RWorksheet_Perez#4c

2024-11-04

1. Use the dataset mpg

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

```
mpg <- read.csv("/cloud/project/mpg.csv")
str(mpg)
```

```
## 'data.frame':    234 obs. of  12 variables:
##  $ X           : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ 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 manufacturer, model, trans, drv, fl, and class are all categorical.

c. Which are continuous variables?

The variables displ, year, cty, and hwy are all continuous.

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.

```
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
```

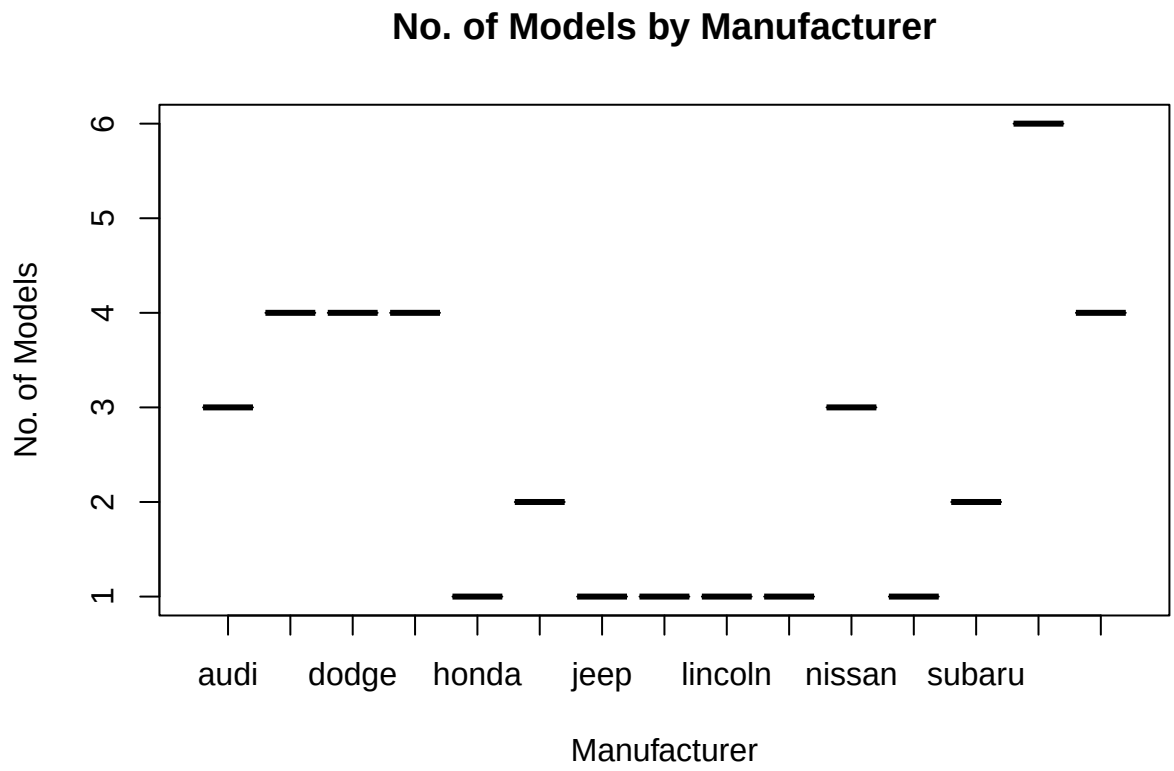
```
manufacturerModels <- mpg %>% group_by(manufacturer) %>% summarize(num_models = n_distinct(model)) %>% a
manufacturerModels
```

```
## # 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
```

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

```
manufacturerModels$manufacturer <- as.factor(manufacturerModels$manufacturer)
plot(manufacturerModels$manufacturer, manufacturerModels$num_models, main = "No. of Models by Manufactu
```

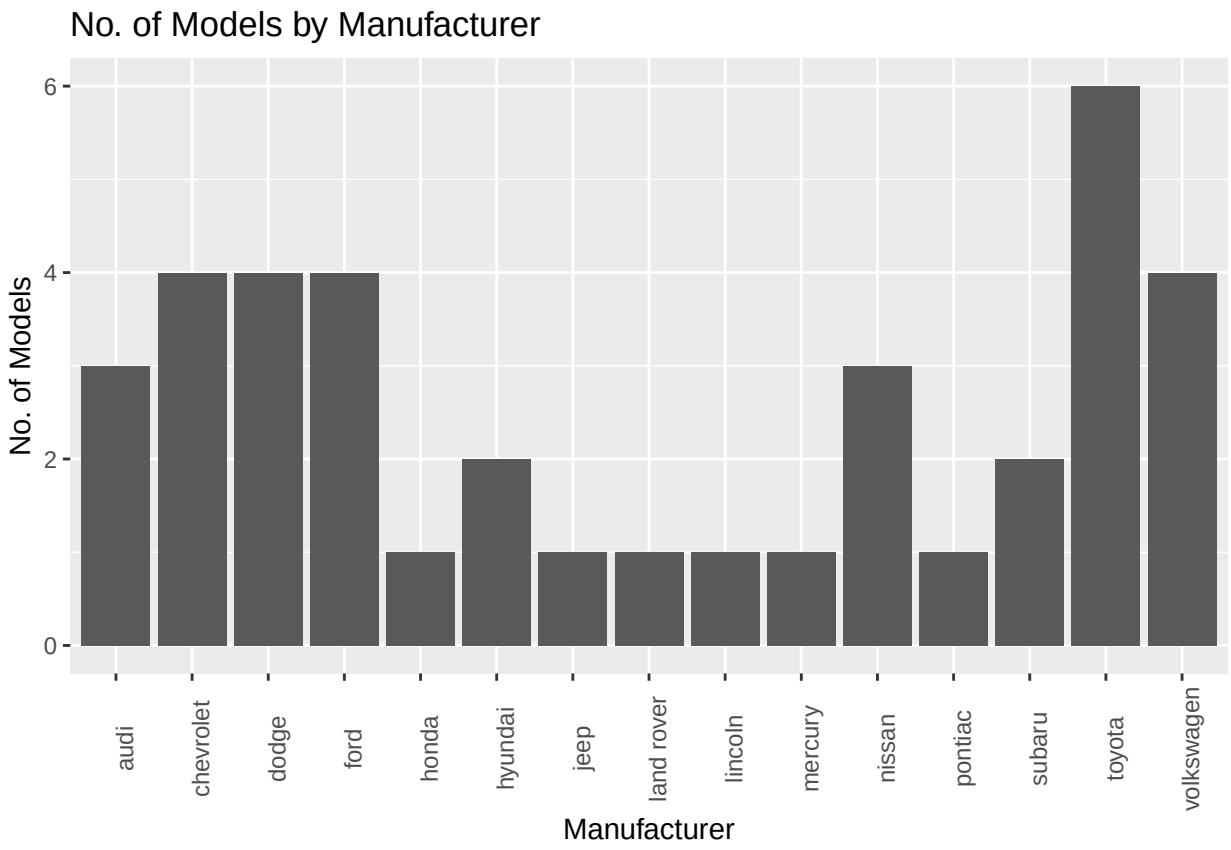



