RWorksheet_asenjo#4c

Samuel Asenjo

2024-11-04

1. a.

```
mpg <- read.csv("/cloud/project/Worksheet 4/mpg.csv")
```

b. The categorical variables in the data set are manufacturer, model name, type of transmission, drive type, fuel type, number of cylinders, and, vehicle class.

c.The continuous variables in the data set are displacement, year of manufacturing, city mileage, highway mileage.

2.1.

```
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
```

```
library(ggplot2)
```

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

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

```
manufacturer_count <- mpg %>%
  group_by(manufacturer) %>%
  summarise(model_count = n_distinct(model)) %>%
  arrange(desc(model_count))

top_manufacturer <- manufacturer_count[1, ]

model_variation <- mpg %>%
  group_by(model) %>%
  summarise(variation_count = n()) %>%
  arrange(desc(variation_count))

top_model <- model_variation[1, ]
```

```r
top_manufacturer
```

```
## # A tibble: 1 x 2
##   manufacturer model_count
##   <chr>              <int>
## 1 toyota                 6
```

```r
top_model
```

```
## # A tibble: 1 x 2
##   model        variation_count
##   <chr>                  <int>
## 1 caravan 2wd               11
```

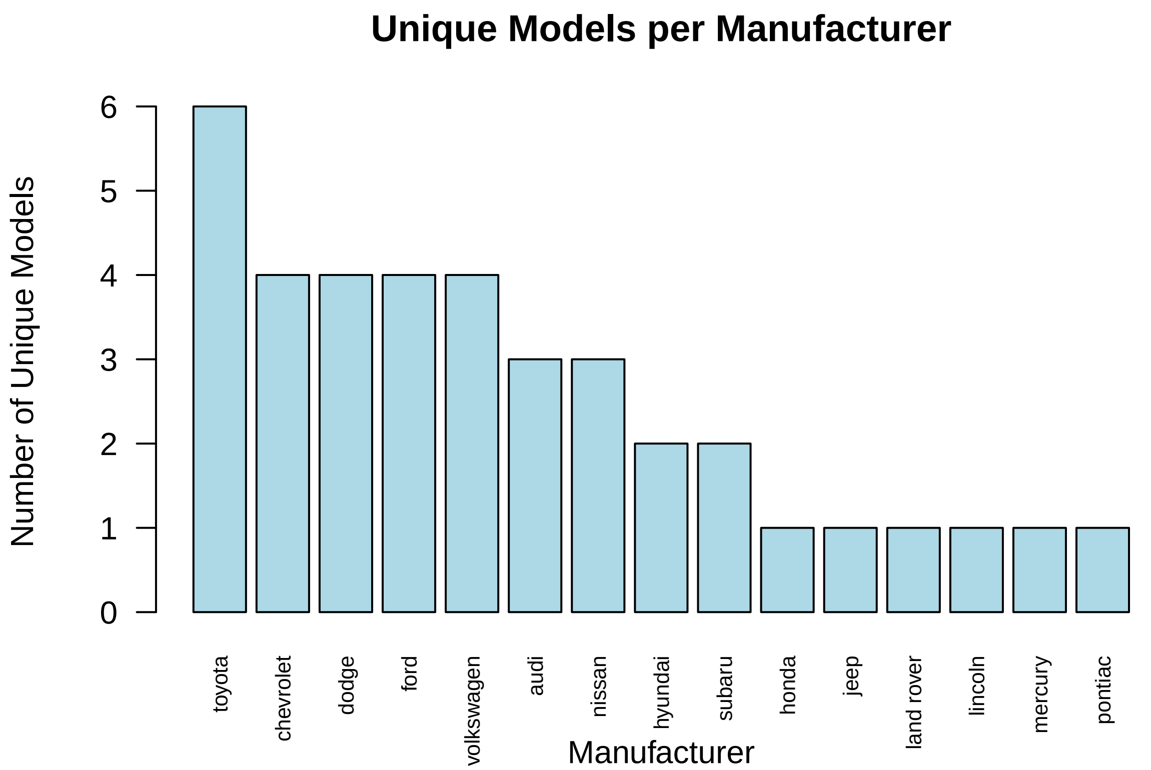
a.

```r
  mile <- mpg %>%
  group_by(manufacturer) %>%
  summarise(unique_models = n_distinct(model)) %>%
  arrange(desc(unique_models))
mile
```

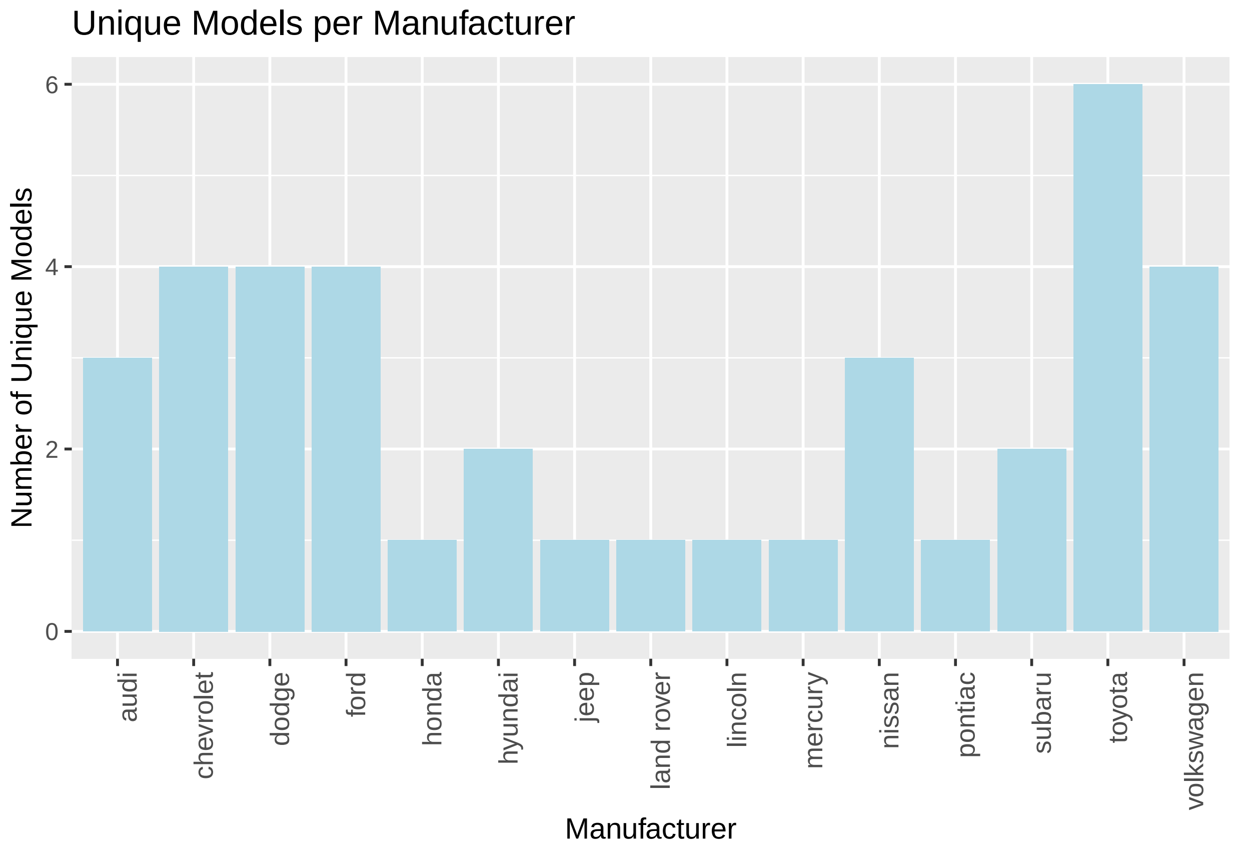
```
## # A tibble: 15 x 2
##    manufacturer unique_models
##    <chr>                <int>
##  1 toyota                   6
##  2 chevrolet                4
##  3 dodge                    4
##  4 ford                     4
##  5 volkswagen               4
##  6 audi                     3
##  7 nissan                   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.

