RWorksheet_Sobusa#4c.Rmd

Nexon Sobusa

2024-11-02

```r
# 1. Using the mpg Dataset
# a. Importing mpg.csv File into R
mpg_data <- read.csv("C:/Users/kurts/Desktop/R-Code/RWorksheet_4/mpg.csv")

# b. Categorical Variables
# Categorical variables in mpg are: manufacturer, model, trans, drv, fl, class.

# c. Continuous Variables
# Continuous variables in mpg are: displ, year, cyl, cty, hwy.
```

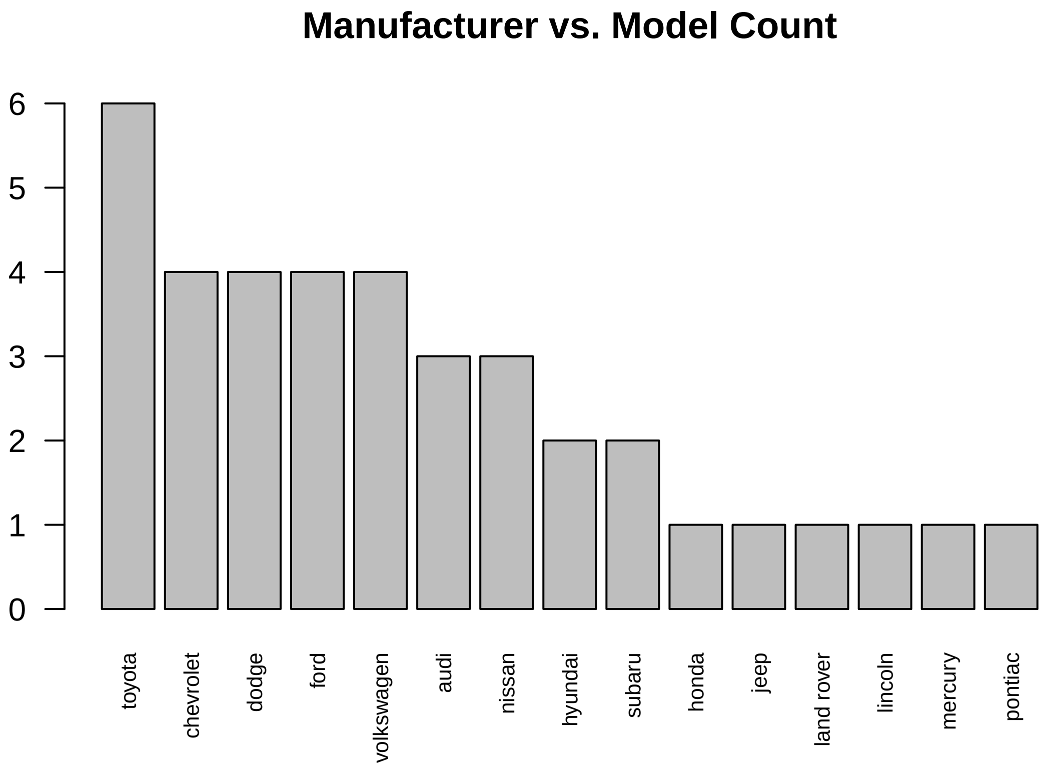
```r
# 2. Analysis of Manufacturers and Models
# a. Find Manufacturer with Most Models and Model with Most Variations
library(dplyr)
```

```
##
## 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
```

