RWorksheet_cadiz#4c

John Dave R. Cadiz

2024-11-03

1. Use the dataset mpg

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

```
library(ggplot2)

mpgdata <- read.csv("mpg.csv")
str(mpgdata)
```

```
## '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 varialbles from mpg data are categorical?

```
# The categorical variables from the mpg data set are manufacture, model, year,
# cyl, trans, drv, fl, and class.
```

c. Which are continouse variables?

```
# The continious variable from the data set mpr are displ, cty, and hwy.
```

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

a. Group the manufacturers and find the unique model. 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
```

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

manufacturer_model
```

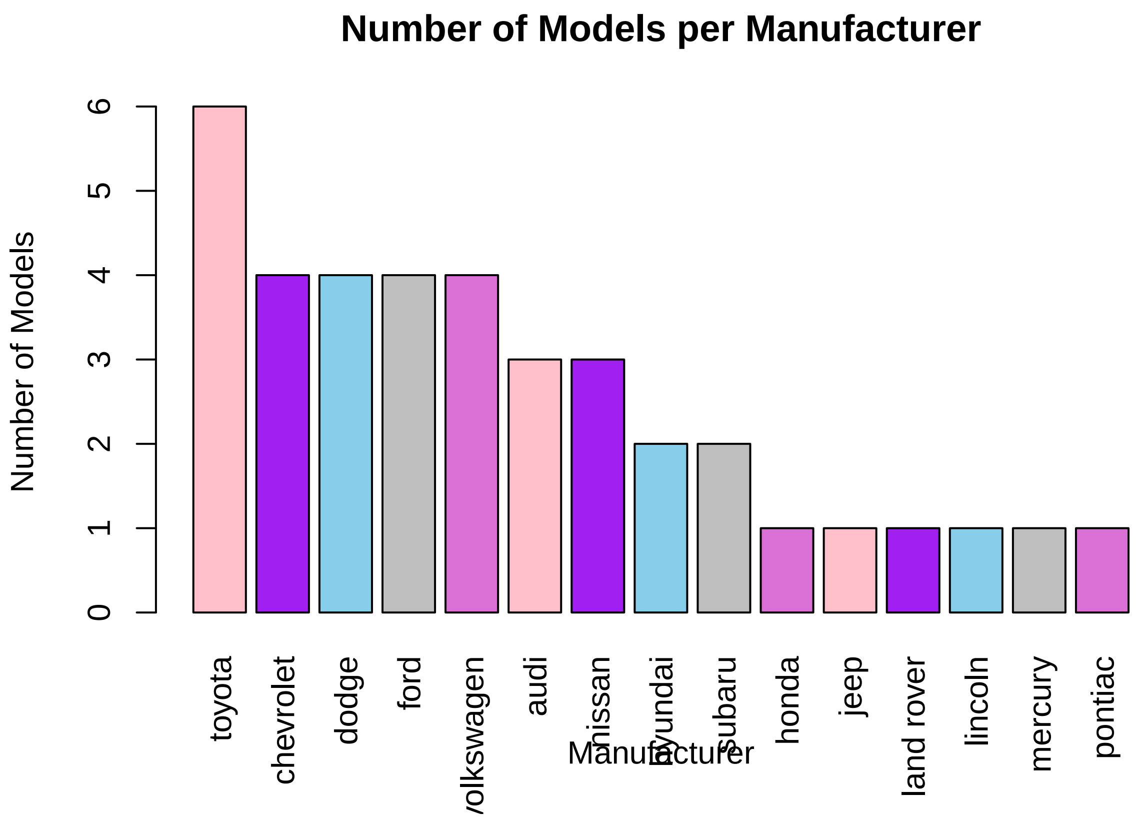
```
## # A tibble: 15 x 2
##    manufacturer model_num
##    <chr>            <int>
##  1 toyota               6
##  2 chevrolet            4
##  3 dodge                4
##  4 ford                 4
##  5 volkswagen           4
##  6 audi                 3
##  7 nissan               3
##  8 hyundai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

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

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

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

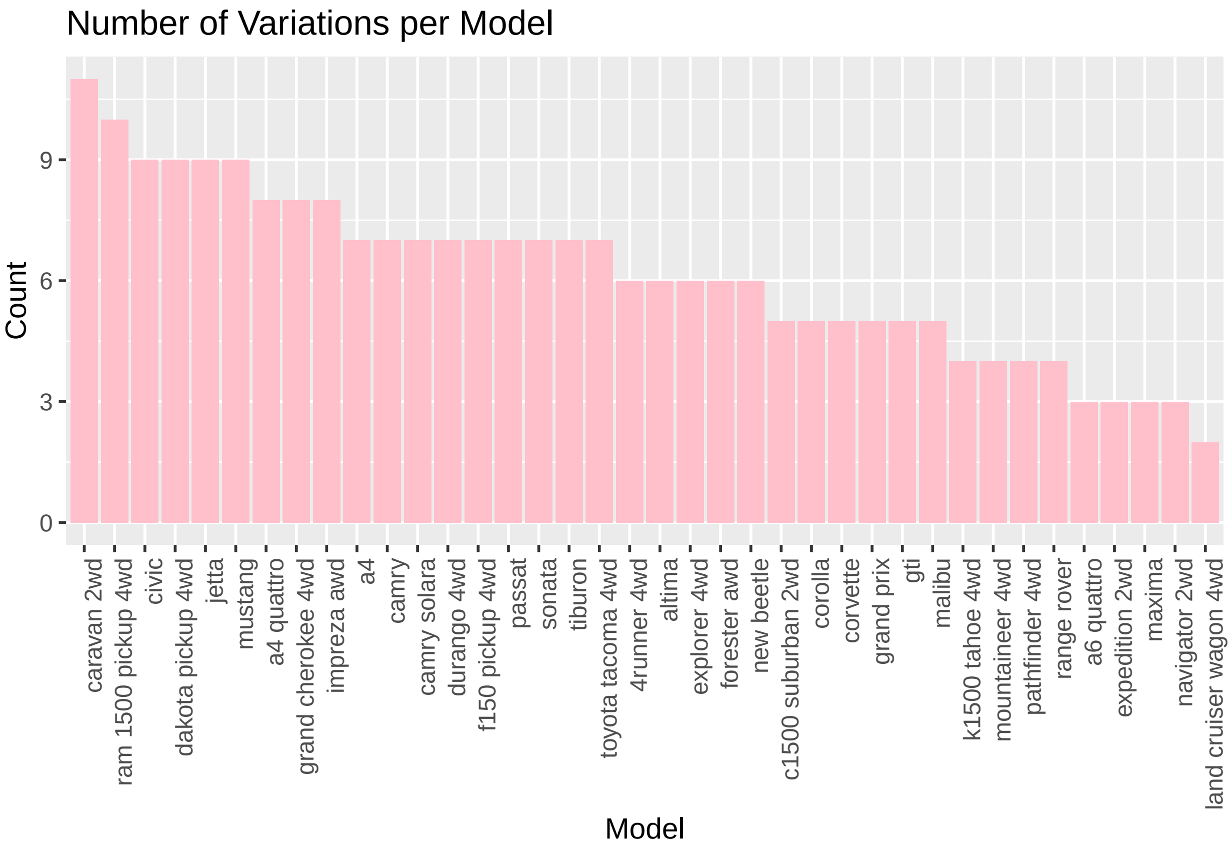
```r
manufacturer_data <- setNames(
  manufacturer_model$model_num,
  manufacturer_model$manufacturer
  )

barplot(manufacturer_data,
        main = "Number of Models per Manufacturer",
        xlab = "Manufacturer",
        ylab = "Number of Models",
        col = c("pink", "purple", "skyblue", "grey", "orchid"),
        las = 3)
```



