Worksheet-4c in R

Riza Angelique Pelaez

2024-11-04

```r
# 1. Importing the dataset
library(readr)
mpg_data <- read_csv("mpg.csv")
```

```
## New names:
## Rows: 234 Columns: 12
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (6): manufacturer, model, trans, drv, fl, class dbl (6): ...1, displ, year,
## cyl, cty, hwy
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

```r
# 1a. Code to import a CSV file into R
mpg_data <- read_csv("mpg.csv")
```

```
## New names:
## Rows: 234 Columns: 12
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (6): manufacturer, model, trans, drv, fl, class dbl (6): ...1, displ, year,
## cyl, cty, hwy
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

```r
# 1b. Identifying categorical variables
cat_vars <- names(mpg_data)[sapply(mpg_data, is.factor)]
cat_vars
```

```
## character(0)
```

```r
# 1c. Identifying continuous variables
cont_vars <- names(mpg_data)[sapply(mpg_data, is.numeric)]
cont_vars
```

```
## [1] "...1"  "displ" "year"  "cyl"   "cty"   "hwy"
```

```r
# 2. Finding manufacturer with the most models
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)
most_models <- mpg_data %>%
  group_by(manufacturer) %>%
  summarize(num_models = n_distinct(model)) %>%
  arrange(desc(num_models))
most_models
```

```
## # A tibble: 15 x 2
##    manufacturer num_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
```

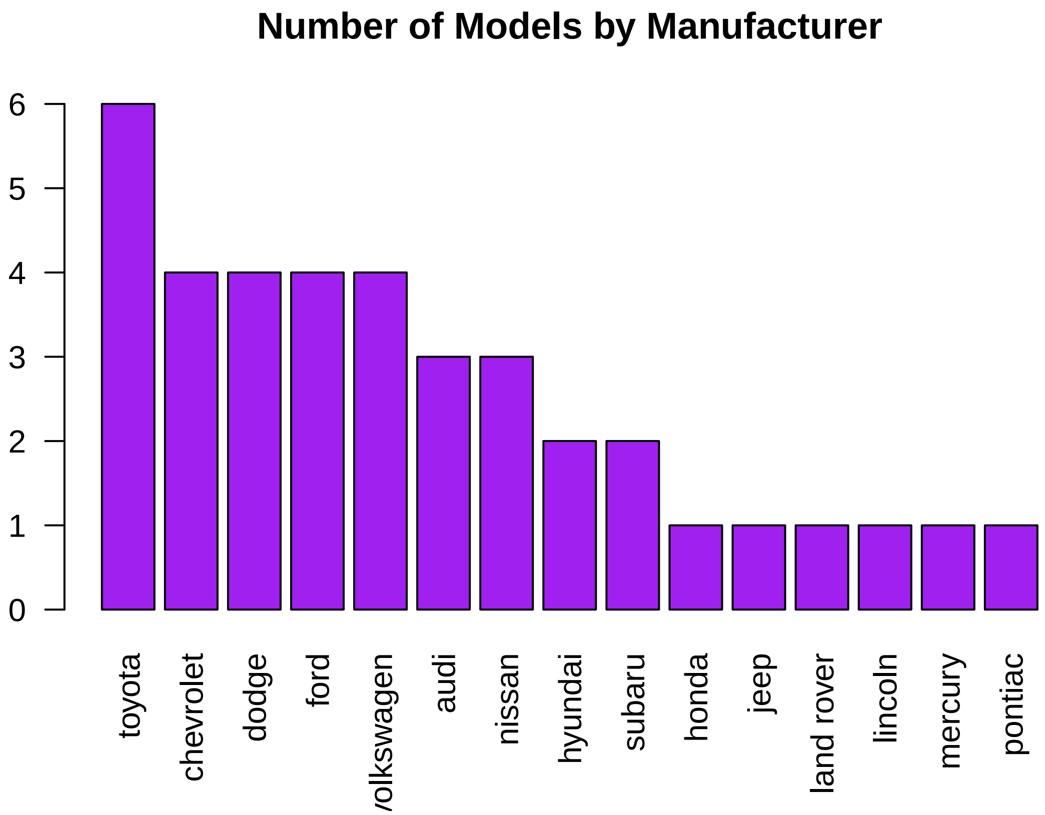
```
# 2a. Code to group manufacturers and find unique models
unique_models <- mpg_data %>%
  group_by(manufacturer, model) %>%
  summarise(count = n()) %>%
  arrange(desc(count))
```

```
## `summarise()` has grouped output by 'manufacturer'. You can override using the
## `.groups` argument.
```