```
library(ggplot2)
```

```
##
## Attaching package: 'ggplot2'
```

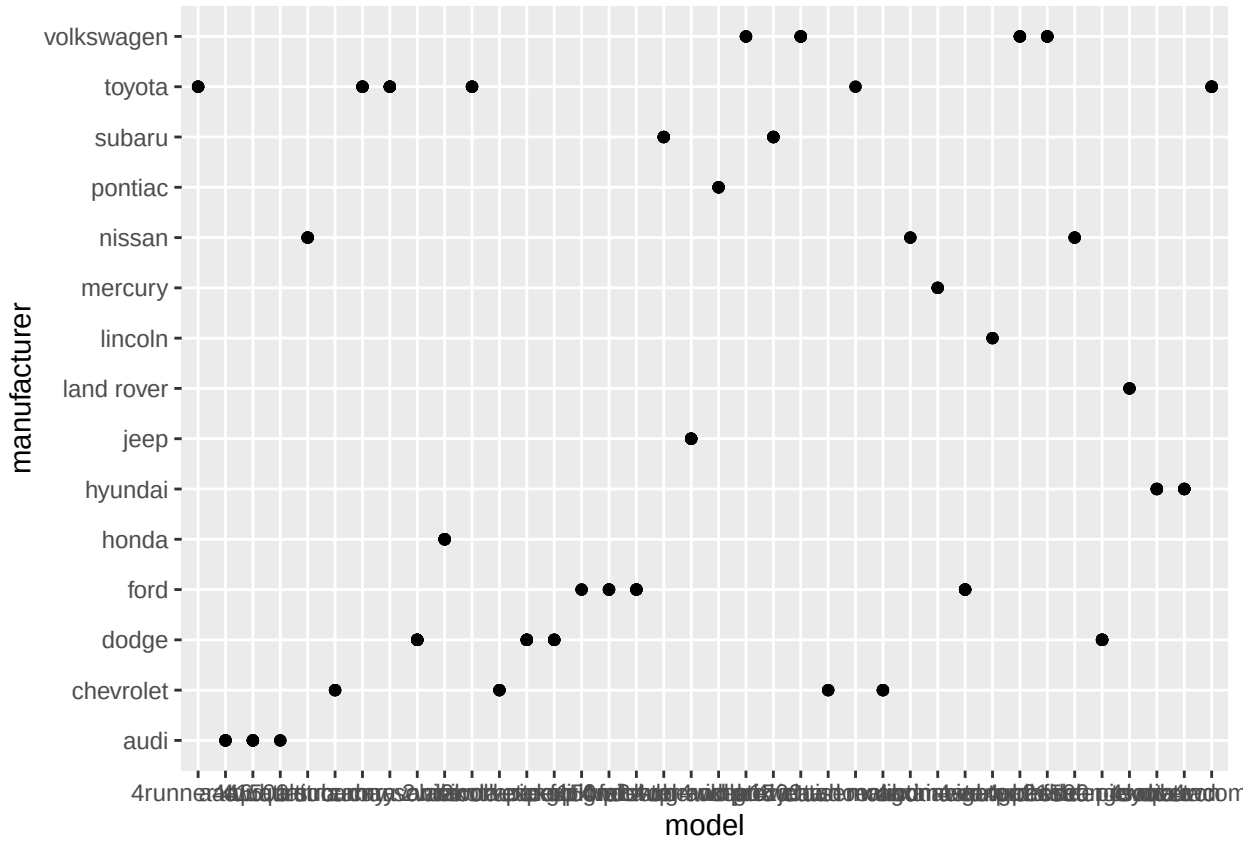
```
## The following object is masked _by_ '.GlobalEnv':
##
##     mpg
```