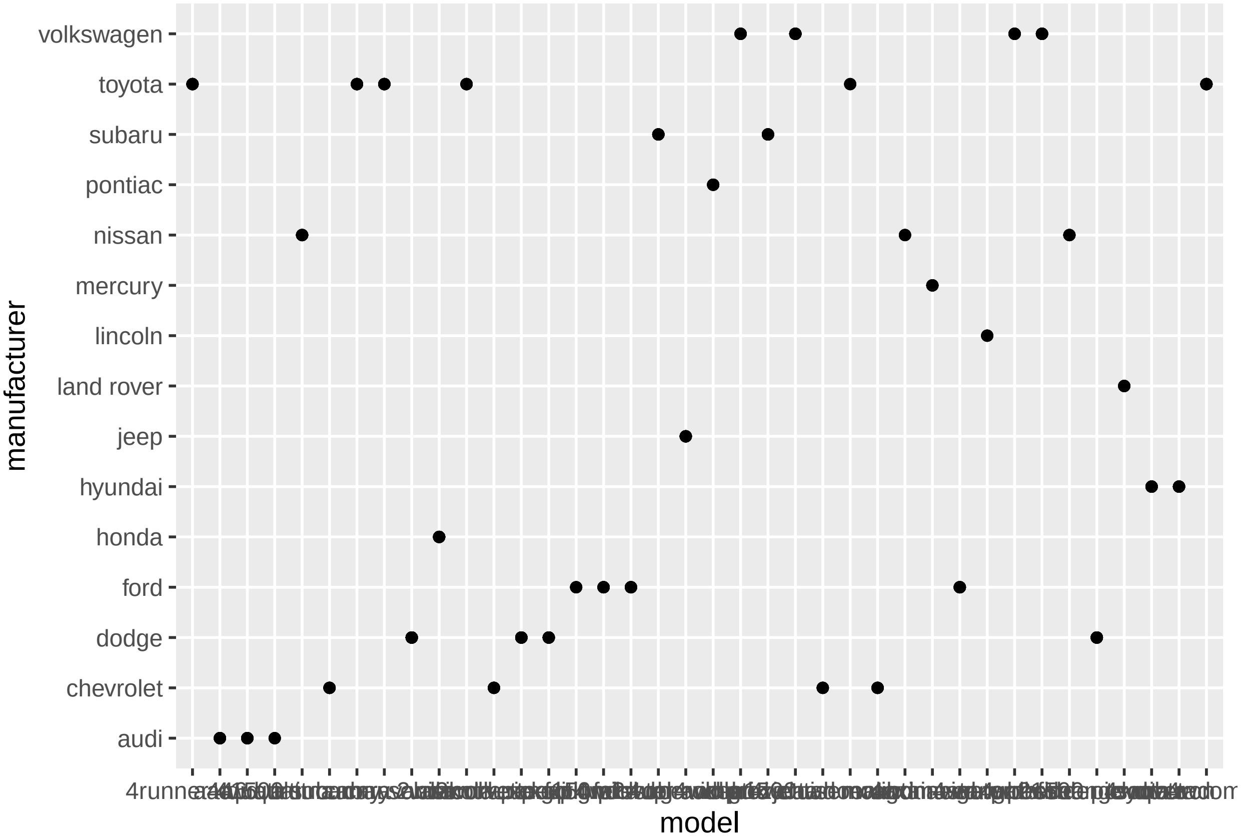
```r
most_models <- mpg_data %>%
  group_by(manufacturer) %>%
  summarize(model_count = n_distinct(model)) %>%
  arrange(desc(model_count))

most_models
```

```
## # A tibble: 15 x 2
##    manufacturer model_count
##    <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. Plotting Manufacturer and Model Counts with barplot()
barplot(most_models$model_count, names.arg = most_models$manufacturer,
        main = "Manufacturer vs. Model Count", las = 2, cex.names = 0.7)
```

