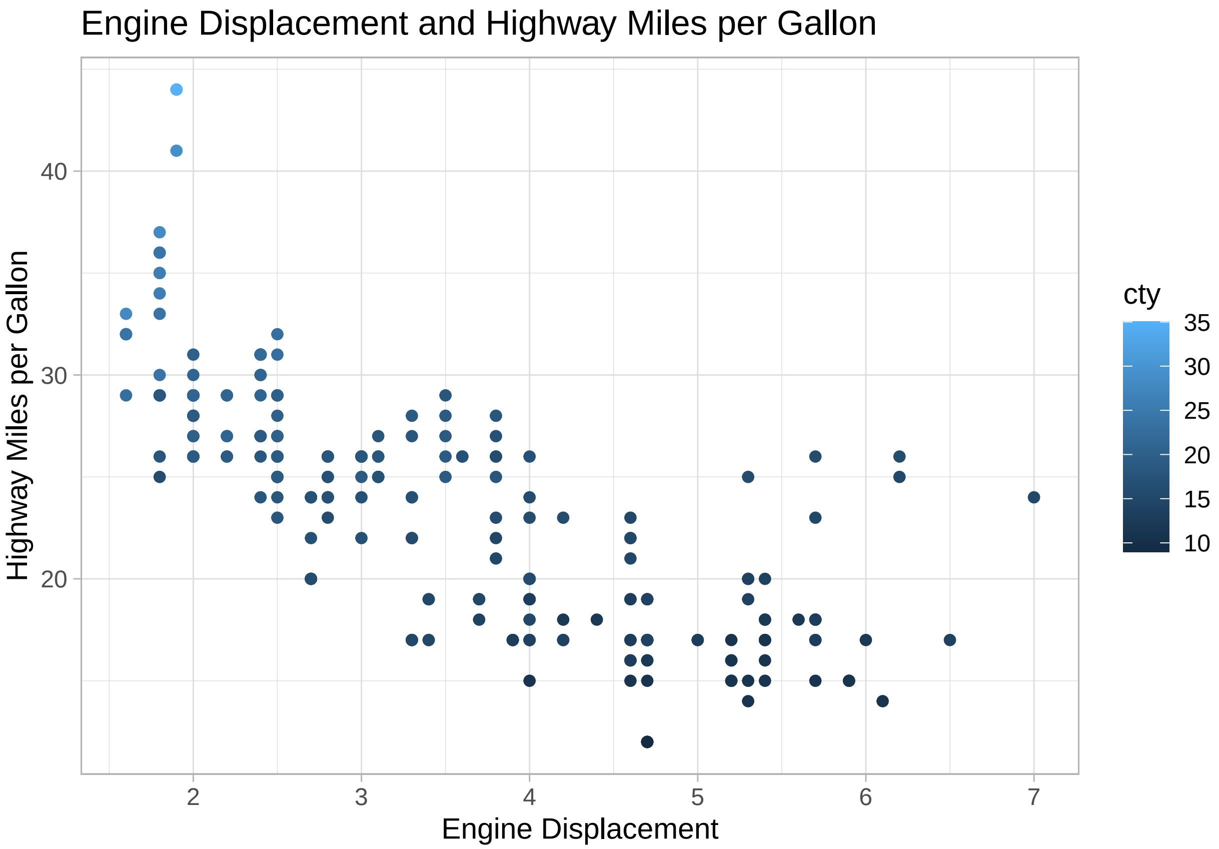
```
library(ggplot2)

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"
  ) +
  theme_light()
```



It shows a positive relationship between cyl and displ. As the number of cylinders increases, the engine displacement generally increases as well.

6. Plot the relationship between displ (engine displacement) and hwy(highway miles per gallon). Mapped it with a continuous variable you have identified in #1-c. What is its result? Why it produced such output?

```
library(ggplot2)

ggplot(mpg, aes(x = displ, y = hwy, color = cty)) +
  geom_point() +
  labs(
    title = "Engine Displacement and Highway Miles per Gallon",
    x = "Engine Displacement",
    y = "Highway Miles per Gallon"
  ) +
  theme_light()
```



The result shows a negative relationship between hwy and displ. As the highway miles per gallon increases, the engine displacement decreases. That is because vehicles with larger engines usually consume more fuel to generate more power, resulting in lower fuel efficiency.

