RWorksheet_Quebral#4c

Myles Andrei Quebral

2024-11-02

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

#b. Which variables from mpg dataset are categorical?

```
# The manufacturer, model, rans, drv, fl, and class
```

c. Which are continuous variables?

```
# The display, cty and hwy
```

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

```
manu_cars <- table(mpgdoc$manufacturer)
manu_cars
```

```
## 
##       audi  chevrolet      dodge       ford      honda    hyundai       jeep 
##         18         19         37         25          9         14          8 
## land rover    lincoln    mercury     nissan    pontiac     subaru     toyota 
##          4          3          4         13          5         14         34 
## volkswagen 
##         27
```

```
# The manufacturer that has the most models is dodge with 37 models.
```

```
model_cars <- table(mpgdoc$model)
model_cars
```

```
## 
##            4runner 4wd                     a4             a4 quattro 
##                      6                      7                      8 
##             a6 quattro                 altima     c1500 suburban 2wd 
##                      3                      6                      5 
##                  camry           camry solara            caravan 2wd 
##                      7                      7                     11 
##                  civic                corolla               corvette 
##                      9                      5                      5 
##      dakota pickup 4wd            durango 4wd         expedition 2wd 
##                      9                      7                      3 
##           explorer 4wd        f150 pickup 4wd           forester awd 
##                      6                      7                      6 
##     grand cherokee 4wd             grand prix                    gti 
##                      8                      5                      5
##            impreza awd                  jetta        k1500 tahoe 4wd
##                      8                      9                      4
## land cruiser wagon 4wd                 malibu                 maxima
##                      2                      5                      3
##        mountaineer 4wd                mustang          navigator 2wd
##                      4                      9                      3
##             new beetle                 passat         pathfinder 4wd
##                      6                      7                      4
##    ram 1500 pickup 4wd            range rover                 sonata
##                     10                      4                      7
##                tiburon      toyota tacoma 4wd
##                      7                      7
```

```
# The model that has the most variations is caravan 2wd with 11 variations.
```

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
```

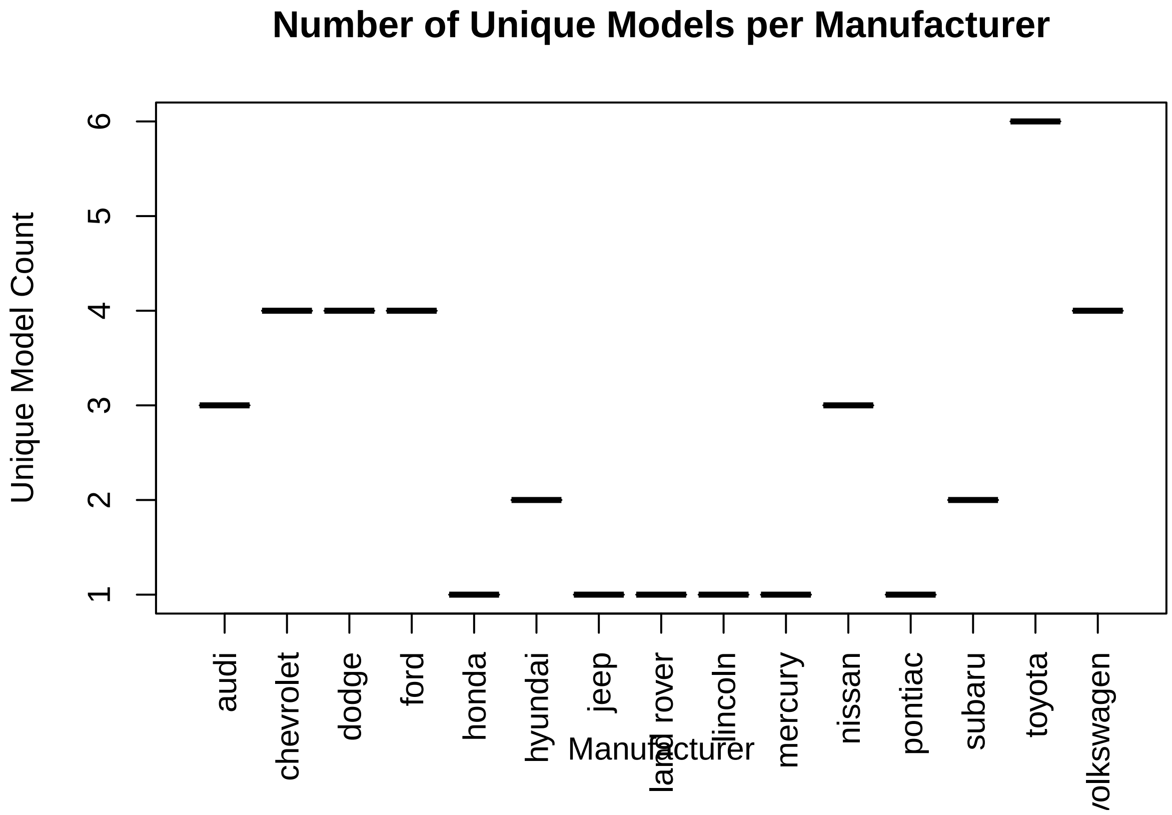
```
unique_model <- mpgdoc %>%
 group_by(manufacturer) %>%
 summarise(models = n_distinct(model))
unique_model
```

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

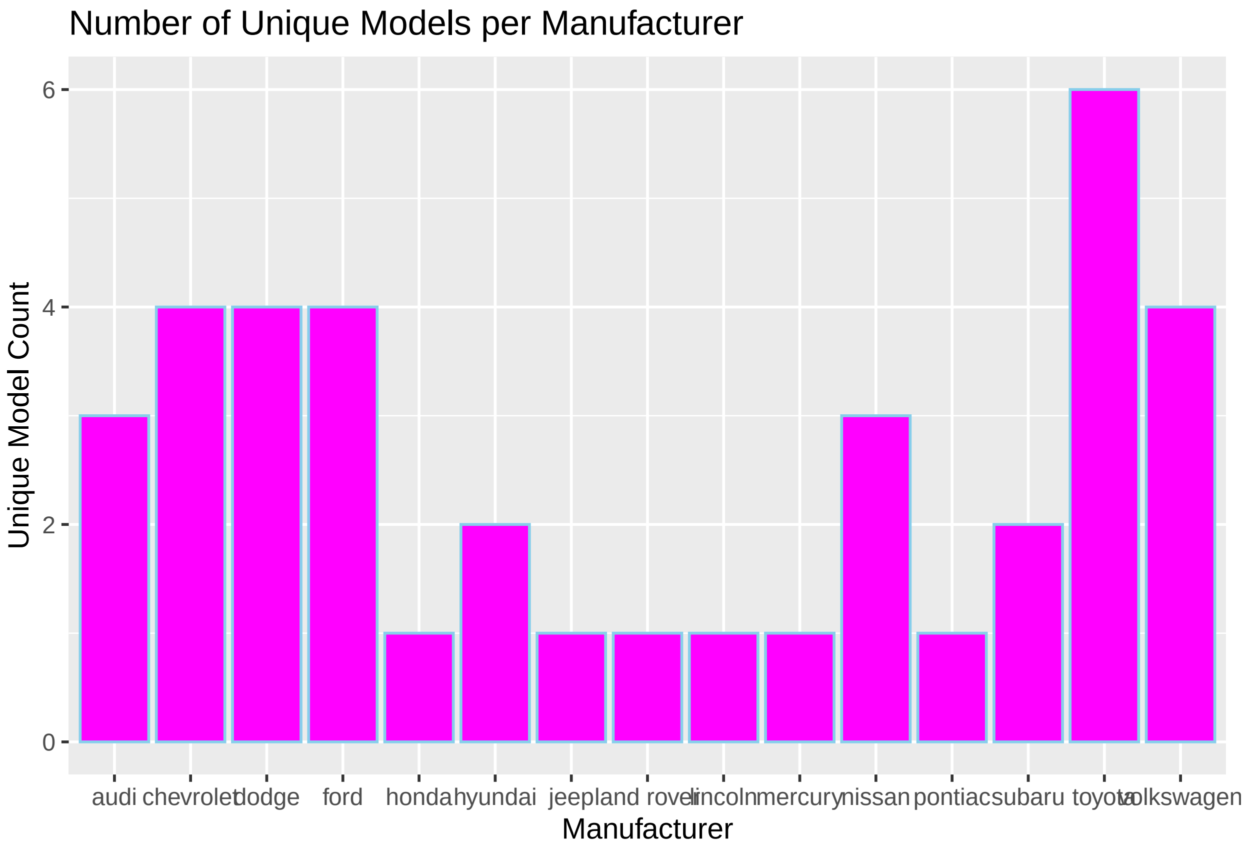
#b. Graph the result by using plot() and ggplot(). Write the codes and its result.

