RWORKSHEET_CATEDRAL4C

RcCatedral

2023-12-08

#1

```
library(readr)
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`
```

```
print(data)
```

```
## # A tibble: 234 x 12
##     ...1 manufacturer model      displ  year   cyl trans drv     cty   hwy fl
##    <dbl> <chr>        <chr>      <dbl> <dbl> <dbl> <chr> <chr> <dbl> <dbl> <chr>
##  1     1 audi         a4           1.8  1999     4 auto~ f        18    29 p
##  2     2 audi         a4           1.8  1999     4 manu~ f        21    29 p
##  3     3 audi         a4           2    2008     4 manu~ f        20    31 p
##  4     4 audi         a4           2    2008     4 auto~ f        21    30 p
##  5     5 audi         a4           2.8  1999     6 auto~ f        16    26 p
##  6     6 audi         a4           2.8  1999     6 manu~ f        18    26 p
##  7     7 audi         a4           3.1  2008     6 auto~ f        18    27 p
##  8     8 audi         a4 quattro   1.8  1999     4 manu~ 4        18    26 p
##  9     9 audi         a4 quattro   1.8  1999     4 auto~ 4        16    25 p
## 10    10 audi         a4 quattro   2    2008     4 manu~ 4        20    28 p
## # i 224 more rows
## # i 1 more variable: class <chr>
```

```
library(ggplot2)

data(mpg)

str(mpg)
```

```
## tibble [234 x 11] (S3: tbl_df/tbl/data.frame)
##  $ manufacturer: chr [1:234] "audi" "audi" "audi" "audi" ...
##  $ model       : chr [1:234] "a4" "a4" "a4" "a4" ...
##  $ displ       : num [1:234] 1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
##  $ year        : int [1:234] 1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
##  $ cyl         : int [1:234] 4 4 4 4 6 6 6 4 4 4 ...
```

```
##  $ trans       : chr [1:234] "auto(l5)" "manual(m5)" "manual(m6)" "auto(av)" ...
##  $ drv         : chr [1:234] "f" "f" "f" "f" ...
##  $ cty         : int [1:234] 18 21 20 21 16 18 18 18 16 20 ...
##  $ hwy         : int [1:234] 29 29 31 30 26 26 27 26 25 28 ...
##  $ fl          : chr [1:234] "p" "p" "p" "p" ...
##  $ class       : chr [1:234] "compact" "compact" "compact" "compact" ...
```

```
#"manufacturer", "model", "trans", "drv", "fl", "class"
```

```
#"displ," "year," "cyl," "cty,", "hwy"
```

#2

```
data(mpg)

manufacturer_most_models <- names(sort(table(mpg$manufacturer), decreasing = TRUE))[1]

model_most_variations <- names(sort(table(mpg$model), decreasing = TRUE))[1]

cat("Manufacturer with the most models:", manufacturer_most_models, "\n")
```

```
## Manufacturer with the most models: dodge
```

```
cat("Model with the most variations:", model_most_variations, "\n")
```

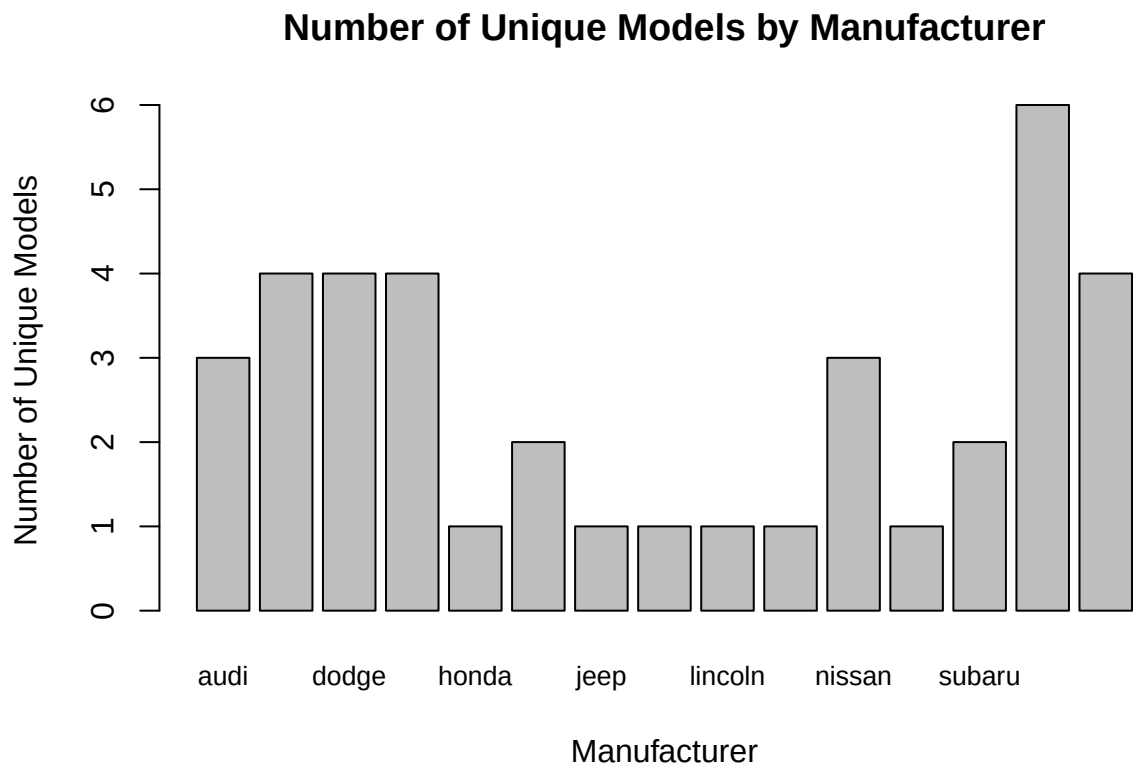
```
## Model with the most variations: caravan 2wd
```

```
data(mpg)

manufacturer_model_counts <- table(mpg$manufacturer, mpg$model)

manufacturer_unique_models <- sapply(rownames(manufacturer_model_counts), function(manufacturer) {
  unique_models <- names(which(manufacturer_model_counts[manufacturer,] > 0))
  return(data.frame(manufacturer = manufacturer, unique_models = length(unique_models)))
})

print(manufacturer_unique_models)
```