```r
library(dplyr)
barplot(mile$unique_models,
        names.arg = mile$manufacturer,
        col = "lightblue",
        main = "Unique Models per Manufacturer",
        xlab = "Manufacturer",
        ylab = "Number of Unique Models",
        las = 2,
        cex.names = 0.7)
```

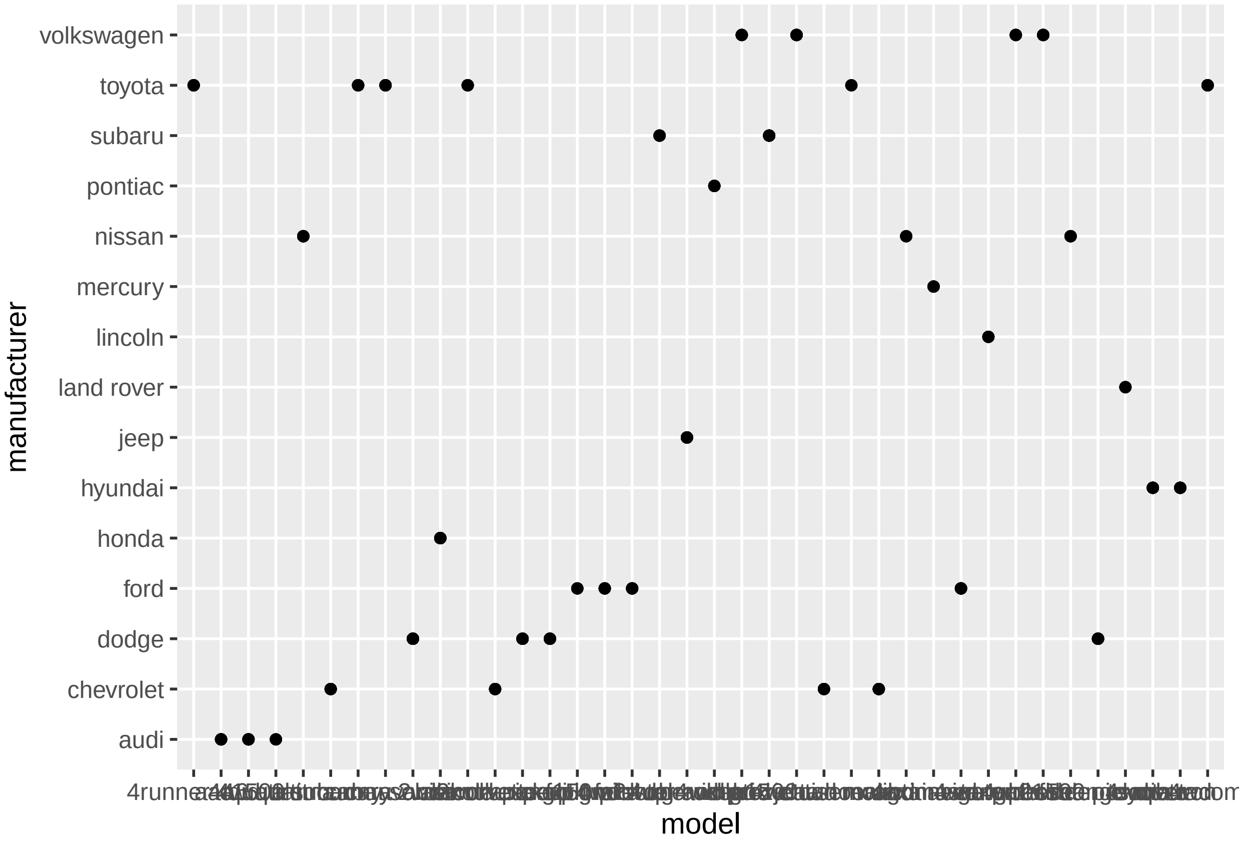