```r
ggplot(manufacturerModels, aes(x = manufacturer, y = num_models)) +
geom_bar(stat = "identity") +
theme(axis.text.x = element_text(angle = 90)) +
ggtitle("No. of Models by Manufacturer") +
xlab("Manufacturer") +
ylab("No. of Models")
```



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

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



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

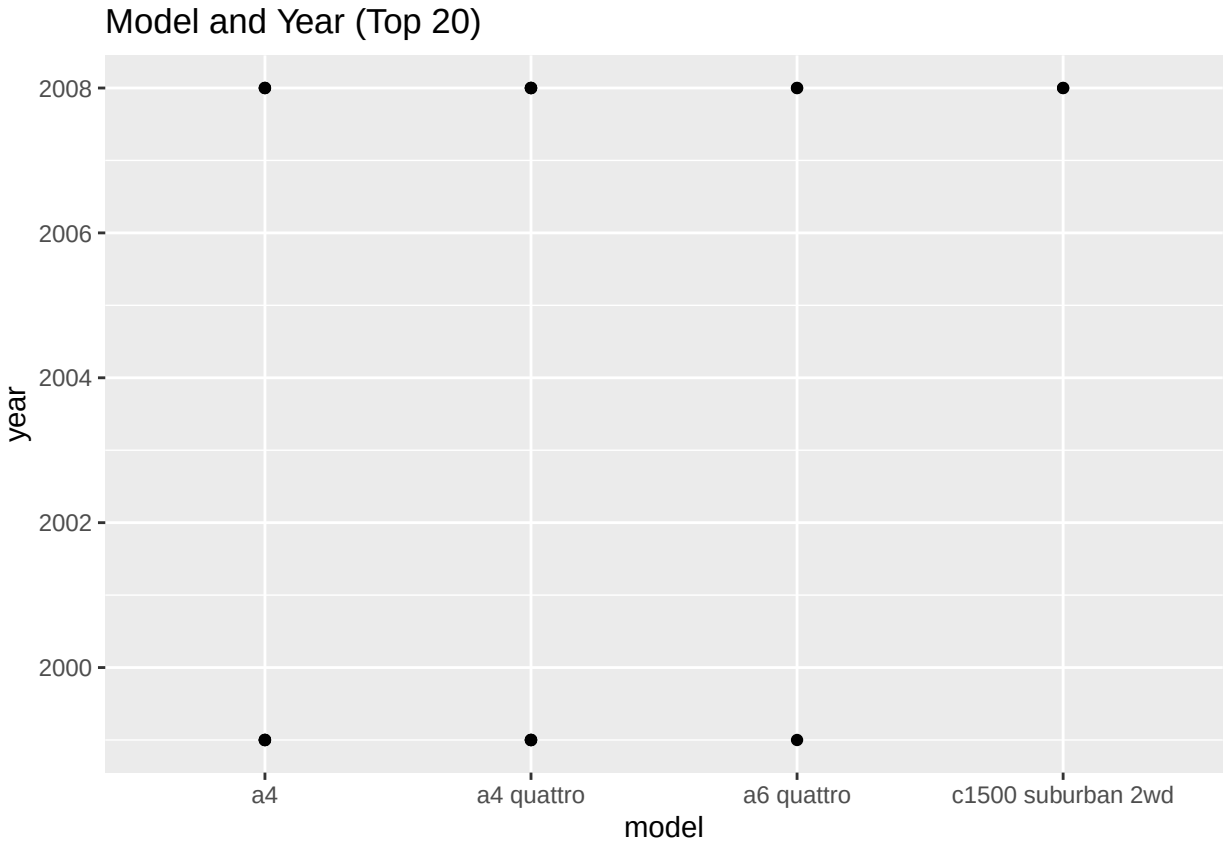
It shows the distribution of each car model across different manufacturers.

b. For you, is it useful? If not, how could you modify the data to make it more informative?

The graph is useful, but it can be better. By summarizing data or adding colors and legends to represent the categories, it can be made more informative.

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

```
top20 <- head(mpg, 20)
ggplot(top20, aes(x = model, y = year)) + geom_point() + ggtitle("Model and Year (Top 20)")
```



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

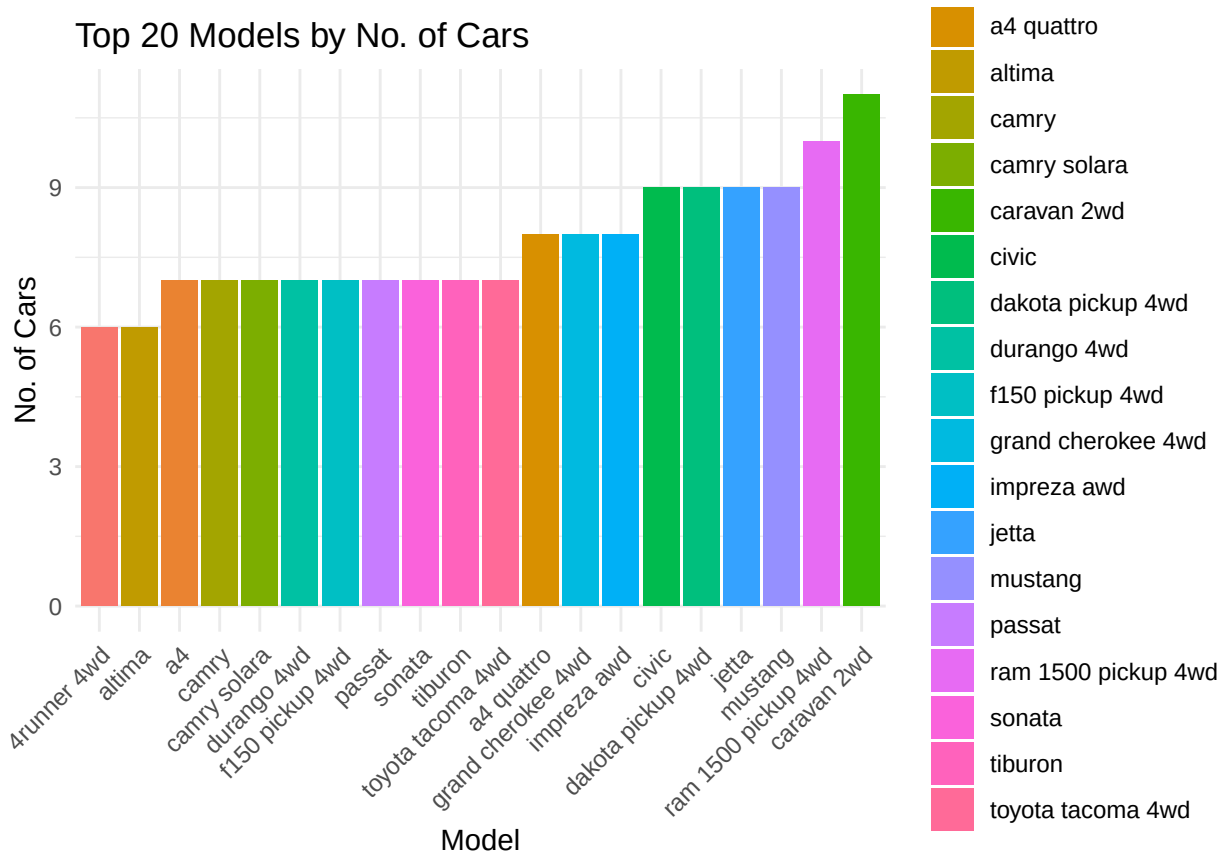
```
modelCount <- mpg %>% group_by(model) %>% summarize(count = n()) %>% arrange(desc(count))
modelCount
```