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

variations_num
```

```
## # 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
```

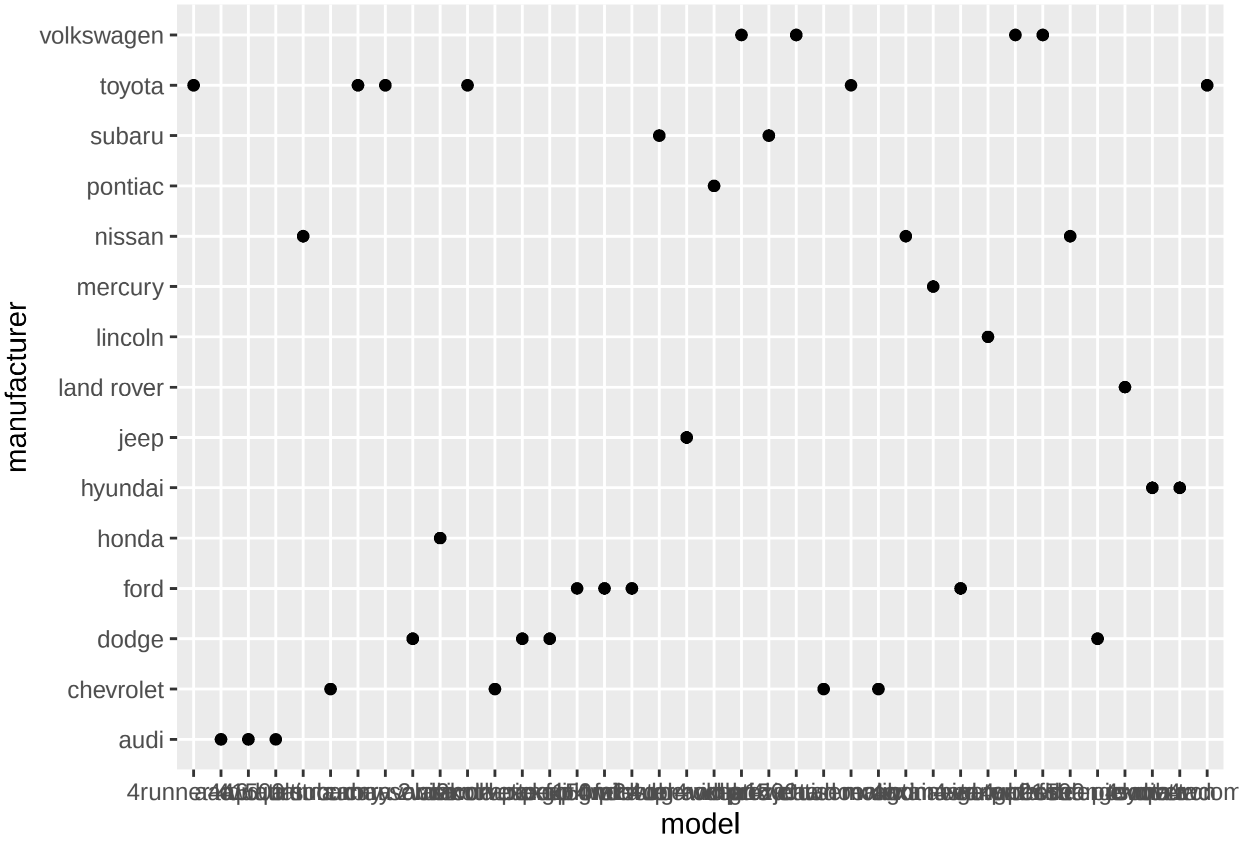
```
ggplot(variations_num,
       aes(x = reorder(model, -count), y = count)) +
       geom_bar(stat = "identity", fill = "pink") +
       labs(title = "Number of Variations per Model", x = "Model", y = "Count") +
       theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



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

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

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



```
# this snippet shows the code of a scatter plot by two data named models and
#manufacturers.
```

