RWorksheet_Sadural#4c

2023-11-22

```
#1.Use the dataset mpg
mpg <- read.csv("mpg.csv")
```

```
#B.
categorical_vars <- c("manufacturer", "model", "trans", "drv", "fl", "class")
categorical_vars
```

```
## [1] "manufacturer" "model"        "trans"        "drv"          "fl"
## [6] "class"
```

```
#1c.
continuous_vars <- c("displ", "year", "cyl", "cty", "hwy")
continuous_vars
```

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

```
#2.
#A.
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
```

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

manufacturer_models[1, ]
```

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

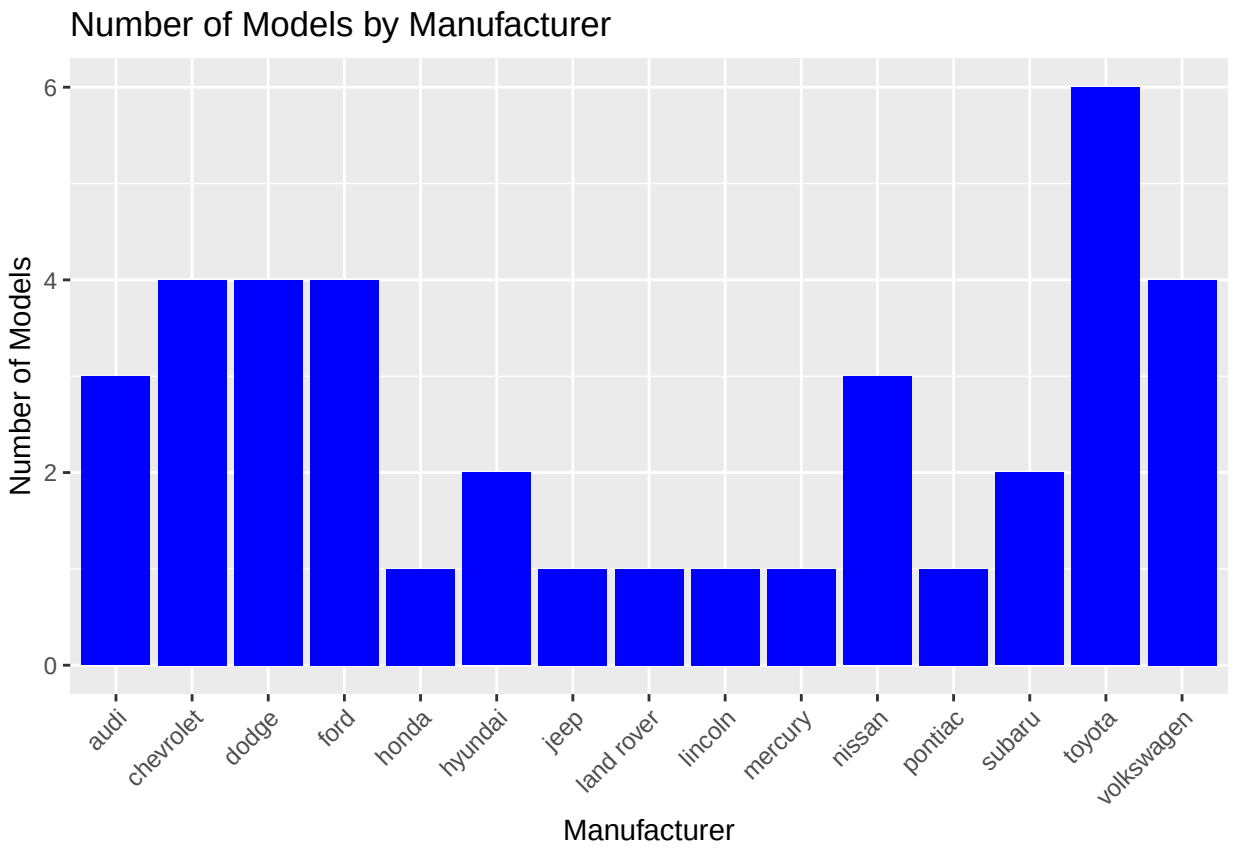
```
#B.
library(ggplot2)
```

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

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

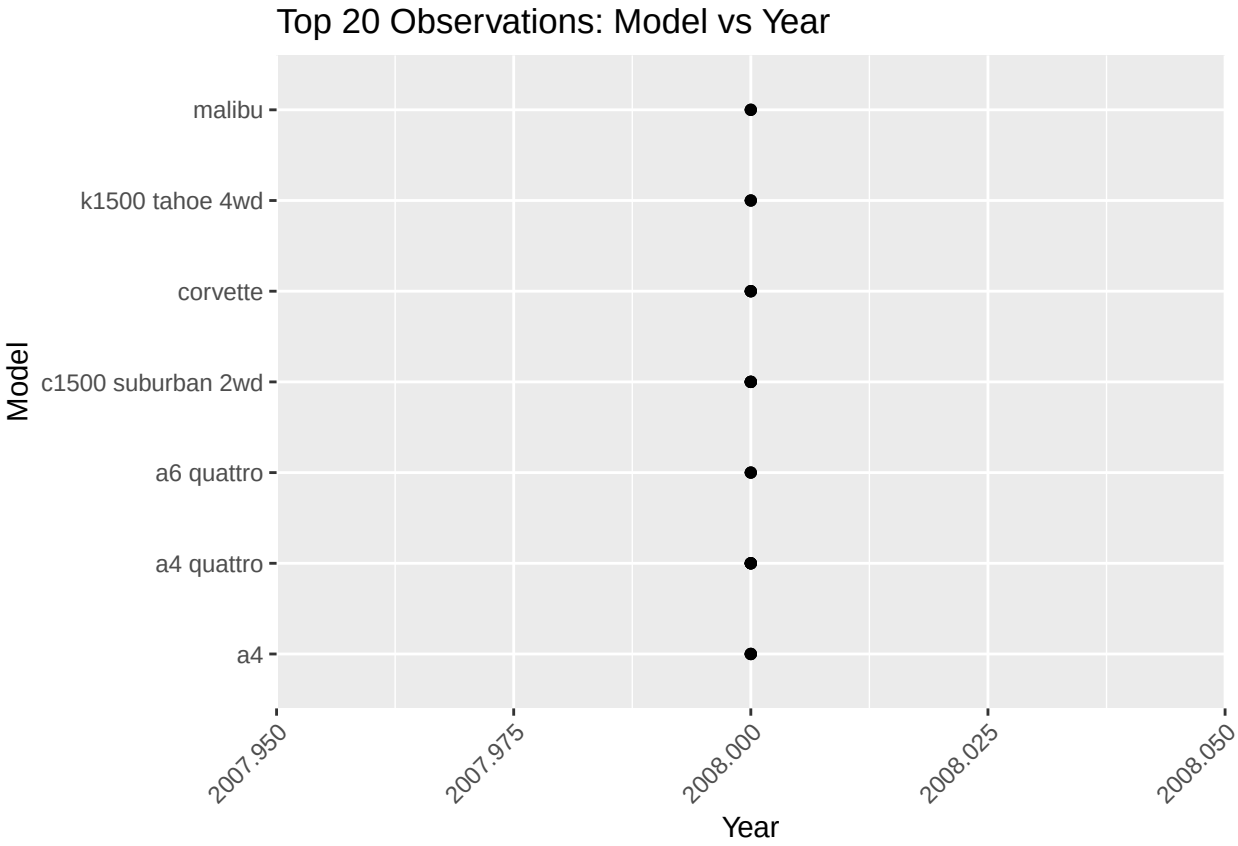
```
##
##     mpg
```