```
## # 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
```

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

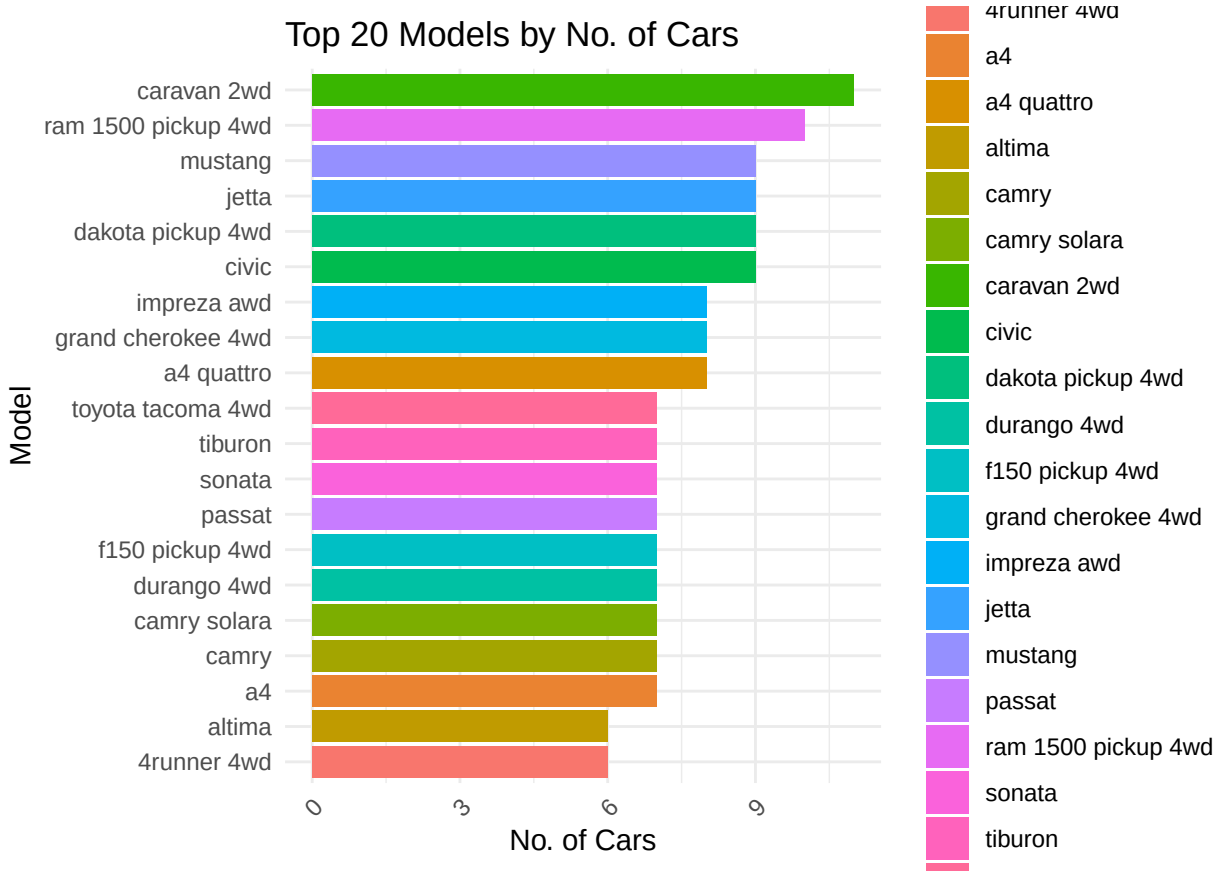
```
top20Models <- modelCount %>% head(20)

ggplot(top20Models, aes(x = reorder(model, count), y = count, fill = model)) +
  geom_bar(stat="identity") +
  labs(title = "Top 20 Models by No. of Cars", x = "Model", y = "No. of Cars") +
```