6. Import the traffic.csv onto your R environment.

a. How many numbers of observation does it have? What are the variables of the traffic dataset the Show your answer.

```
TrafficData <- read.csv("traffic.csv")

str(TrafficData)
```

```
## 'data.frame':    48120 obs. of  4 variables:
##  $ DateTime: chr  "2015-11-01 00:00:00" "2015-11-01 01:00:00" "2015-11-01 02:00:00" "2015-11-01 03:0
##  $ Junction: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Vehicles: int  15 13 10 7 9 6 9 8 11 12 ...
##  $ ID      : num  2.02e+10 2.02e+10 2.02e+10 2.02e+10 2.02e+10 ...
```

The traffic dataset has a total of 48120 observations or rows. And 4 variables or columns which are DateTime, Junction, Vehicles, and ID.

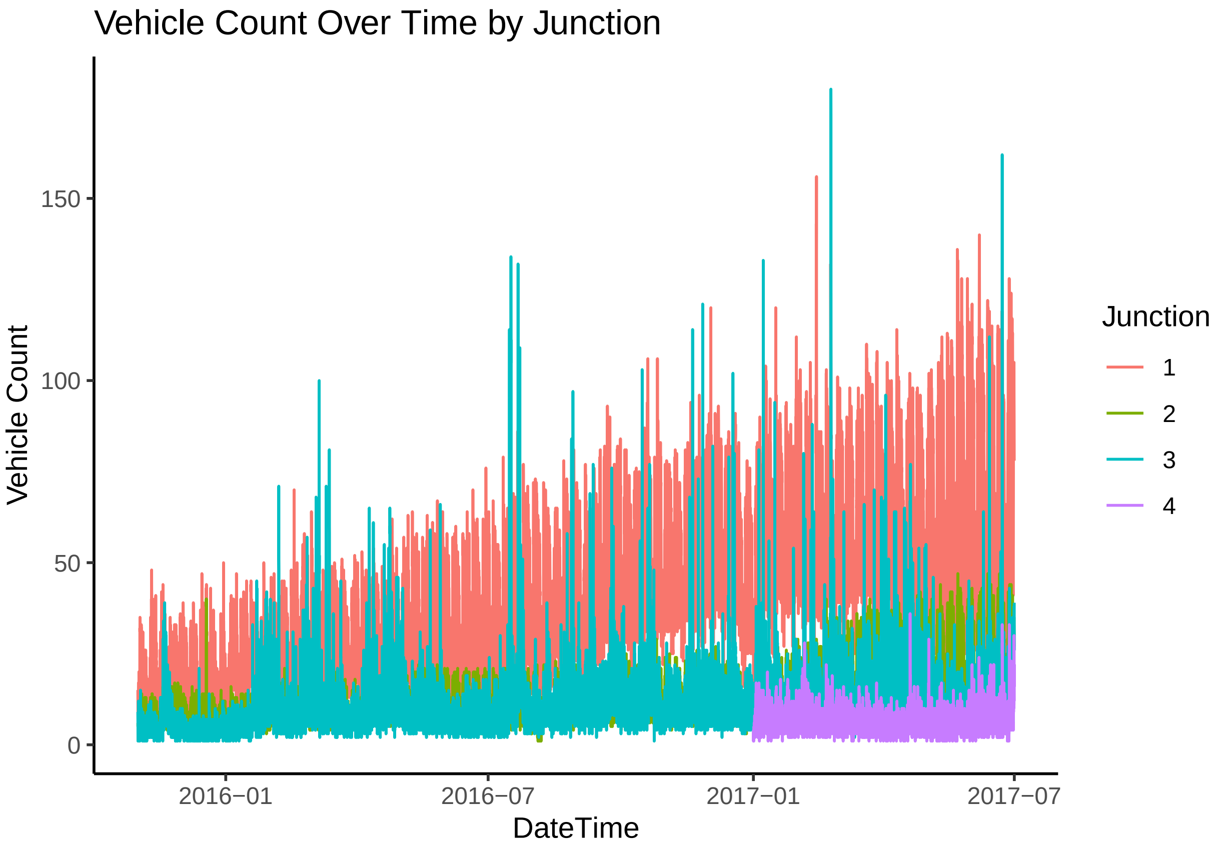
b. subset the traffic dataset into junctions. What is the R codes and its output?

```
trafficJunction <- TrafficData$Junction
```