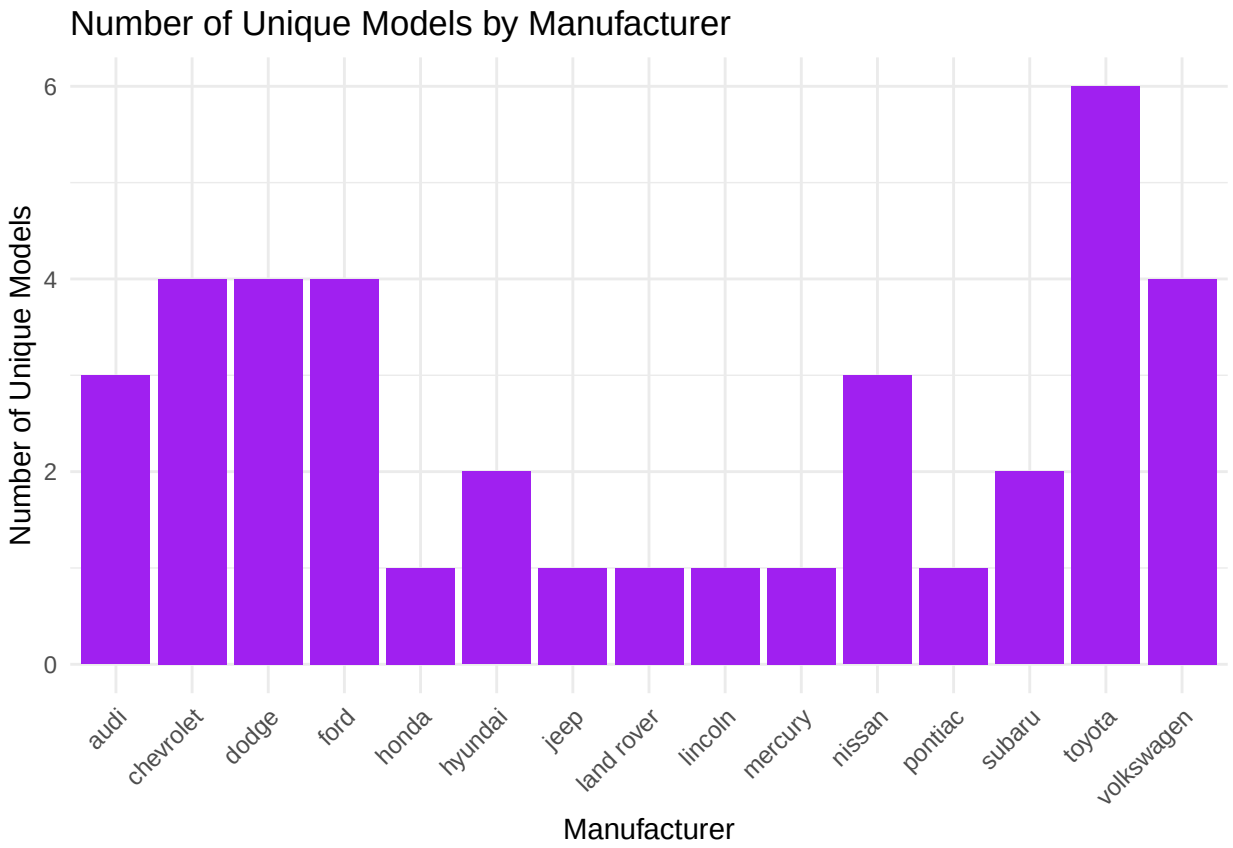
```
##               audi   chevrolet   dodge   ford   honda   hyundai   jeep
## manufacturer  "audi" "chevrolet" "dodge" "ford" "honda" "hyundai" "jeep"
## unique_models 3      4           4       4      1       2         1
##               land rover   lincoln   mercury   nissan   pontiac   subaru
## manufacturer  "land rover" "lincoln" "mercury" "nissan" "pontiac" "subaru"
## unique_models 1            1         1         3        1         2
##               toyota   volkswagen
## manufacturer  "toyota" "volkswagen"
## unique_models 6        4
```

```
data(mpg)

manufacturer_model_counts <- table(mpg$manufacturer, mpg$model)

manufacturer_unique_models <- sapply(rownames(manufacturer_model_counts), function(manufacturer) {
  unique_models <- names(which(manufacturer_model_counts[manufacturer, ] > 0))
  return(length(unique_models))
})

result_df <- data.frame(manufacturer = names(manufacturer_unique_models), unique_models = manufacturer_
```

