RWorksheet_Sorenio#4c

2024-11-04

```
# 1a
library(readr)
mpg <- read.csv("C:/Users/User/Downloads/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" ...
```

```
# 1b Categorical Variables
#The manufacturer, model, trans, drv, fl, and class.

# 1c Continuous Variables
#The displ, cty, and hwy.
```

```
# 2
# Manufacturer that has most models

# 2a
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
manufacturerMod <- mpg
manufacturerMod <- aggregate(model ~ manufacturer, data = manufacturerMod, FUN = function(x) length(uni
manufacturerMod <- manufacturerMod[order(-manufacturerMod$model), ]
manufacturerMod
```

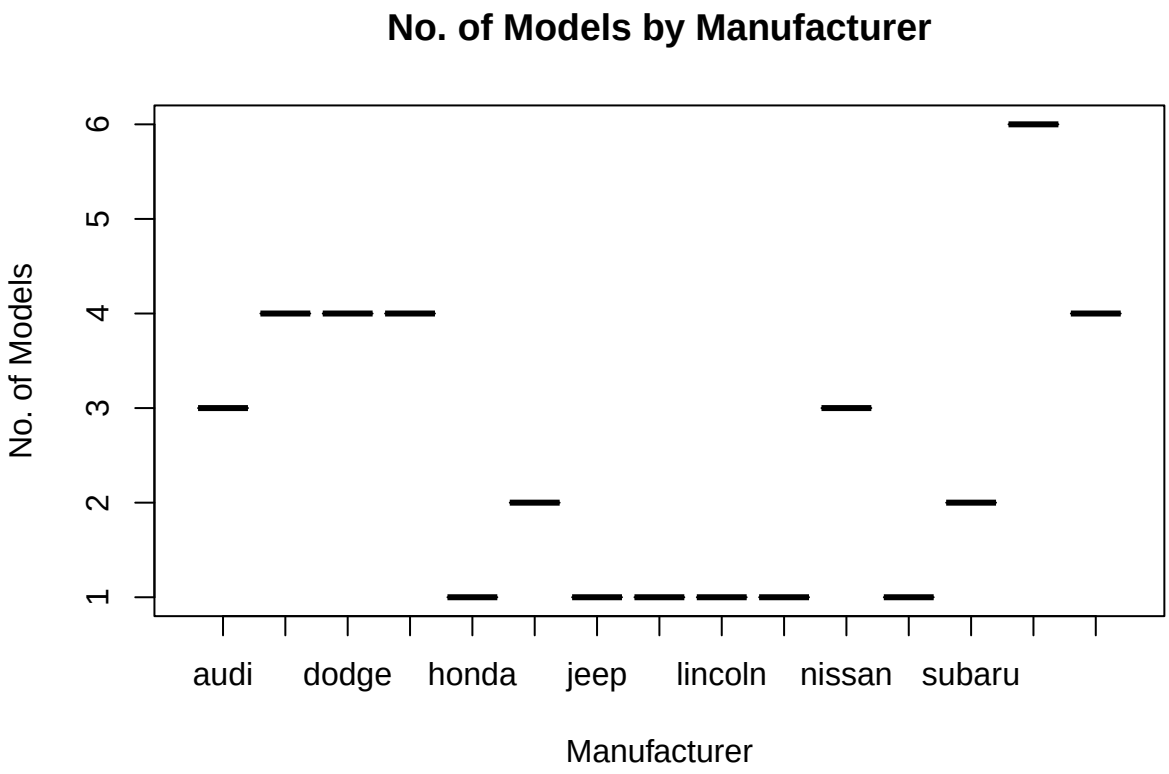
```
##    manufacturer model
## 14       toyota     6
## 2     chevrolet     4
## 3         dodge     4
## 4          ford     4
## 15   volkswagen     4
## 1           audi     3
## 11       nissan     3
## 6       hyundai     2
## 13       subaru     2
## 5         honda     1
## 7          jeep     1
## 8    land rover     1
## 9       lincoln     1
## 10      mercury     1
## 12      pontiac     1
```

```r
# 2b
manufacturerMod$manufacturer <- as.factor(manufacturerMod$manufacturer)

library(ggplot2)
```

