worksheet_4c

Jessamine Paula Orada

2025-11-24

```
install.packages("readxl")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.5'
## (as 'lib' is unspecified)
```

```
# Load necessary libraries
library(ggplot2)
library(dplyr)
```

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

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

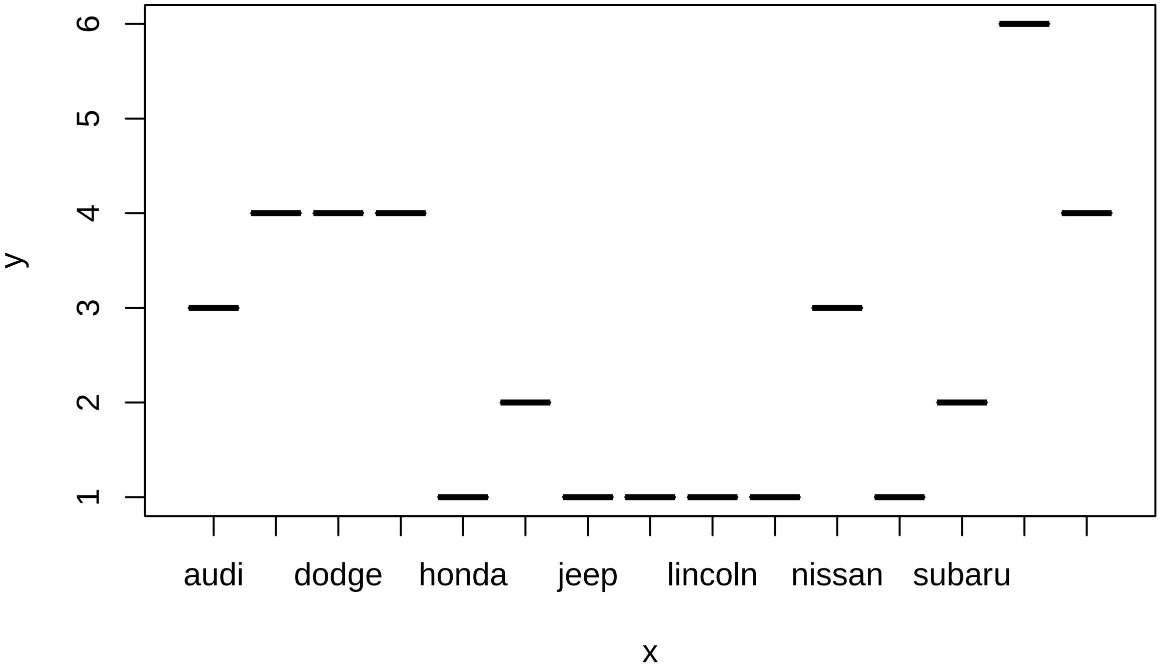
```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(readxl)
```

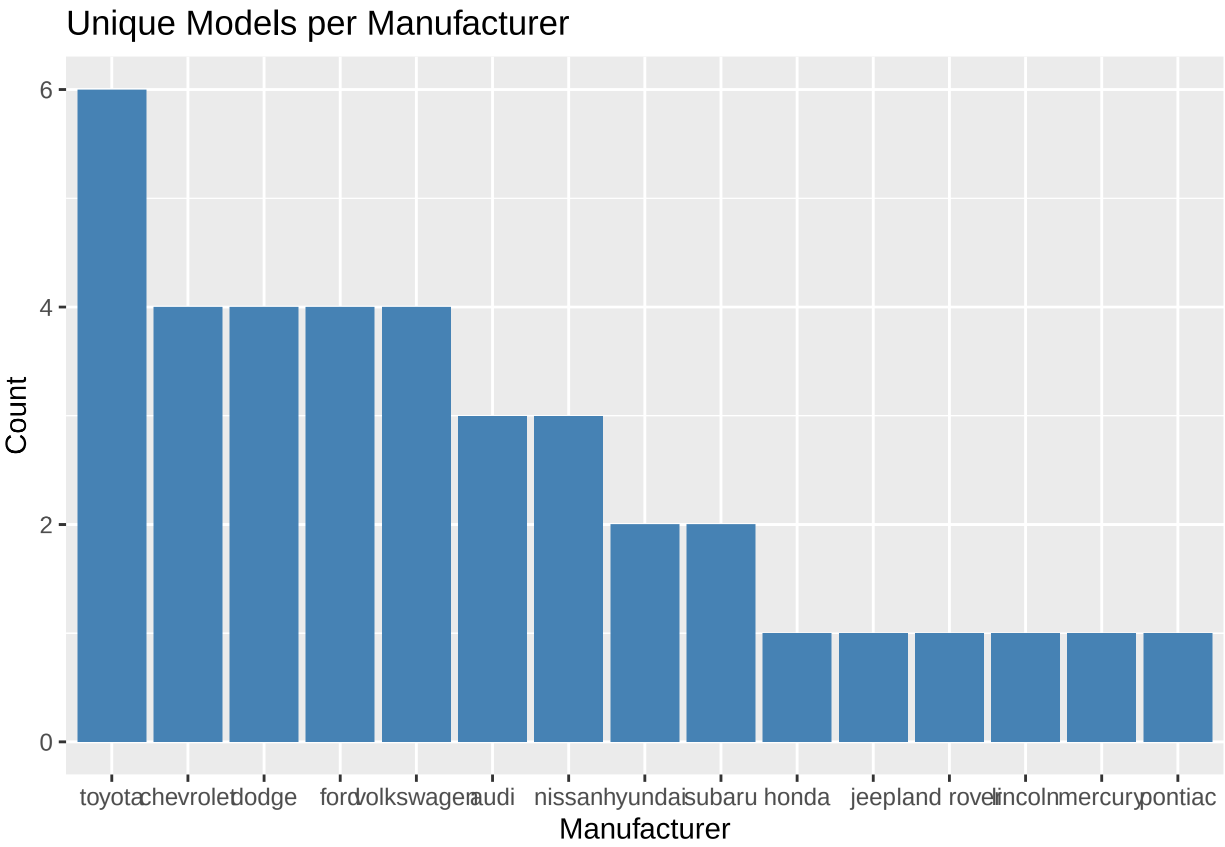
```
# --- 1. Use the dataset mpg --- [cite: 11]
# a. Importing a csv file into the environment [cite: 14]
# Note: Use write.csv(mpg, "mpg.csv") first if you need to create the file locally.
mpg_data <- read.csv("mpg.csv") #[cite: 14]

# b. Categorical variables[cite: 15]:
# manufacturer, model, trans, drv, fl, class [cite: 12]

# c. Continuous variables[cite: 16]:
# displ, cty, hwy [cite: 12]

# --- 2. Manufacturer and Model Analysis --- [cite: 18]
# a. Group manufacturers and find unique models [cite: 19]
manufacturer_unique <- mpg %>%
  group_by(manufacturer) %>%
  summarise(unique_models = n_distinct(model)) #[cite: 19]
manufacturer_unique
```

```
## # A tibble: 15 x 2
##    manufacturer unique_models
##    <chr>                <int>
##  1 audi                     3
##  2 chevrolet                4
##   3 dodge                 4
##   4 ford                  4
##   5 honda                 1
##   6 hyundai               2
##   7 jeep                  1
##   8 land rover            1
##   9 lincoln               1
##  10 mercury               1
##  11 nissan                3
##  12 pontiac               1
##  13 subaru                2
##  14 toyota                6
##  15 volkswagen            4
```