```
ggplot(manufacturer_models, aes(x = manufacturer, 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))
```




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

#This ggplot command creates a scatter plot where each point represents a car model, positioned along t

#B. Is it useful? If not, how could you modify the data to make it more informative?

#The plot may be useful for visualizing the distribution of models across manufacturers, but it could b
```

```
#3.
top_20_data <- head(mpg[order(mpg$year, decreasing = TRUE), ], 20)
ggplot(top_20_data, aes(x = year, y = model)) +
  geom_point() +
  labs(title = "Top 20 Observations: Model vs Year", x = "Year", y = "Model") + theme(axis.text.x = ele
```

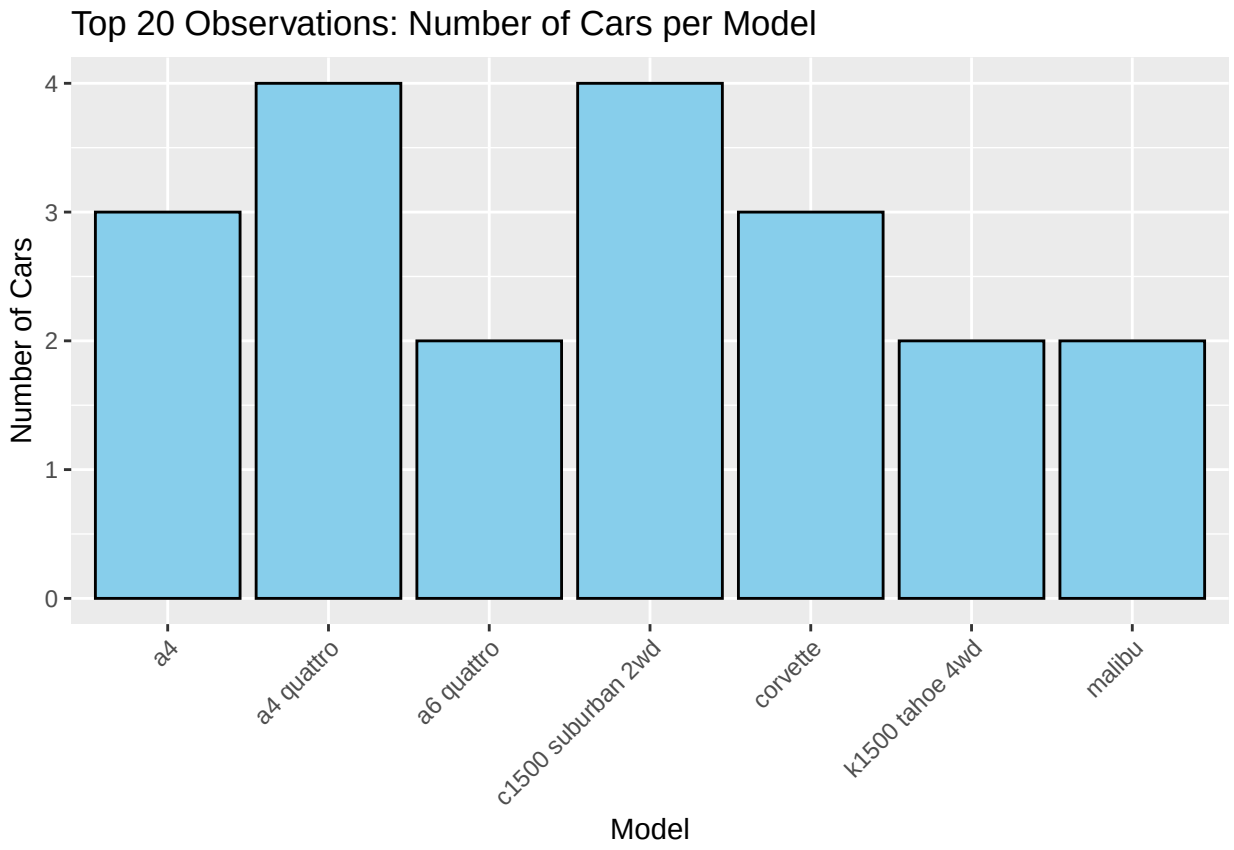