```
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_viridis_d(aesthetics = "lightpink")
```



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

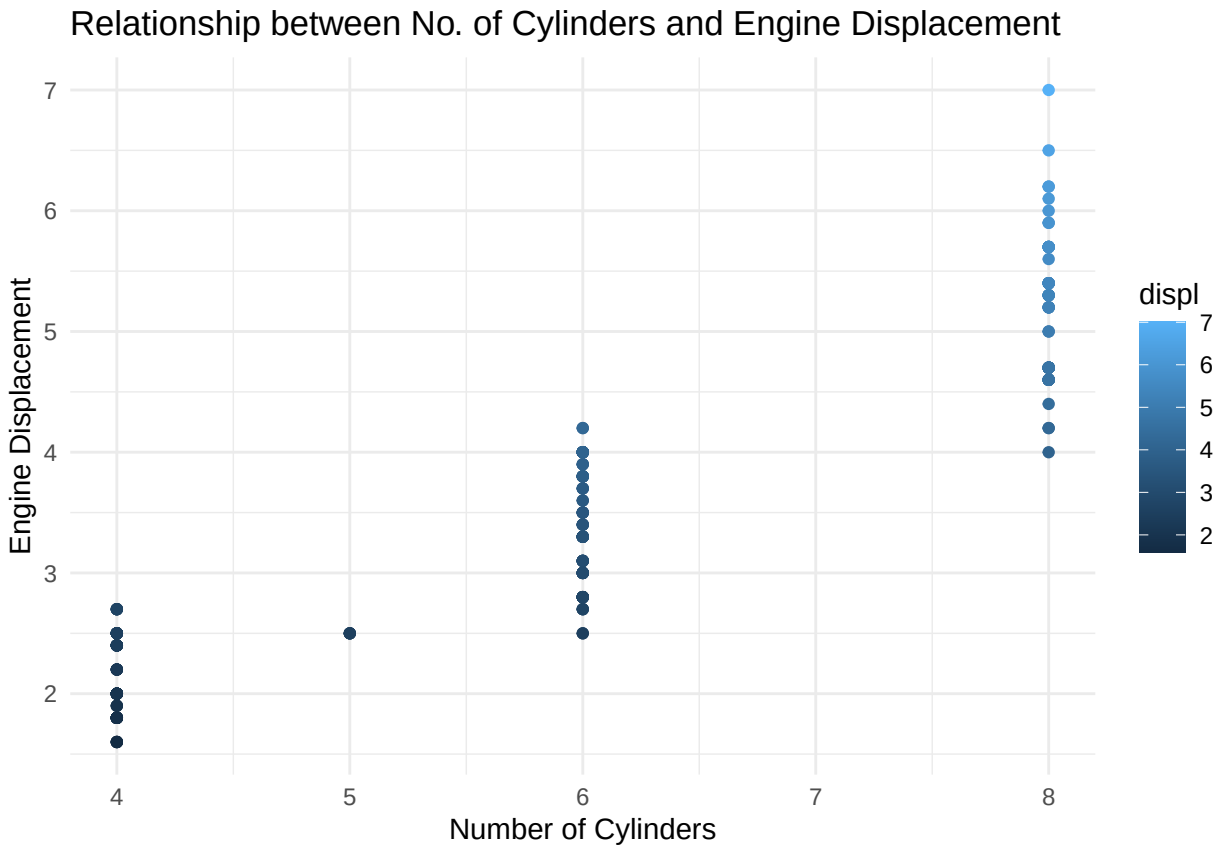
```
ggplot(top20Models, aes(x = reorder(model, count), y = count, fill = model)) +
  geom_bar(stat="identity") +
  labs(title = "Top 20 Models by No. of Cars", x = "Model", y = "No. of Cars") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_viridis_d(aesthetics = "lightpink") +
  coord_flip()
```



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.

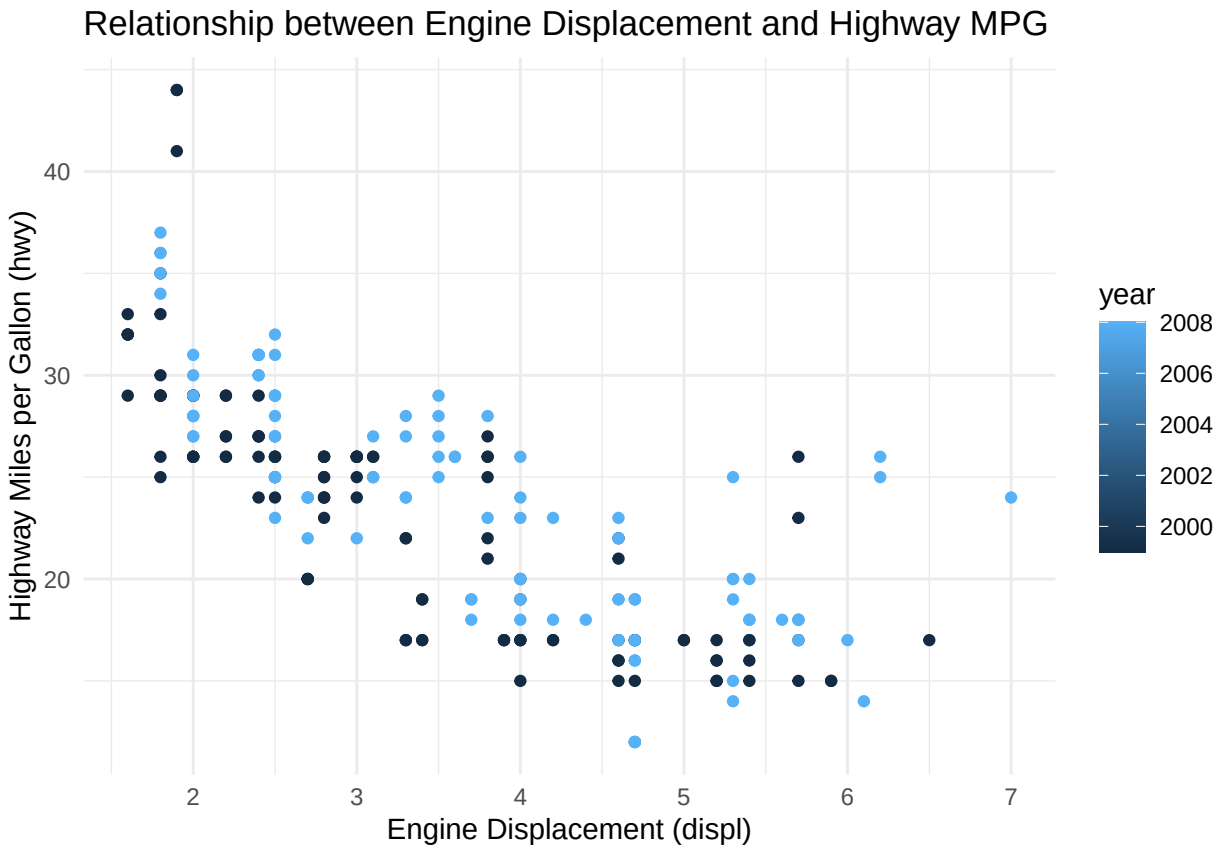
```
ggplot(mpg, aes(x = cyl, y = displ, color = displ)) +
  geom_point() +
  labs(title = "Relationship between No. of Cylinders and Engine Displacement",
    x = "Number of Cylinders",
    y = "Engine Displacement"
  ) +
  theme_minimal()
```



As its displacement increases, so does the number of cylinders.

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?