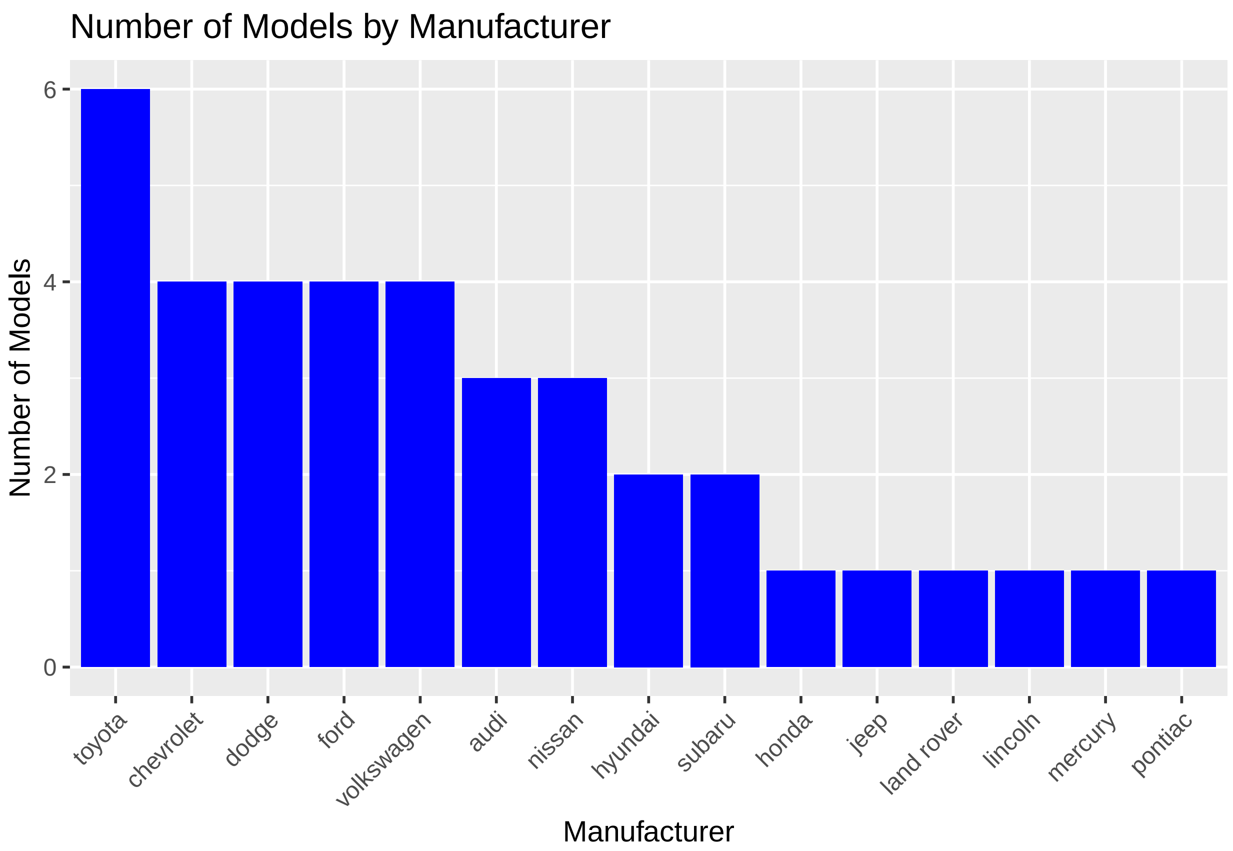
```
unique_models
```

```
## # A tibble: 38 x 3
## # Groups:   manufacturer [15]
##    manufacturer model               count
##    <chr>        <chr>               <int>
##  1 dodge        caravan 2wd            11
##  2 dodge        ram 1500 pickup 4wd    10
##  3 dodge        dakota pickup 4wd       9
##  4 ford         mustang                 9
##  5 honda        civic                   9
##  6 volkswagen   jetta                   9
##  7 audi         a4 quattro              8
##  8 jeep         grand cherokee 4wd      8
##  9 subaru       impreza awd             8
## 10 audi         a4                      7
## # i 28 more rows
```