```
library(ggplot2)
unique_model$manufacturer <- as.factor(unique_model$manufacturer)
unique_model$models <- as.numeric(unique_model$models)

plot(unique_model$manufacturer, unique_model$models,
     type = "p",
     col = "red",
     main = "Number of Unique Models per Manufacturer",
     xlab = "Manufacturer",
     ylab = "Unique Model Count",
     las = 3)
```

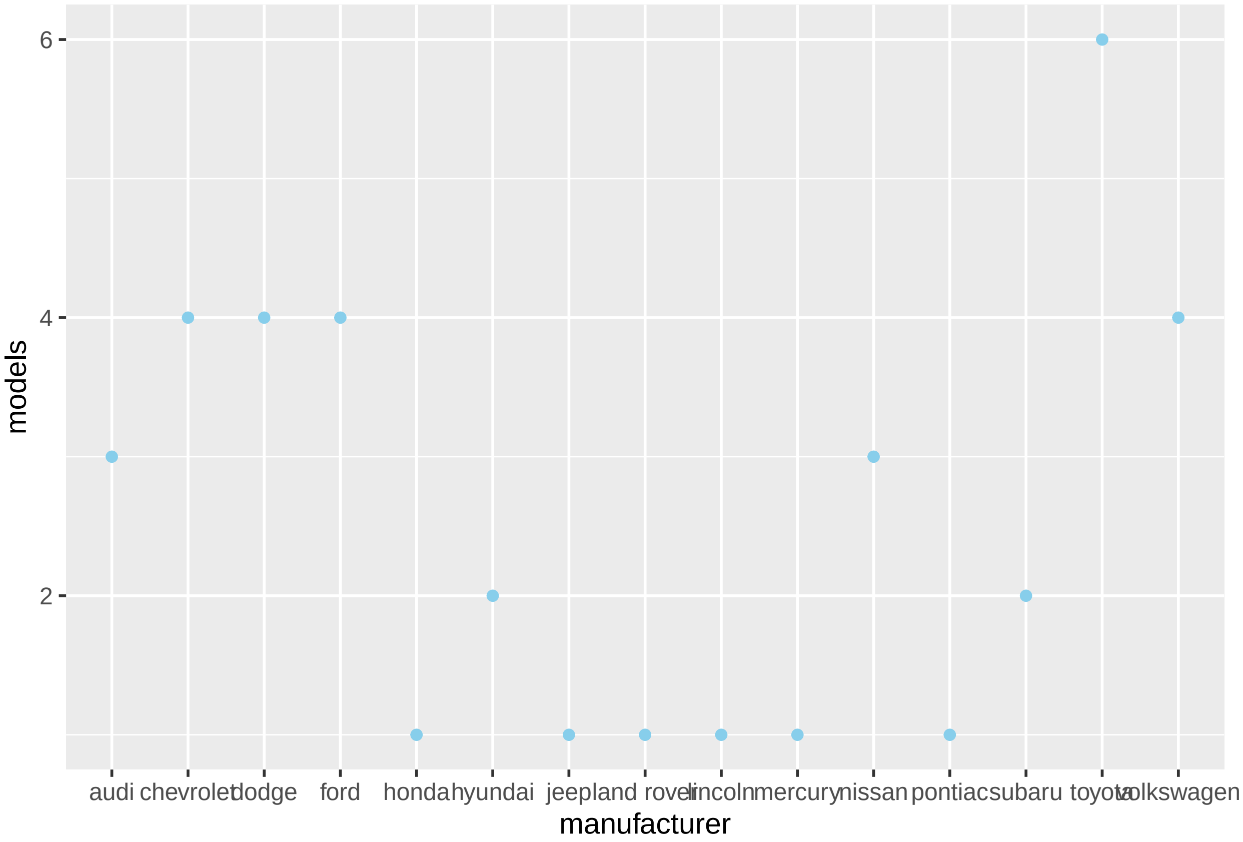


```
ggplot(unique_model, aes(manufacturer, models), y = models) +
  geom_bar(stat = "identity", fill = "magenta", color = "skyblue") +
  labs(title = "Number of Unique Models per Manufacturer", x = "Manufacturer", y = "Unique Model Count"
```



# 2. Same dataset will be used. You are going to show the relationship of the model and the manufacturer. a. What does ggplot(mpg, aes(model, manufacturer)) + geom_point() show?