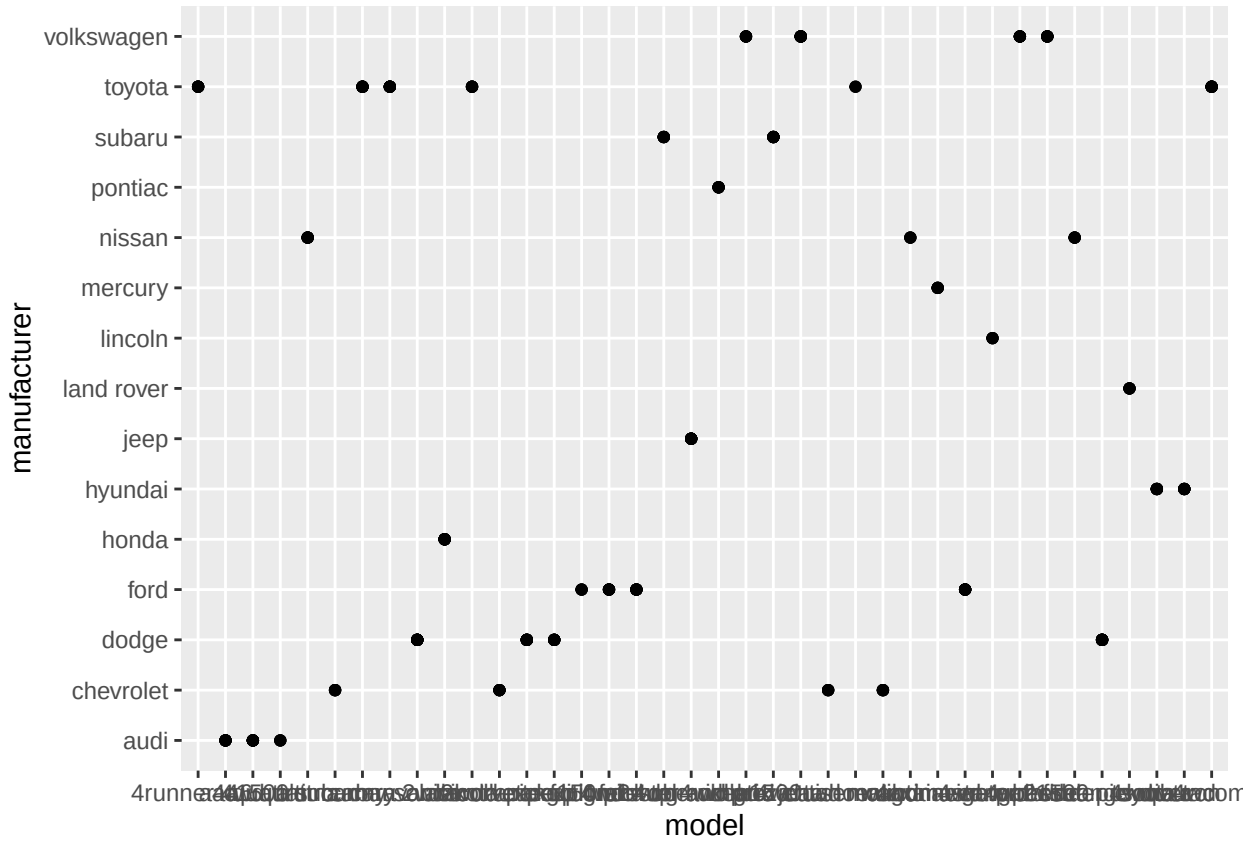
```r
barplot(result_df$unique_models,
        names.arg = result_df$manufacturer,
        col = "grey",
        xlab = "Manufacturer",
        ylab = "Number of Unique Models",
        main = "Number of Unique Models by Manufacturer",
        cex.names = 0.8)
```




```r
data(mpg)

manufacturer_model_counts <- table(mpg$manufacturer, mpg$model)

manufacturer_unique_models <- sapply(rownames(manufacturer_model_counts), function(manufacturer) {
  unique_models <- names(which(manufacturer_model_counts[manufacturer, ] > 0))
  return(length(unique_models))
})

result_df <- data.frame(manufacturer = names(manufacturer_unique_models), unique_models = manufacturer_u

ggplot(data = result_df, aes(x = manufacturer, y = unique_models)) +
  geom_bar(stat = "identity", fill = "purple") +
  labs(x = "Manufacturer", y = "Number of Unique Models",
       title = "Number of Unique Models by Manufacturer") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

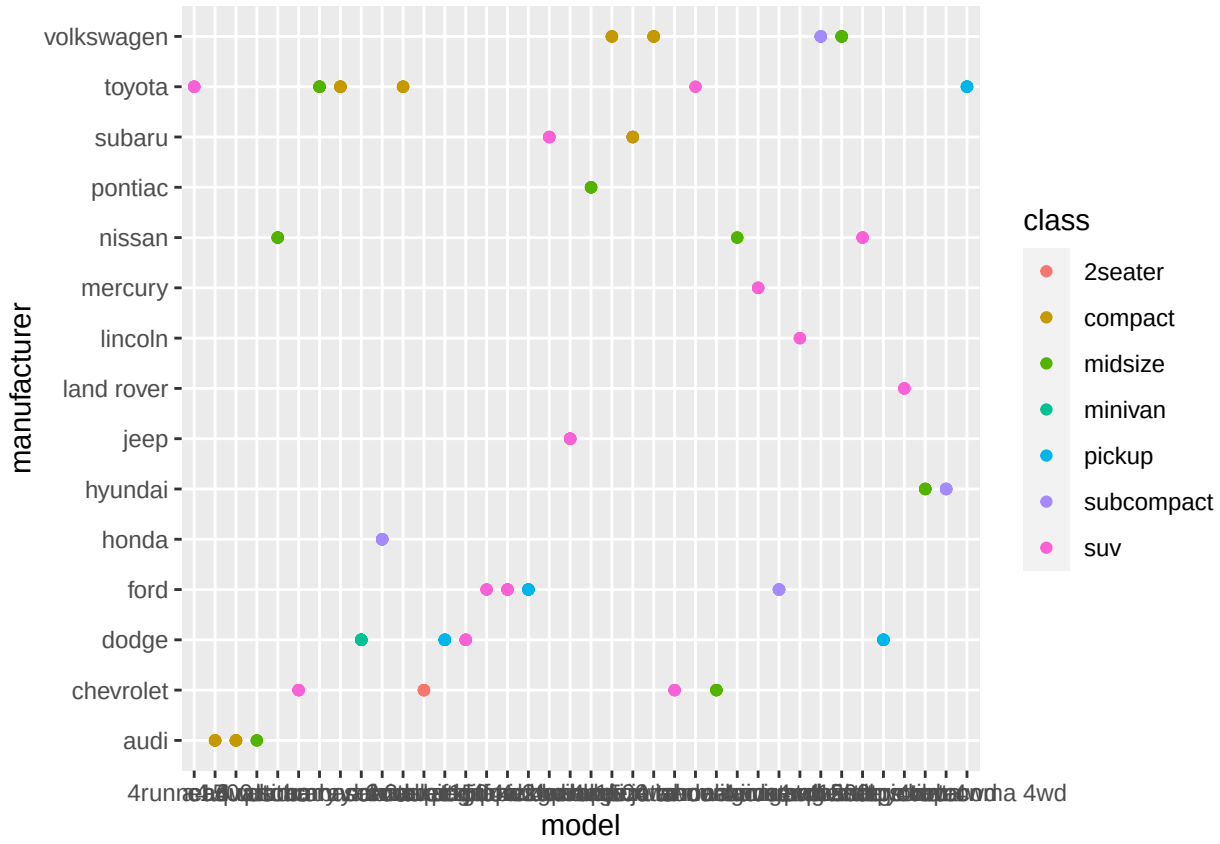



```
ggplot(mpg, aes(model, manufacturer)) + geom_point()
```

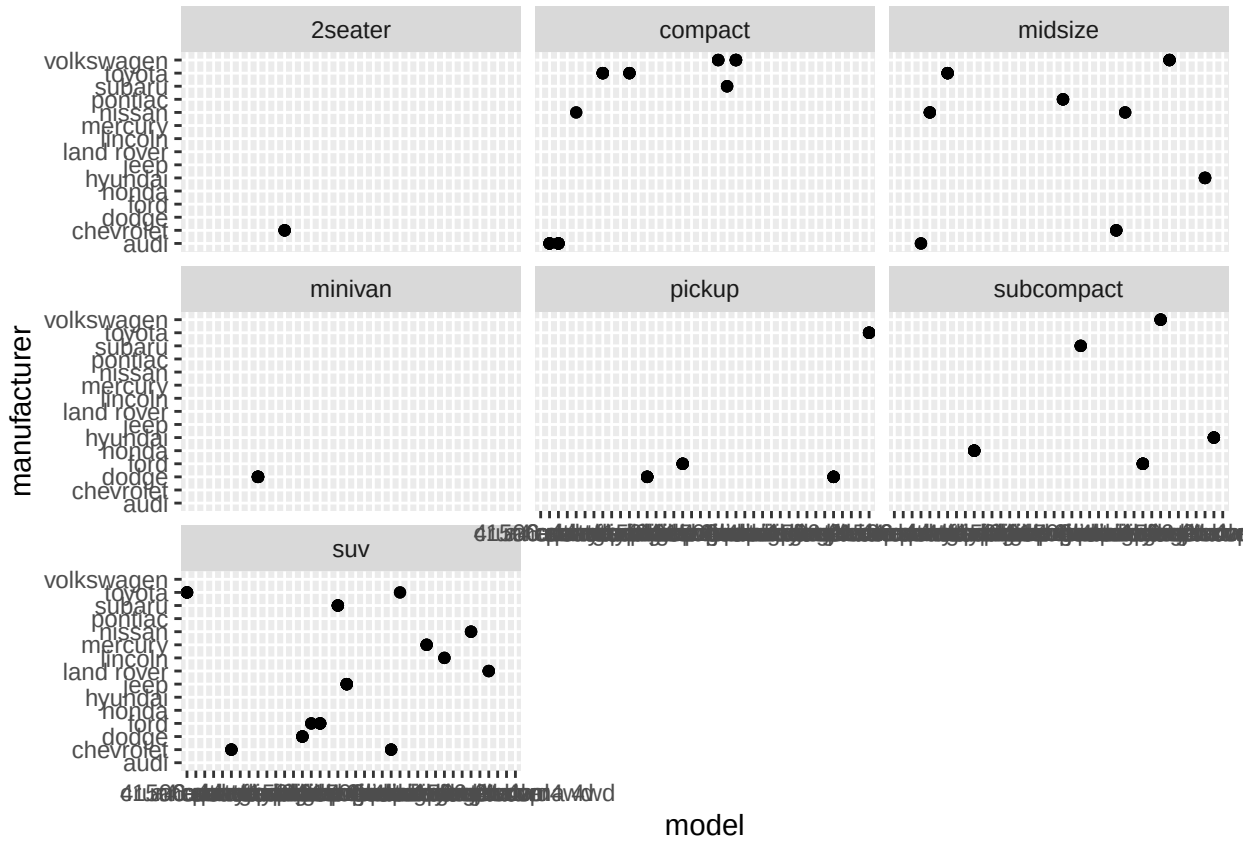