```r
library(ggplot2)

ggplot(mile, aes(x = manufacturer, y = unique_models)) +
  geom_bar(stat = "identity", fill = "lightblue") +
  ggtitle("Unique Models per Manufacturer") +
  xlab("Manufacturer") +
  ylab("Number of Unique Models") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, size = 10))
```



2.2. a. The code The plot created by ggplot(mpg, aes(model, manufacturer)) + geom_point() is a scatter plot that shows the relationship between car manufacturers (y-axis) and their models (x-axis), with each point representing a specific model from a manufacturer. It visually displays which models belong to which manufacturer, but due to many categorical values, it can appear crowded.

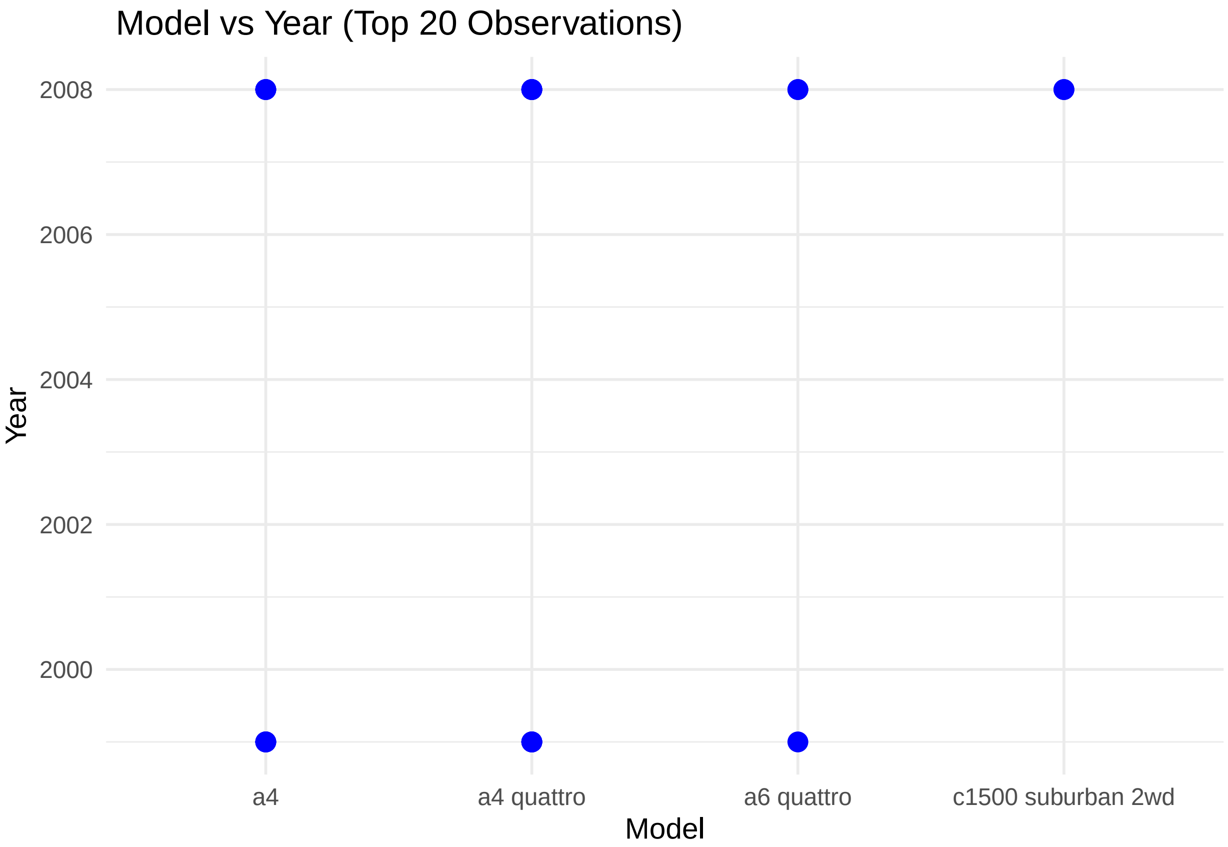
```
ggplot(mpg, aes(model, manufacturer)) + geom_point()
```



b. No, the original plot isn't very useful because it is overcrowded and doesn't offer clear insights with categorical data on both axes. A better approach would be to group the data by manufacturer, count the number of unique models, and create a bar plot to show how many models each manufacturer produces.

3.

```
library(ggplot2)
library(dplyr)


mpg_top20 <- mpg %>% head(20)

ggplot(mpg_top20, aes(x = model, y = year)) +
  geom_point(color = "blue", size = 3) +
  labs(title = " Model vs Year (Top 20 Observations)",
       x = "Model",
       y = "Year") +
  theme_minimal()
```



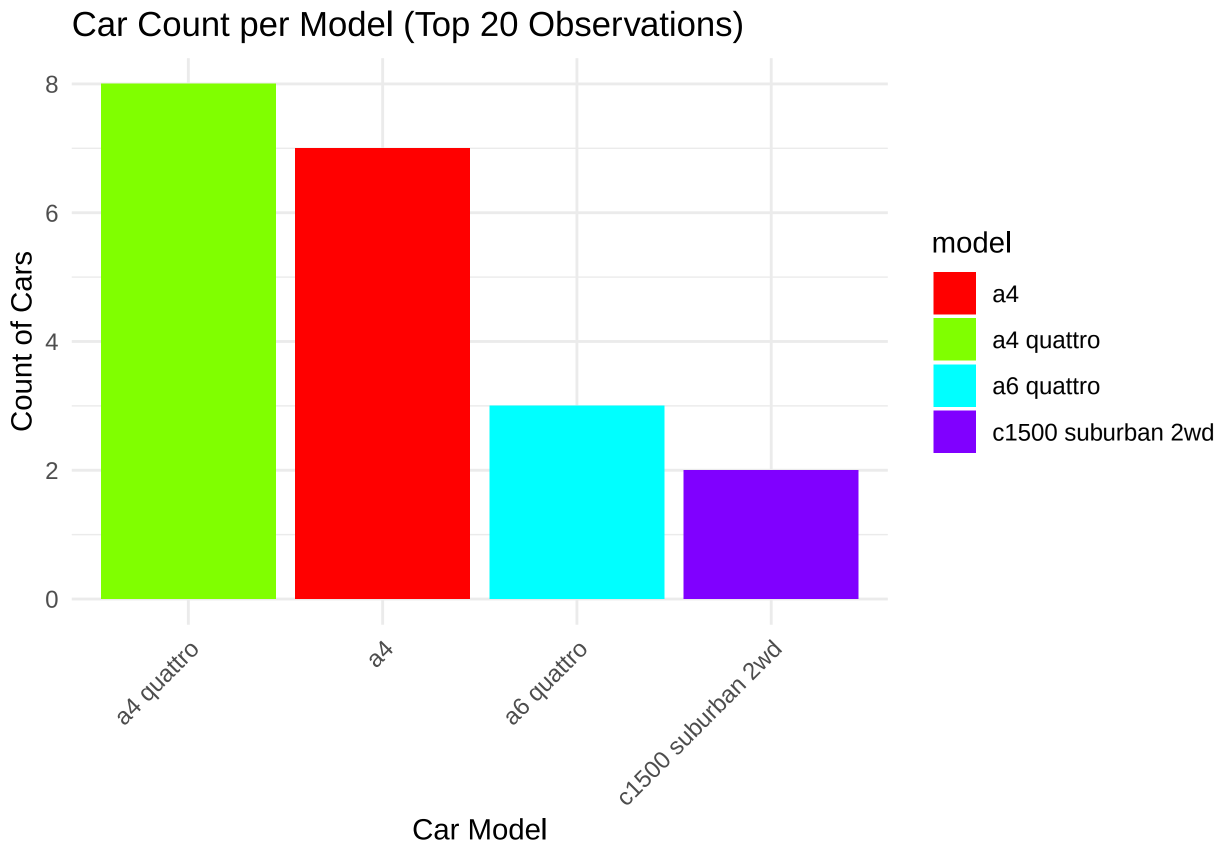
4.

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


cars_per_model <- mpg %>%
  group_by(model) %>%
  summarise(car_count = n()) %>%
  arrange(desc(car_count))

cars_per_model
```

```
## # A tibble: 38 x 2
##    model                car_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
```