```r
library(ggplot2)
ggplot(unique_model, aes(manufacturer, models), y = models) +
  geom_point( color = "skyblue")
```



```
# it shows the representation of the data using points
```

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

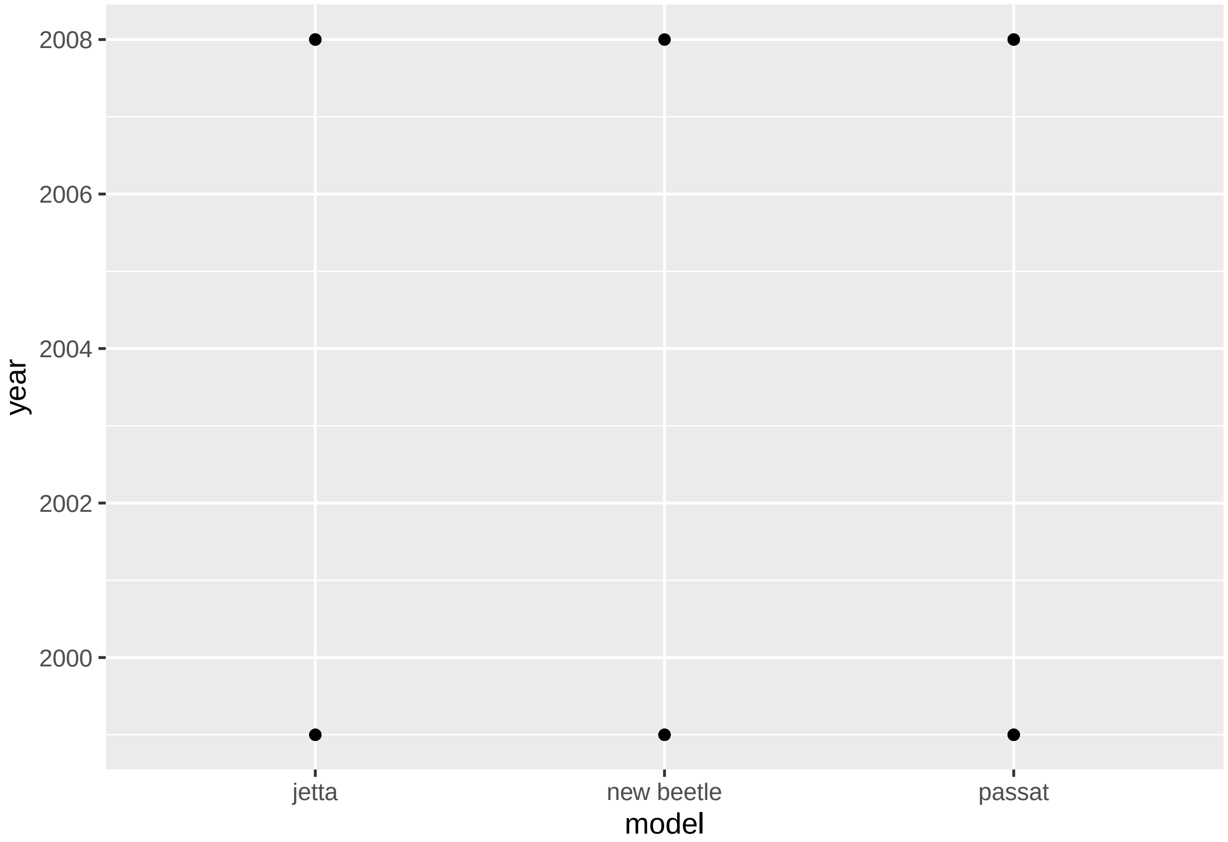
```
# It is very useful. but if not, you can improve it by transforming the data, summarizing it, using col
```

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

```
library(ggplot2)
top_20_mpgdoc <- mpgdoc %>%
  arrange(desc(mpg)) %>%  # Replace 'mpg' with the appropriate column for ranking
  head(20)
top_20_mpgdoc
```

```
##      X manufacturer      model displ year cyl      trans drv cty hwy fl
## 1  234   volkswagen     passat   3.6 2008   6   auto(s6)   f  17  26  p
## 2  233   volkswagen     passat   2.8 1999   6 manual(m5)   f  18  26  p
## 3  232   volkswagen     passat   2.8 1999   6   auto(l5)   f  16  26  p
## 4  231   volkswagen     passat   2.0 2008   4 manual(m6)   f  21  29  p
## 5  230   volkswagen     passat   2.0 2008   4   auto(s6)   f  19  28  p
## 6  228   volkswagen     passat   1.8 1999   4 manual(m5)   f  21  29  p
## 7  229   volkswagen     passat   1.8 1999   4   auto(l5)   f  18  29  p
## 8  226   volkswagen new beetle   2.5 2008   5 manual(m5)   f  20  28  r
## 9  227   volkswagen new beetle   2.5 2008   5   auto(s6)   f  20  29  r
## 10 224   volkswagen new beetle   2.0 1999   4 manual(m5)   f  21  29   r
## 11 225   volkswagen new beetle   2.0 1999   4   auto(l4)   f  19  26   r
## 12 222   volkswagen new beetle   1.9 1999   4 manual(m5)   f  35  44   d
## 13 223   volkswagen new beetle   1.9 1999   4   auto(l4)   f  29  41   d
## 14 221   volkswagen      jetta   2.8 1999   6 manual(m5)   f  17  24   r
## 15 220   volkswagen      jetta   2.8 1999   6   auto(l4)   f  16  23   r
## 16 219   volkswagen      jetta   2.5 2008   5 manual(m5)   f  21  29   r
## 17 218   volkswagen      jetta   2.5 2008   5   auto(s6)   f  21  29   r
## 18 217   volkswagen      jetta   2.0 2008   4 manual(m6)   f  21  29   p
## 19 216   volkswagen      jetta   2.0 2008   4   auto(s6)   f  22  29   p
## 20 214   volkswagen      jetta   2.0 1999   4 manual(m5)   f  21  29   r
##         class
## 1     midsize
## 2     midsize
## 3     midsize
## 4     midsize
## 5     midsize
## 6     midsize
## 7     midsize
## 8  subcompact
## 9  subcompact
## 10 subcompact
## 11 subcompact
## 12 subcompact
## 13 subcompact
## 14    compact
## 15    compact
## 16    compact
## 17    compact
## 18    compact
## 19    compact
## 20    compact
```