```
#The code creates a scatter plot using the ggplot2 package in R. In this specific plot, the x-axis repr
```

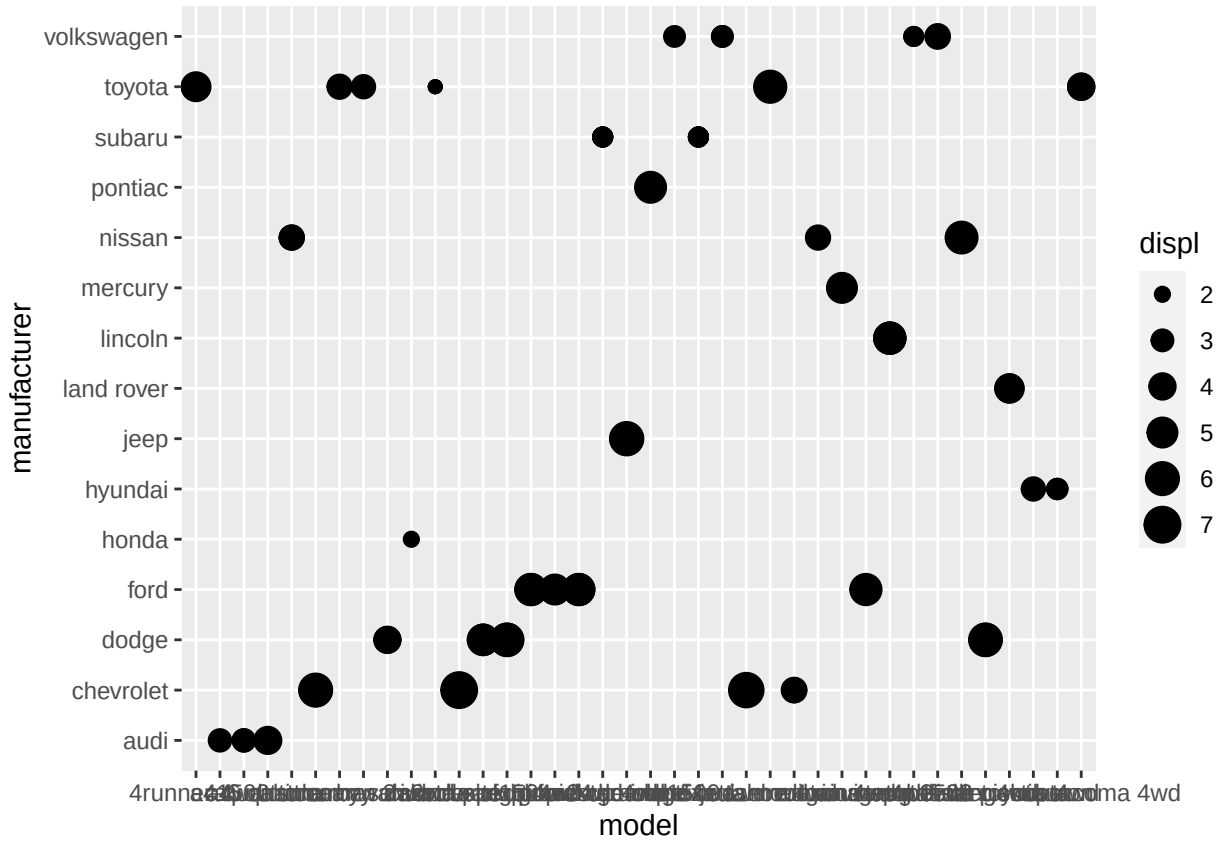
```
#Coloring by a categorical variable
ggplot(mpg, aes(model, manufacturer, color = class)) + geom_point()
```




```
#Faceting by a categorical variable
ggplot(mpg, aes(model, manufacturer)) +
  geom_point() +
  facet_wrap(~class)
```

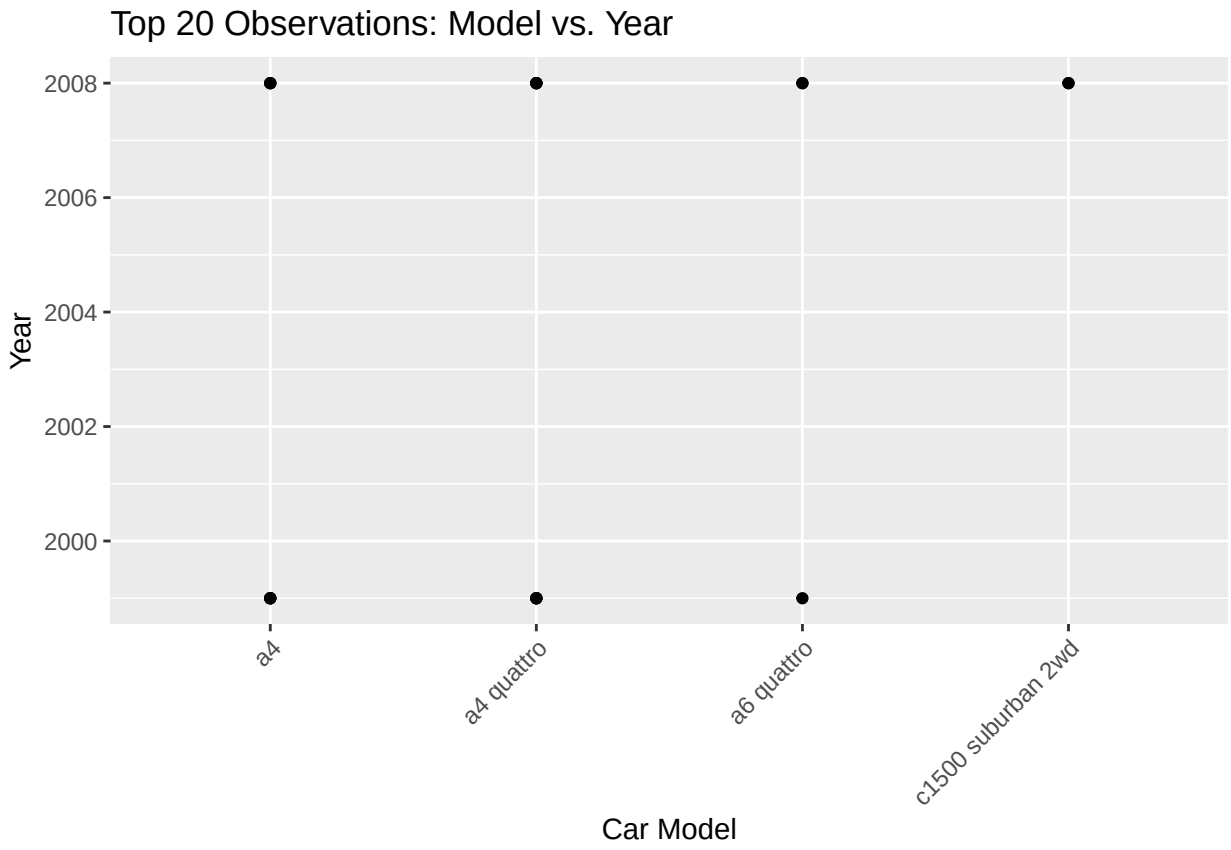