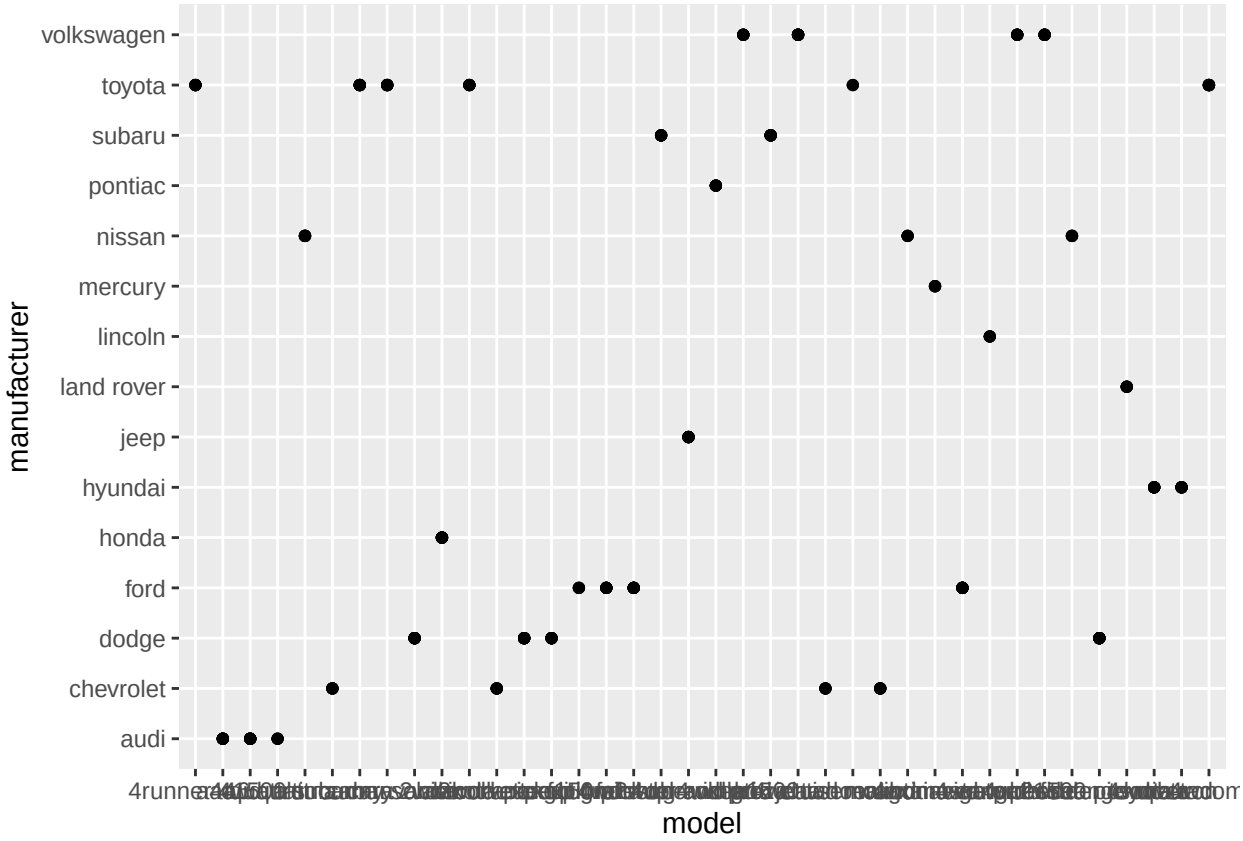
```
##
## Attaching package: 'ggplot2'
```

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

```r
plot(manufacturerMod$manufacturer, manufacturerMod$model,
     main = "No. of Models by Manufacturer",
     xlab = "Manufacturer",
     ylab = "No. of Models")
```