```r
ggplot(top_20_mpgdoc, aes(model,year)) + geom_point()
```



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

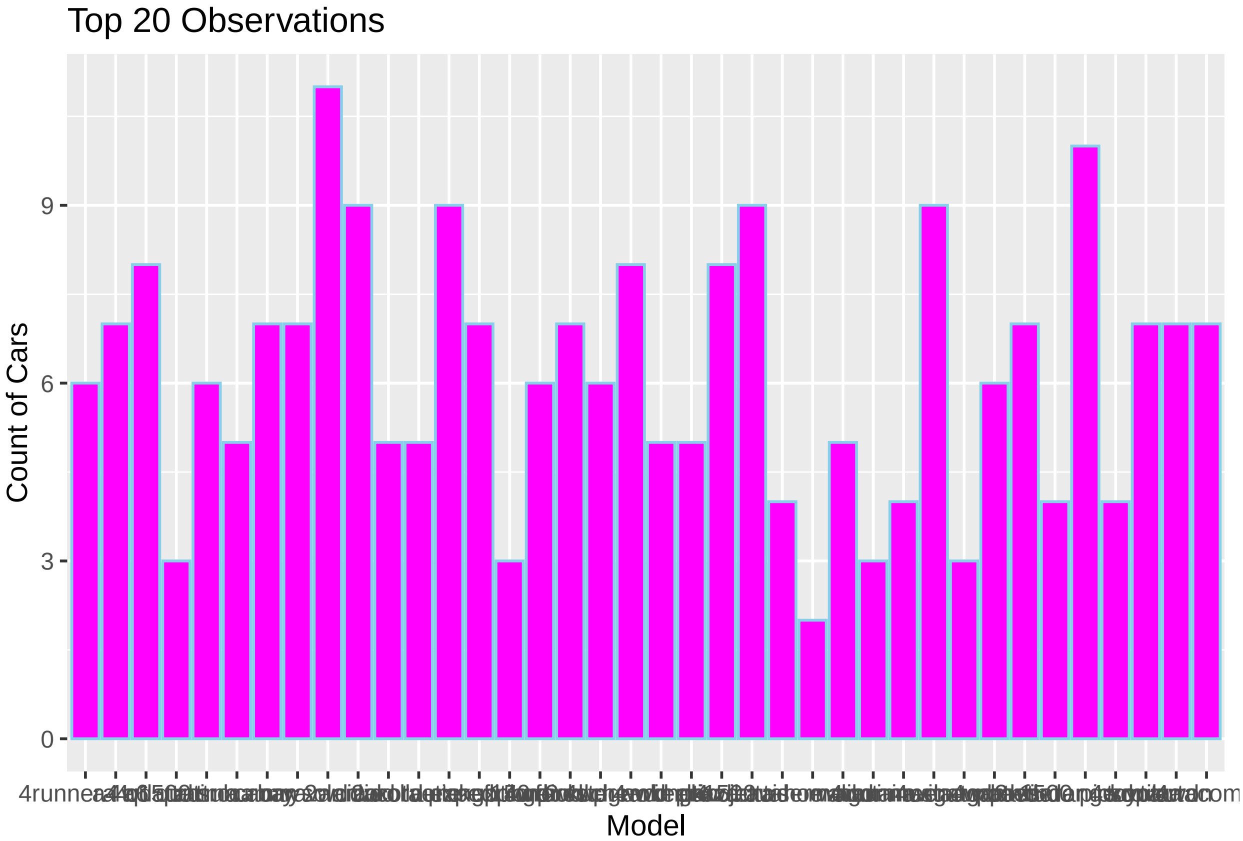
```
library(dplyr)
car_counts <- mpgdoc %>%
  group_by(model) %>%         # Group the data by the model
  summarise(count = n()) %>% # Count the number of cars in each model
  arrange(desc(count))        # Arrange the results in descending order

car_counts
```

```
## # 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.

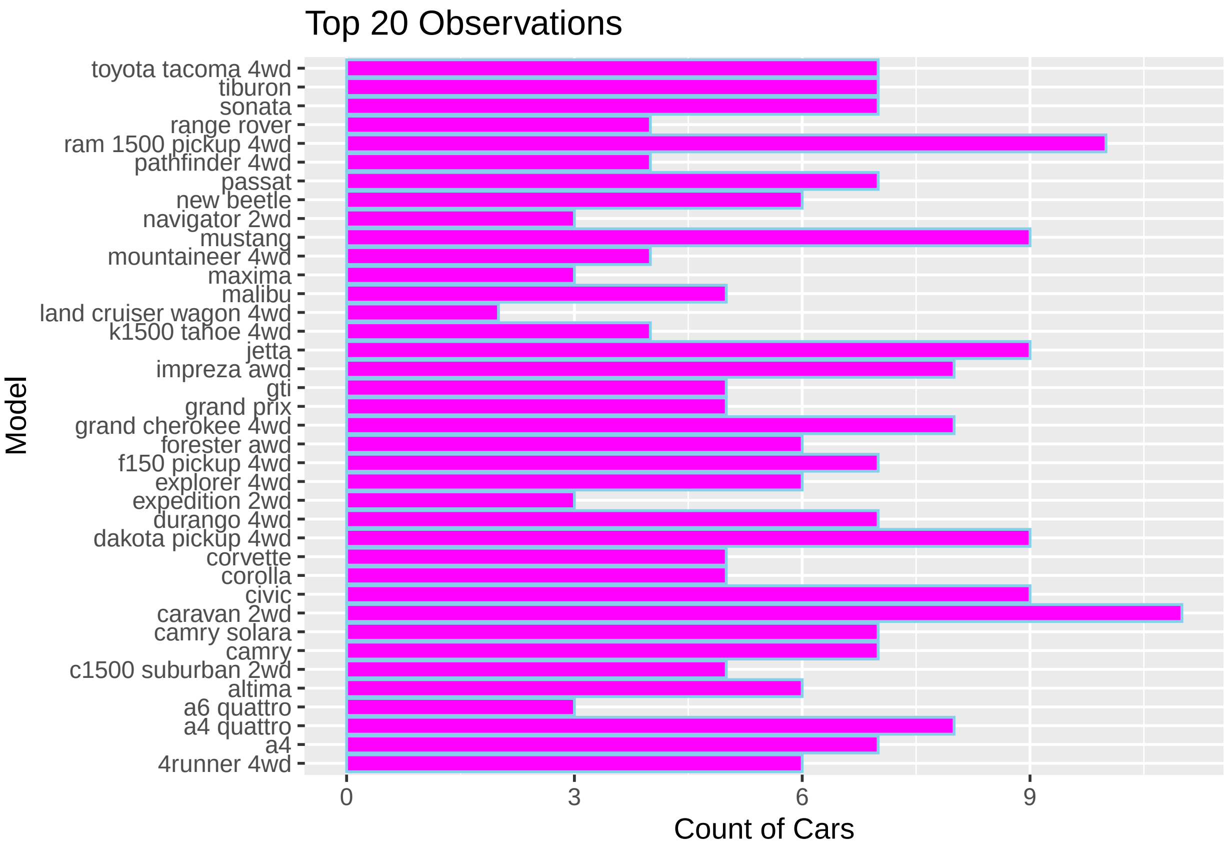
```r
library(ggplot2)
ggplot(car_counts, aes(x = model, y = count)) +
  geom_bar(stat = "identity", fill = "magenta", color = "skyblue") +
  labs(title = "Top 20 Observations", x = "Model", y = "Count of Cars")
```