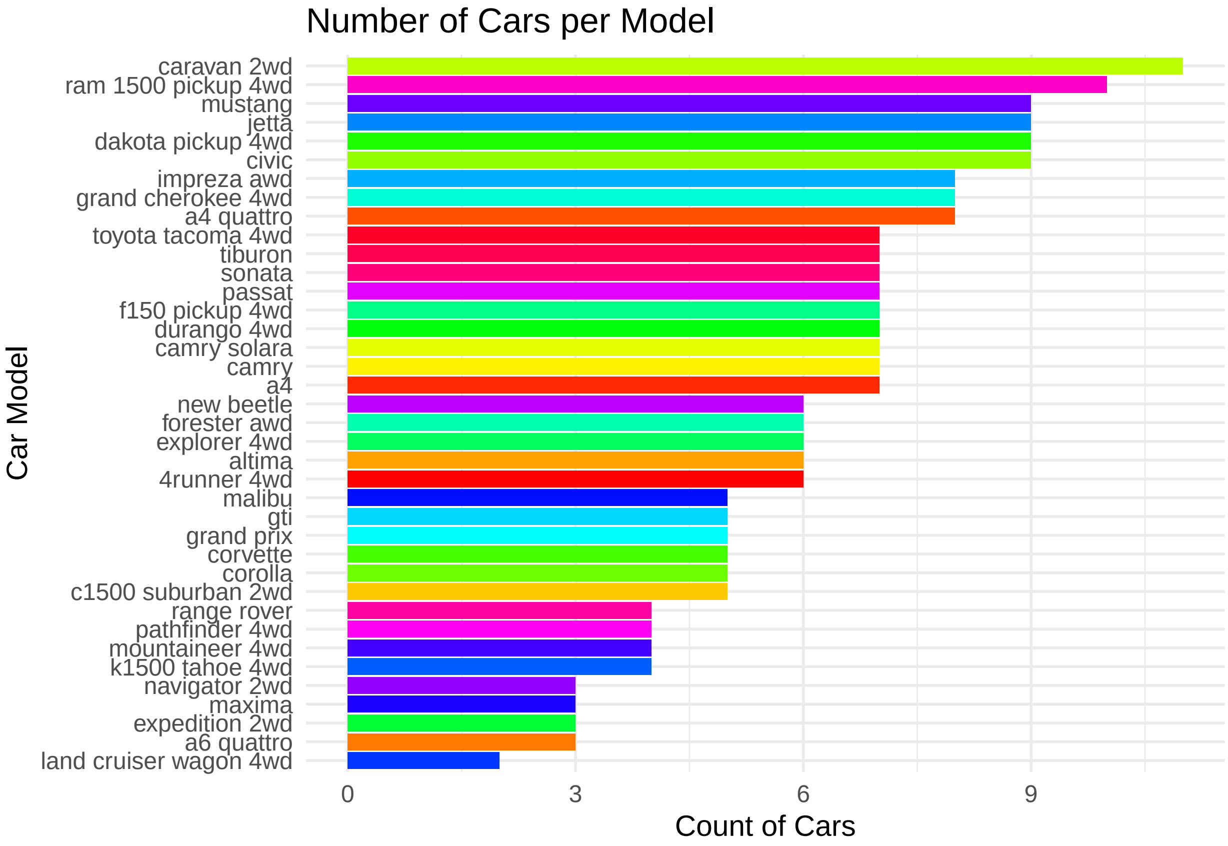
a.

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



mpg_top20 <- mpg %>% head(20)

model_counts <- mpg_top20 %>%
  group_by(model) %>%
  summarise(car_count = n()) %>%
  arrange(desc(car_count))