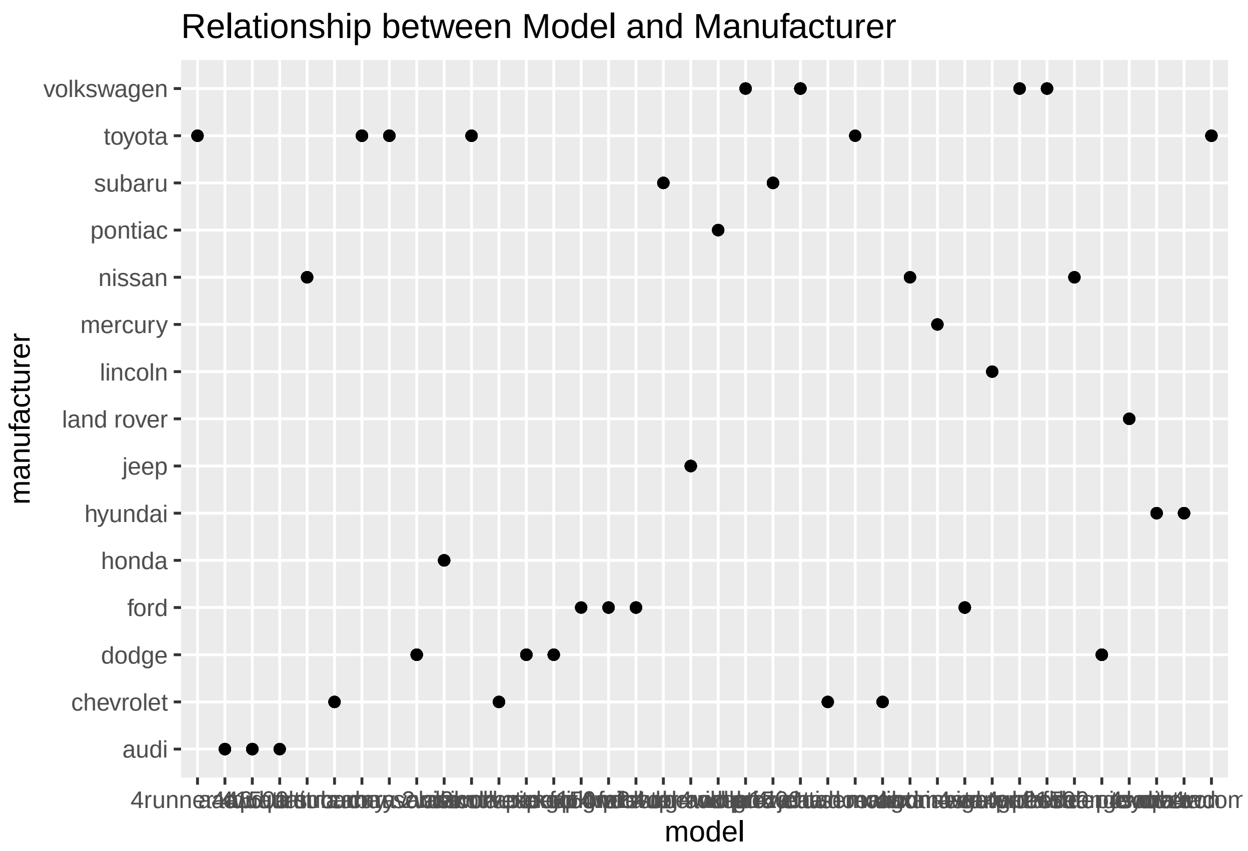
```
# 2b. Plotting manufacturers by number of models

barplot(most_models$num_models, names.arg = most_models$manufacturer, las = 2, col = "purple",
        main = "Number of Models by Manufacturer")
```



```
ggplot(most_models, aes(x = reorder(manufacturer, -num_models), y = num_models)) +
  geom_bar(stat = "identity", fill = "blue") +
  labs(title = "Number of Models by Manufacturer", x = "Manufacturer", y = "Number of Models") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

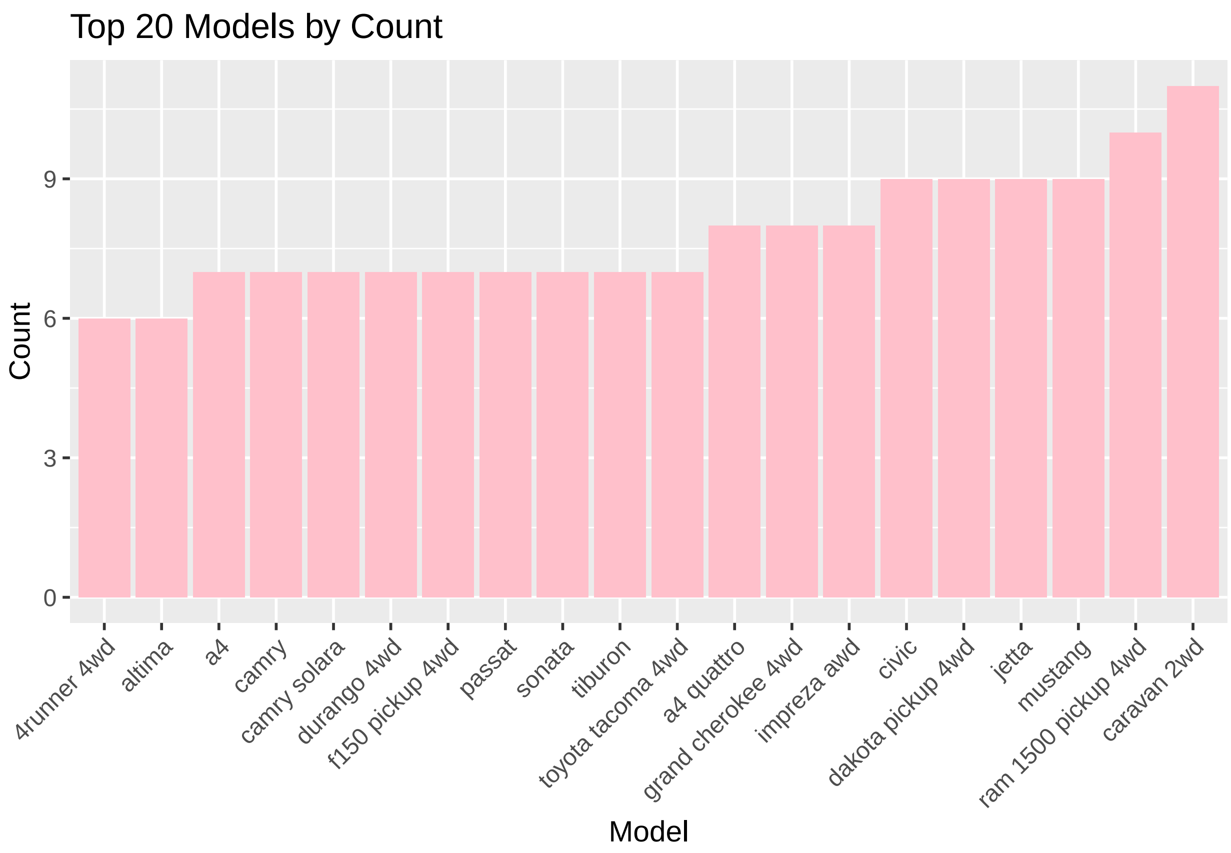