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

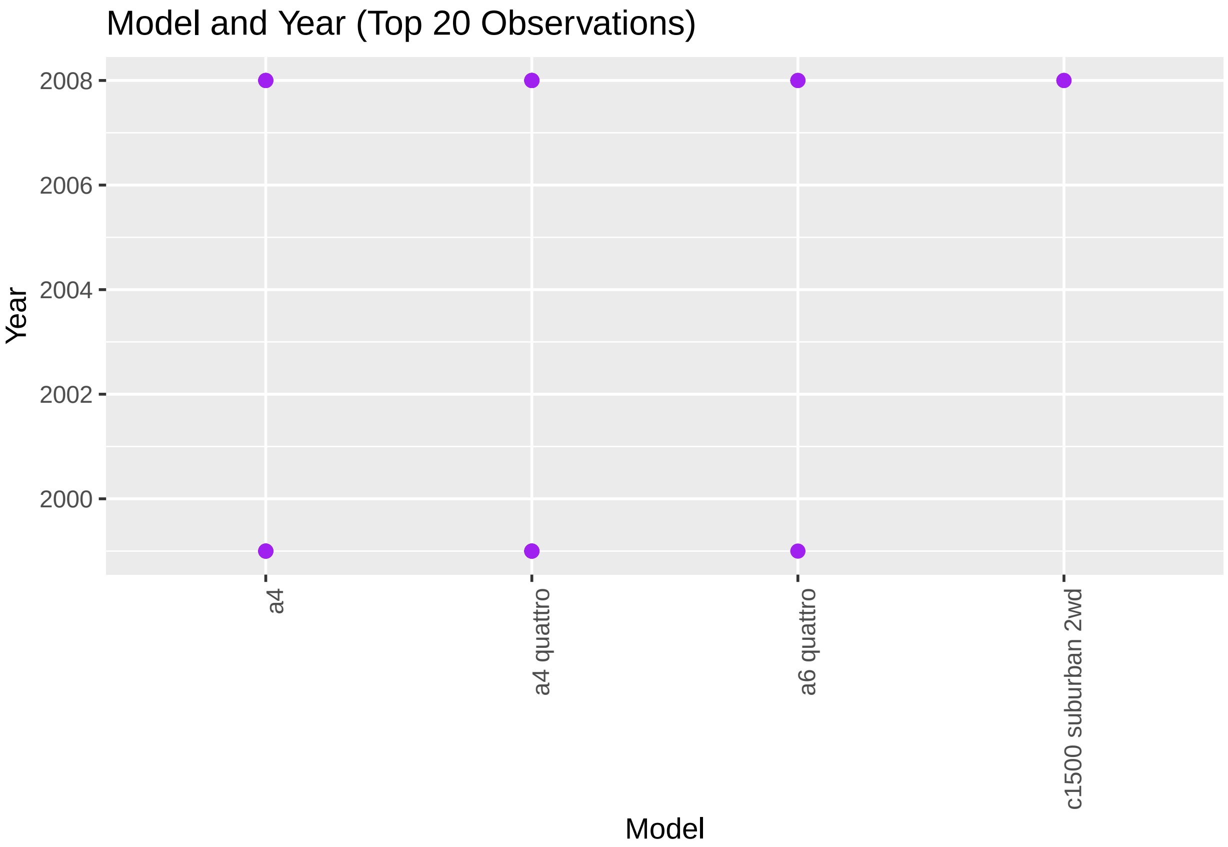
##more informative?

```
# For me, it could be better to not use the scatter plot in this type of
#visualization because it is difficult to track and interpret the given data.
#I will modify it and change to bar graph, by this I could show the difference
#between each models and manufacturer by using the labels and specific colors
#in this way it can help me to track and interpret the data without error.
```

3.Plot the mdoel and the year using ggplot(). Use only the top 20

##observations. Write the codes and its results.

```
topobservation <- mpg[1:20, ]

ggplot(topobservation,
       aes(x = model, y = year)) +
       geom_point(color = "purple", size = 2) +
       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.

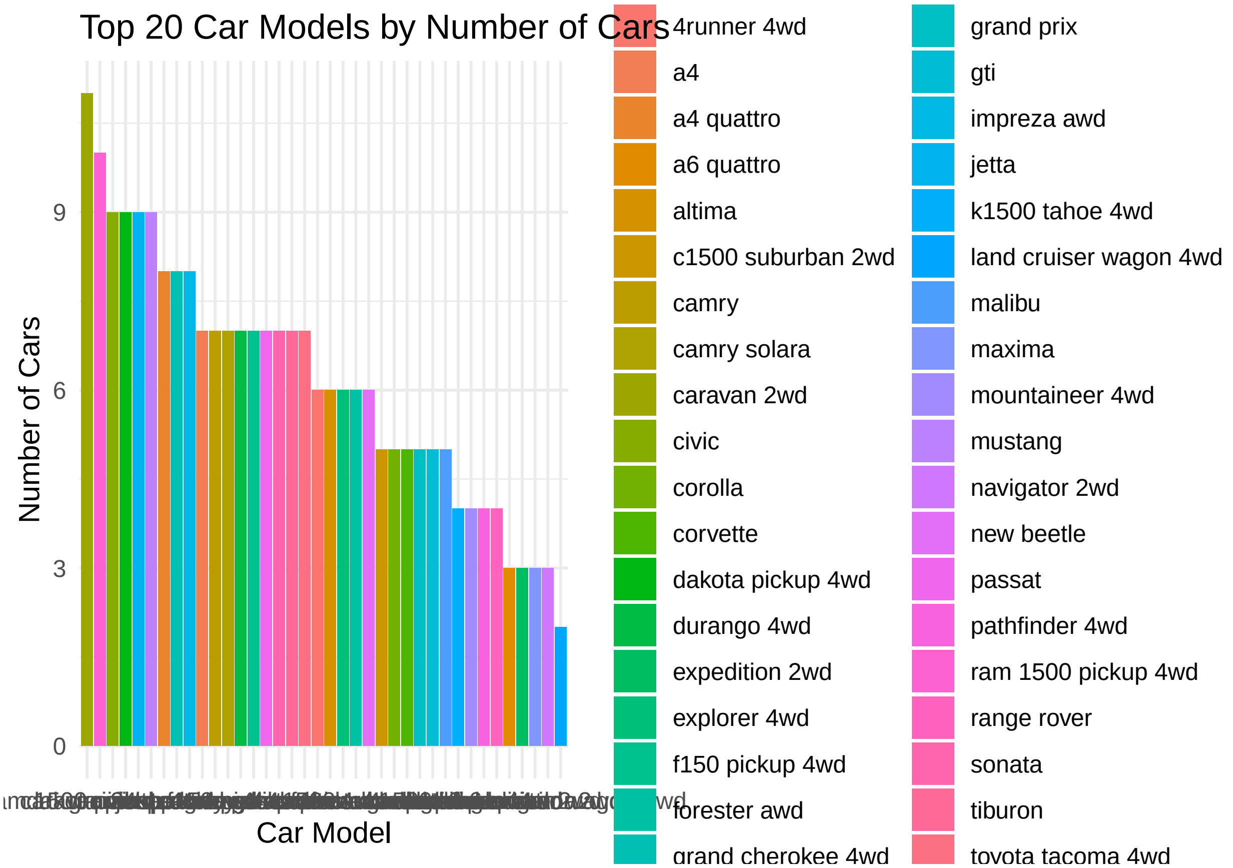
```
library(dplyr)

carcountpermodel <- mpg %>%
  group_by(model) %>%
  summarise(car_count = n())

print(carcountpermodel)
```

```
## # A tibble: 38 x 2
##    model              car_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 observation only.