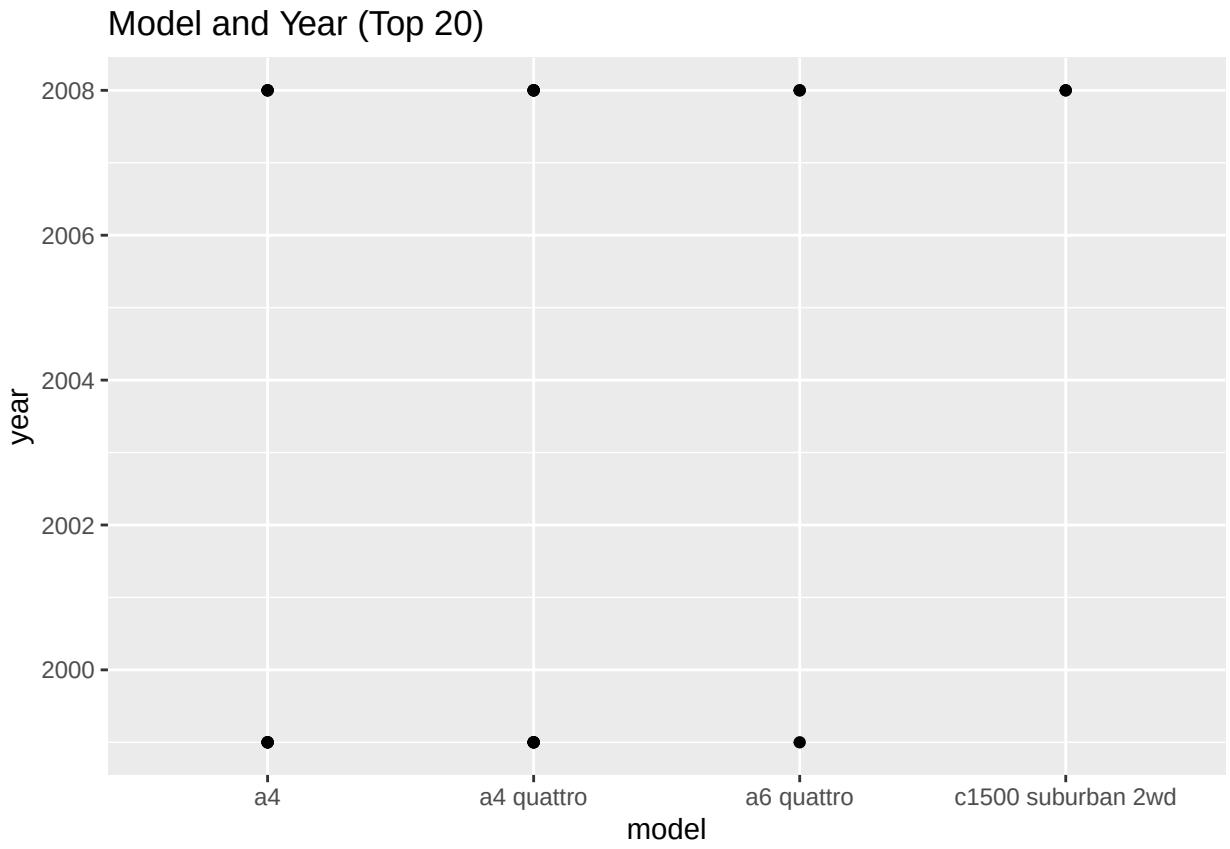

```
# 2
ggplot(mpg, aes(model, manufacturer)) + geom_point()
```




```
# 2a. What does ggplot(mpg, aes(model, manufacturer)) + geom_point() show?

# The distribution of each car model in different manufacturers.

# 2b. For you, is it useful? If not, how could you modify the data to make it more informative?

# The graph is useful, however, improvement  is reccomended. It can be better with the help of proper u
```

3.

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



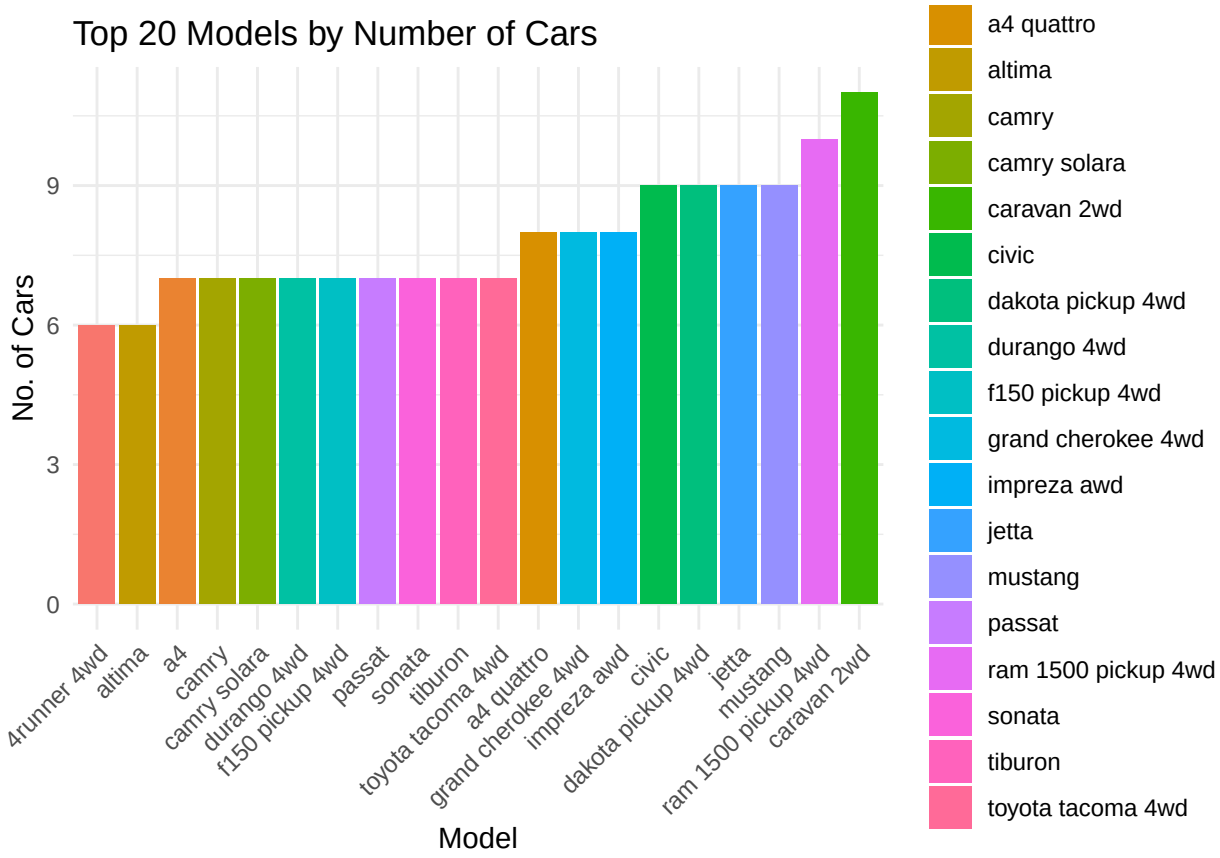
4.