c. Plot each junction in a using geom_line(). Show your solution and output.

```
library(ggplot2)
library(dplyr)
```

```
trafficJunctionPlot <- TrafficData %>% select(DateTime, Junction, Vehicles)

trafficJunctionPlot$DateTime <- as.POSIXct(trafficJunctionPlot$DateTime, format="%Y-%m-%d %H:%M:%S")

ggplot(trafficJunctionPlot, aes(x = DateTime, y = Vehicles, color = factor(Junction))) +
  geom_line() +
  labs(title = "Vehicle Count Over Time by Junction",
       x = "DateTime",
       y = "Vehicle Count",
       color = "Junction") +
  theme_classic()
```



7. From alexa_file.xlsx, import it to your environment

a. How many observations does alexa_file has? What about the number of columns? Show your solution and answer.

```
library(readxl)
alexaData <- read_xlsx("alexa_file.xlsx")
str(alexaData)
```

```
## tibble [3,150 x 5] (S3: tbl_df/tbl/data.frame)
##  $ rating          : num [1:3150] 5 5 4 5 5 5 3 5 5 5 ...
##  $ date            : POSIXct[1:3150], format: "2018-07-31" "2018-07-31" ...
##  $ variation       : chr [1:3150] "Charcoal Fabric" "Charcoal Fabric" "Walnut Finish" "Charcoal Fabr
##  $ verified_reviews: chr [1:3150] "Love my Echo!" "Loved it!" "Sometimes while playing a game, you ca
##  $ feedback        : num [1:3150] 1 1 1 1 1 1 1 1 1 1 ...
```

Alexa file has a total of 3150 observations or rows and 5 variables or columns which are rating, date, variation, verified_reviews, and feedback.

b. group the variations and get the total of each variations. Use dplyr package. Show solution and answer.

```
allVariations <- alexaData %>%
  group_by(variation) %>%
   summarise(Total = n())

allVariations
```