ggplot(model_counts, aes(x = reorder(model, -car_count), y = car_count, fill = model)) +
  geom_bar(stat = "identity") +
  labs(title = "Car Count per Model (Top 20 Observations)",
       x = "Car Model",
       y = "Count of Cars") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_manual(values = rainbow(nrow(model_counts)))
```



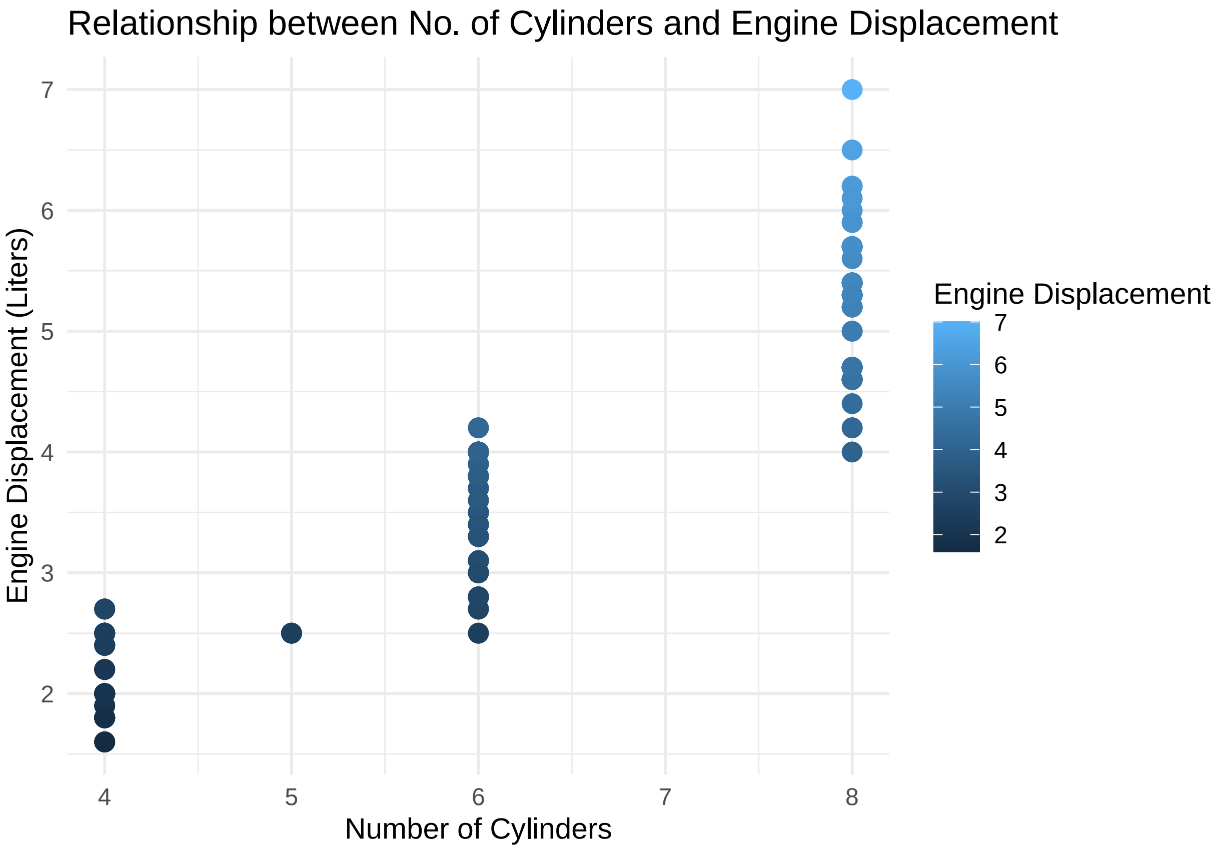
b.

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



model_counts <- mpg %>%
  group_by(model) %>%
  summarise(car_count = n()) %>%
  arrange(desc(car_count))

ggplot(model_counts, aes(x = reorder(model, car_count), y = car_count, fill = model)) +
  geom_bar(stat = "identity") +
  coord_flip() +  # Flips the coordinates to make horizontal bars
  labs(title = "Number of Cars per Model",
       x = "Car Model",
       y = "Count of Cars") +
  theme_minimal() +
  theme(legend.position = "none") +  # Hide the legend
  scale_fill_manual(values = rainbow(nrow(model_counts)))
```



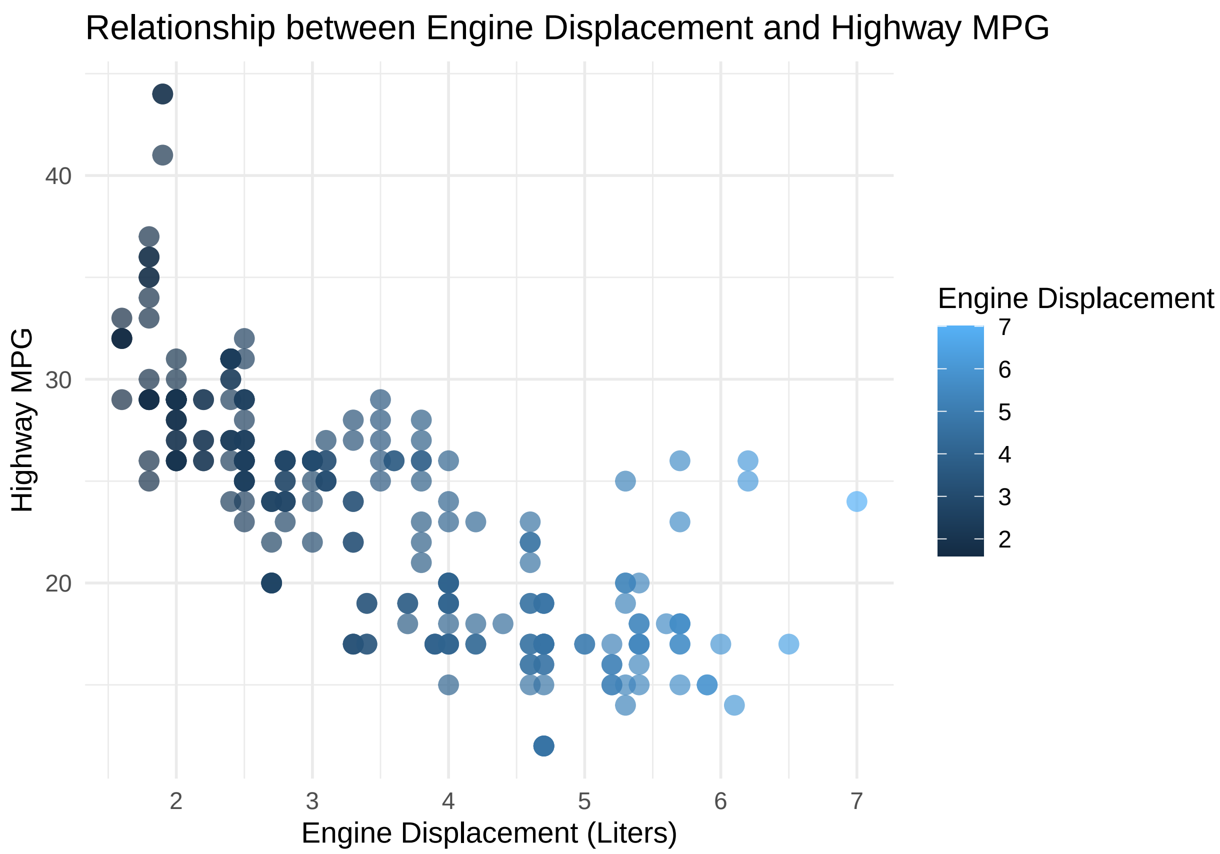
5. a. The relationship between the number of cylinders and engine displacement is generally positive, meaning that as the number of cylinders increases, the engine displacement tends to increase as well. This suggests that cars with more cylinders typically have larger engines in terms of displacement.

```r
library(ggplot2)


ggplot(mpg, aes(x = cyl, y = displ, color = displ)) +
  geom_point(size = 3) +
  labs(title = "Relationship between No. of Cylinders and Engine Displacement",
       x = "Number of Cylinders",
       y = "Engine Displacement (Liters)",
       color = "Engine Displacement") +
  theme_minimal()
```