```
#Sizing by a numerical variable
ggplot(mpg, aes(model, manufacturer, size = displ)) + geom_point()
```



#3

```
data(mpg)

top20 <- head(mpg, 20)

ggplot(top20, aes(x = model, y = year)) +
  geom_point() +
  labs(x = "Car Model", y = "Year", title = "Top 20 Observations: Model vs. Year") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



#4

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

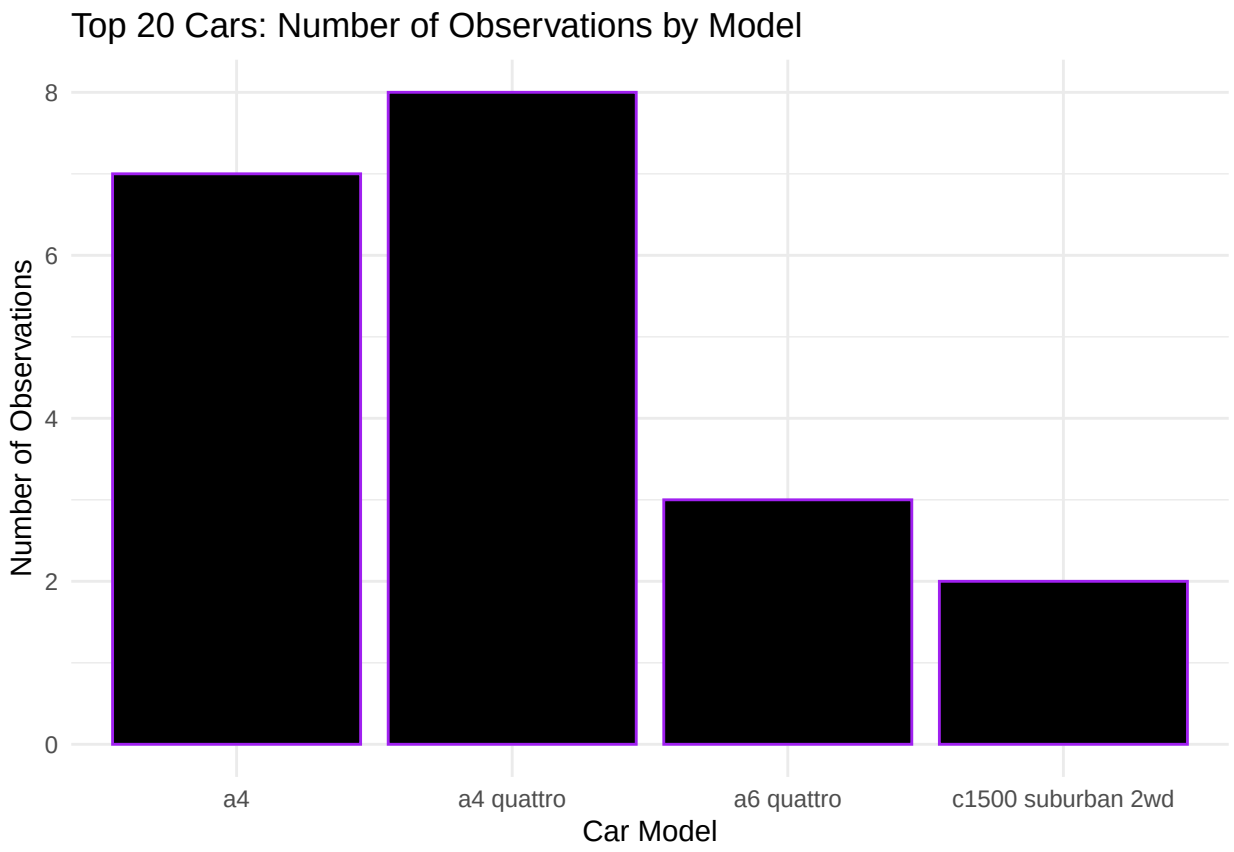
```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
data(mpg)

cars_per_model <- mpg %>%
  group_by(model) %>%
  summarize(number_of_cars = n())

print(cars_per_model)
```

```
## # A tibble: 38 x 2
##    model              number_of_cars
##    <chr>                       <int>
##  1 4runner 4wd                     6
##  2 a4                              7
##  3 a4 quattro                      8
```

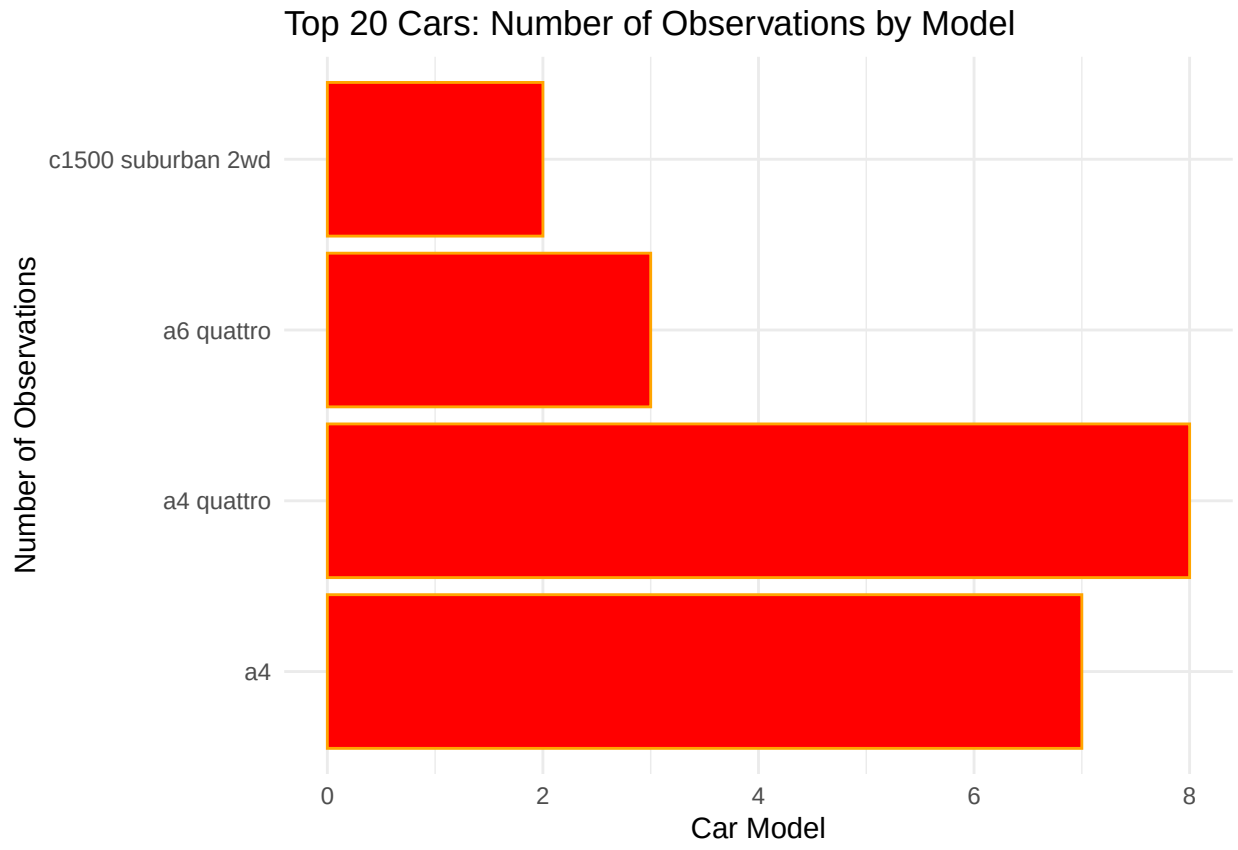
```
##  4 a6 quattro                     3
##  5 altima                         6
##  6 c1500 suburban 2wd             5
##  7 camry                          7
##  8 camry solara                   7
##  9 caravan 2wd                   11
## 10 civic                          9
## # i 28 more rows
```

```r
library(ggplot2)

data(mpg)

top20 <- head(mpg, 20)

ggplot(top20, aes(x = model)) +
  geom_bar(fill = "black", color = "purple") +
  labs(
    title = "Top 20 Cars: Number of Observations by Model",
    x = "Car Model",
    y = "Number of Observations"
  ) +
  theme_minimal()
```