```
ggplot(mpg, aes(x = displ, y = hwy, color = year)) +
  geom_point() +
  labs(
    title = "Relationship between Engine Displacement and Highway MPG",
    x = "Engine Displacement (displ)",
    y = "Highway Miles per Gallon (hwy)"
  ) +
  theme_minimal()
```



The result displays a scatter plot of engine displacement versus highway miles per gallon, with points colored according to the car's manufacturing year.

As engine displacement (displ) increases, fuel efficiency (hwy) tends to decrease, thus presenting us with a downward slope of the points. Mapping the year variable to color may reveal slight trends in fuel efficiency improvements over time.

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? Show your answer.

```
traffic <- read.csv("/cloud/project/traffic.csv")
str(traffic)
```

```
## '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 ...
```

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

```
junction <- subset(traffic, select = Junction)
head(junction)
```

```
##   Junction
## 1        1
## 2        1
## 3        1
```

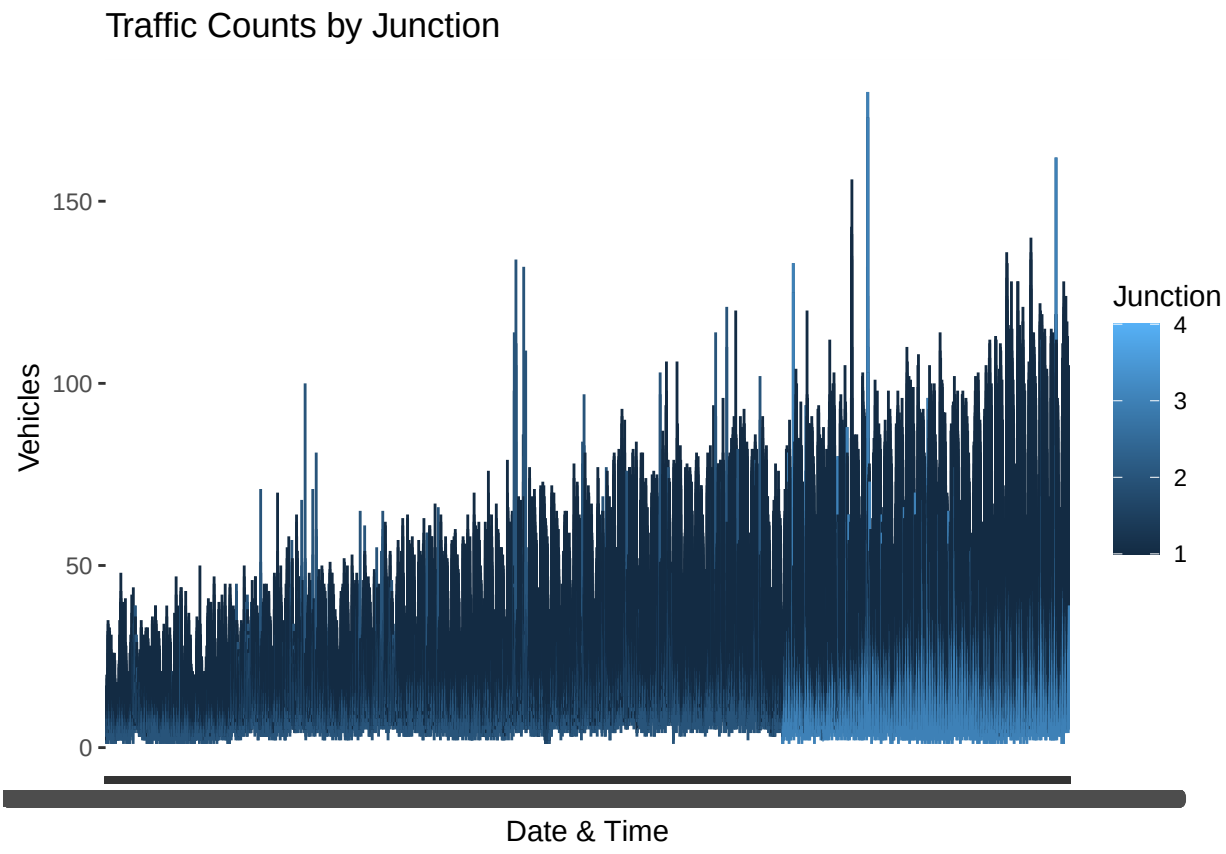
```
## 4         1
## 5         1
## 6         1
```

```
tail(junction)
```

```
##       Junction
## 48115        4
## 48116        4
## 48117        4
## 48118        4
## 48119        4
## 48120        4
```

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

```
library(ggplot2)
ggplot(traffic, aes(x = DateTime, y = Vehicles, color = Junction)) +
  geom_line() +
  labs(title = "Traffic Counts by Junction", x = "Date & Time", y = "Vehicles")
```



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

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

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