```
#4.
library(dplyr)

cars_per_model <- mpg %>%
  group_by(model) %>%
  summarise(num_cars = n())

  print(cars_per_model)
```

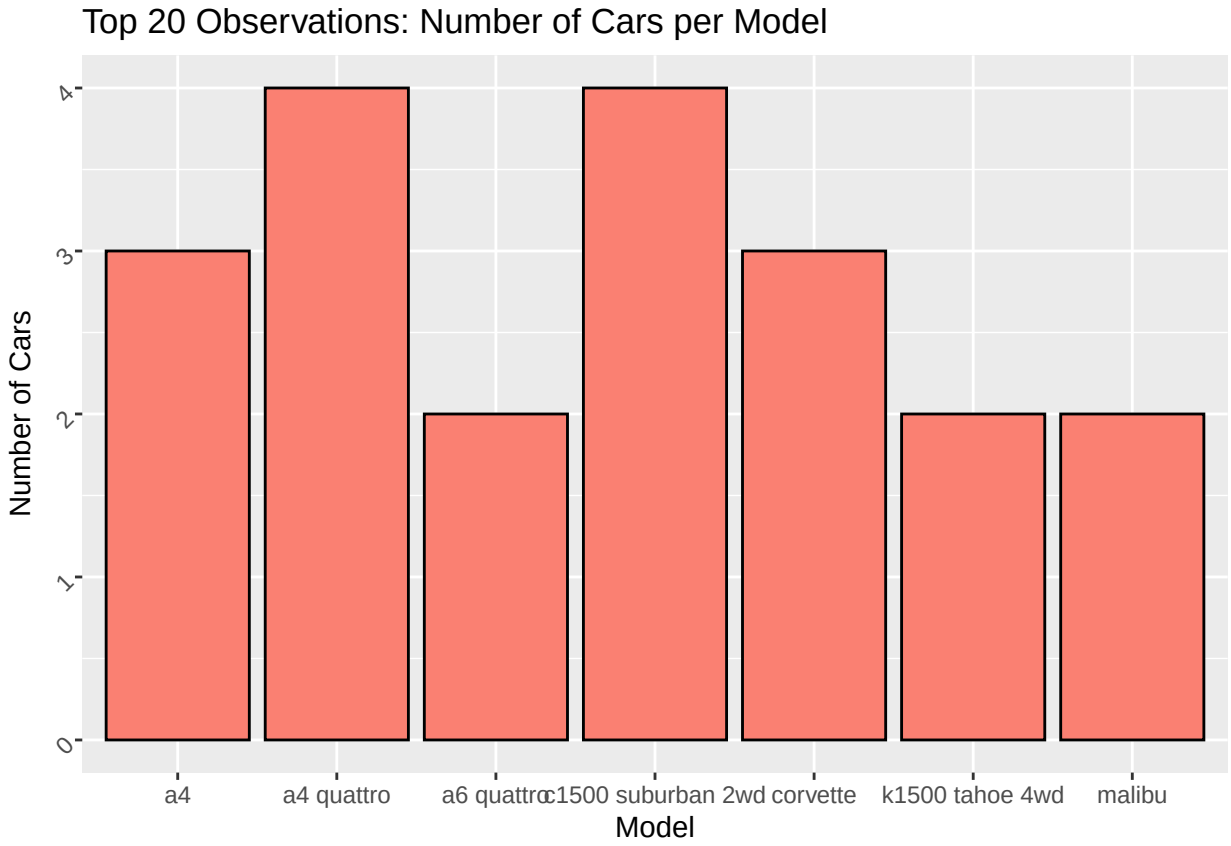
```
## # A tibble: 38 x 2
##    model              num_cars
##    <chr>                 <int>
##  1 4runner 4wd               6
##  2 a4                        7
##  3 a4 quattro                8
##  4 a6 quattro                3
##  5 altima                    6
##  6 c1500 suburban 2wd        5
##  7 camry                     7
##  8 camry solara              7
##  9 caravan 2wd              11
## 10 civic                     9
## # i 28 more rows
```

```
#4A.
top_20_data <- head(mpg[order(mpg$year, decreasing = TRUE), ], 20)

ggplot(top_20_data, aes(x = model)) +
```