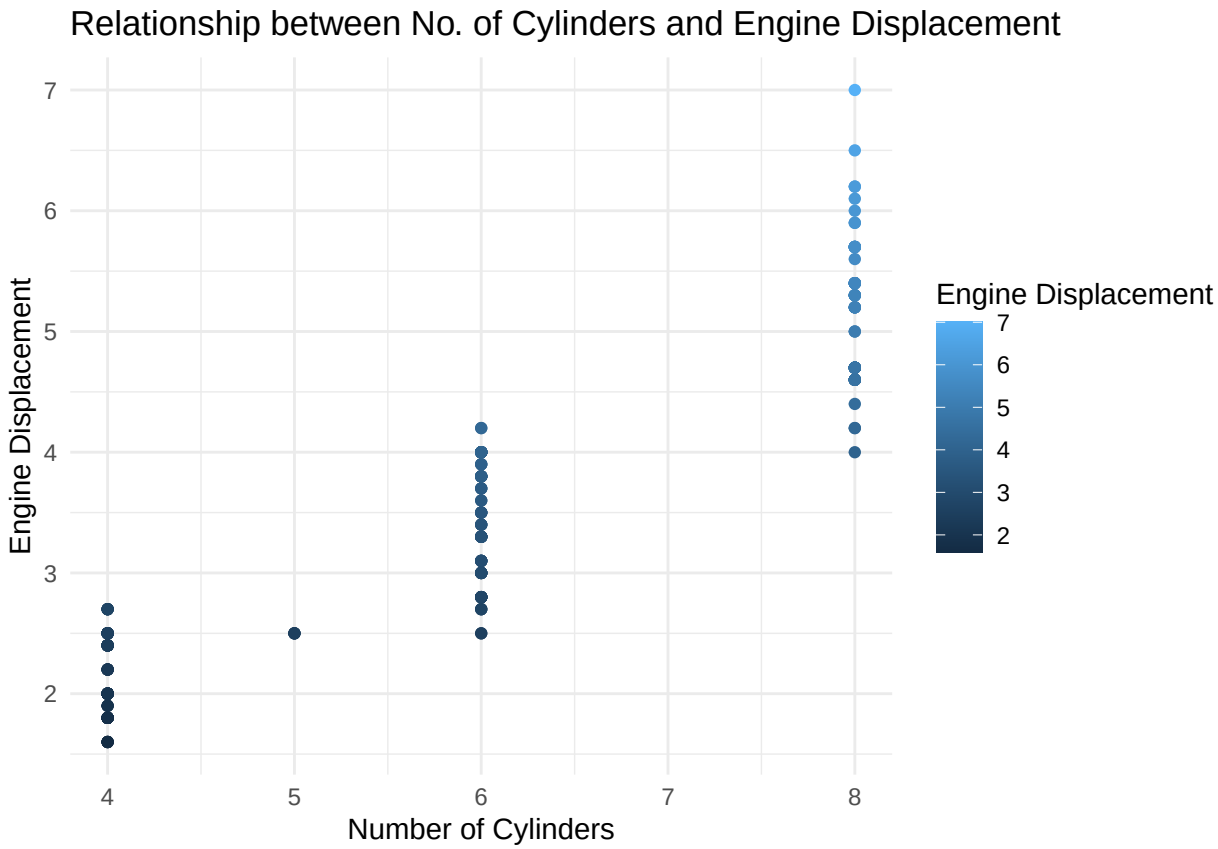

```r
data(mpg)

top20 <- head(mpg, 20)
```

```r
ggplot(top20, aes(x = model)) +
  geom_bar(fill = "red", color = "orange") +
  labs(
    title = "Top 20 Cars: Number of Observations by Model",
    x = "Number of Observations",
    y = "Car Model"
  ) +
  theme_minimal() +
  coord_flip()
```