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

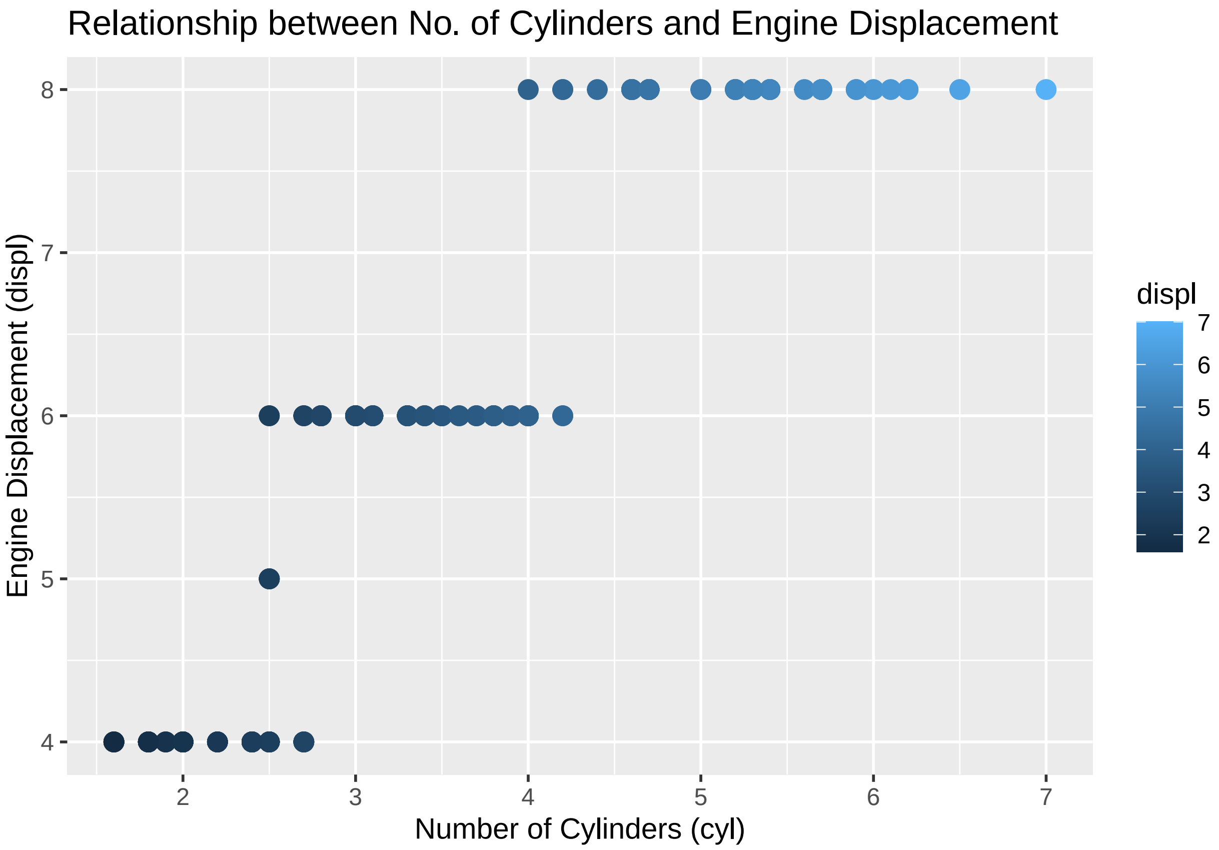
```r
library(ggplot2)

ggplot(car_counts, aes(x = model, y = count)) +
  geom_bar(stat = "identity", fill = "magenta", color = "skyblue") +
  labs(title = "Top 20 Observations", x = "Model", y = "Count of Cars") +
  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.

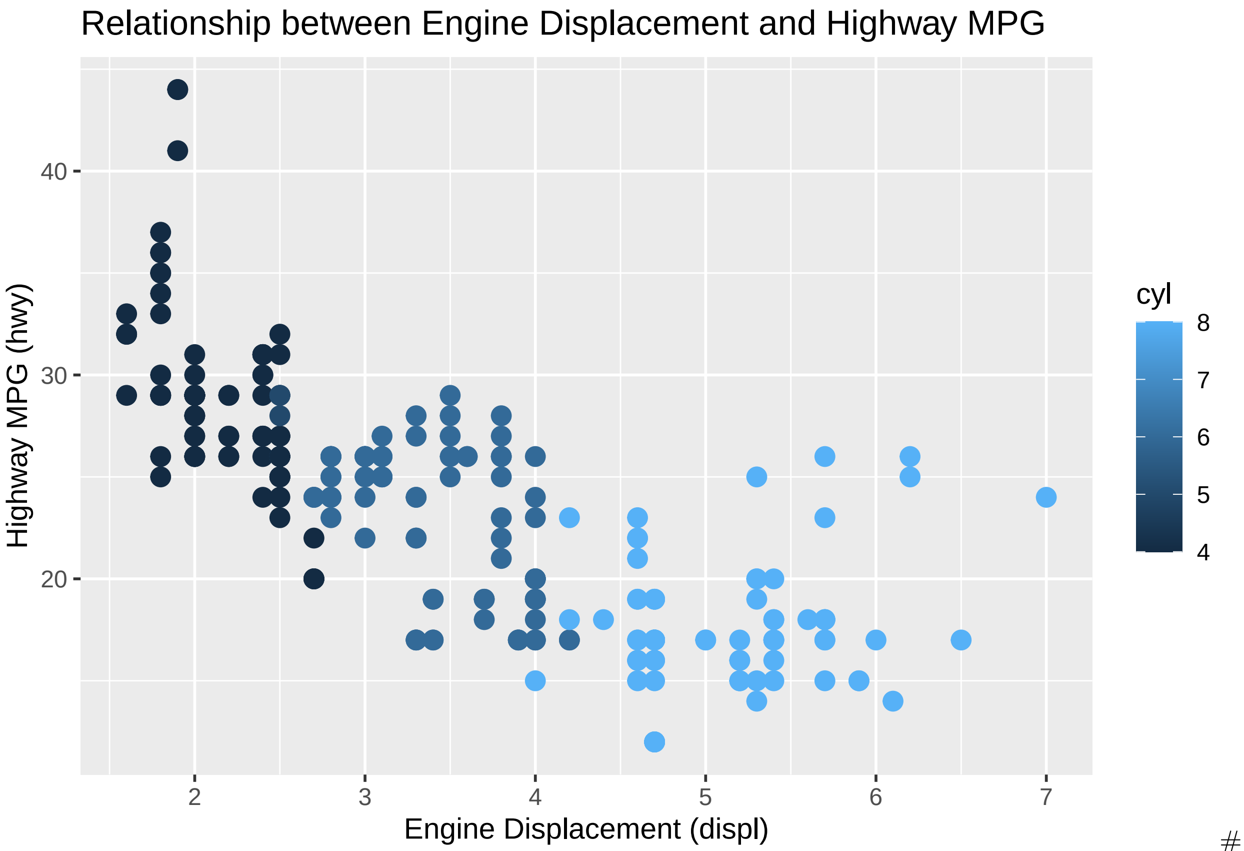
```
library(ggplot2)
ggplot(mpgdoc, aes(x = displ, y = cyl, color = displ)) + geom_point(size = 3) +  labs(title = "Relations
       x = "Number of Cylinders (cyl)",
       y = "Engine Displacement (displ)")
```



#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(mpgdoc, aes(x = displ, y = hwy, color = cyl)) +
  geom_point(size = 3) +
  labs(title = "Relationship between Engine Displacement and Highway MPG",
       x = "Engine Displacement (displ)",
       y = "Highway MPG (hwy)")
```