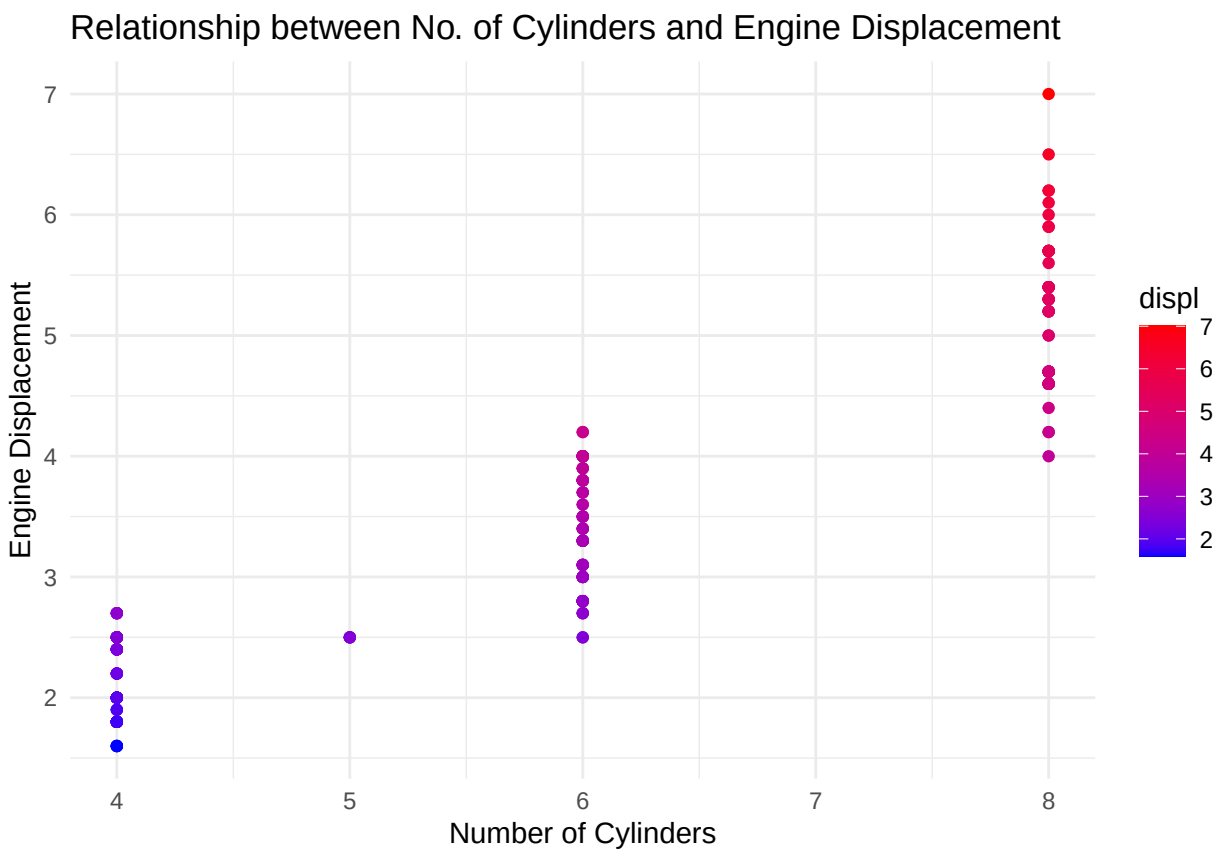
```
  geom_bar(fill = "skyblue", color = "black") +
  labs(title = "Top 20 Observations: Number of Cars per Model",
       x = "Model", y = "Number of Cars") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```