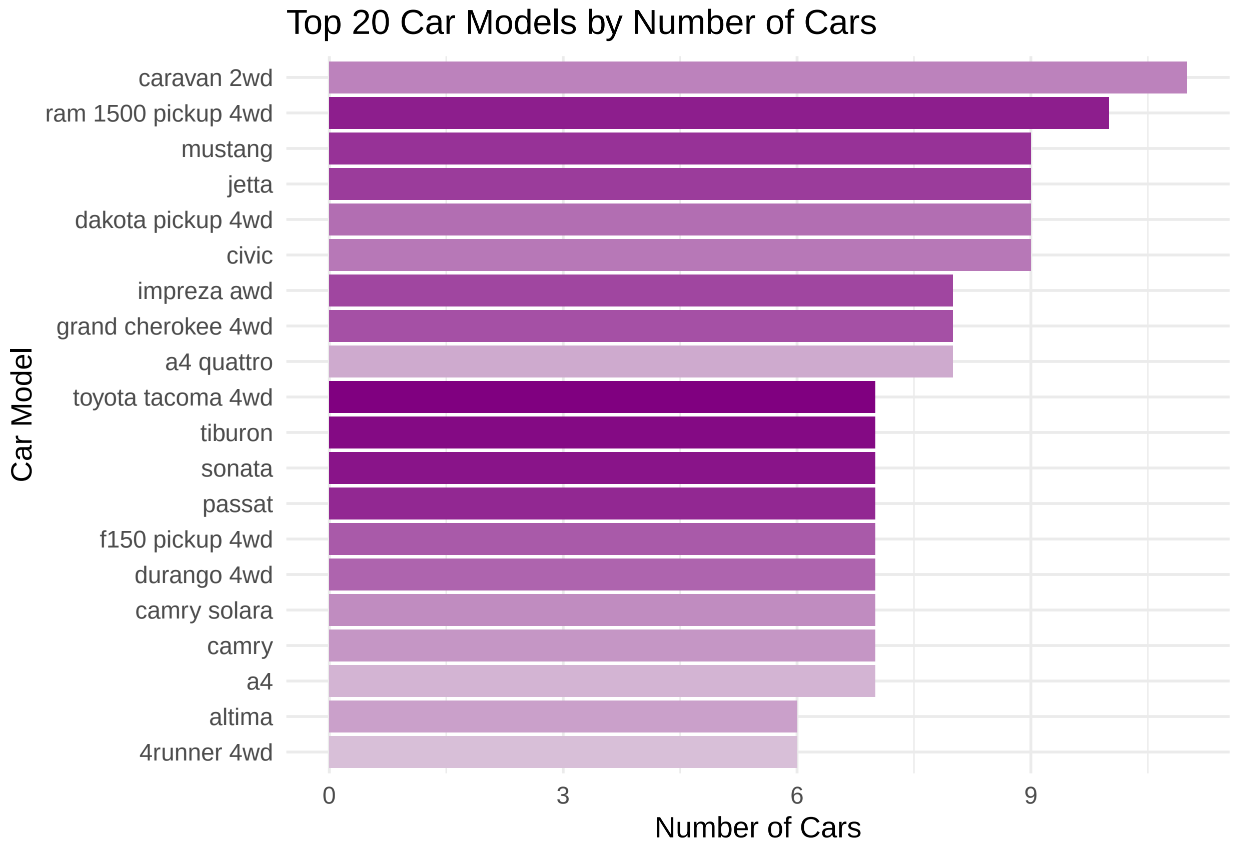
##The graphs should have title, labels and colors. show code and its results.

```r
ggplot(carcountpermodel, aes(x = reorder(model, -car_count), y = car_count, fill = model)) +
  geom_bar(stat = "identity") +
  labs(
    title = "Top 20 Car Models by Number of Cars",
    x = "Car Model",
    y = "Number of Cars"
  ) +
  theme_minimal()
```



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

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

top_20_models <- carcountpermodel %>%
  arrange(desc(car_count)) %>%
  head(20)