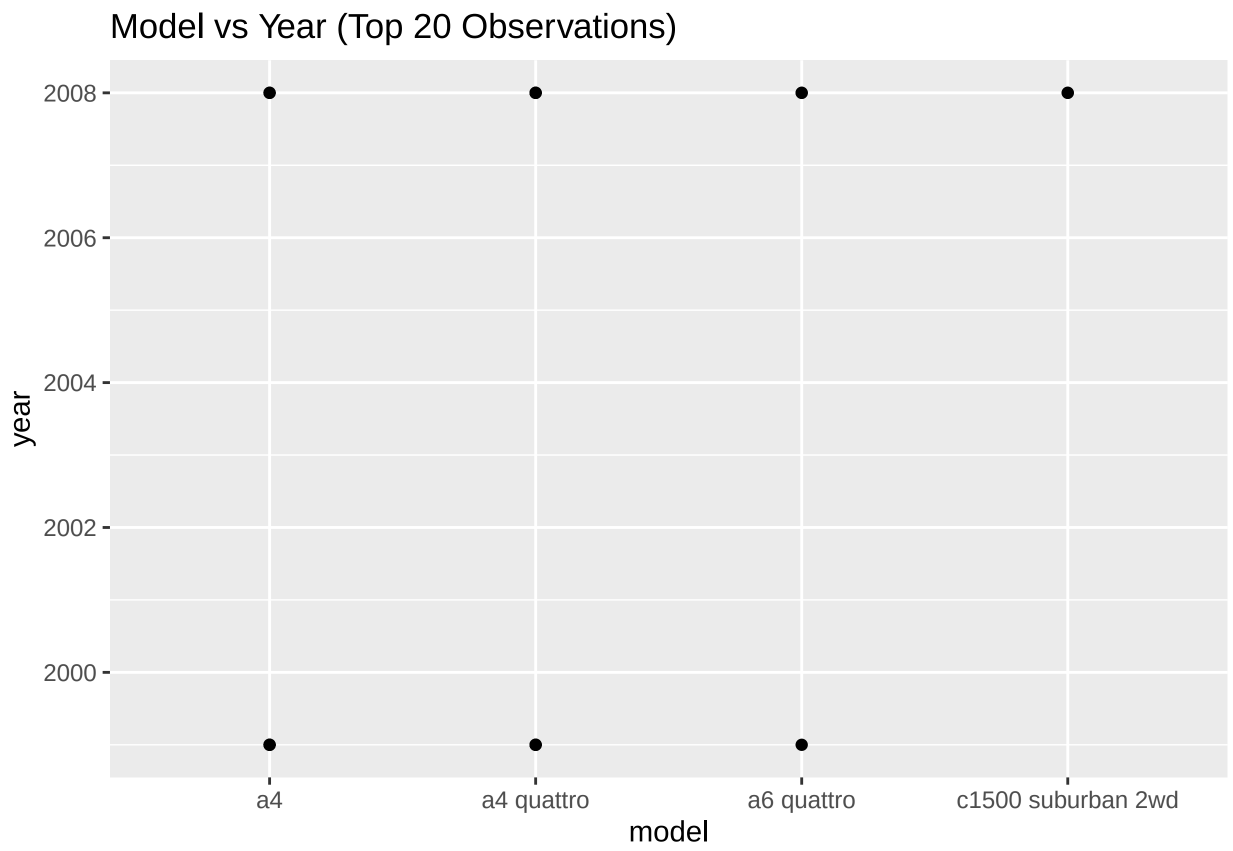


```
#3. Exploring Model and Manufacturer Relationship
#a. Plotting the Relationship with ggplot
library(ggplot2)
ggplot(mpg_data, aes(x = model, y = manufacturer)) + geom_point()
```



```
# b. Improving Data Presentation
# This scatterplot is likely cluttered due to many models. To make it more readable, consider summarizi
```

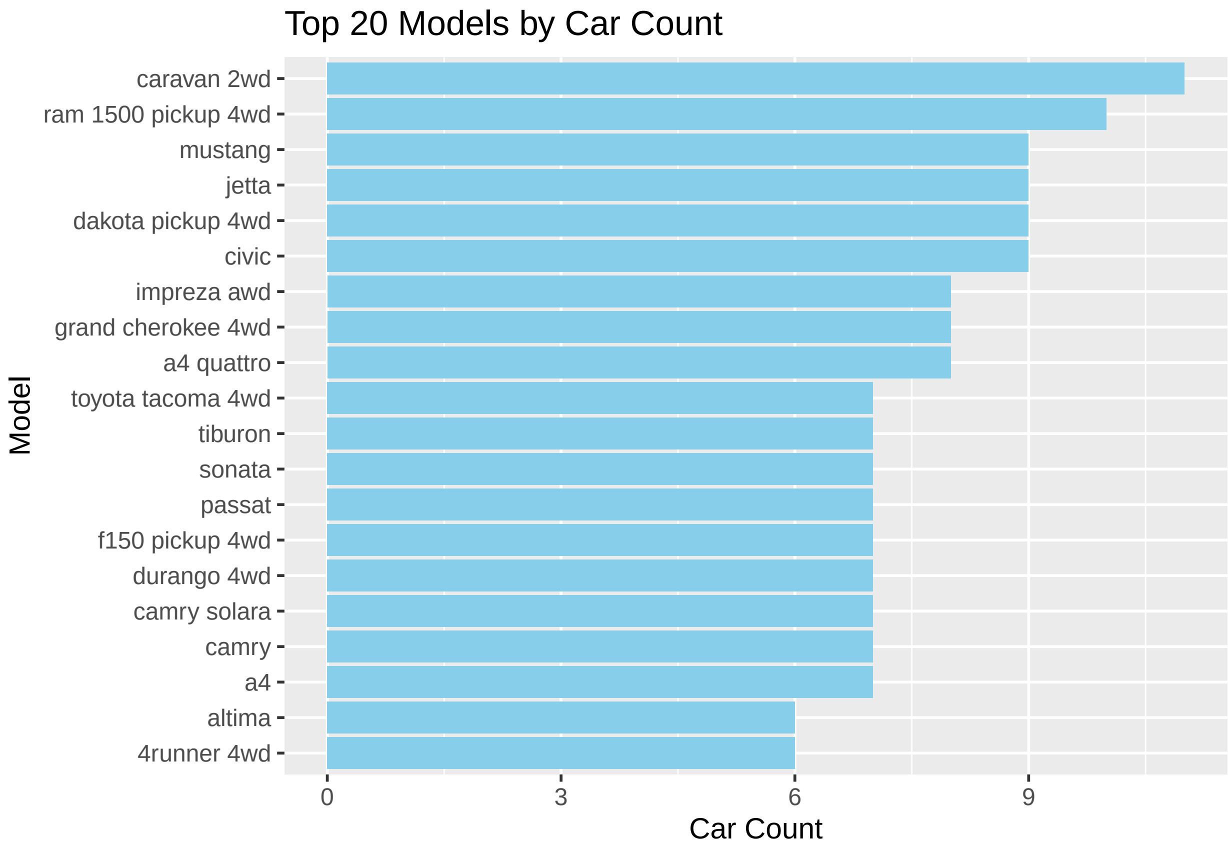
```
# 4. Plotting Model and Year (Top 20 Observations)
top20_data <- head(mpg_data, 20)
ggplot(top20_data, aes(x = model, y = year)) + geom_point() +
  labs(title = "Model vs Year (Top 20 Observations)")
```