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

```
## # 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
```

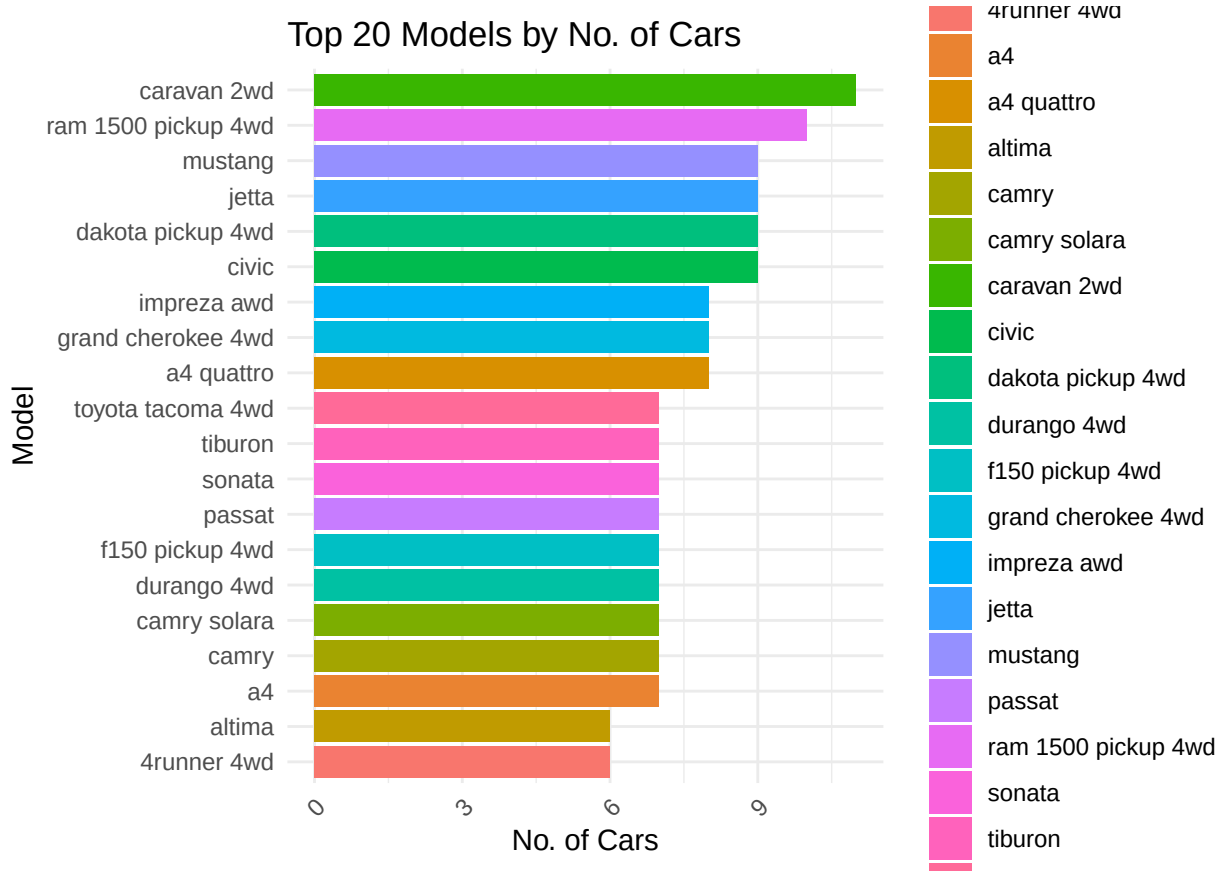
a. Plot using geom_bar() using the top 20 observations only. The graphs shoudl have a title, labels and colors. Show code and results.

```
top20Mod <- modelC %>% head(20)

ggplot(top20Mod, aes(x = reorder(model, count), y = count, fill = model)) +
  geom_bar(stat="identity") +
  labs(title = "Top 20 Models by Number 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 = "lightgreen")
```