```
ggplot(top_20_data, aes(y = model)) +
  geom_bar(fill = "salmon", color = "black") +
  labs(title = "Top 20 Observations: Number of Cars per Model",
       x = "Number of Cars", y = "Model") +
  coord_flip() +
  theme(axis.text.y = element_text(angle = 45, hjust = 1))
```

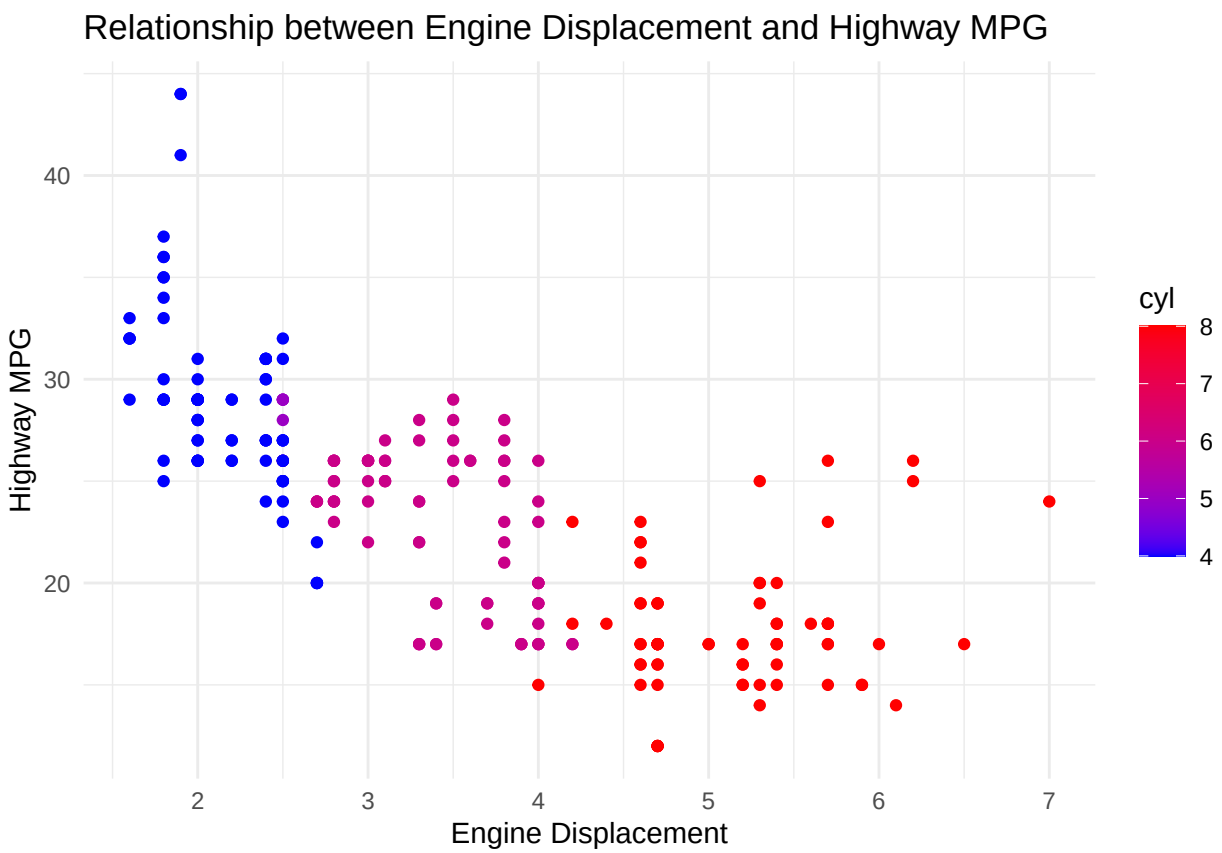



```
#5.