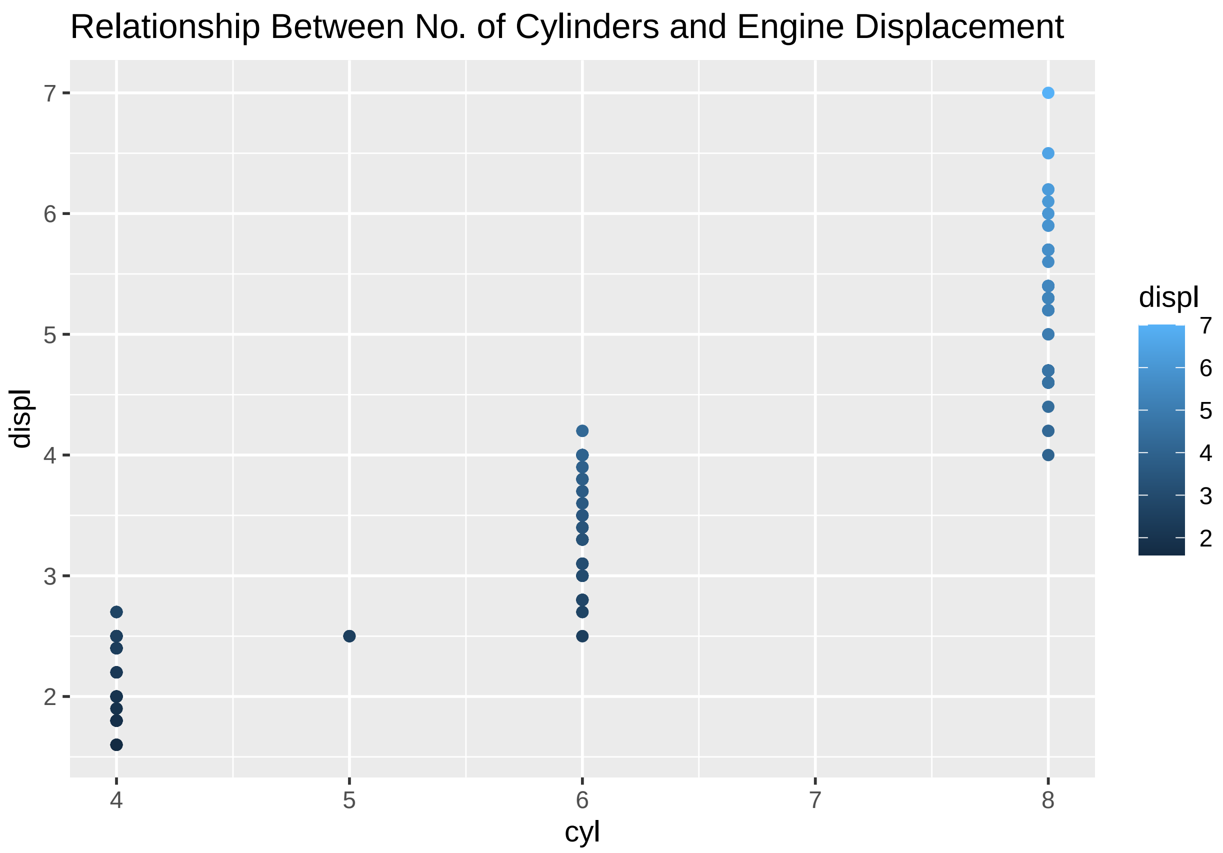
```
# 5. Count Cars per Model with Pipe and Plot
# a. Grouping and Counting Models
model_count <- mpg_data %>%
  group_by(model) %>%
  summarize(car_count = n()) %>%
  arrange(desc(car_count))
model_count
```

```
## # 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
```

```
# b. Plotting with geom_bar() and coord_flip()
ggplot(model_count[1:20, ], aes(x = reorder(model, car_count), y = car_count)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  coord_flip() +
  labs(title = "Top 20 Models by Car Count", x = "Model", y = "Car Count")
```