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

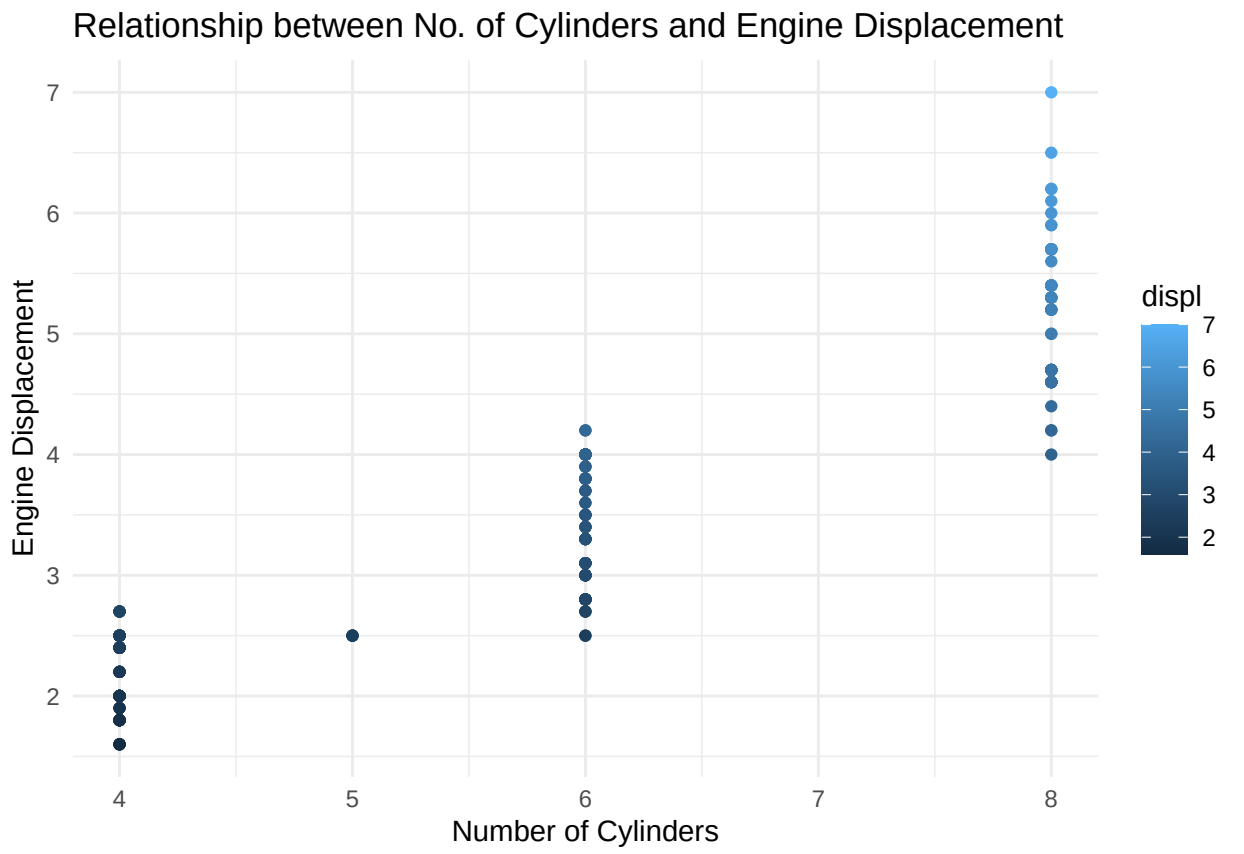
```
ggplot(top20Mod, aes(x = reorder(model, count), y = count, fill = model)) +
  geom_bar(stat="identity") +
  labs(title = "Top 20 Models by No. of Cars", x = "Model", y = "No. of Cars") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_viridis_d(aesthetics = "blue") +
  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.

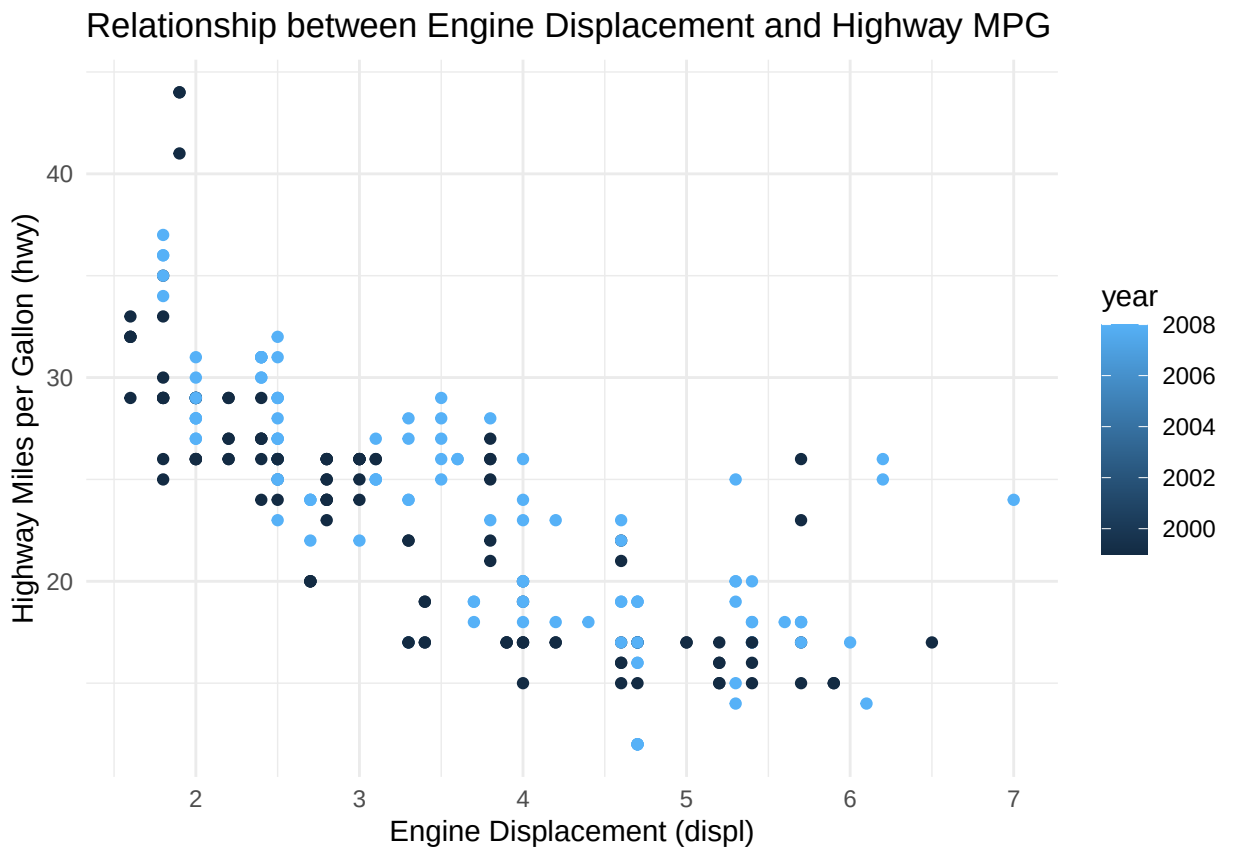
```
ggplot(mpg, aes(x = cyl, y = displ, color = displ)) +
  geom_point() +
  labs(title = "Relationship between No. of Cylinders and Engine Displacement",
    x = "Number of Cylinders",
    y = "Engine Displacement"
  ) +
  theme_minimal()
```



If the displacement increases, the number of cylinders also increases.

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 is displaying a scatter plot of engine displacement vs highway miles per gallon, the points in colors are according to the manufacturing year of the car.

As the engine displacement (displ) increases, fuel efficiency (hwy) decreases. This results in a downward slope of the points. Assigning the year variable to color might highlight subtle trends in fuel efficiency improvements over time.

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

a.