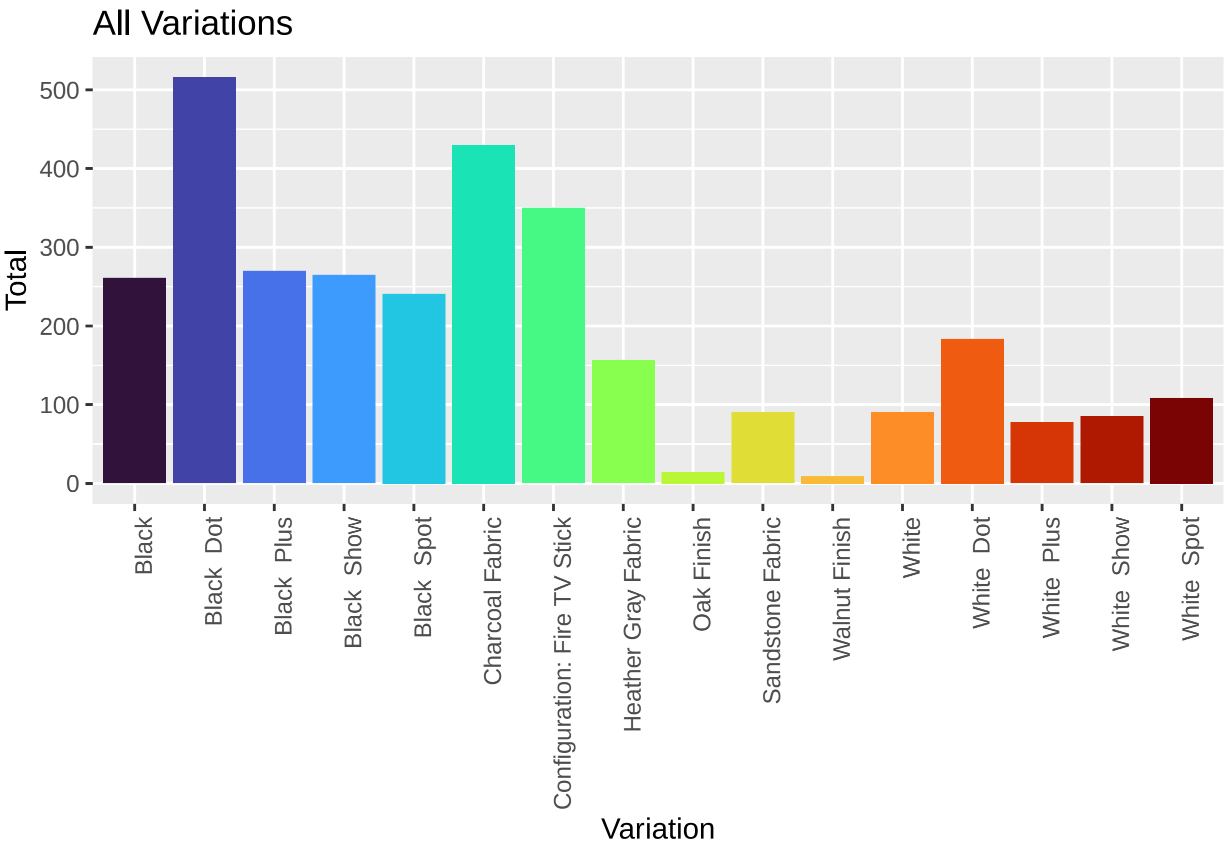
```
## # A tibble: 16 x 2
##    variation                    Total
##    <chr>                        <int>
##  1 Black                          261
##  2 Black  Dot                     516
##  3 Black  Plus                    270
##  4 Black  Show                    265
##  5 Black  Spot                    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
```

c. Plot the variations using the ggplot() function. What did you observe? Complete the details of the graph. Show solution and answer.

```
library(viridis)
```

```
## Loading required package: viridisLite
```

```
library(ggplot2)

ggplot(allVariations, aes(x = variation, y = Total, fill = variation)) +
  geom_bar(stat = "identity") +
  labs(title = "All Variations",
       x = "Variation",
       y = "Total") +
       theme(legend.position = "none") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  scale_fill_viridis_d(option = "turbo")
```