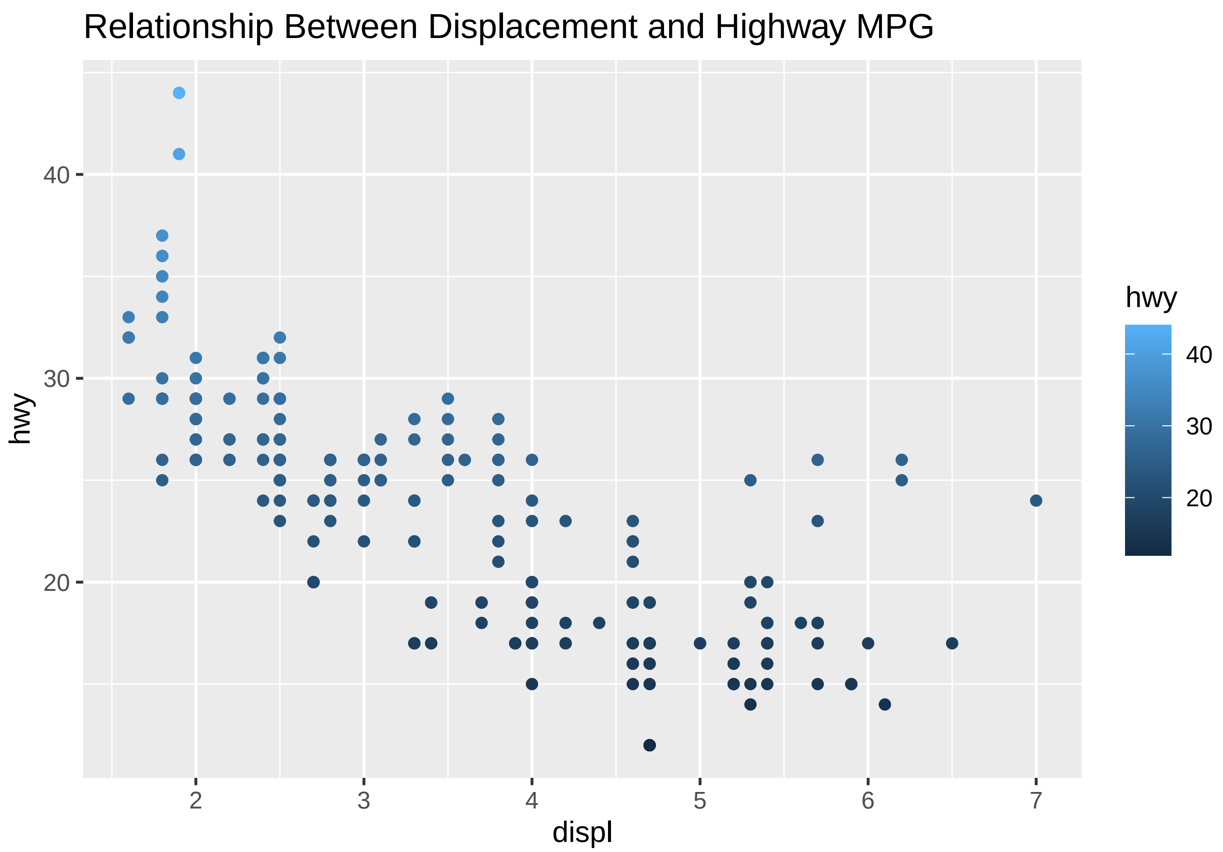


```
# 6. Plotting Cylinders vs. Displacement with Color
ggplot(mpg_data, aes(x = cyl, y = displ, color = displ)) +
  geom_point() +
  labs(title = "Relationship Between No. of Cylinders and Engine Displacement")
```



```
# This plot displays the relationship between the number of cylinders (cyl) and engine displacement (di
```

```
# 7. Plotting Displacement vs. Highway MPG
ggplot(mpg_data, aes(x = displ, y = hwy, color = hwy)) + geom_point() +
  labs(title = "Relationship Between Displacement and Highway MPG")
```



```
# This plot shows how highway miles per gallon (hwy) varies with engine displacement (displ), with colo
```

```
# 8. Importing traffic.csv into R
# a. Number of Observations and Variables
traffic_data <- read.csv("C:/Users/kurts/Desktop/R-Code/RWorksheet_4/traffic.csv")
dim(traffic_data)  # Returns the number of rows and columns
```

```
## [1] 48120     4
```