6.1. The scatter plot reveals a negative correlation between engine displacement (displ) and highway miles per gallon (hwy), indicating that as engine displacement increases, highway MPG tends to decrease. This output arises because larger engines typically consume more fuel, leading to lower fuel efficiency on the highway, reflecting the trade-off between engine size and fuel economy in vehicles.

```r
library(ggplot2)



ggplot(mpg, aes(x = displ, y = hwy, color = displ)) +
  geom_point(size = 3, alpha = 0.7) +
  labs(title = "Relationship between Engine Displacement and Highway MPG",
       x = "Engine Displacement (Liters)",
       y = "Highway MPG",
       color = "Engine Displacement") +
  theme_minimal()
```



6.2.

```
traffic <- read.csv("/cloud/project/Worksheet 4/traffic.csv")
```

a.

```
num_observations <- nrow(traffic)
print(paste("Number of observations:", num_observations))
```

```
## [1] "Number of observations: 48120"
```

```
variables <- colnames(traffic)
print("Variables in the dataset:")
```

```
## [1] "Variables in the dataset:"
```

```
print(variables)
```

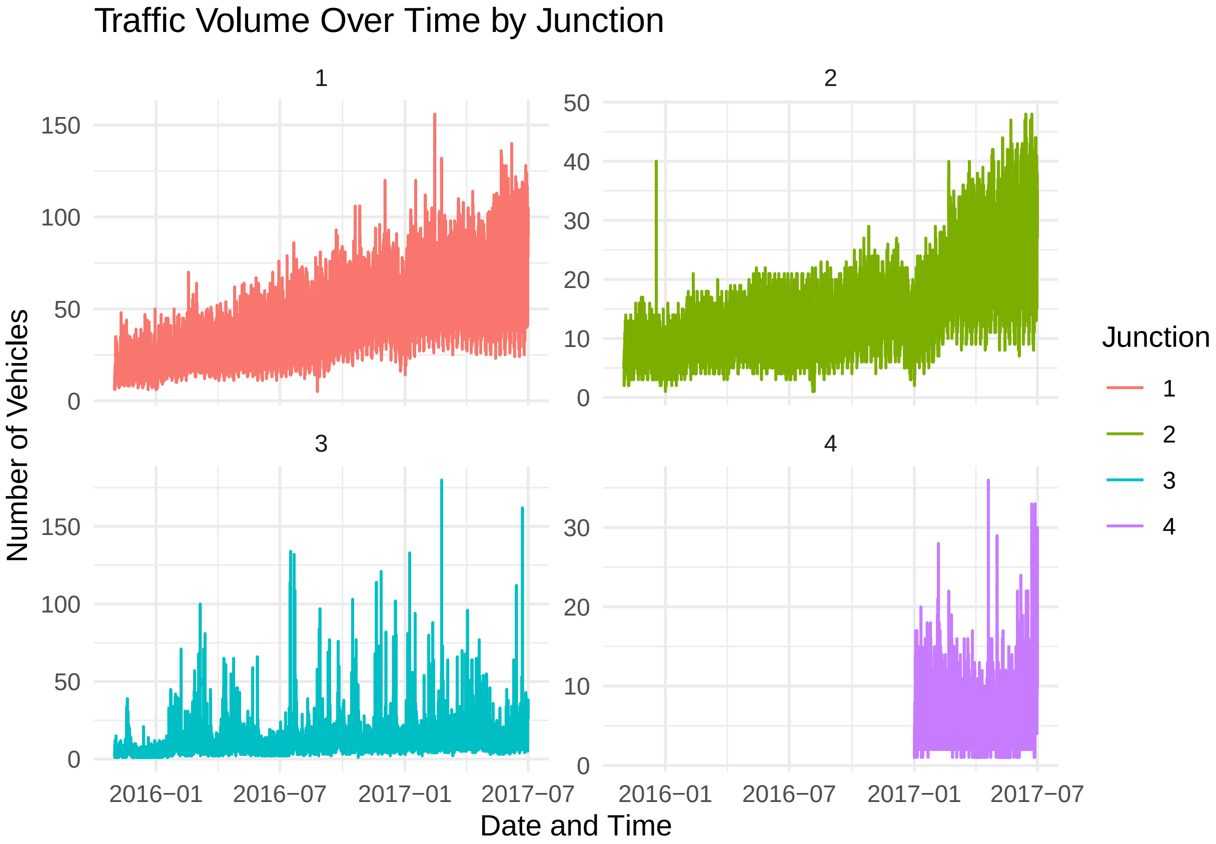
```
## [1] "DateTime" "Junction" "Vehicles" "ID"
```

b.

```
junction_subsets <- split(traffic, traffic$Junction)
```

```
library(ggplot2)
library(dplyr)


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

ggplot(traffic, aes(x = DateTime, y = Vehicles, color = factor(Junction))) +
```

```r
  geom_line() +
  labs(title = "Traffic Volume Over Time by Junction",
       x = "Date and Time",
       y = "Number of Vehicles",
       color = "Junction") +
  theme_minimal() +
  facet_wrap(~ Junction, scales = "free_y")
```



7.

```r
library(readxl)

alexa <- read_excel("/cloud/project/Worksheet 4/alexa_file.xlsx")
```

a.

```r
n <- nrow(alexa)
print(paste("Number of observations:", n))
```

```
## [1] "Number of observations: 3150"
```

```r
v <- colnames(alexa)
print("Variables in the dataset:")
```

```
## [1] "Variables in the dataset:"
```

```r
print(variables)
```

```
## [1] "DateTime" "Junction" "Vehicles" "ID"
```

b.