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_gradient(low = "blue", high = "red") +
  theme_minimal()
```




```
#A.
#The plot shows that there is a positive relationship between the number of cylinders and engine displa
```

```
#5.
ggplot(mpg, aes(x = displ, y = hwy, color = cyl)) +
  geom_point() +
  labs(title = "Relationship between Engine Displacement and Highway MPG",
       x = "Engine Displacement", y = "Highway MPG") +
  scale_color_gradient(low = "blue", high = "red") +
  theme_minimal()
```




```
#6.
traffic <- read.csv("traffic.csv")
```

```
#A.
dim(traffic)
```

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

```
colnames(traffic)
```

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

```
#B.
junctions <- unique(traffic$junction_name)
junctions
```

```
## NULL
```

```
#C..
library(ggplot2)

for (j in junctions) {
 subset_traffic <- traffic[traffic$junction_name == j, ]
 ggplot(subset_traffic, aes(x = timestamp, y = congestion_value, group = ID)) +
    geom_line() +
    labs(title = paste("Congestion over time for junction", j)) +
    theme_minimal()
}
```

```r
#7

library(openxlsx)

alexafile <- read.xlsx("alexa_file.xlsx")
```

```r
#A.
alexa_obser <- nrow(alexafile)

alexacol_obser <- ncol(alexafile)

cat("The number of observations on alexa is:",alexa_obser,"\n")
```

```
## The number of observations on alexa is: 3150
```

```r
cat("The number of coloumns on alexa is:",alexacol_obser,"\n")
```

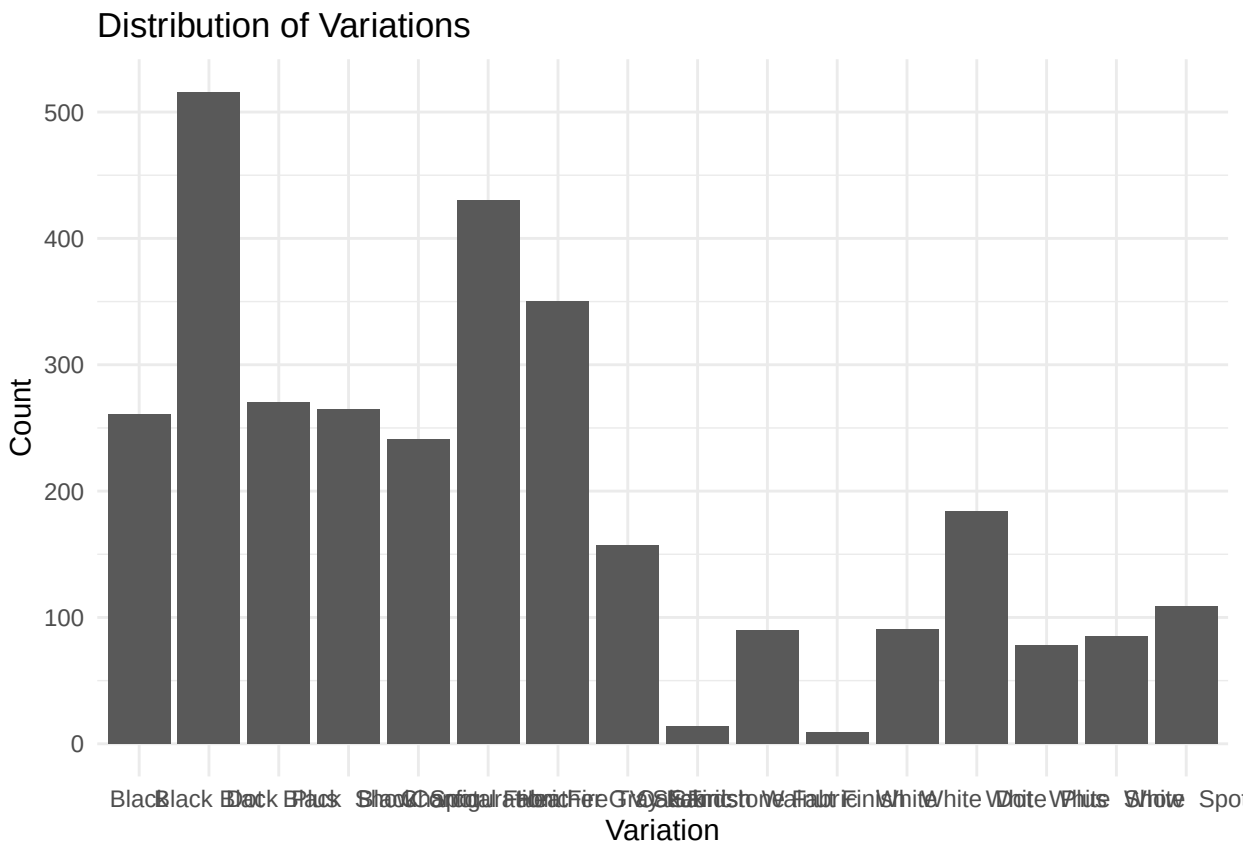
```
## The number of coloumns on alexa is: 5
```