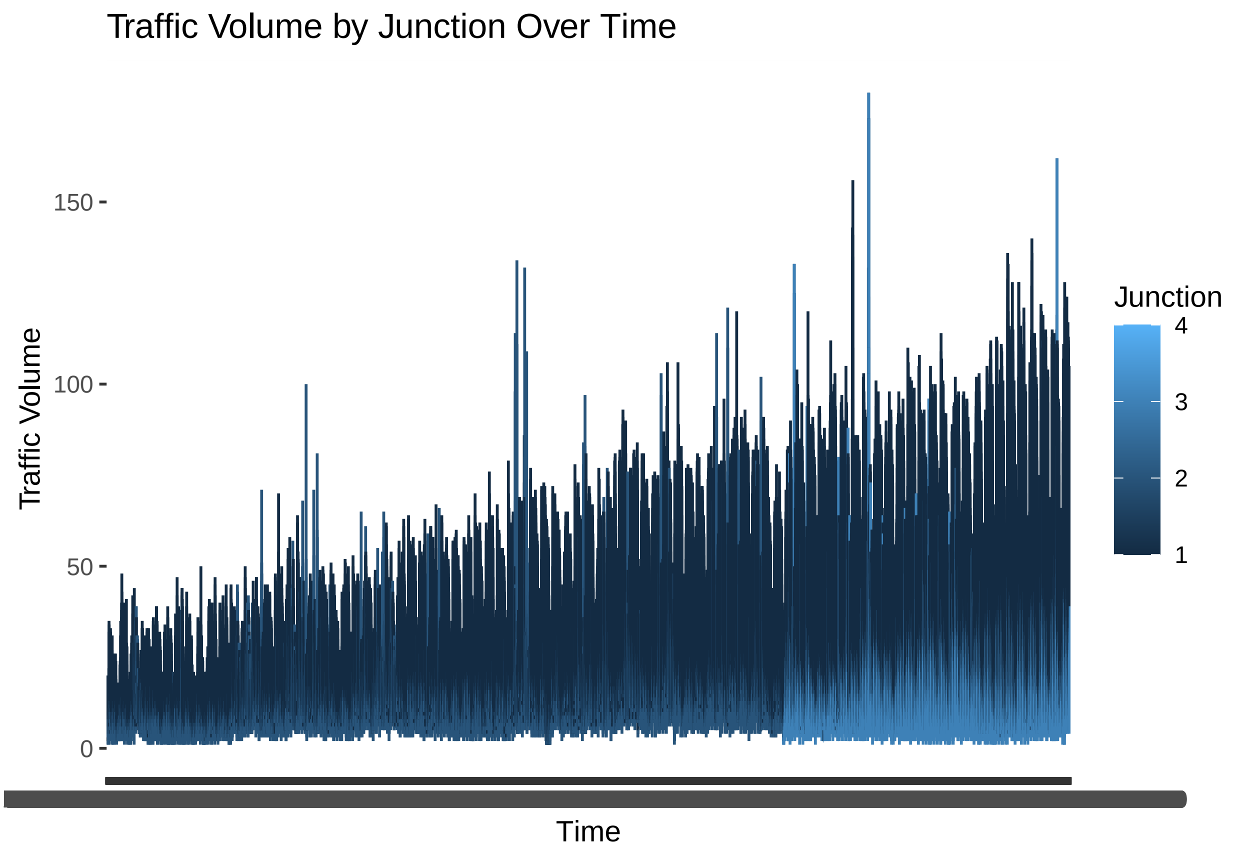
```
names(traffic_data)  # Lists the variable names
```

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

```
# b. Subsetting Traffic Dataset by Junctions
traffic_junctions <- traffic_data %>%
  group_by(Junction) %>%
  summarize(count = n())
traffic_junctions
```

```
## # A tibble: 4 x 2
##   Junction count
##      <int> <int>
## 1        1 14592
## 2        2 14592
## 3        3 14592
## 4        4  4344
```

```
# If the column names are correct, create the plot
ggplot(traffic_data, aes(x = DateTime, y = Vehicles, color = Junction)) +
  geom_line() +
  labs(title = "Traffic Volume by Junction Over Time", x = "Time", y = "Traffic Volume")
```