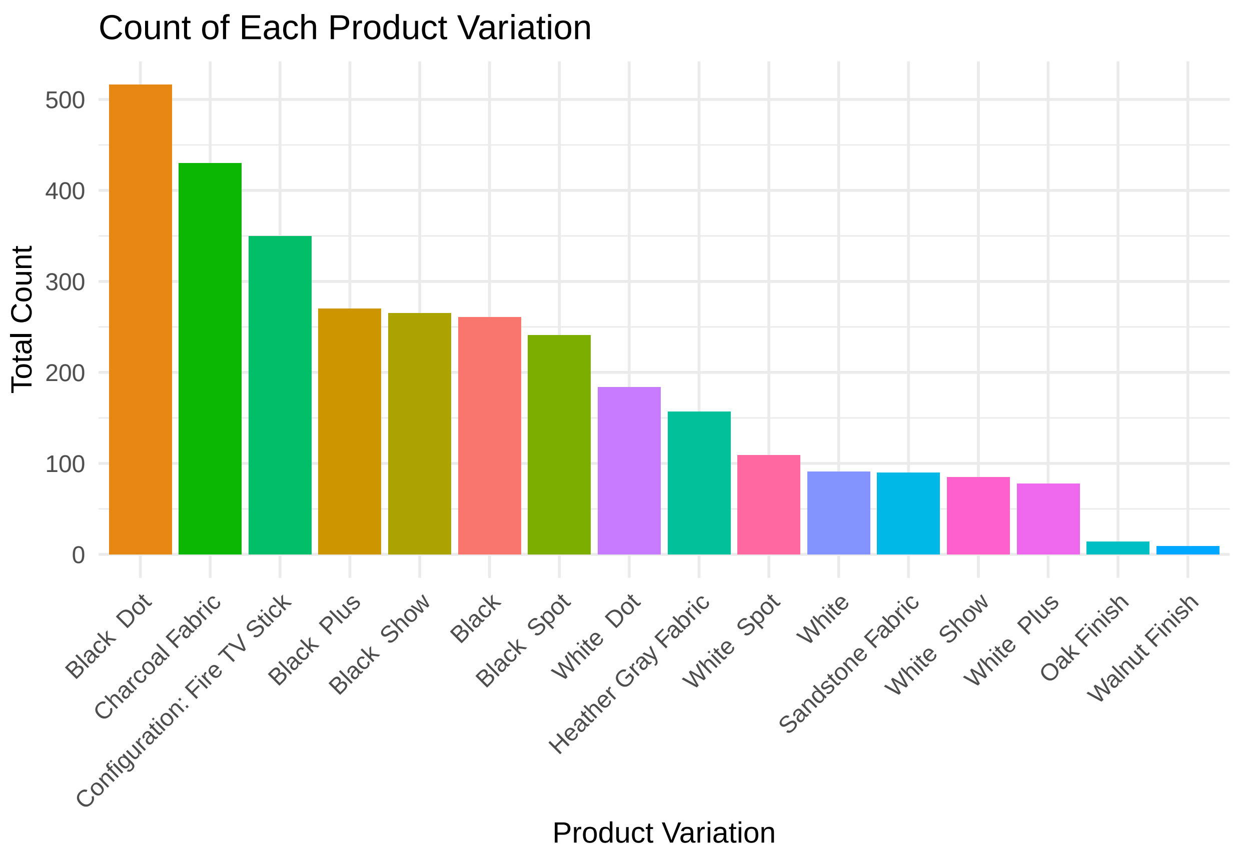
```r
library(dplyr)


variation_counts <- alexa %>%
  group_by(variation) %>%
  summarise(total = n())

print(variation_counts)
```

```
## # 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. The plot shows which product variations are most popular, with a clear lead for some variations over others. It highlights consumer preferences for specific variations in the data.

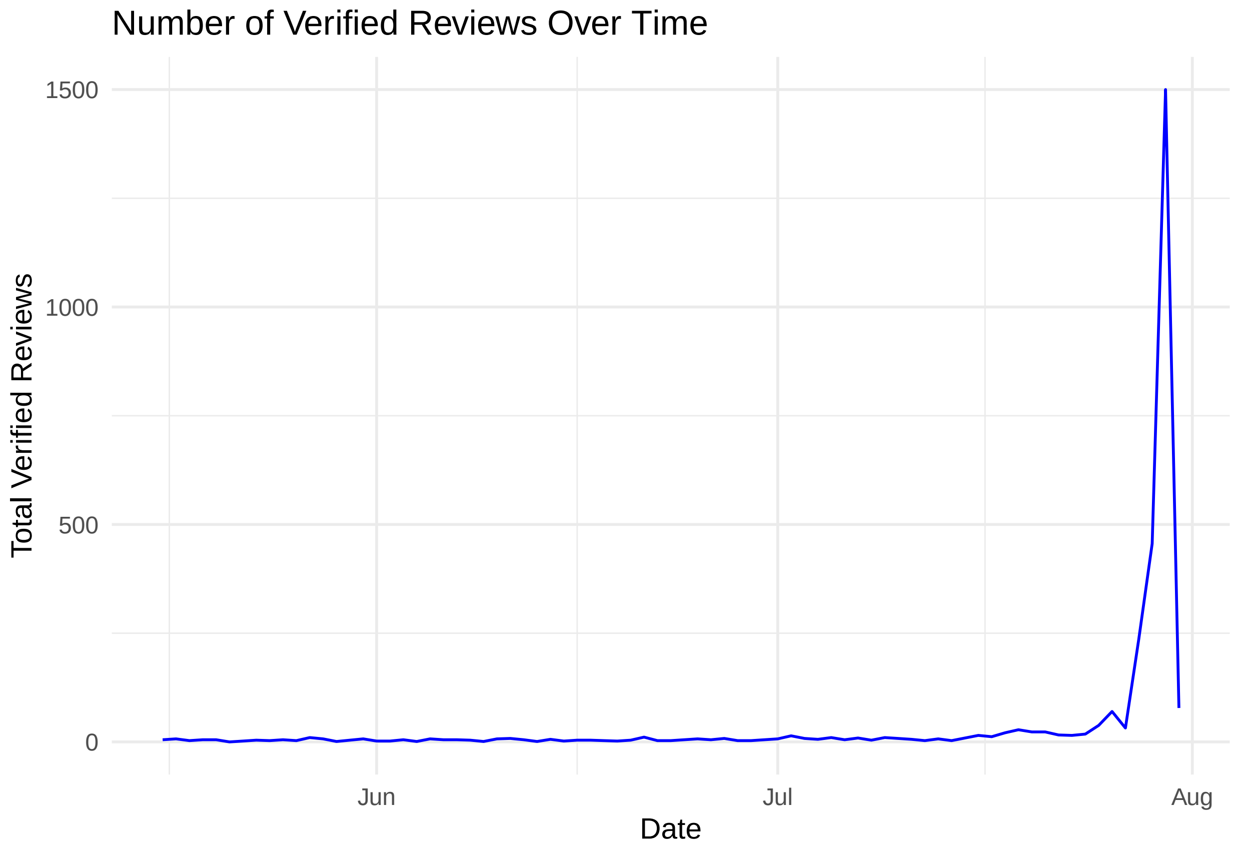
```r
ggplot(variation_counts, aes(x = reorder(variation, -total), y = total, fill = variation)) +
  geom_bar(stat = "identity") +
  labs(title = "Count of Each Product Variation",
       x = "Product Variation",
       y = "Total Count") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  guides(fill = FALSE)
```