```
traff <- read.csv("C:/Users/User/Downloads/traffic.csv")
str(traff)
```

```
## '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.

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

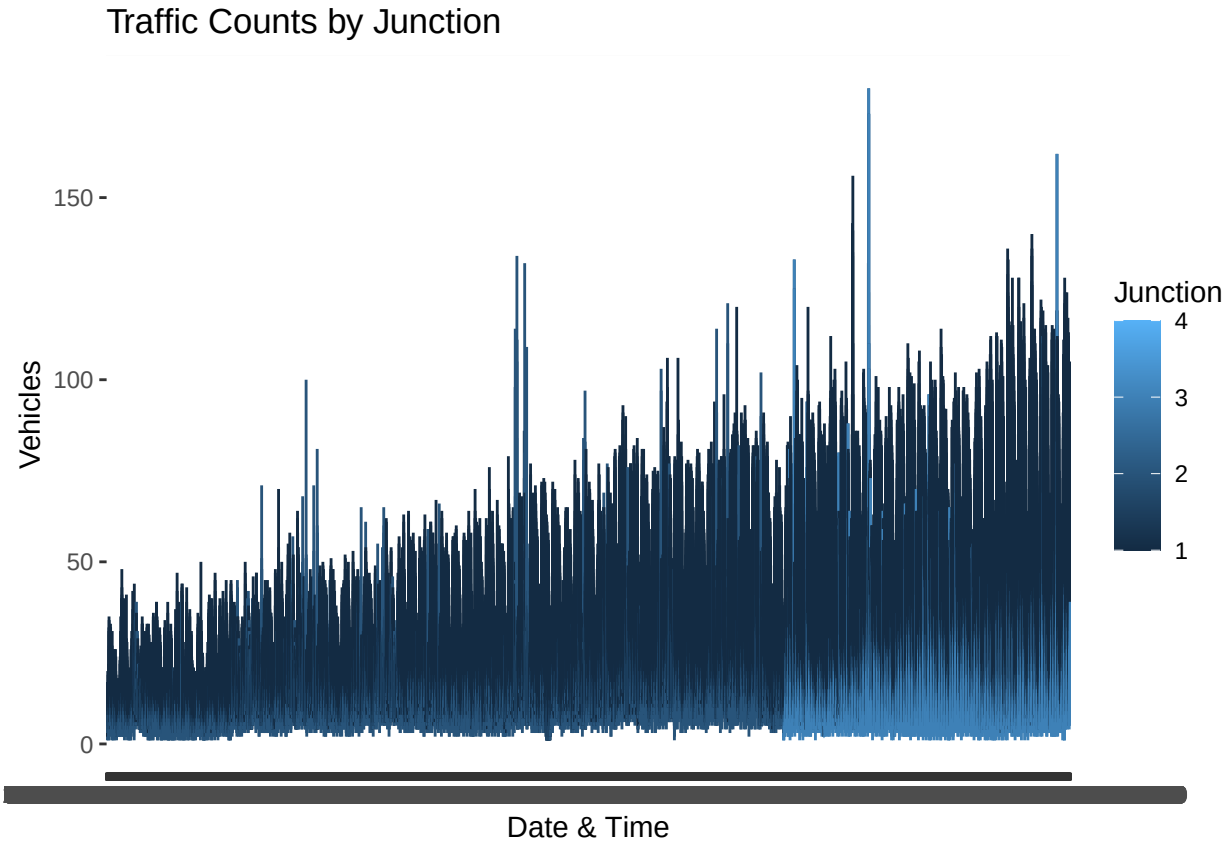
```
##   Junction
## 1        1
## 2        1
## 3        1
## 4        1
## 5        1
## 6        1
```

```
tail(junction)
```

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

c.

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



7.

```
library("readxl")
alexa <- read_xlsx("C:/Users/User/Downloads/alexa_file.xlsx")
alexa
```

```
## # A tibble: 3,150 x 5
##    rating date                variation           verified_reviews      feedback
##     <dbl> <dttm>              <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
```

a.

```
nrow(alexa)
```

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

```
ncol(alexa)
```

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

It has a total of 3150 observations and 5 columns.

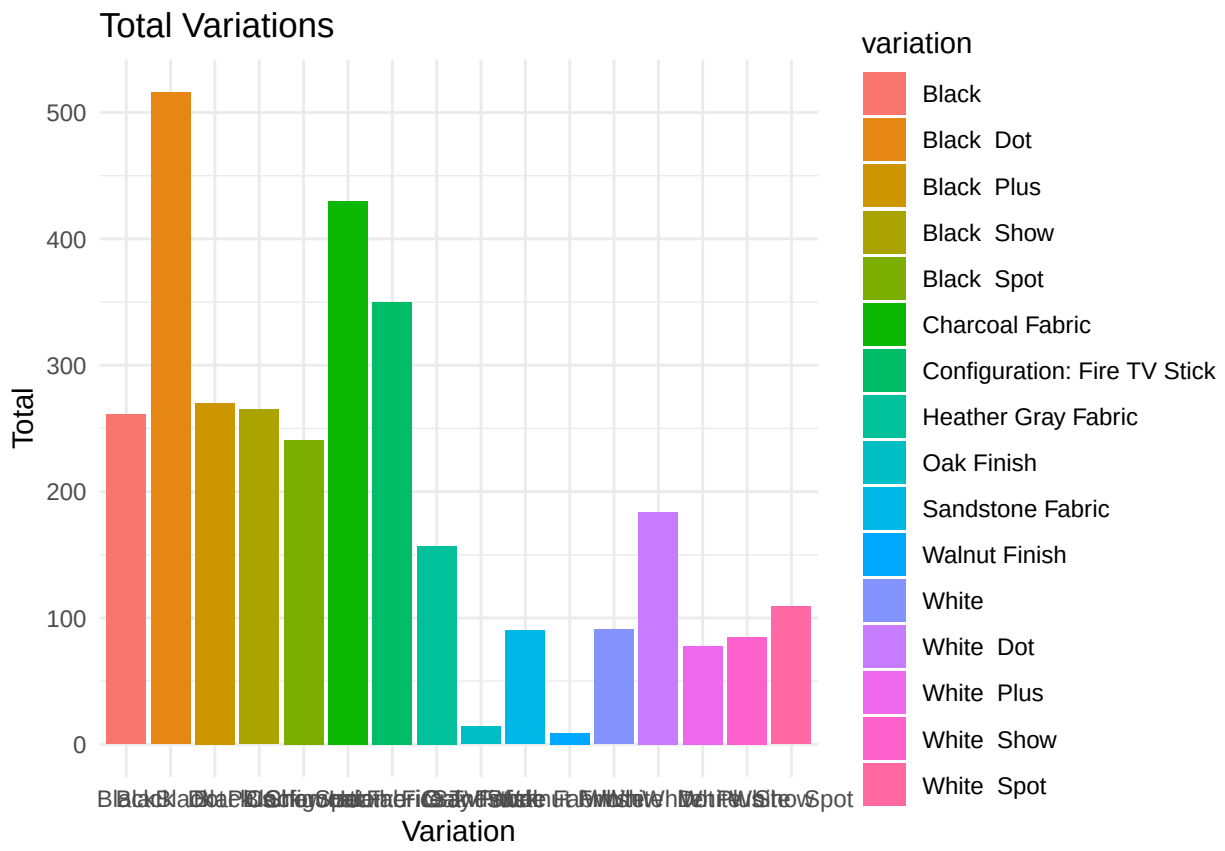
b.

```
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.