```r
# 3. Relationship between model and manufacturer
ggplot(mpg_data, aes(x = model, y = manufacturer)) +
  geom_point() +
  labs(title = "Relationship between Model and Manufacturer")
```

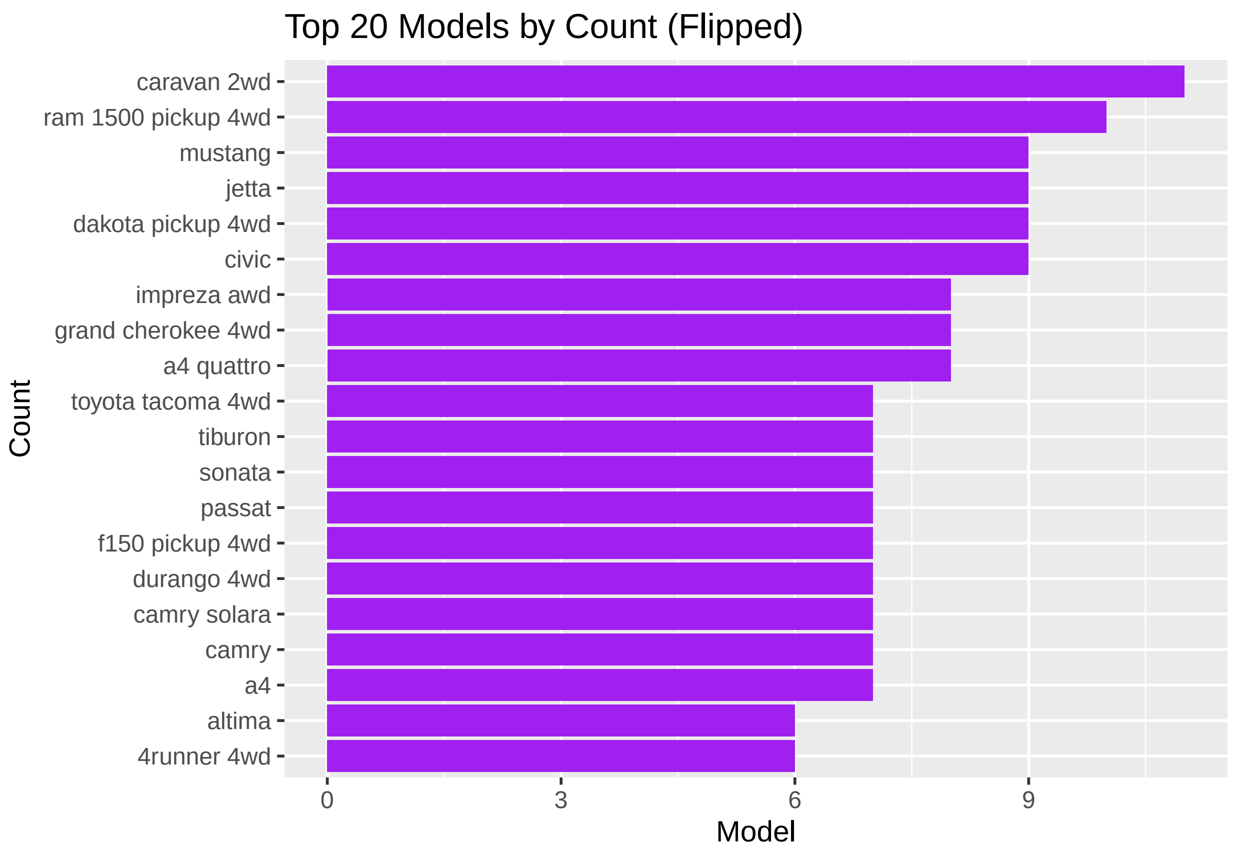