ggplot(top_20_models, aes(x = reorder(model, car_count), y = car_count, fill = model)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(
    title = "Top 20 Car Models by Number of Cars",
    x = "Car Model",
    y = "Number of Cars"
  ) +
  theme_minimal() +
  theme(legend.position = "none") +
scale_fill_manual(values = colorRampPalette(c("#D8BFD8", "#800080"))(20))
```

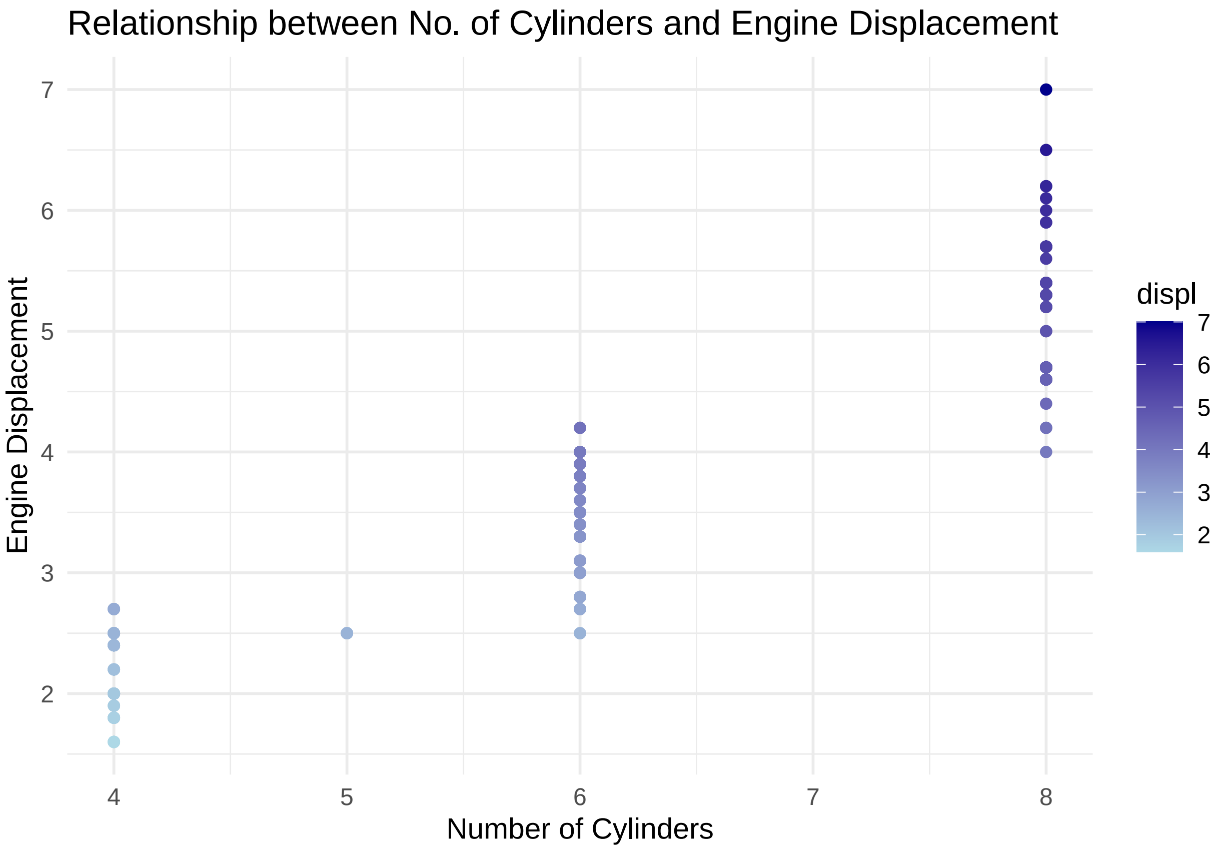


5. Plot the relationship between cyl - number of cylinders and disply -

##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 code 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_gradient(low = "lightblue", high = "darkblue") +
  theme_minimal()
```

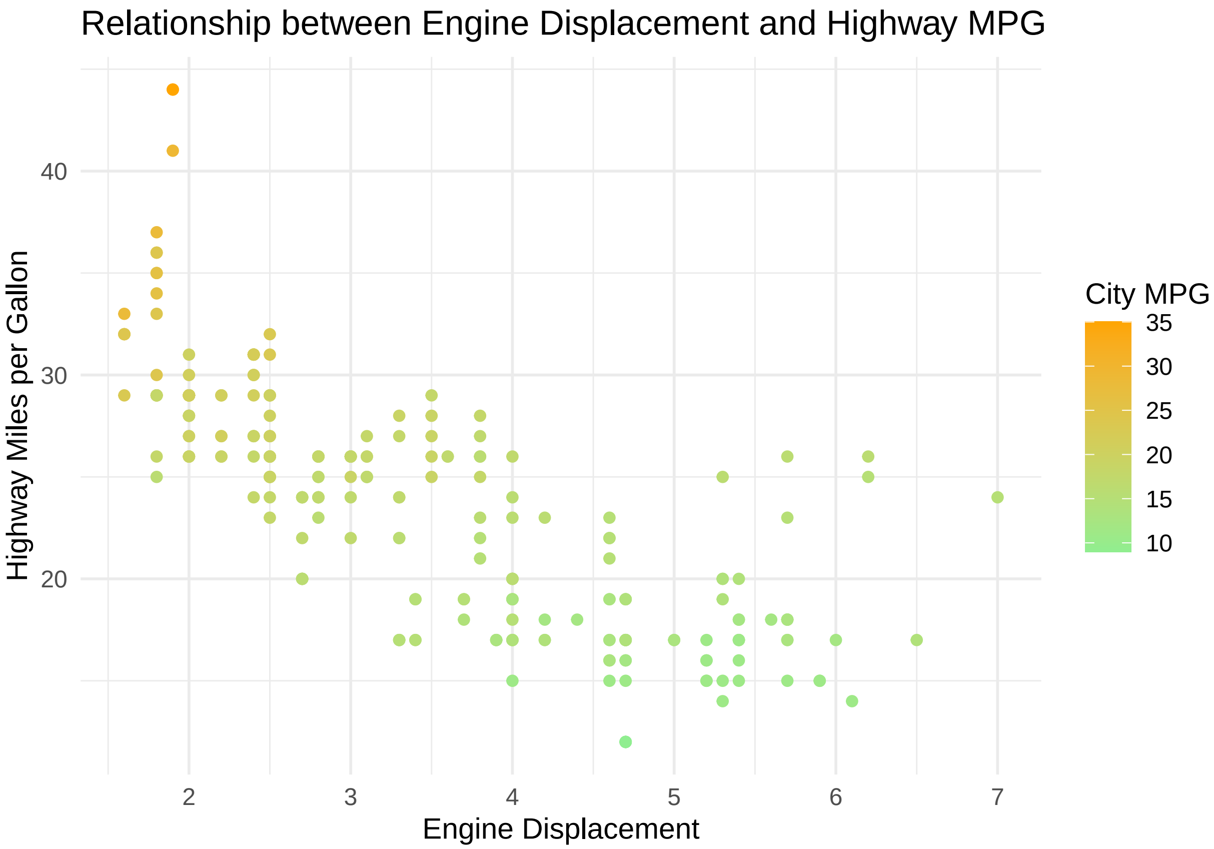


```
# It shows that as the number of cylinders increases, engine displacement also
#increases, indicating a positive relationship. However, variability exist
#within each cylinder group, and the relationship is not perfectly linear due
#to the differences in engine design and vehicle type.
```

6. Plot the relationship between displ (engine displacement) and hwy

##(highway miles per gallon). Mapped it with a continous 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 = "Relationship between Engine Displacement and Highway MPG",
    x = "Engine Displacement",
    y = "Highway Miles per Gallon",
    color = "City MPG"
  ) +
  scale_color_gradient(low = "lightgreen", high = "orange") +
  theme_minimal()
```