```
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 shows the total counts of different "Variations," with some being much more common than others. The Black Dot variation, shown by the tallest orange bar, has over 500 instances, making it the most popular. Overall, the chart highlights a big difference in how often each variation is chosen, with some being favored much more than others.

d.

```
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 = "red") +
```

```
  labs(title = "Verified Reviews Over Time", x = "Date", y = "Verified Reviews") +
  theme_minimal()
```

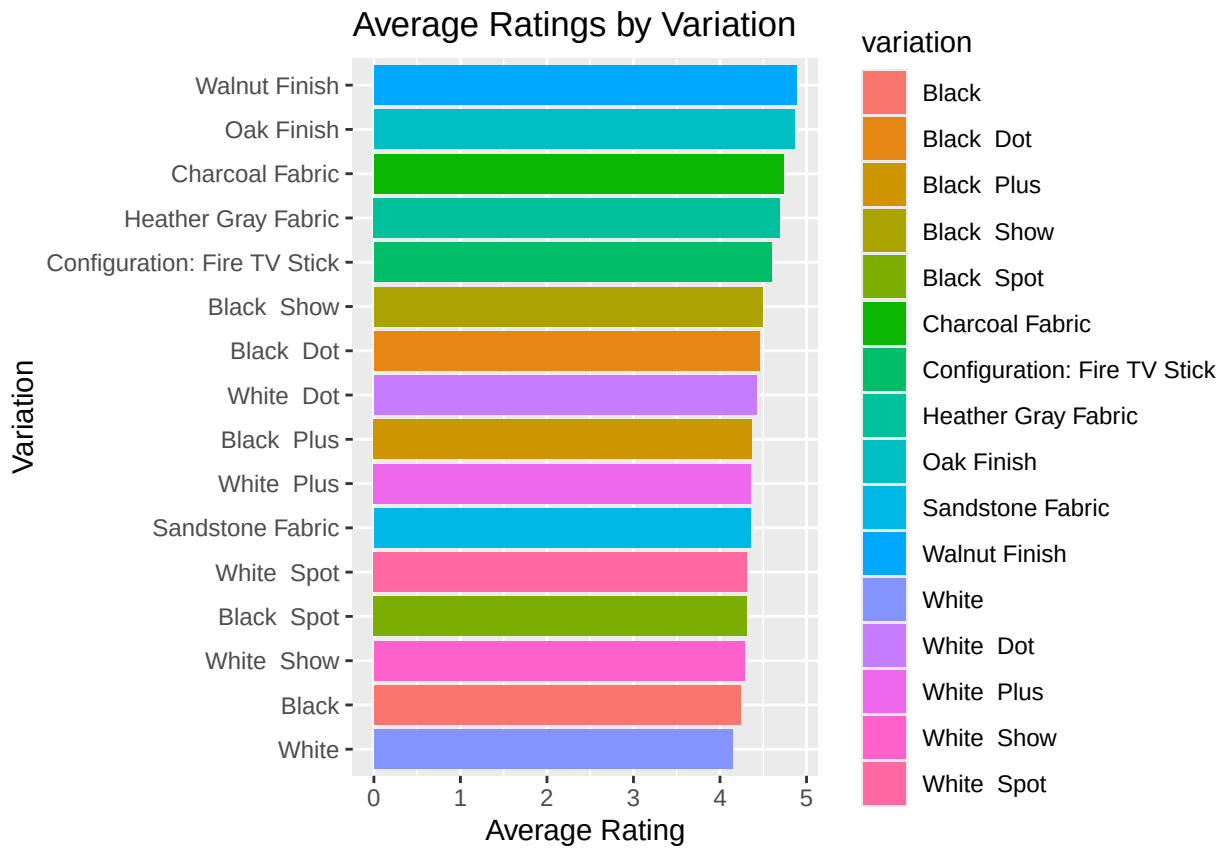
e.

```
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
```

```
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()
```



Highest ratings are Walnut Finish, Oak Finish, and Charcoal Fabric.