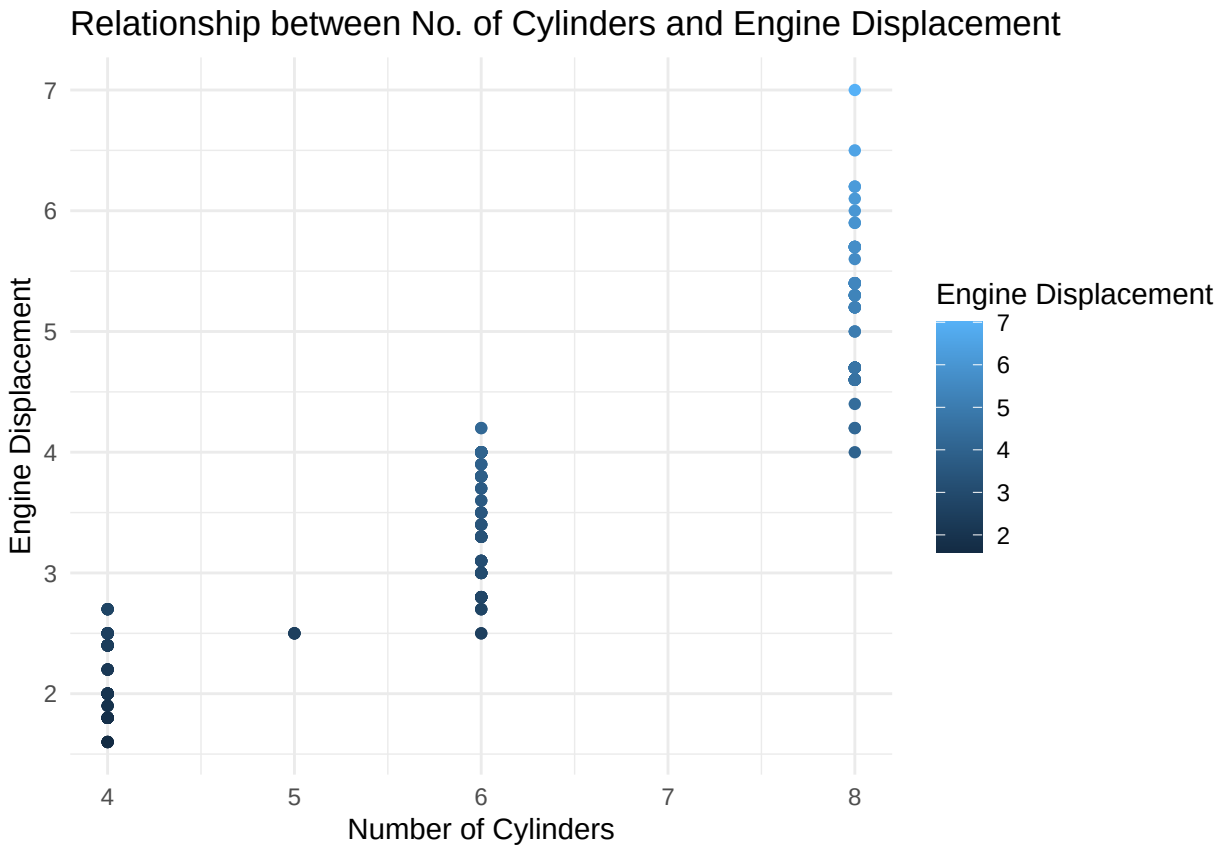


#5

```r
data(mpg)

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_continuous(name = "Engine Displacement") +
  theme_minimal()
```




```
data(mpg)

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_continuous(name = "Engine Displacement") +
  theme_minimal()
```

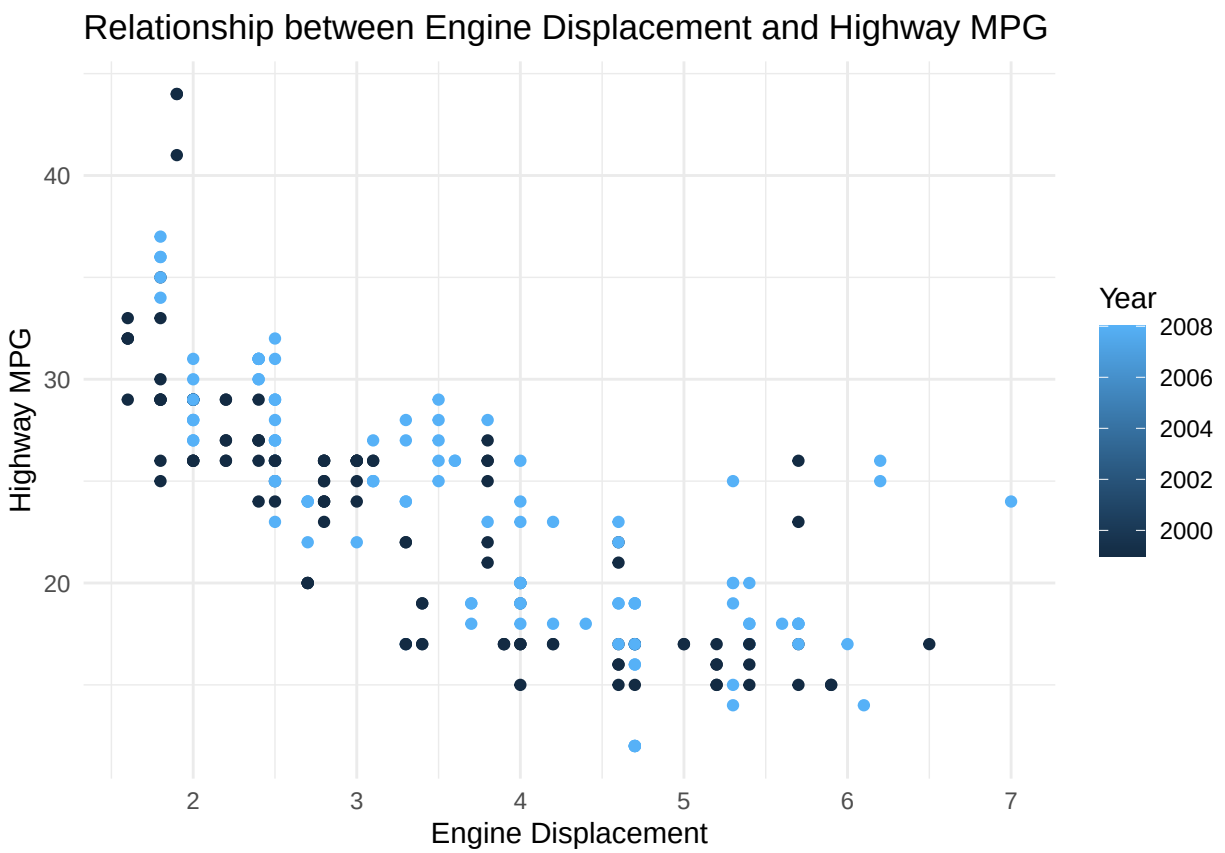



```
correlation <- cor(mpg$cyl, mpg$displ)
cat("Correlation Coefficient:", correlation, "\n")
```

```
## Correlation Coefficient: 0.9302271
```

#6

```
data(mpg)

ggplot(mpg, aes(x = displ, y = hwy, color = year)) +
  geom_point() +
  labs(
    title = "Relationship between Engine Displacement and Highway MPG",
    x = "Engine Displacement",
    y = "Highway MPG"
  ) +
  scale_color_continuous(name = "Year") +
  theme_minimal()
```



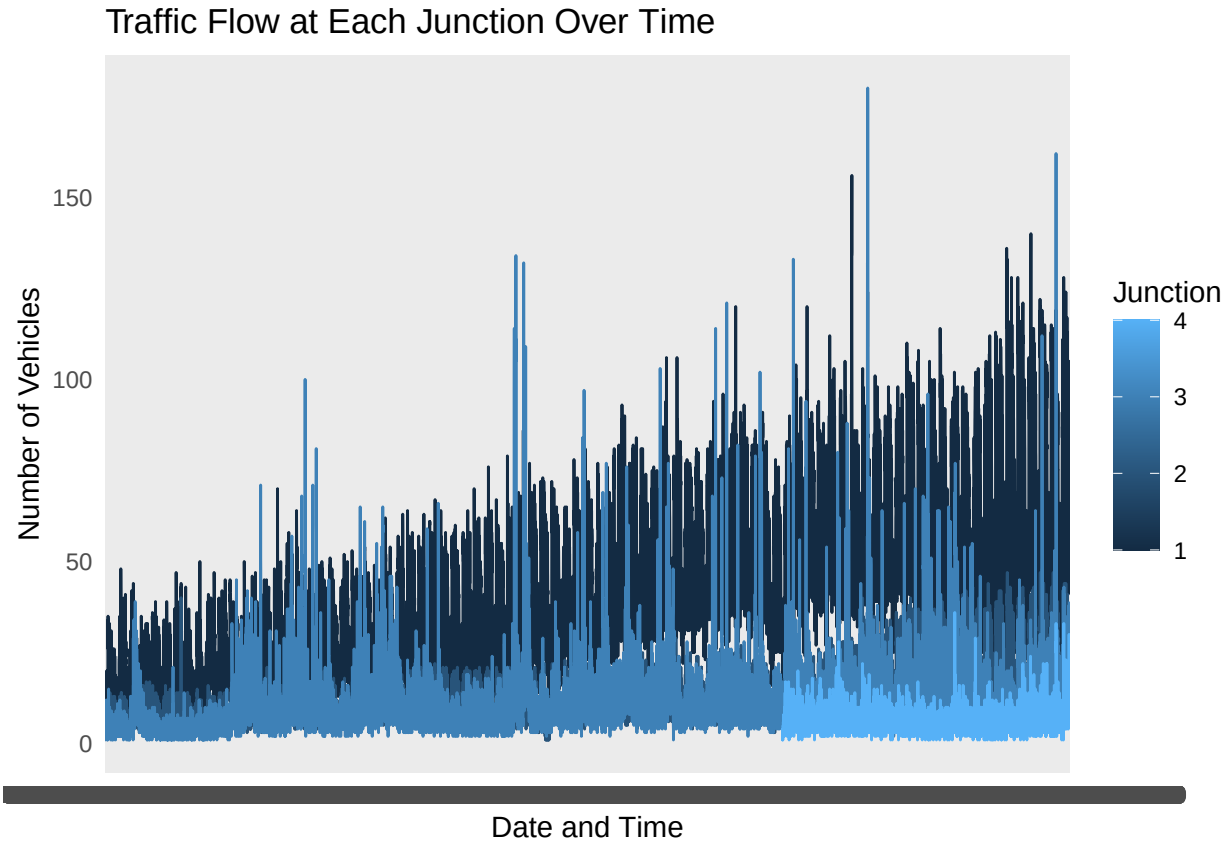

```
num_observations <- read.csv("traffic.csv")

nrow(num_observations)
```