```
## To answer it, the plot reveals the negative relationship between engine
##displacement(displ) and highway mpg(hwy), with larger engines generally
##achieving lower highway mileage. in addition, cars with lower city MPG(cty)
##tend to have large displacements, as these engines consumes more fuel,
##leading to lower fuel efficiency overall.
```

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

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

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

```
numberof_observation <- nrow(trafficdata)
variables <- colnames(trafficdata)

cat("Number of observations:", numberof_observation, "\n")
```

```
## Number of observations: 48120
```

```
cat("Variables:", variables, "\n")
```

```
## Variables: DateTime Junction Vehicles ID
```

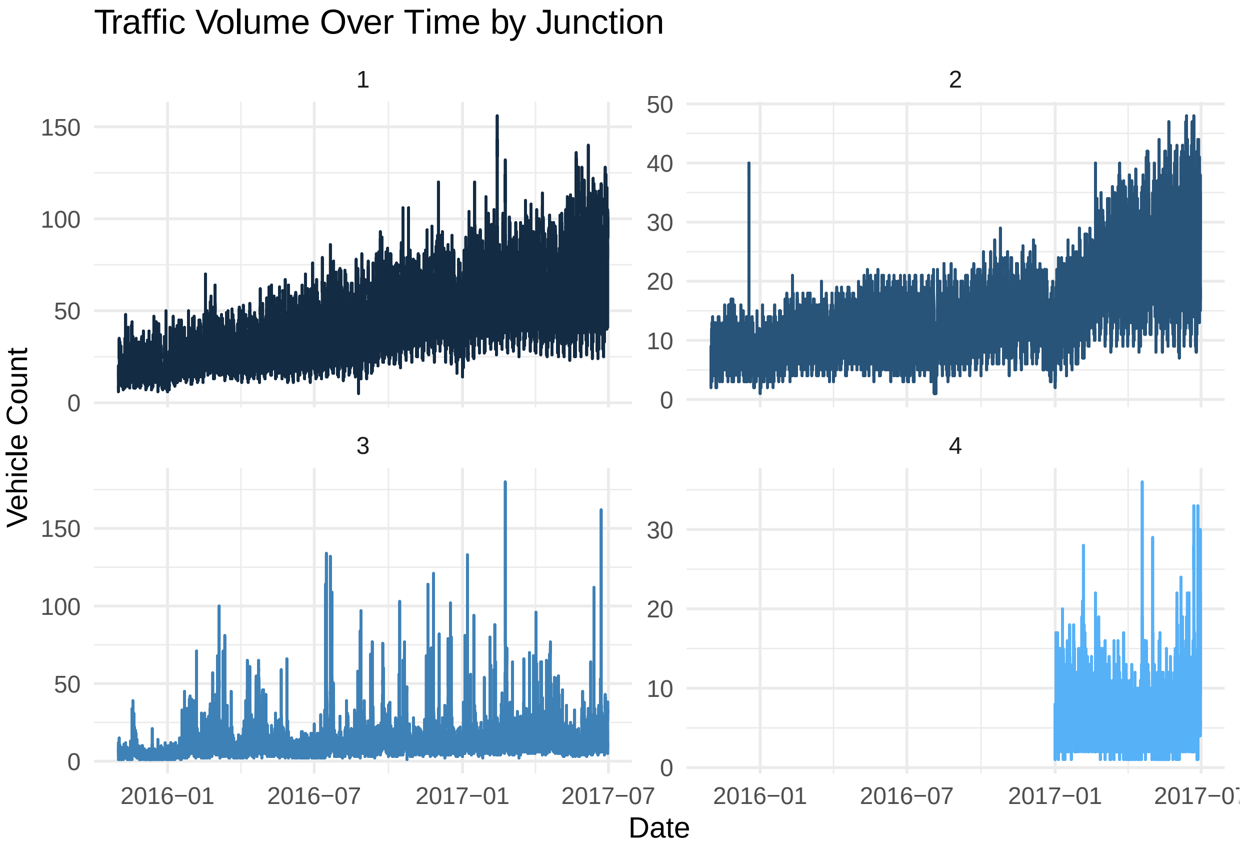
b.Subset the dataet into the junctions. What is the R code and its output.

```
junctionslist <- split(trafficdata, trafficdata$Junction)

lapply(junctionslist, head)
```

```
## $`1`
##              DateTime Junction Vehicles          ID
## 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
##
## $`2`
##                  DateTime Junction Vehicles          ID
## 14593 2015-11-01 00:00:00        2        6 20151101002
## 14594 2015-11-01 01:00:00        2        6 20151101012
## 14595 2015-11-01 02:00:00        2        5 20151101022
## 14596 2015-11-01 03:00:00        2        6 20151101032
## 14597 2015-11-01 04:00:00        2        7 20151101042
## 14598 2015-11-01 05:00:00        2        2 20151101052
##
## $`3`
##                  DateTime Junction Vehicles          ID
## 29185 2015-11-01 00:00:00        3        9 20151101003
## 29186 2015-11-01 01:00:00        3        7 20151101013
## 29187 2015-11-01 02:00:00        3        5 20151101023
## 29188 2015-11-01 03:00:00        3        1 20151101033
## 29189 2015-11-01 04:00:00        3        2 20151101043
## 29190 2015-11-01 05:00:00        3        2 20151101053
##
## $`4`
##                  DateTime Junction Vehicles          ID
## 43777 2017-01-01 00:00:00        4        3 20170101004
## 43778 2017-01-01 01:00:00        4        1 20170101014
## 43779 2017-01-01 02:00:00        4        4 20170101024
## 43780 2017-01-01 03:00:00        4        4 20170101034
## 43781 2017-01-01 04:00:00        4        2 20170101044
## 43782 2017-01-01 05:00:00        4        1 20170101054
```

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

```
library(dplyr)
library(ggplot2)

trafficdata$DateTime <- as.Date(trafficdata$DateTime, format = "%Y-%m-%d")

ggplot(trafficdata, aes(x = DateTime, y = Vehicles, color = Junction)) +
  geom_line() +
  labs(
    title = "Traffic Volume Over Time by Junction",
    x = "Date",
    y = "Vehicle Count"
  ) +
  theme_minimal() +
  facet_wrap(~ Junction, scales = "free_y") +
  theme(legend.position = "none")
```



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

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

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

```
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 Fabri
##  $ 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 ...
```

```
# The alexaq file has 3,150 number of observations and 5 numbers of variables or columns, these are the
```

b. group the variations and get the total of each variations.

##Use dplyr package. Show solution and answer.