```r
#B.
library(dplyr)
groupvari <- alexafile%>%
  group_by(variation)%>%
  summarise(totalcount_ = n())


groupvari
```

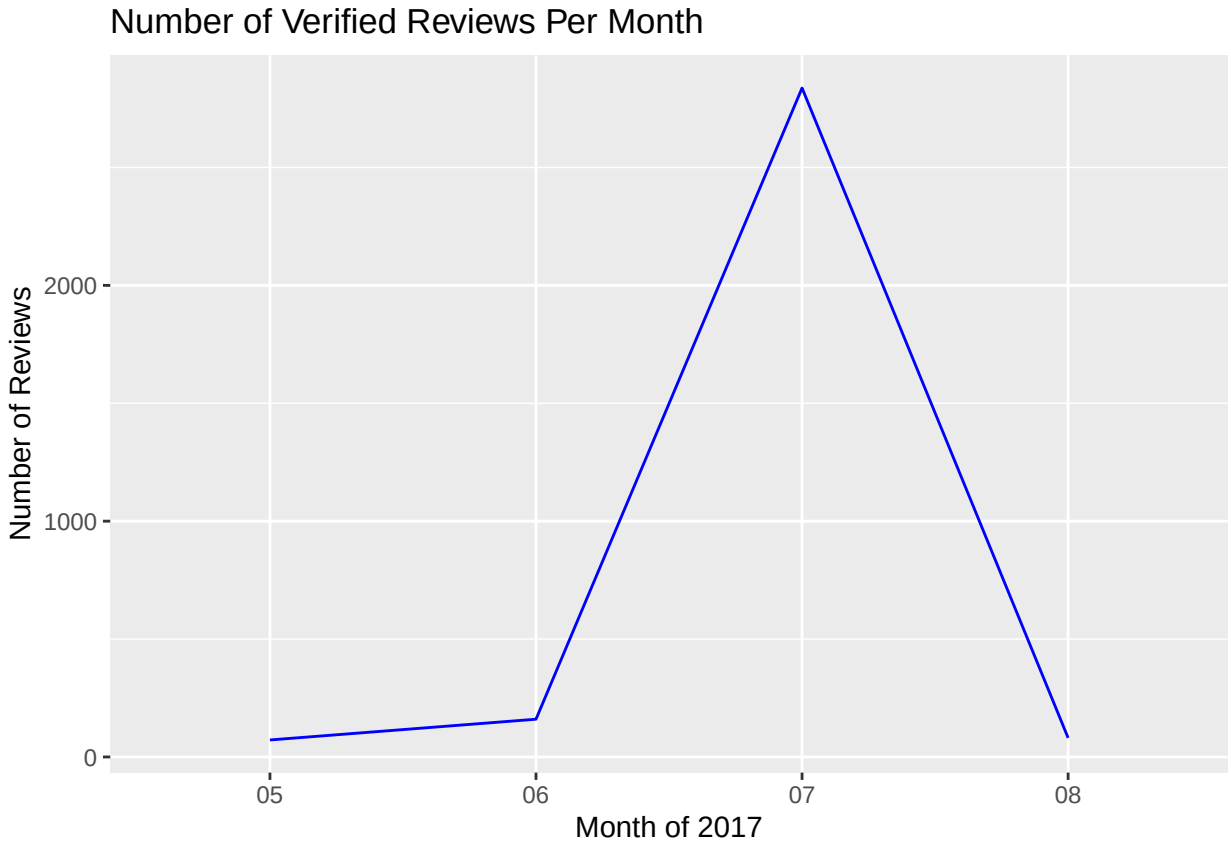
```
## # A tibble: 16 x 2
##    variation                       totalcount_
##    <chr>                                 <int>
##  1 "Black"                                 261
##  2 "Black  Dot"                            516
##  3 "Black  Plus"                           270
##  4 "Black  Show"                           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
#C.
library(ggplot2)