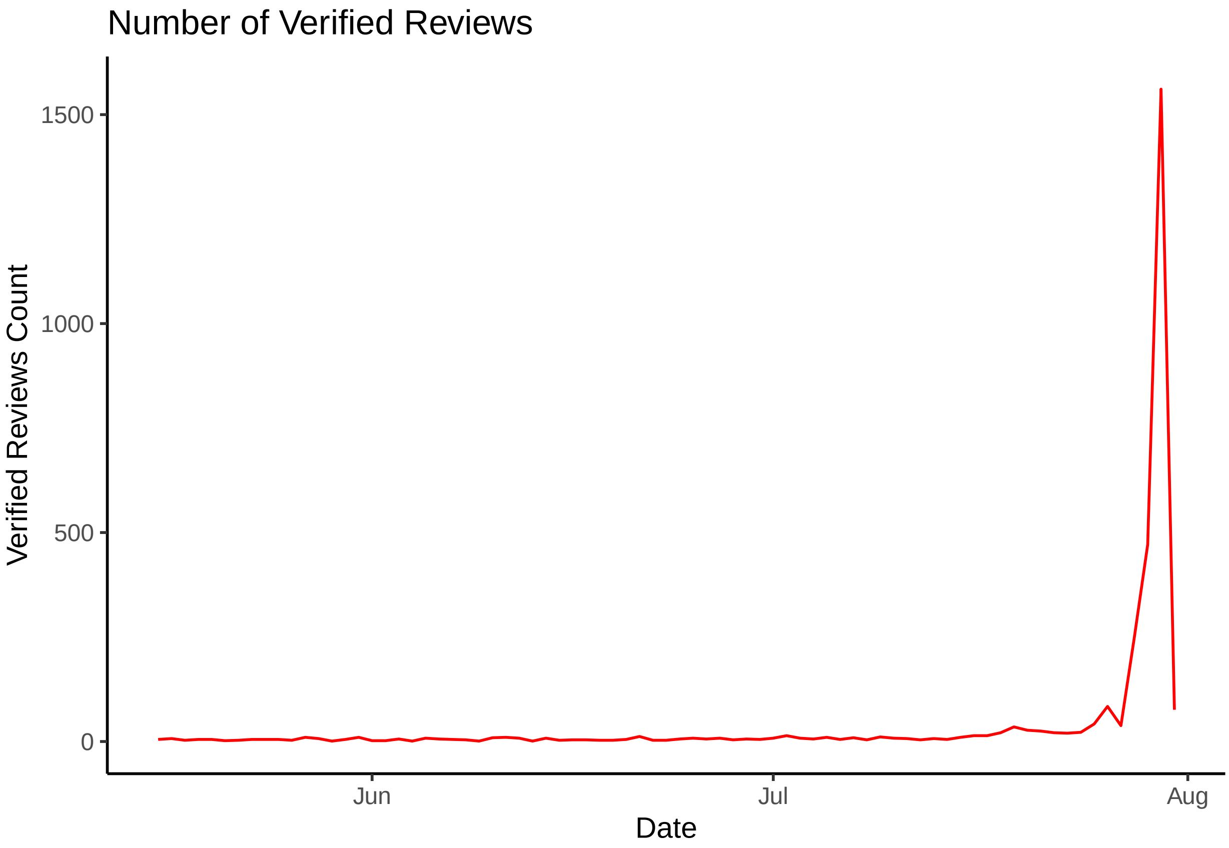
I observed that the variation that has the most total is the Black Dot followed by Charcoal Fabric and Fire TV Stick. The variation with the least total or less common is the Walnut Finish.

d. Plot a geom_line() with the date and the number of verified reviews. Complete the details of the graphs. Show your answer and solution.

```
library(ggplot2)
library(dplyr)

dailyReviews <- alexaData %>%
  filter(!is.na(verified_reviews)) %>%
  group_by(date) %>%
  summarise(verifiedReviewsCount = n())

dailyReviews
```