```
alexavariation <- alexadata %>%
  group_by(variation) %>%
  summarise(total = n())

print(alexavariation)
```

```
## # 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
```

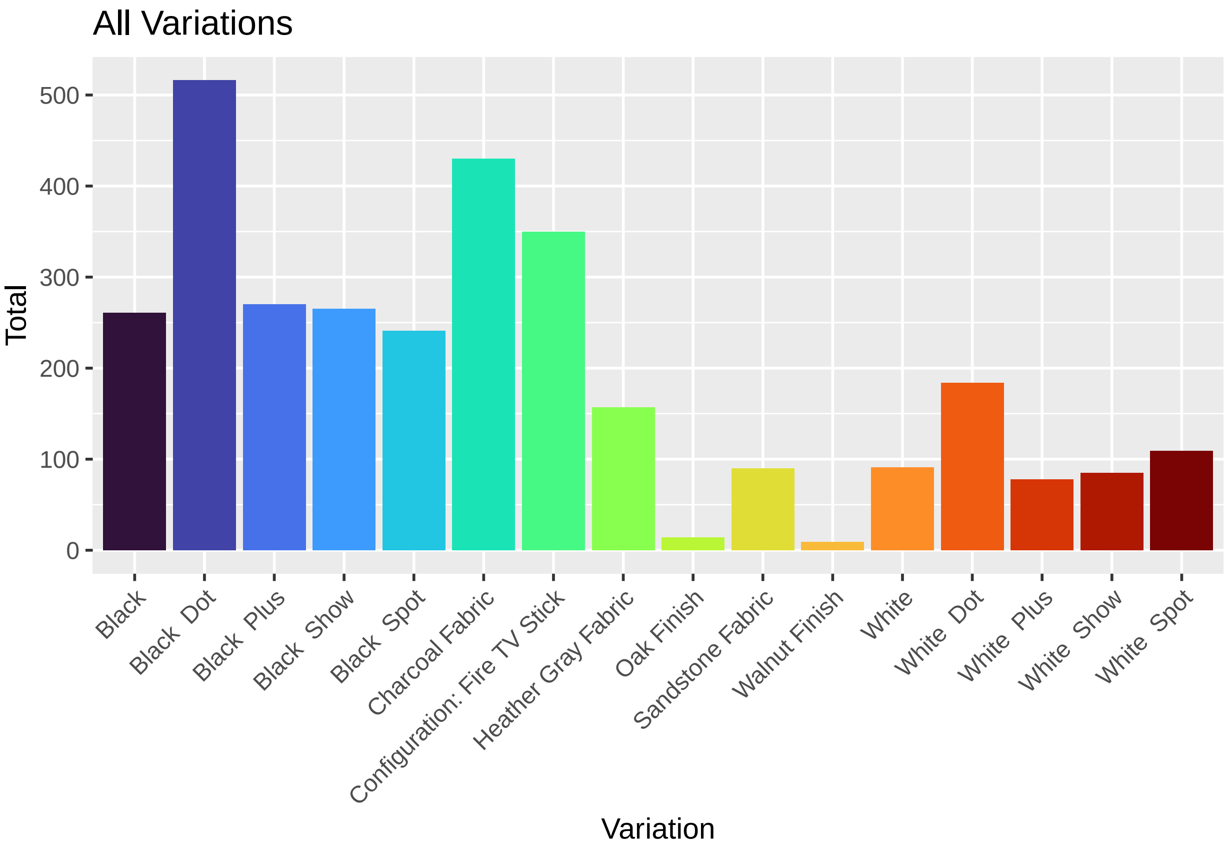
c. Plot the variations using the ggplot() function. What did you observed?

##Complete the details of the graph. Show solution and answer.

```
library(viridis)
```

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

```
library(ggplot2)

ggplot(alexavariation, 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 = 45, hjust = 1)) +
  scale_fill_viridis_d(option = "turbo")
```



```
# Based on my insight, the dark colored variations are the most dominant one,
#most of it have a higher total than those in the white or light colored
#variations.
```

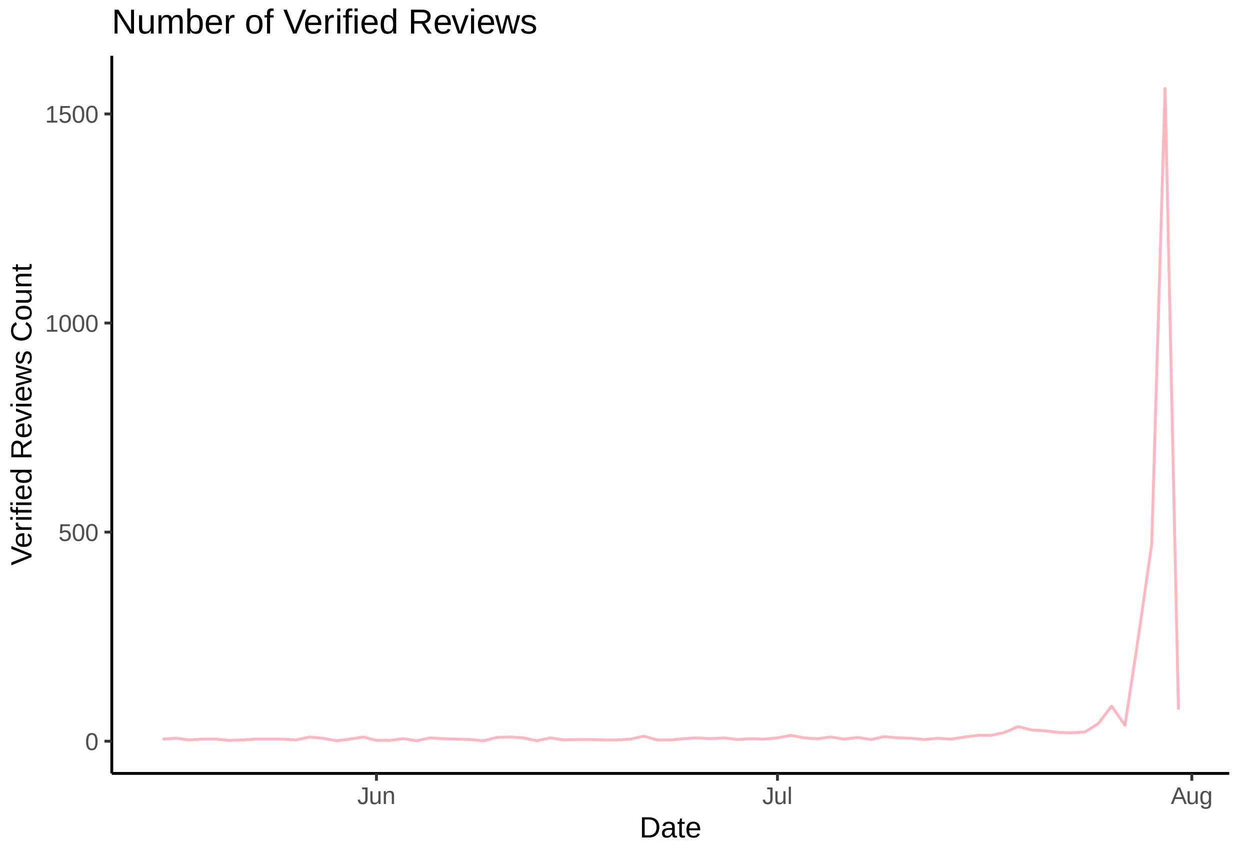
d. Plot a geom_line() with the date and the number of verified reviews.

##Complete the details of the graphs. Show your answer and solution.