```
# The graph shows model and manufacturer relationships, but lacks interpret ability.
```

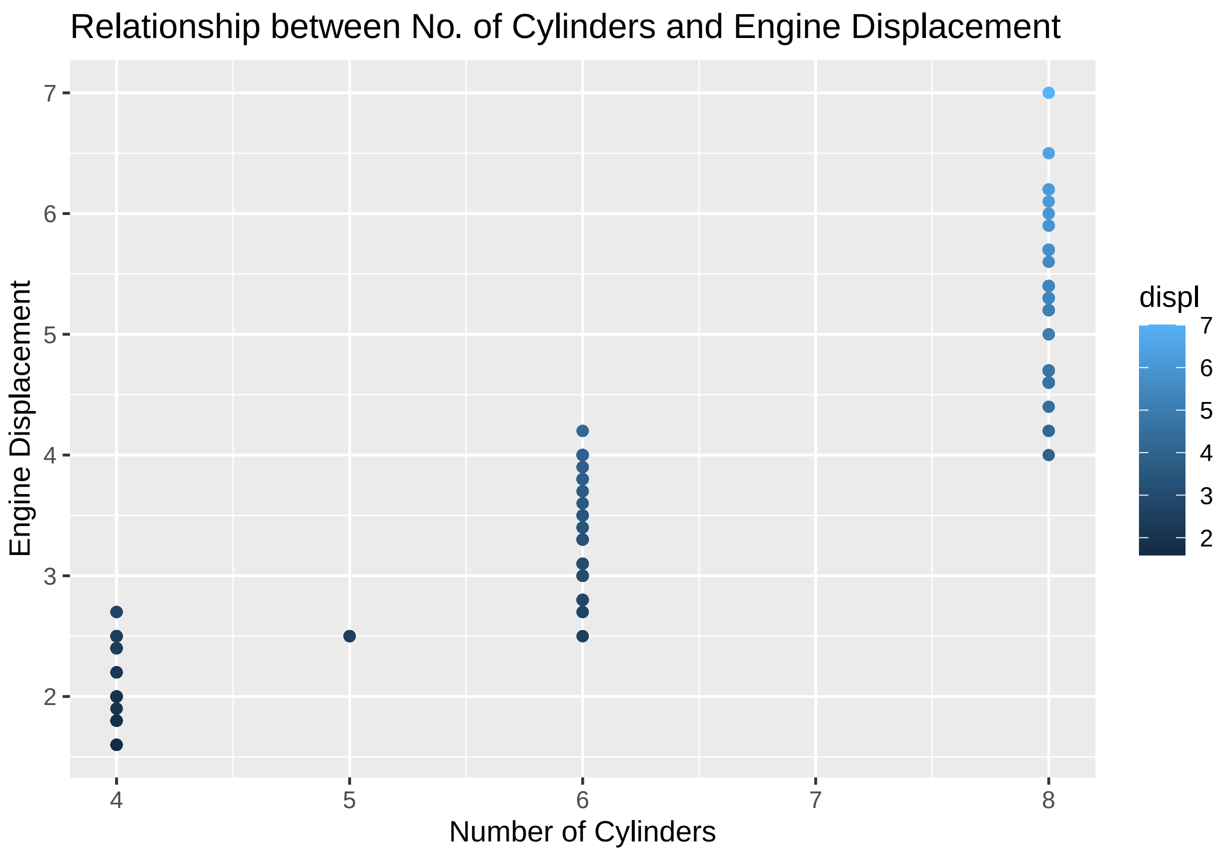
```
# 4. Using the pipe (%>%), group the model and get the number of cars per model
model_counts <- mpg_data %>%
  group_by(model) %>%
  summarize(count = n()) %>%
  arrange(desc(count))

# 4a. Plot using geom_bar() with the top 20 observations
# The graph should include a title, labels, and colors
top_20_models <- model_counts %>% slice_head(n = 20)
ggplot(top_20_models, aes(x = reorder(model, count), y = count)) +
  geom_bar(stat = "identity", fill = "pink") +
  labs(title = "Top 20 Models by Count", x = "Model", y = "Count") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```
# 4b. Plot using geom_bar() + coord_flip() for top 20 observations
ggplot(top_20_models, aes(x = reorder(model, count), y = count)) +
  geom_bar(stat = "identity", fill = "purple") +
  coord_flip() +
  labs(title = "Top 20 Models by Count (Flipped)", x = "Count", y = "Model")
```

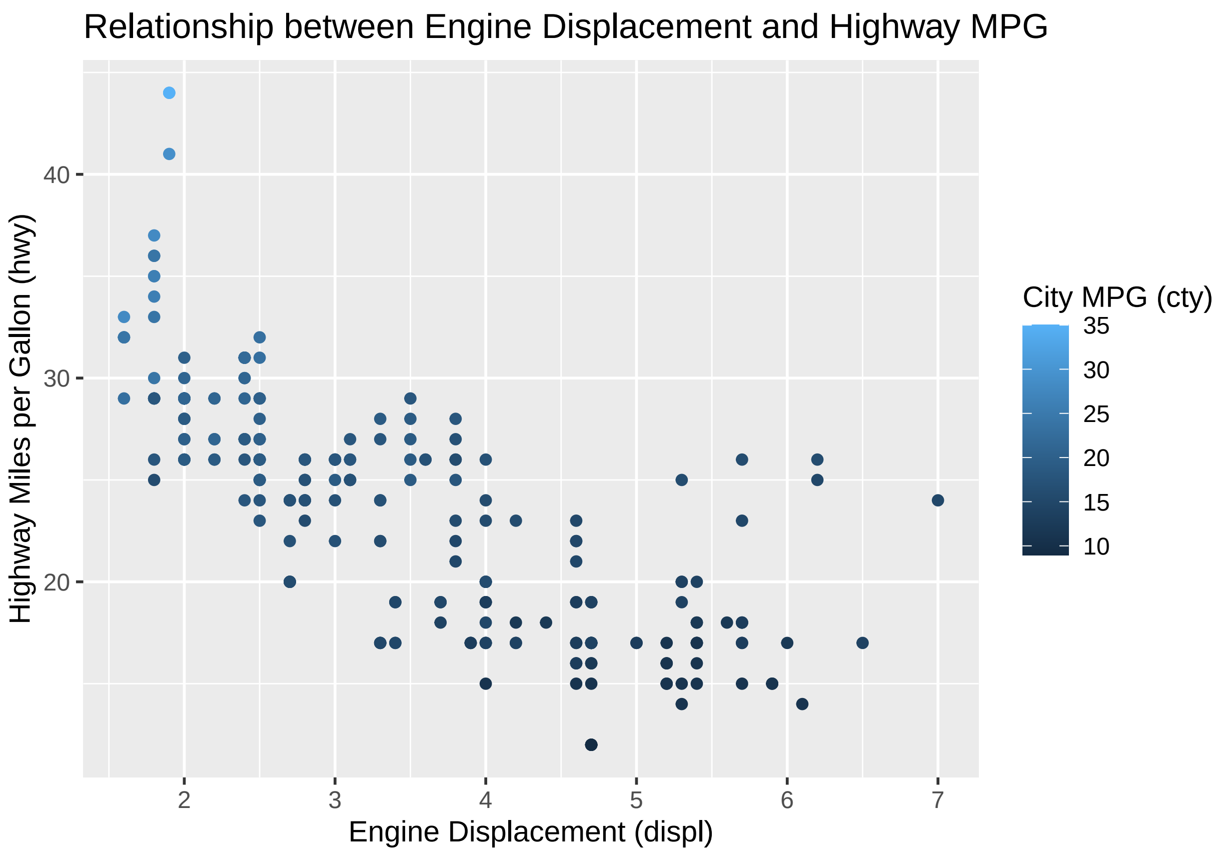