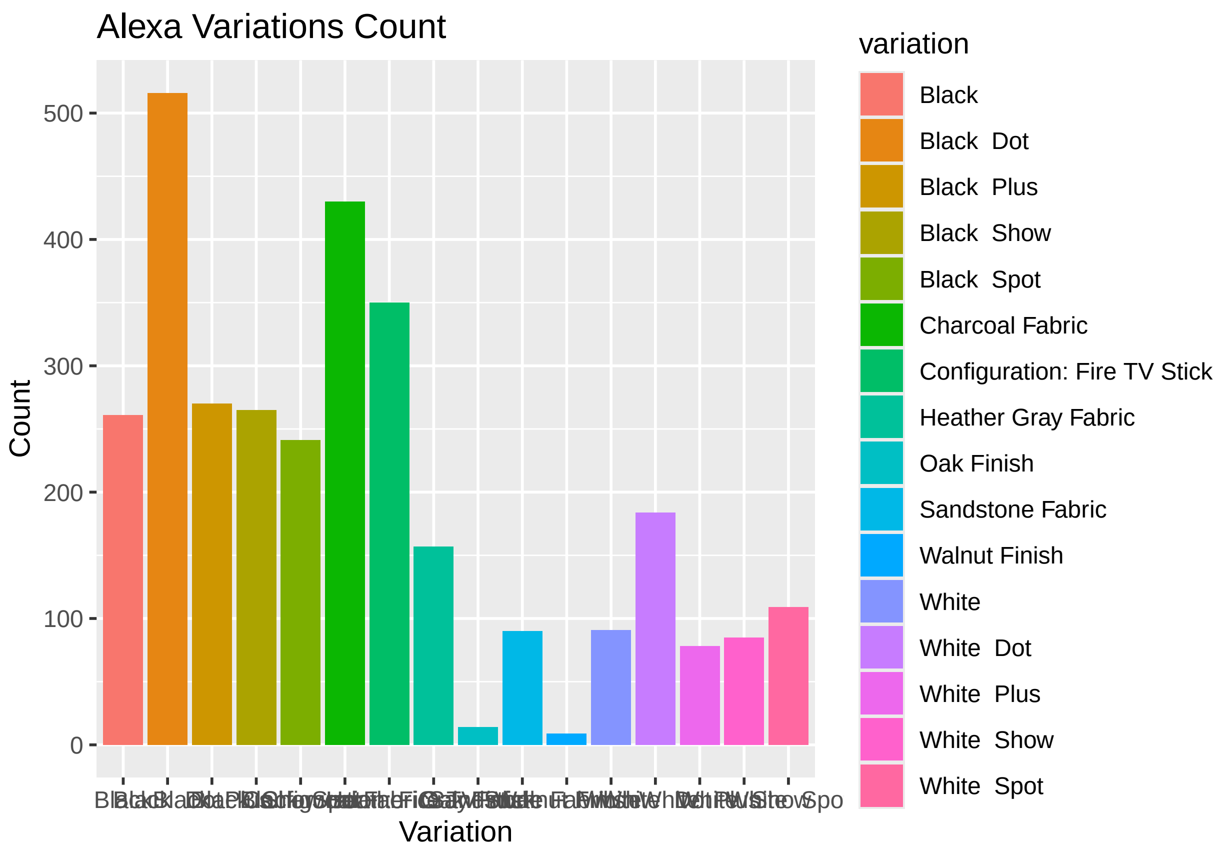
```
# 9. Importing alexa_file.xlsx
# a. Number of Observations and Columns
library(readxl)
alexa_data <- read_excel("C:/Users/kurts/Desktop/R-Code/RWorksheet_4/alexa_file.xlsx")
dim(alexa_data)   # Shows rows and columns
```

```
## [1] 3150    5
```

```
# b. Grouping and Summing Variations
variation_counts <- alexa_data %>%
  group_by(variation) %>%
  summarize(total_count = n())
variation_counts
```

```
## # A tibble: 16 x 2
##    variation                    total_count
##    <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. Plotting Variations with ggplot
ggplot(variation_counts, aes(x = variation, y = total_count, fill = variation)) +
  geom_bar(stat = "identity") +
  labs(title = "Alexa Variations Count", x = "Variation", y = "Count")
```



```
# d. Plotting Date vs. Verified Reviews with geom_line()
ggplot(alexa_data, aes(x = date, y = verified_reviews)) +
  geom_line() +
  labs(title = "Verified Reviews Over Time", x = "Date", y = "Verified Reviews")
```