```r
nrow(alexa)
```

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

```r
ncol(alexa)
```

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

The alexa_file has a total of 3150 observations and 5 columns.

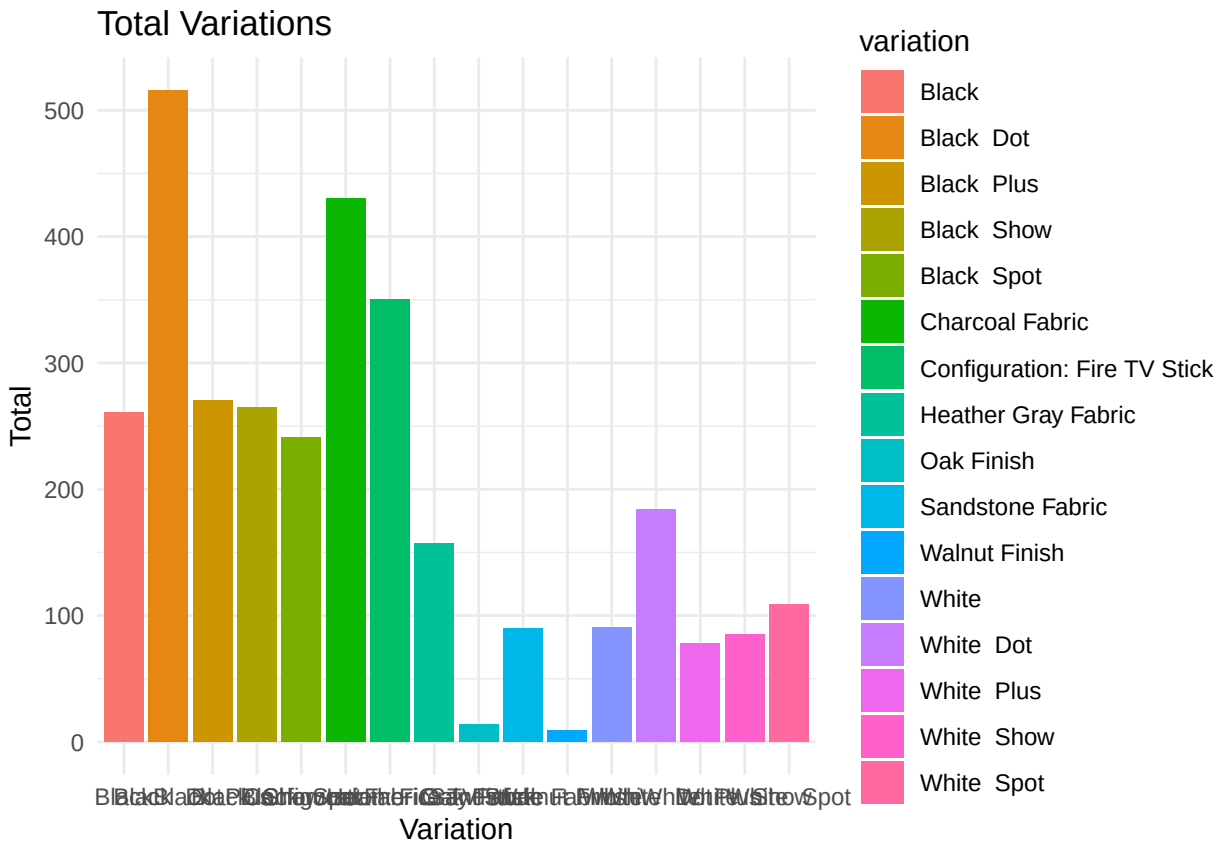
b. group the variations and get the total of each variations. Use dplyr package. Show solution and answer.

```r
library(dplyr)
variationTotal <- alexa %>%
  group_by(variation) %>%
  summarize(total = n())
print(variationTotal)
```

```
## # 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.

```r
ggplot(variationTotal, aes(x = variation, y = total, fill = variation)) +
  geom_bar(stat = "identity") +
  labs(title = "Total Variations", x = "Variation", y = "Total") +
  theme_minimal()
```

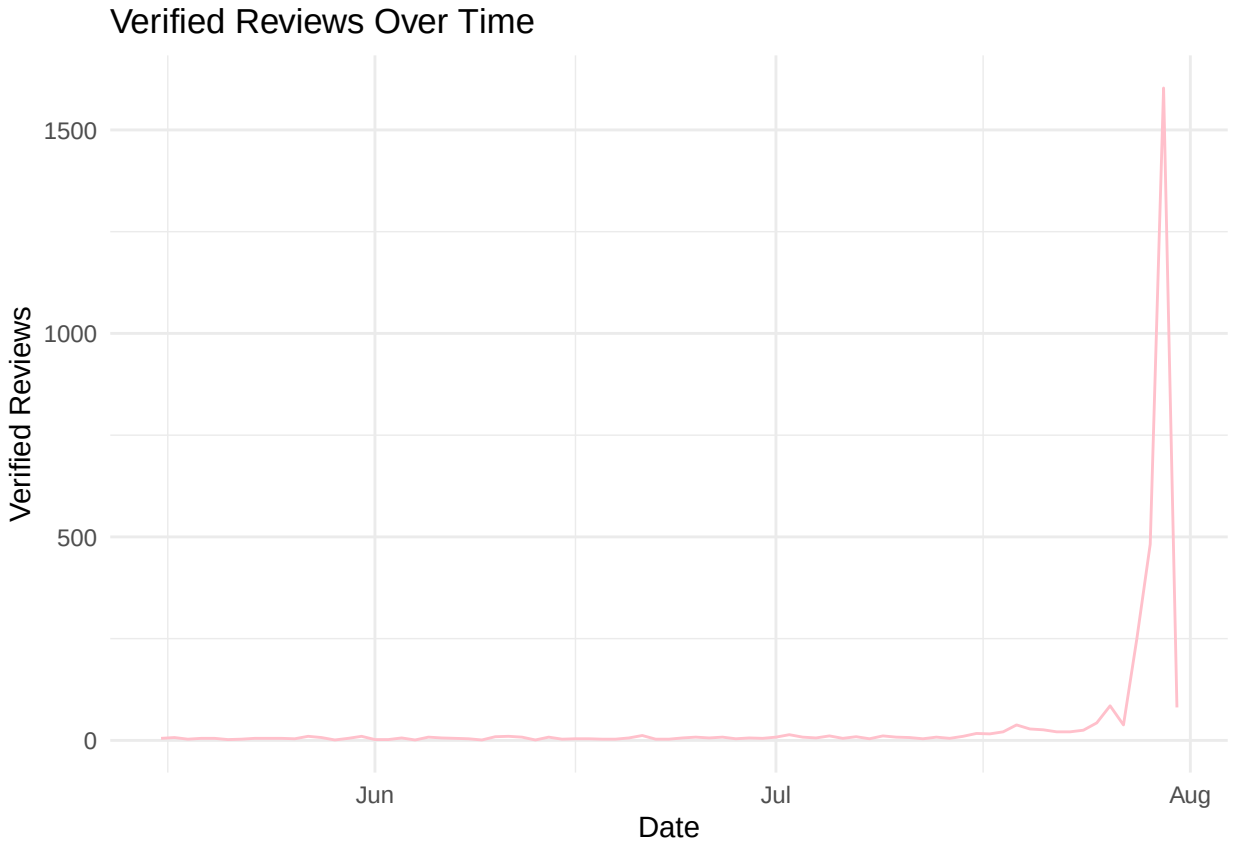


The chart displays the total counts of various "Variations", revealing that some variations are significantly more common than others, as shown by the taller bars. Black Dot, represented by the tallest orange bar, has over 500 instances, indicating a strong preference or popularity. Overall, there is a clear difference in counts from the most popular to the less popular variations, highlighting that certain variations are much more frequently chosen or favored than others.