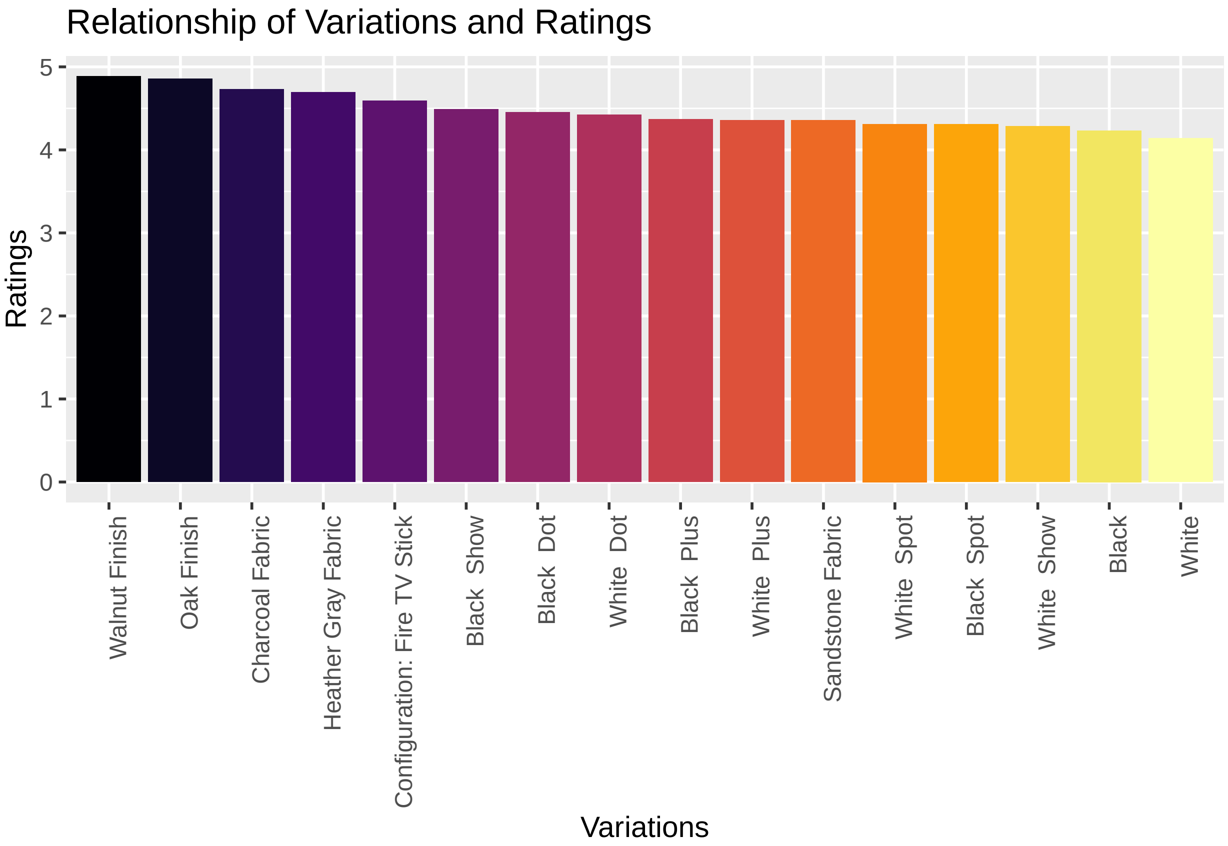
```
## # A tibble: 77 x 2
##    date                verifiedReviewsCount
##    <dttm>                             <int>
##  1 2018-05-16 00:00:00                    5
##  2 2018-05-17 00:00:00                    7
##  3 2018-05-18 00:00:00                    3
##  4 2018-05-19 00:00:00                    5
##  5 2018-05-20 00:00:00                    5
##  6 2018-05-21 00:00:00                    2
##  7 2018-05-22 00:00:00                    3
##  8 2018-05-23 00:00:00                    5
##  9 2018-05-24 00:00:00                    5
## 10 2018-05-25 00:00:00                    5
## # i 67 more rows
```

```
ggplot(dailyReviews, aes(x = date, y = verifiedReviewsCount)) +
  geom_line(color = "red") +
  labs(title = "Number of Verified Reviews",
       x = "Date",
       y = "Verified Reviews Count") +
  theme_classic()
```



e. Get the relationship of variations and ratings. Which variations got the most highest in rating? Plot a graph to show its relationship. Show your solution and answer.

```
library(ggplot2)
library(dplyr)
library(viridis)
library(forcats)

averageRatings <- alexaData %>%
  group_by(variation) %>%
  summarise(avgRating = mean(rating))

averageRatings <- averageRatings %>%
  mutate(variation = fct_reorder(variation, avgRating, .desc = TRUE))

ggplot(averageRatings, aes(x = variation, y = avgRating, fill = variation)) +
  geom_bar(stat = "identity") +
  labs(
    title = "Relationship of Variations and Ratings",
    x = "Variations",
    y = "Ratings"
  ) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  theme(legend.position = "none") +
  scale_fill_viridis_d(option = "inferno")
```



The top 5 variations that got the most highest in ratings are Walnut Finish, Oak Finish, Charcoal Fabric, Heather Gray Fabric, and Fire TV Stick.