#

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.

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

```
str(traffic_docs)
```

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

```
# There are 48120 observations and 4 variables named DateTime, Junction, Vehicles, and ID
```

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

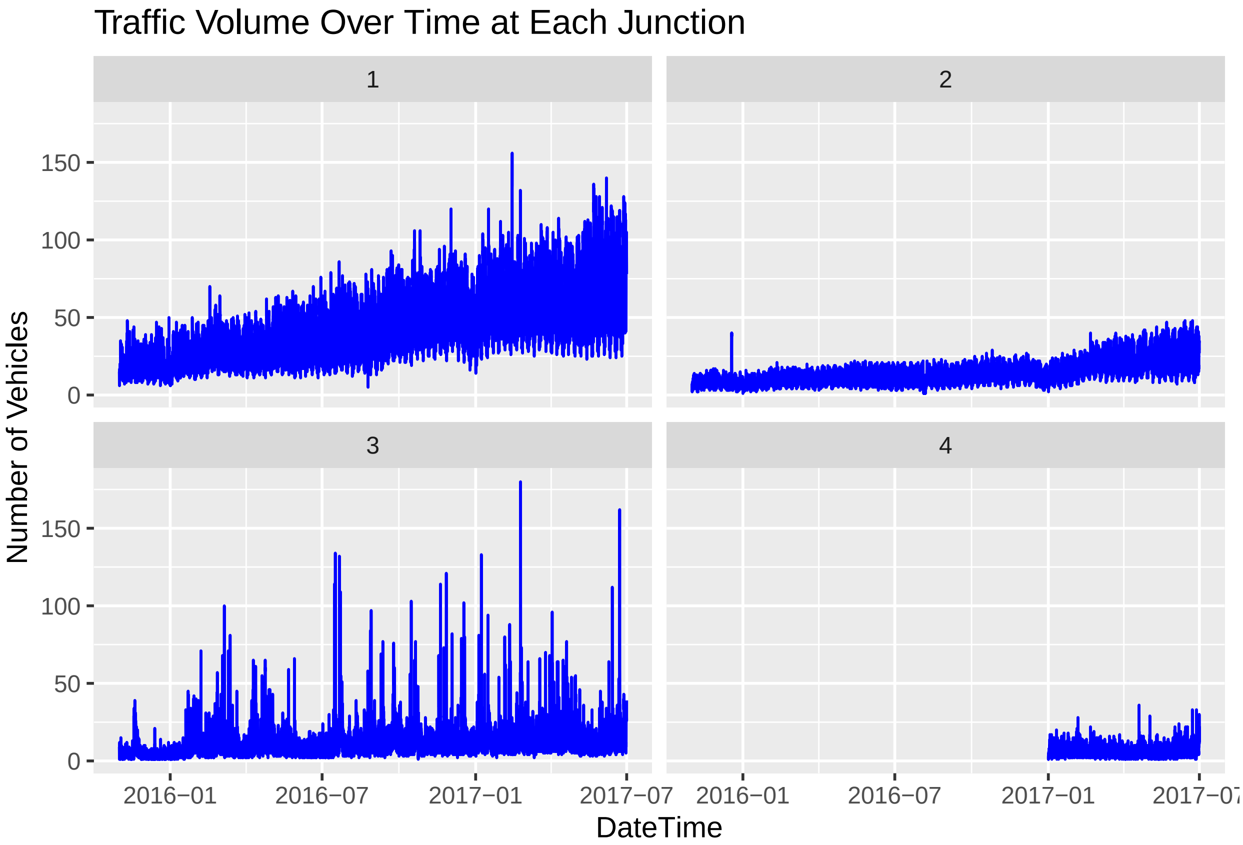
```
junc_list <- split(traffic_docs, traffic_docs$Junction)
```

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

```r
library(ggplot2)

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

# Plot each junction using geom_line() and facet_wrap()
ggplot(traffic_docs, aes(x = DateTime, y = Vehicles)) +
  geom_line(color = "blue") +
  labs(title = "Traffic Volume Over Time at Each Junction",
       x = "DateTime",
       y = "Number of Vehicles") +
  facet_wrap(~ Junction)
```