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

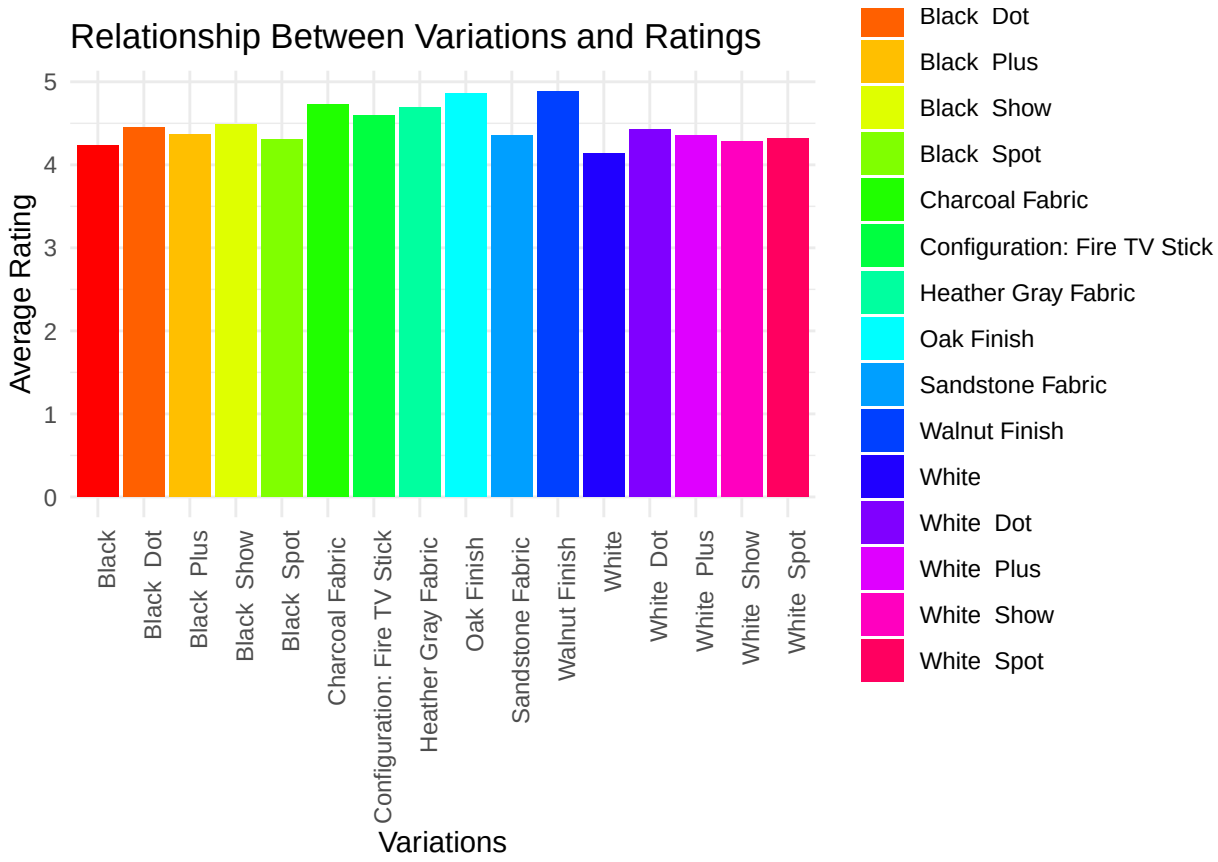
```
  theme_minimal()
```




```
#D.
alexafile$date <- as.Date(alexafile$date)

alexafile$month <- format(alexafile$date, "%m")

monthscount <- alexafile %>%
  group_by(month) %>%
  summarise(num_reviews = n())


monthlyrev <- table(monthscount)

ggplot(monthscount, aes(x = month, y = num_reviews, group = 1)) +
  geom_line(color = "blue") +
  labs(title = "Number of Verified Reviews Per Month",
       x = "Month of 2017", y = "Number of Reviews")
```




```
#E.
library(dplyr)
ggplot(alexafile, aes(x = variation, y = rating, fill = variation)) +
  geom_bar(stat = "summary", fun = "mean", position = "dodge") +
  scale_fill_manual(values = rainbow(n = length(unique(alexafile$variation)))) +
  labs(title = "Relationship Between Variations and Ratings",
       x = "Variations",
       y = "Average Rating") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```




```
var_ratings <-  alexafile %>%
  group_by(variation)%>%
  summarise(avearage_rating = mean(rating, na.rm = TRUE))

max_rating <- max(var_ratings$average_rating, na.rm = TRUE)
```

```
## Warning: Unknown or uninitialised column: `average_rating`.
```

```
## Warning in max(var_ratings$average_rating, na.rm = TRUE): no non-missing
## arguments to max; returning -Inf
```

```
highrate <- alexafile %>%
  filter(rating == max_rating)
print(highrate)
```

```
## [1] rating           date             variation        verified_reviews
## [5] feedback         month
## <0 rows> (or 0-length row.names)
```