```r
# 5. Plot the relationship between cyl - number of cylinders and displ - engine displacement
ggplot(mpg_data, aes(x = cyl, y = displ, color = displ)) +
  geom_point() +
  labs(title = "Relationship between No. of Cylinders and Engine Displacement",
       x = "Number of Cylinders", y = "Engine Displacement")
```



```
# 5a. Description:
# This plot shows that as the number of cylinders increases,
# the engine displacement also tends to increase.
# This suggests a positive correlation between these two variables.
```

```
# 6(1). Plotting the relationship between displ and hwy, mapped with cty as a continuous variable
ggplot(mpg_data, aes(x = displ, y = hwy, color = cty)) +
  geom_point() +
  labs(title = "Relationship between Engine Displacement and Highway MPG",
       x = "Engine Displacement (displ)", y = "Highway Miles per Gallon (hwy)",
       color = "City MPG (cty)")
```



```
# The result, and it produced such output:
# The engine displacement (displ) increases, highway MPG (hwy) decreases.
# This is because larger engines generally consume more fuel, reducing fuel efficiency.
# The color gradient for city MPG (cty) reinforces this trend, as city and highway efficiencies tend to
```

```
# 6(2). Import the traffic.csv dataset
traffic_data <- read.csv("traffic.csv")

# 6a. Check the number of observations and variables
num_observations <- nrow(traffic_data)
num_variables <- ncol(traffic_data)
variables <- names(traffic_data)

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

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

```
cat("Number of variables:", num_variables, "\n")
```

```
## Number of variables: 4
```

```
cat("Variables in the dataset:", variables, "\n")
```

```
## Variables in the dataset: DateTime Junction Vehicles ID
```

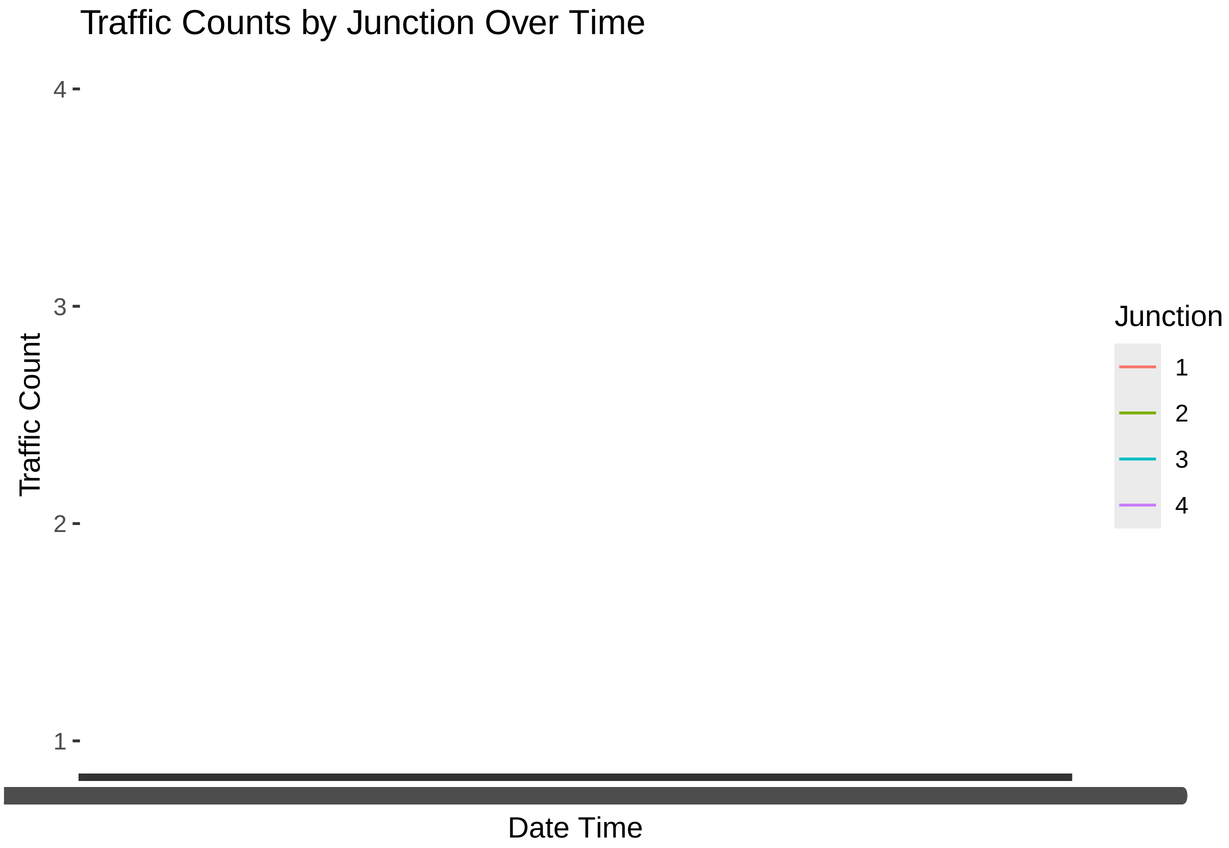
```
# 6b. Subset the traffic dataset by junctions
junction_data <- traffic_data %>%
  group_by(Junction) %>%
  summarize(Junction = n())
```

```
print(junction_data)
```