# 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.

```r
setwd("/cloud/project")
alexa_file <- read.csv("alexa.csv")


str(alexa_file)
```

```
## 'data.frame':    3150 obs. of  5 variables:
##  $ rating          : int  5 5 4 5 5 5 3 5 5 5 ...
##  $ date            : chr  "31-Jul-18" "31-Jul-18" "31-Jul-18" "31-Jul-18" ...
##  $ variation       : chr  "Charcoal Fabric " "Charcoal Fabric " "Walnut Finish " "Charcoal Fabric "
##  $ verified_reviews: chr  "Love my Echo!" "Loved it!" "Sometimes while playing a game, you can answe
##  $ feedback        : int  1 1 1 1 1 1 1 1 1 1 ...
```

```
ncol(alexa_file)
```

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

```
# There are 3150 observations and 5 columns in alexa_file
```

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

```
print(colnames(alexa_file))
```

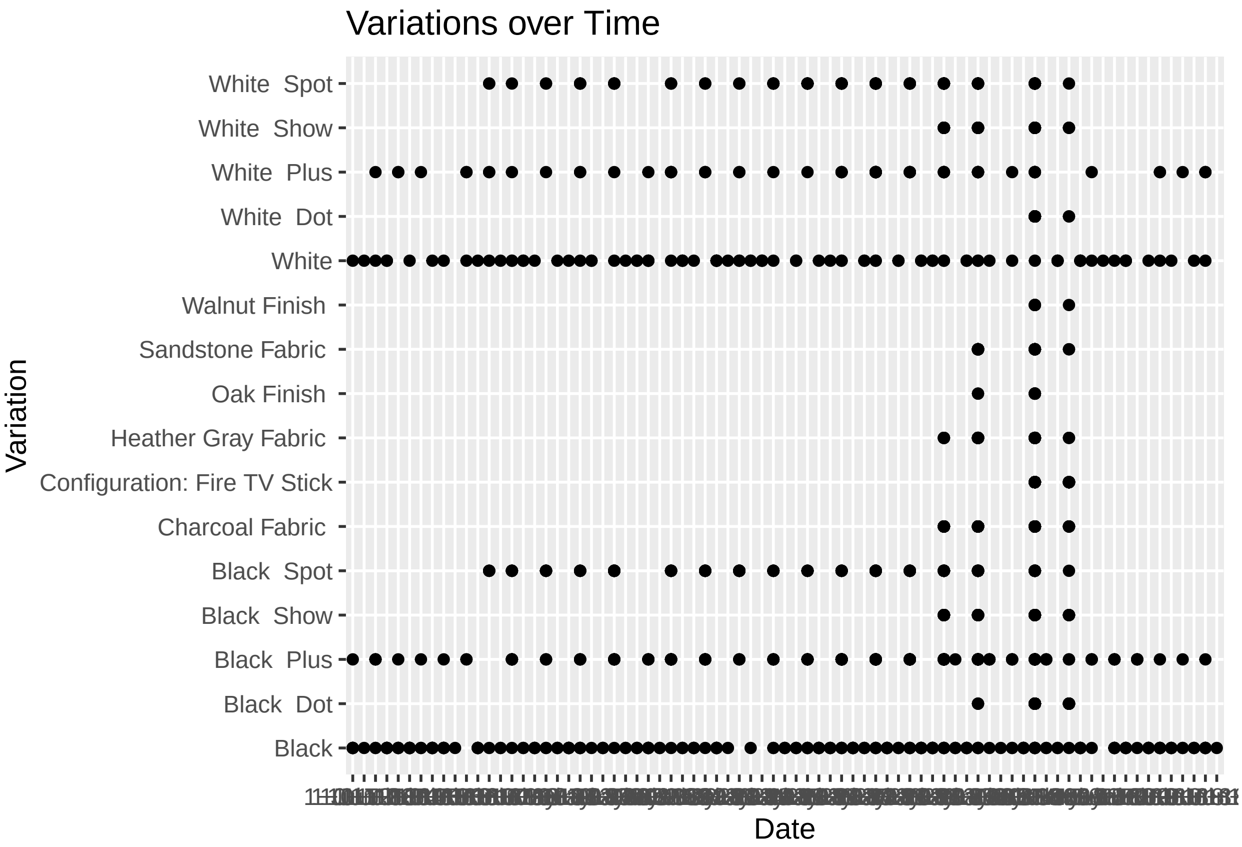
```
## [1] "rating"           "date"             "variation"        "verified_reviews"
## [5] "feedback"
```

```
variation_totals <- alexa_file %>%
 group_by(variation) %>%
  summarise(total = n())
variation_totals
```

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

```
library(ggplot2)
ggplot(alexa_file, aes(x = date, y = variation)) +
geom_point() + labs(title = "Variations over Time", x = "Date", y = "Variation", color = "Verified")
```