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

```
library(dplyr)

junction_data <- num_observations %>%
  filter(!is.na(Junction))

head(junction_data)
```

```
##              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
```

```
junction_data <- num_observations %>%
  filter(!is.na(Junction))

ggplot(junction_data, aes(x = DateTime, y = Vehicles, group = Junction, color = Junction)) +
  geom_line() +
  labs(title = "Traffic Flow at Each Junction Over Time", x = "Date and Time", y = "Number of Vehicles")
  theme_minimal()
```



#7

```
library(readxl)

alexa_file <- read_excel("alexa_file.xlsx")
alexa_file
```

```
## # 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
```

```
num_rows <- nrow(alexa_file)
num_columns <- ncol(alexa_file)
num_rows
```

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

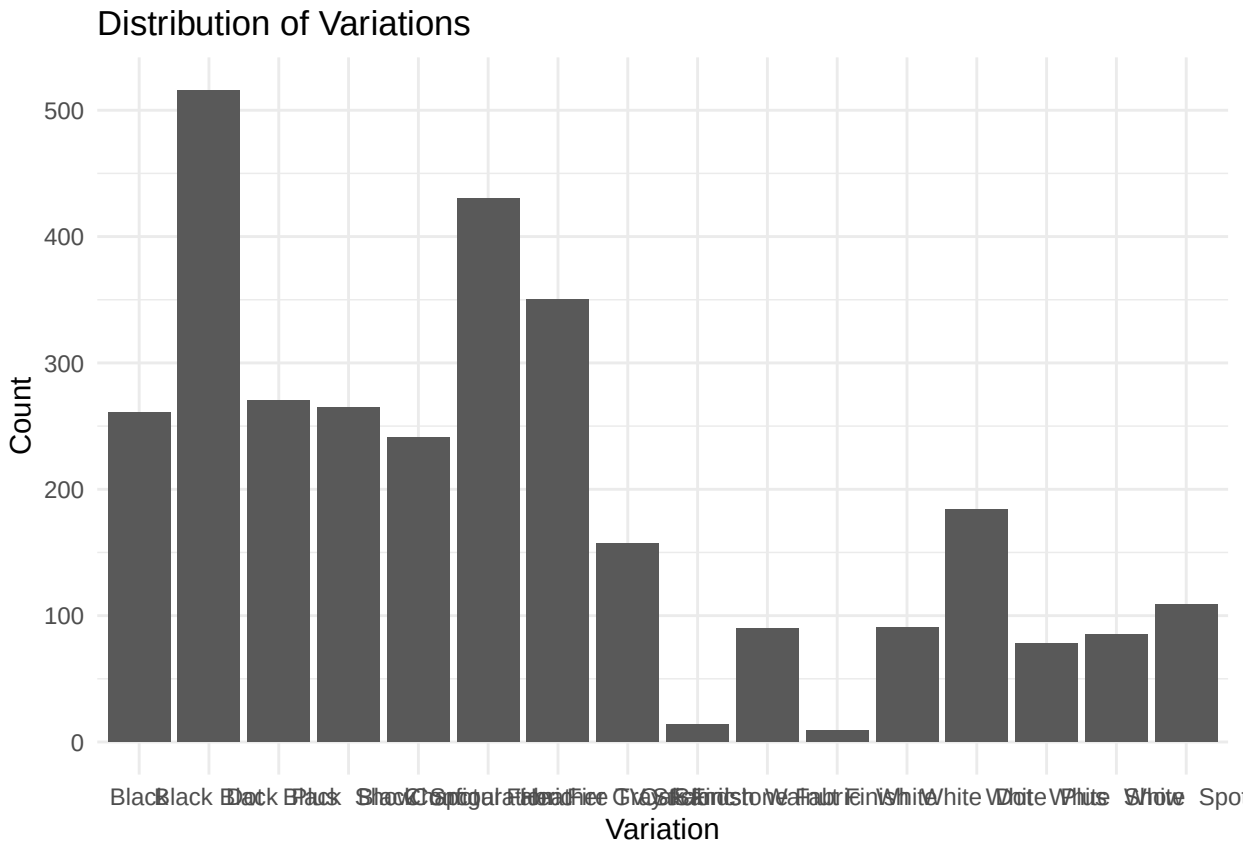
```
num_columns
```

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

```
library(dplyr)

result <- alexa_file %>%
  group_by(variation) %>%
  summarize(total_count = n())

print(result)
```

```
## # 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
```

```
library(ggplot2)

ggplot(alexa_file, aes(x = variation)) +
  geom_bar() +
  labs(title = "Distribution of Variations", x = "Variation", y = "Count") +
  theme_minimal()
```




```
library(ggplot2)

alexa_file$verified_reviews <- iconv(alexa_file$verified_reviews, to = "ASCII", sub = " ")


ggplot(alexa_file, aes(x = date, y = verified_reviews, group = 1)) +
  geom_line(color = "black") +
  labs(title = "Verified Reviews Over Time",
       x = "Date",
       y = "Number of Verified Reviews") +
  theme_minimal()
```

serious flaws, particularly if you are the last one to bed or the first to wake. It doesn t seem like the engineers actuall

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

one of the lights by saying "Alexa, turn off the second light". In the Alexa app, I created a 'Group' with &#34
getting terrible support. The guy that took my call just rambled off a (completely unhelpful) script and I ended up settir

nting 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

```
library(ggplot2)
```

```r
ggplot(alexa_file, aes(x = variation, y = rating, fill = variation)) +
  geom_boxplot() +
  labs(title = "Relationship Between Variations and Ratings",
       x = "Variation",
       y = "Rating") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```