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(dplyr)
no_of_verified_reviews <- alexa %>%
  group_by(date) %>%
  summarize(count = n()) %>%
  arrange(date)
```

```
library(ggplot2)
ggplot(no_of_verified_reviews, aes(x = date, y = count)) +
  geom_line(color = "pink") +
  labs(title = "Verified Reviews Over Time", x = "Date", y = "Verified Reviews") +
  theme_minimal()
```

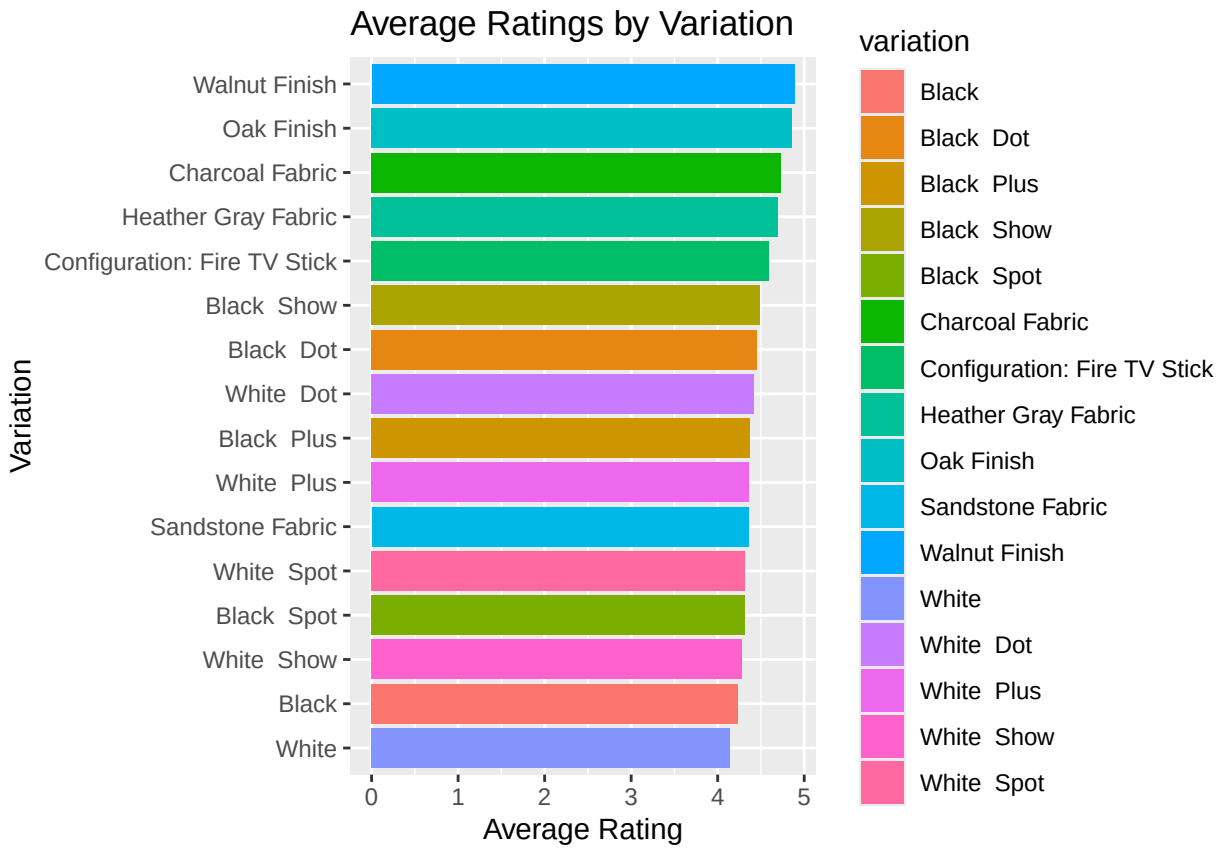


e. Get the relationship of variations and ratings. Which variations got the most highest in rating? Plot a graph to show its relationship. Show your solution and answer.

```
variationRating <- alexa %>%
  group_by(variation) %>%
  summarize(avg_rating = mean(rating, na.rm = TRUE)) %>%
  arrange(desc(avg_rating))
print(variationRating)
```

```
## # A tibble: 16 x 2
##    variation                    avg_rating
##    <chr>                             <dbl>
##  1 Walnut Finish                      4.89
##  2 Oak Finish                         4.86
##  3 Charcoal Fabric                    4.73
##  4 Heather Gray Fabric                4.69
##  5 Configuration: Fire TV Stick       4.59
##  6 Black  Show                        4.49
##  7 Black  Dot                         4.45
##  8 White  Dot                         4.42
##  9 Black  Plus                        4.37
## 10 White  Plus                        4.36
## 11 Sandstone Fabric                   4.36
## 12 White  Spot                        4.31
## 13 Black  Spot                        4.31
## 14 White  Show                        4.28
## 15 Black                              4.23
## 16 White                              4.14
```

```r
ggplot(variationRating, aes(x = reorder(variation, avg_rating), y = avg_rating, fill = variation)) +
  geom_bar(stat = "identity") +
  labs(title = "Average Ratings by Variation", x = "Variation", y = "Average Rating") +
  coord_flip()
```



Walnut Finish,Oak Finish, and Charcoal Fabric are a few of the variations with the highest ratings.