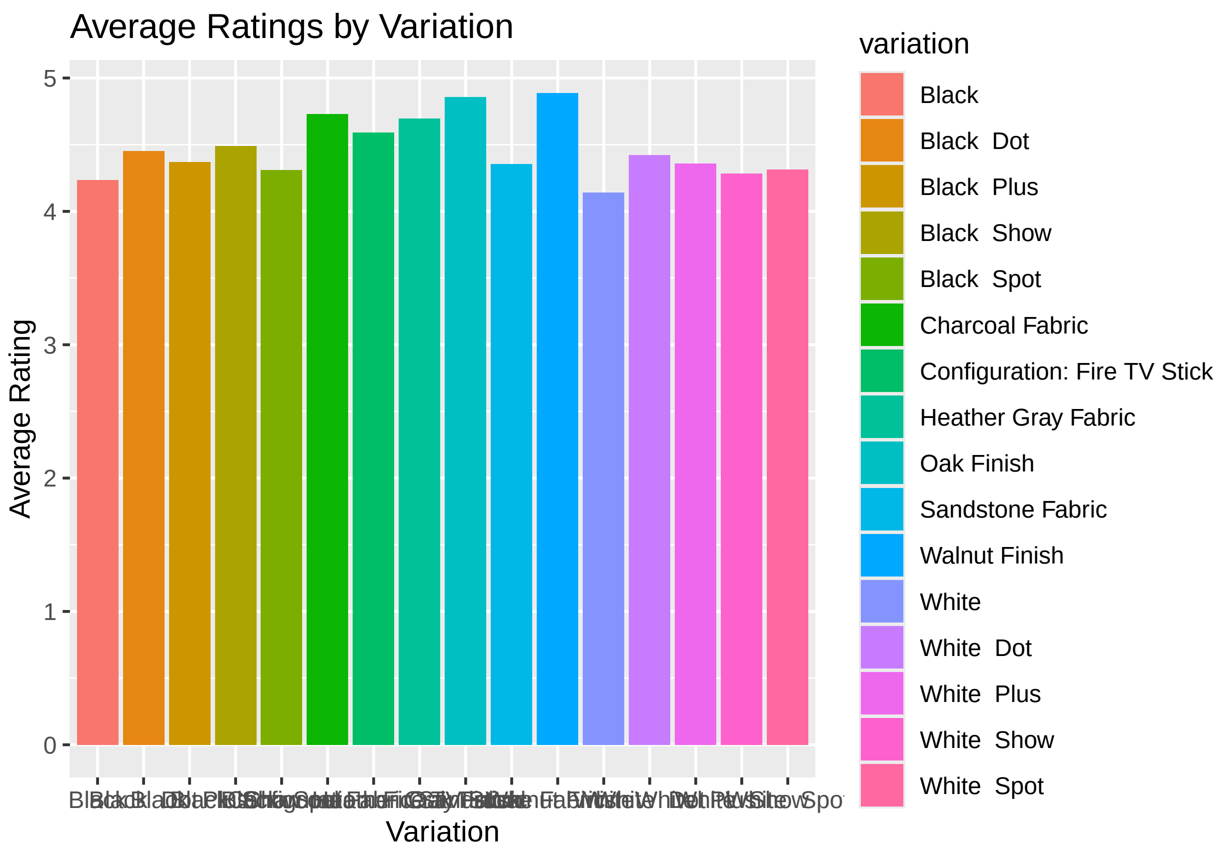
are some serious flaws, particularly if you are the last one to bed or the first to wake. It doesn’t seem like the engineer

ıexpensive alternative option to fill the gap. Ordered the Amazon Fire Stick from Best Buy. Instructions were short and

ˢ one of the lights by saying &#34;Alexa, turn off the second light&#34;. In the Alexa app, I created a 'Group' with &#34
ɔut lately I’ve been getting terrible support. The guy that took my call just rambled off a (completely unhelpful) script an

npting to add this bulb to my Alexa Echo Plus. Everything I tried, ended in a Discovery Failed message. I tried to set ɛ
gh multiple pages. The one thing that I am not a fan of is the home screen cards do not really rotate that much - they a

```r
# e. Plotting Relationship of Variations and Ratings
variation_ratings <- alexa_data %>%
  group_by(variation) %>%
  summarize(avg_rating = mean(rating, na.rm = TRUE))

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


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