```
## Warning: The `<scale>` argument of `guides()` cannot be `FALSE`. Use "none" instead as
## of ggplot2 3.3.4.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



d.

```r
alexa$date <- as.Date(alexa$date)

daily_reviews <- alexa %>%
  group_by(date) %>%
  summarise(total_verified_reviews = sum(feedback))

ggplot(daily_reviews, aes(x = date, y = total_verified_reviews)) +
  geom_line(color = "blue") +
  labs(title = "Number of Verified Reviews Over Time",
       x = "Date",
       y = "Total Verified Reviews") +
  theme_minimal()
```



e.

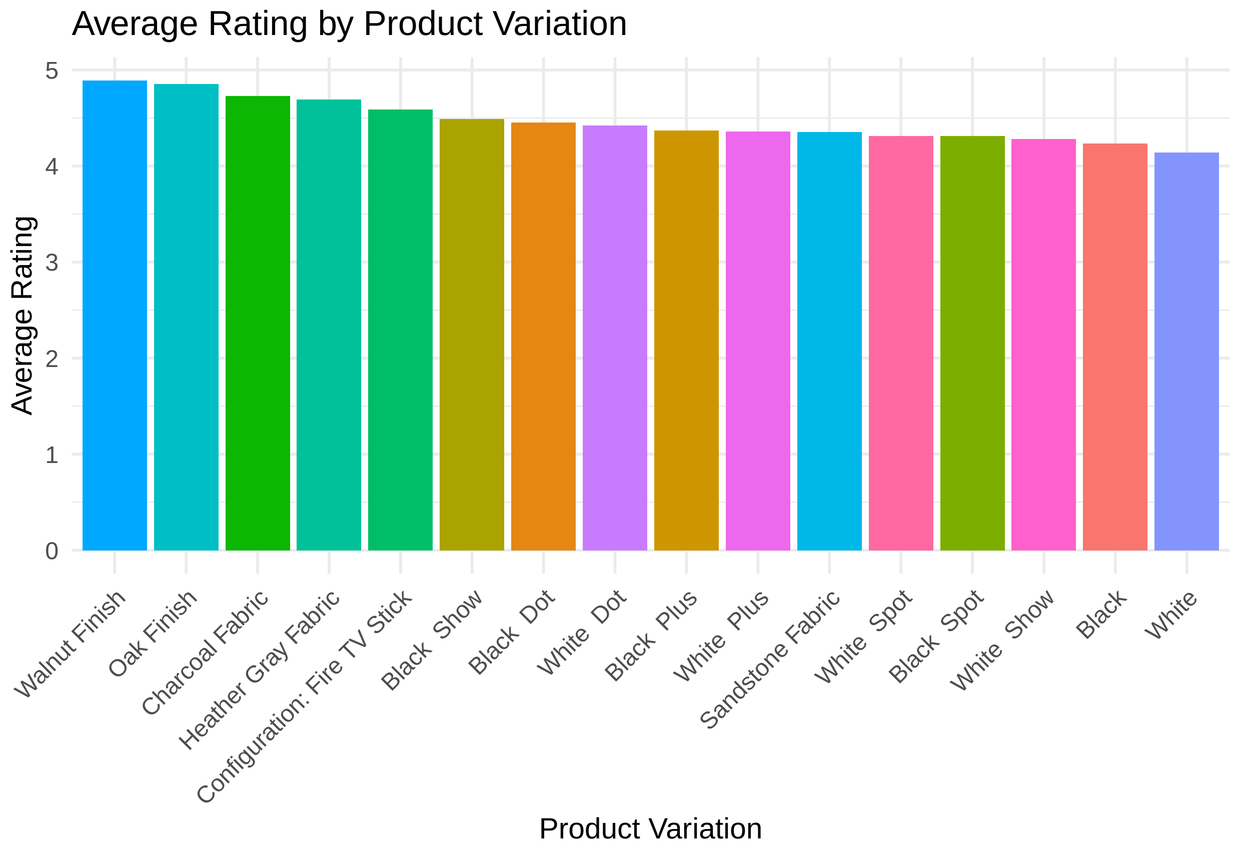
```
library(dplyr)
library(ggplot2)

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

arrange(desc(average_rating))

variation_ratings
```

```
## # A tibble: 16 x 2
##    variation                    average_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
```

```
hv <- variation_ratings %>%
  slice(1)
hv
```

```
## # A tibble: 1 x 2
##   variation     average_rating
##   <chr>                  <dbl>
## 1 Walnut Finish           4.89
```

```
ggplot(variation_ratings, aes(x = reorder(variation, -average_rating), y = average_rating, fill = varia
  geom_bar(stat = "identity") +
  labs(title = "Average Rating by Product Variation",
       x = "Product Variation",
       y = "Average Rating") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  guides(fill = FALSE)
```


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