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

```
# 6c. Plot each junction over time using geom_line()
ggplot(traffic_data, aes(x = DateTime, y = Junction , color = as.factor(Junction))) +
  geom_line() +
  labs(title = "Traffic Counts by Junction Over Time",
       x = "Date Time", y = "Traffic Count",
       color = "Junction")
```

```
## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
```



```
# 7. Import the alexa_file.xlsx dataset
library(readxl)
alexa_data <- read_excel("alexa_file.xlsx")

# 7a. Check the number of observations and columns
num_observations <- nrow(alexa_data)
num_columns <- ncol(alexa_data)
```

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

```
## Number of observations: 3150
```

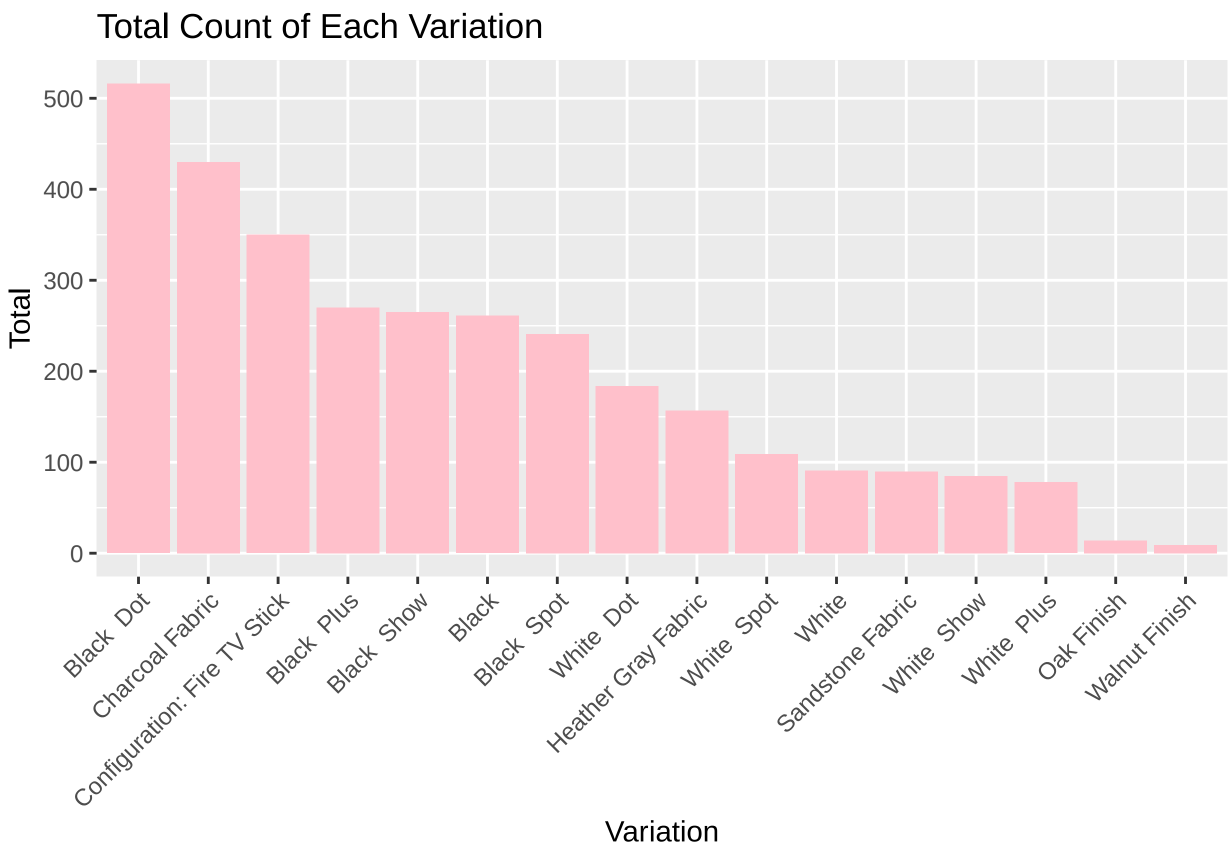
```r
cat("Number of columns:", num_columns, "\n")
```

```
## Number of columns: 5
```

```r
# 7b. Group by 'variation' and get the total count of each variation
variation_totals <- alexa_data %>%
  group_by(variation) %>%
  summarise(total = n())
print(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
```

```r
# 7c. Plot the variations using ggplot()
ggplot(variation_totals, aes(x = reorder(variation, -total), y = total)) +
  geom_bar(stat = "identity", fill = "pink") +
  labs(title = "Total Count of Each Variation", x = "Variation", y = "Total") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```
# Observation:
# The plot shows the distribution of different variations. Some variations have significantly higher co
# indicating they are more common or popular.

# 7d. Plot a geom_line() with date and number of verified reviews
ggplot(alexa_data, aes(x = date, y = verified_reviews)) +
  geom_line(color = "purple") +
  labs(title = "Date vs Verified Reviews", x = "Date", y = "Number of Verified Reviews")
```