```
library(ggplot2)
library(dplyr)

reviews <- alexadata %>%
  filter(!is.na(verified_reviews)) %>%
  group_by(date) %>%
  summarise(reviews_num = n())
print(reviews)
```

```
## # A tibble: 77 x 2
##    date                reviews_num
##    <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
```

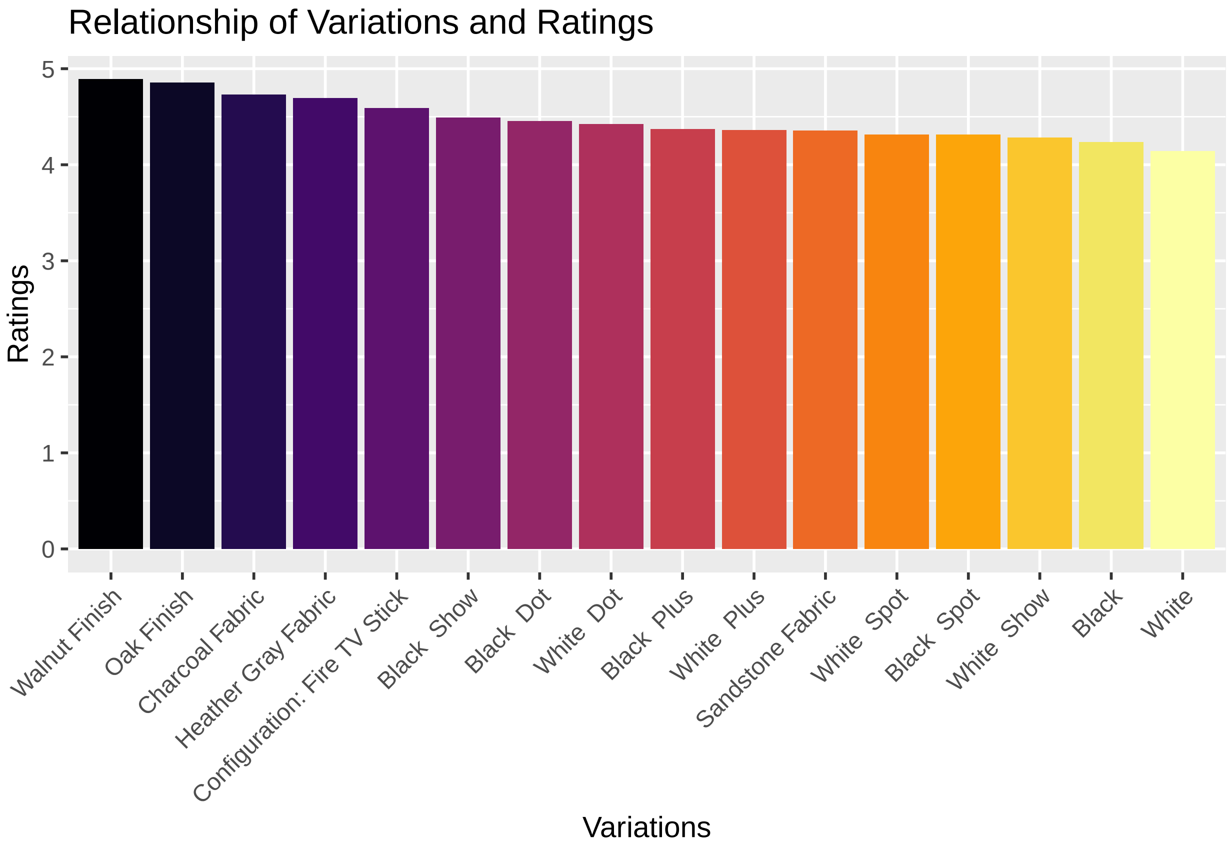
```
ggplot(reviews, aes(x = date, y = reviews_num)) +
  geom_line(color = "lightpink") +
  labs(title = "Number of Verified Reviews",
       x = "Date",
       y = "Verified Reviews Count") +
  theme_classic()
```



e. Get the relationship of variations and ratings. Which variations got the

##most highest rating? Plot a graph to show its relationship. ## Show your solution and answer.

```
library(forcats)
ratings_data <- alexadata %>%
  group_by(variation) %>%
  summarise(avg_rating = mean(rating))

ratings_data <- ratings_data %>%
  mutate(variation = fct_reorder(variation, avg_rating, .desc = TRUE))

ggplot(ratings_data, aes(x = variation, y = avg_rating, fill = variation)) +
  geom_bar(stat = "identity") +
  labs(
    title = "Relationship of Variations and Ratings",
    x = "Variations",
    y = "Ratings"
  ) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  theme(legend.position = "none") +
  scale_fill_viridis_d(option = "inferno")
```



```
# The top 3 variations that got highest ratings are the Walnut Finish followed
# by Oak Finish and Charcoal Fabric.
```