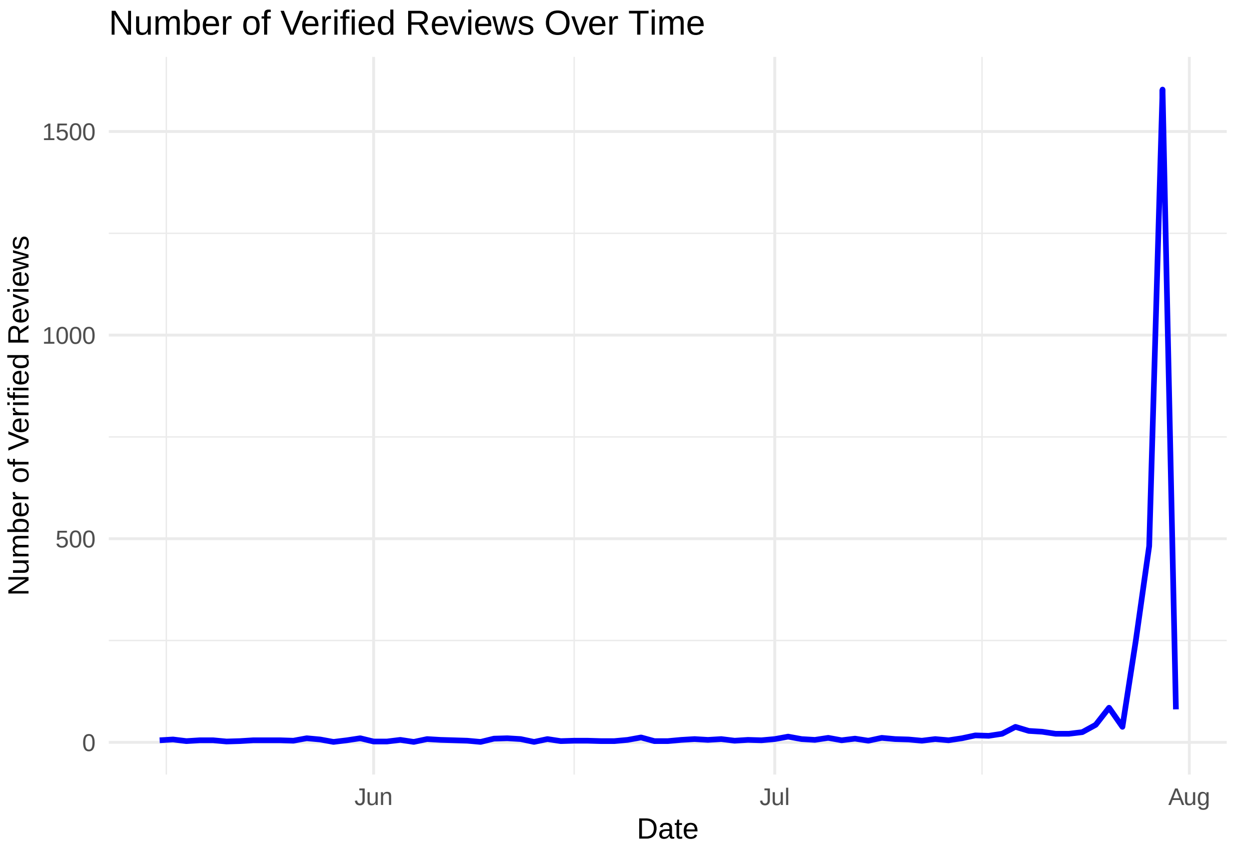


# d. Plot a geom_line() with the date and the number of verified reviews. Complete the details of the graphs. Show your answer and solution.

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

alexa_file$date <- as.Date(alexa_file$date, format = "%d-%b-%y")
review_counts <- alexa_file %>%
  group_by(date) %>%
  summarise(review_count = n())
ggplot(data = review_counts, aes(x = date, y = review_count)) +
  geom_line(color = "blue", size = 1) +
  labs(
    title = "Number of Verified Reviews Over Time",
    x = "Date",
    y = "Number of Verified Reviews"
  ) +
  theme_minimal()
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

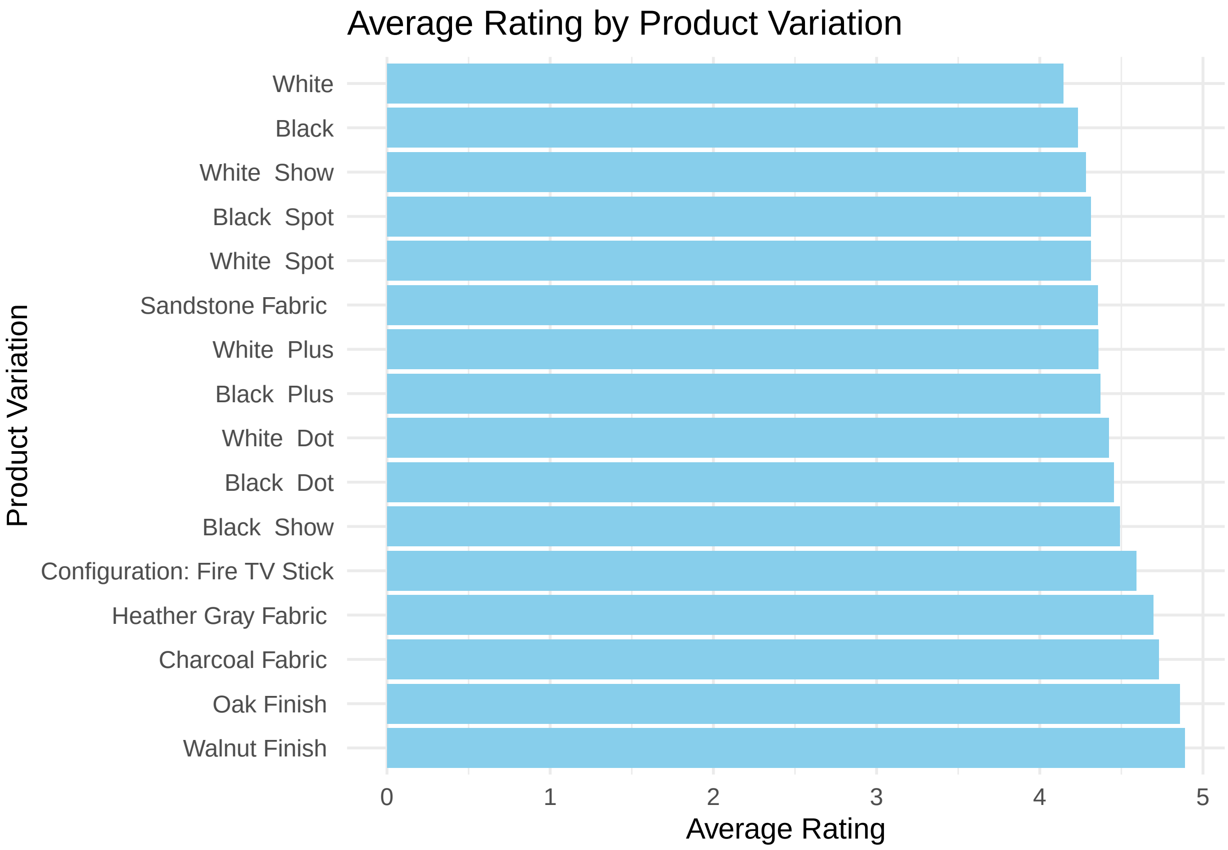


#e. Get the relationship of variations and ratings. Which variations got the most highest in rating? Plot a graph to show its relationship. Show your solution and answer.

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


variation_ratings <- alexa_file %>%
  group_by(variation) %>%
  summarise(average_rating = mean(rating, na.rm = TRUE)) %>%
  arrange(desc(average_rating))


ggplot(variation_ratings, aes(x = reorder(variation, -average_rating), y = average_rating)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  coord_flip() +  # Flip coordinates for better readability
  labs(title = "Average Rating by Product Variation",
       x = "Product Variation",
       y = "Average Rating") +
  theme_minimal()
```


Average Rating by Product Variation
Product Variation
White
Black
White Show
Black Spot
White Spot
Sandstone Fabric
White Plus
Black Plus
White Dot
Black Dot
Black Show
Configuration: Fire TV Stick
Heather Gray Fabric
Charcoal Fabric
Oak Finish
Walnut Finish
0
1
2
3
4
5
Average Rating