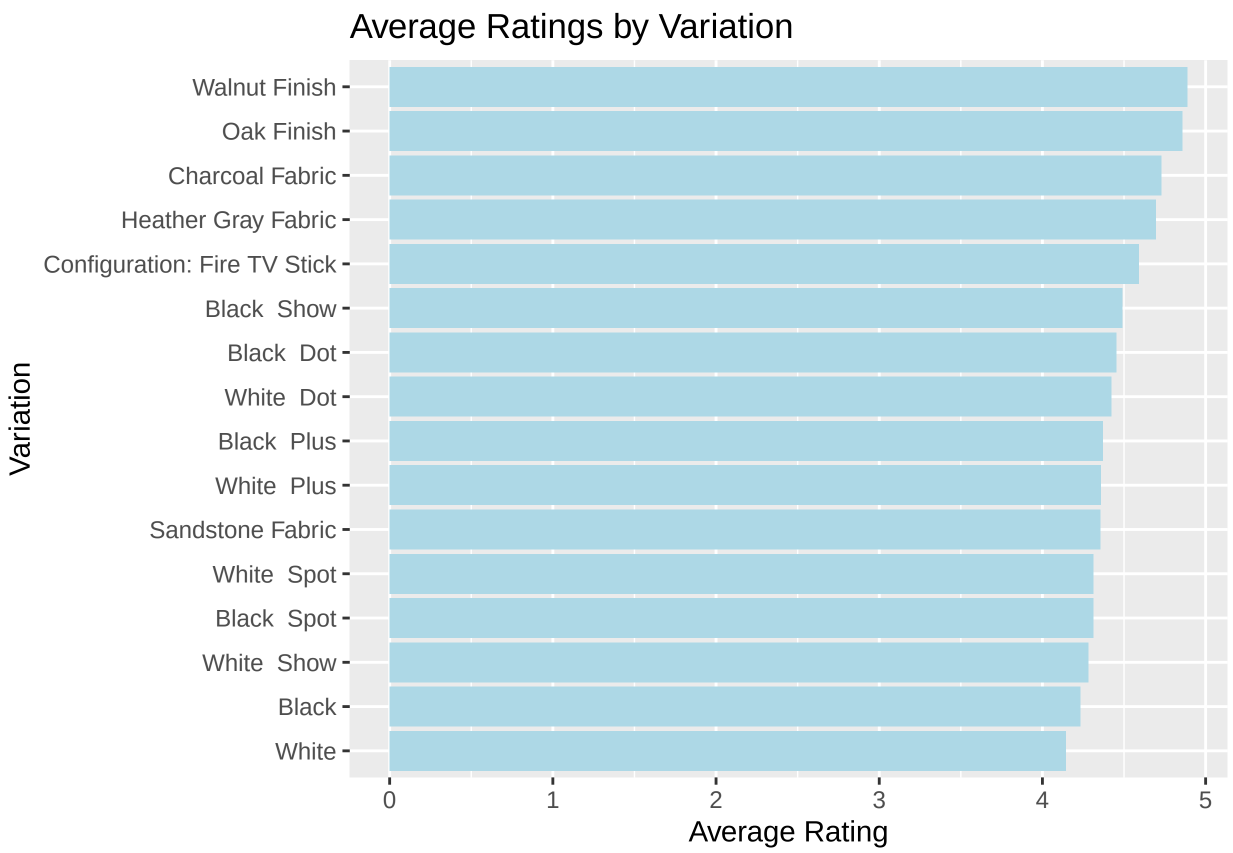
are some serious flaws, particularly if you are the last one to bed or the first to wake. It doesn’t seem like the engineer

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

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

mpting to add this bulb to my Alexa Echo Plus. Everything I tried, ended in a Discovery Failed message. I tried to set
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
# 7e. Analyze the relationship of variations and ratings, and find the highest-rated variation
variation_ratings <- alexa_data %>%
  group_by(variation) %>%
  summarize(avg_rating = mean(rating, na.rm = TRUE)) %>%
  arrange(desc(avg_rating))

# Plot the relationship of variations and their average ratings
ggplot(variation_ratings, aes(x = reorder(variation, avg_rating), y = avg_rating)) +
  geom_bar(stat = "identity", fill = "lightblue") +
  coord_flip() +
  labs(title = "Average Ratings by Variation", x = "Variation", y = "Average Rating")
```


Average Ratings by Variation
Variation
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
0
1
2
3
4
5
Average Rating