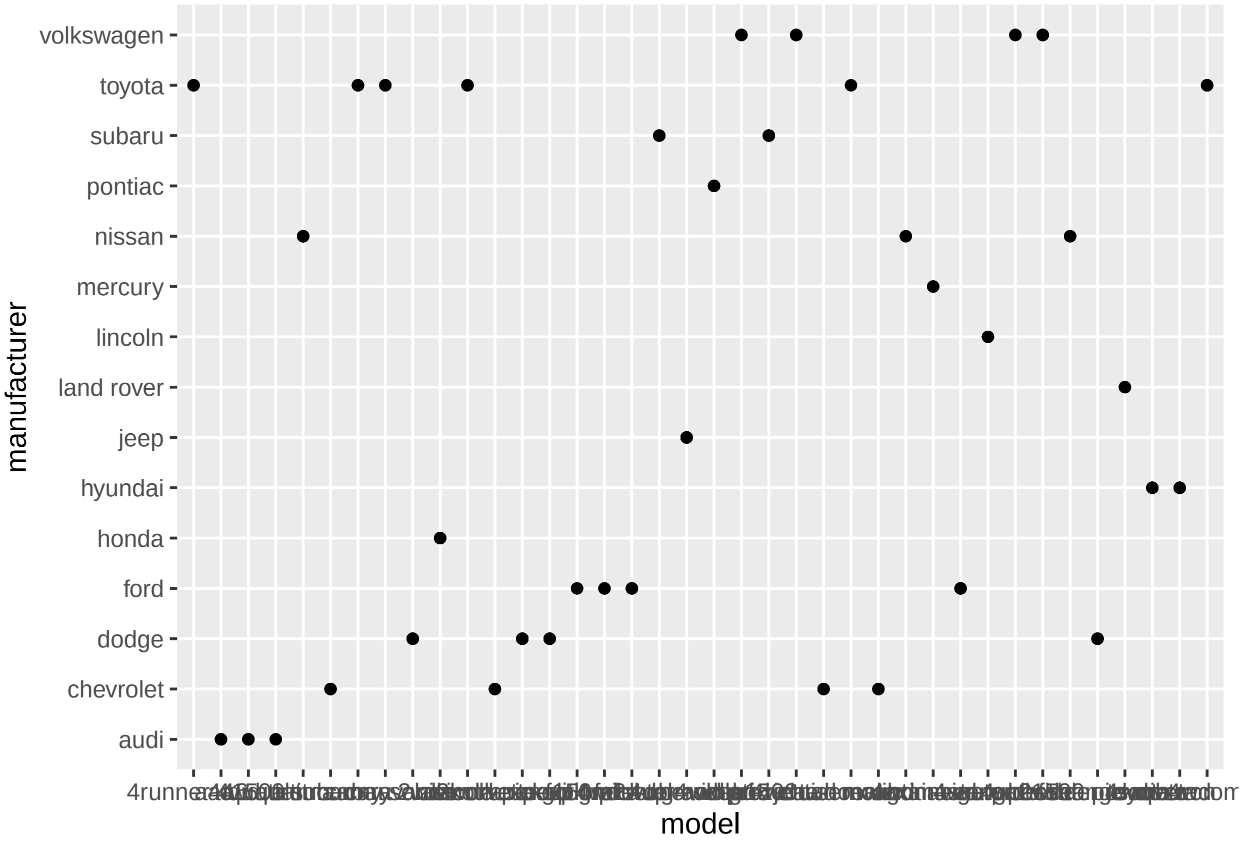
```
# b. Graph the result using plot() and ggplot() [cite: 20]
# Base R plot
plot(as.factor(manufacturer_unique$manufacturer), manufacturer_unique$unique_models) #[cite: 20]
```



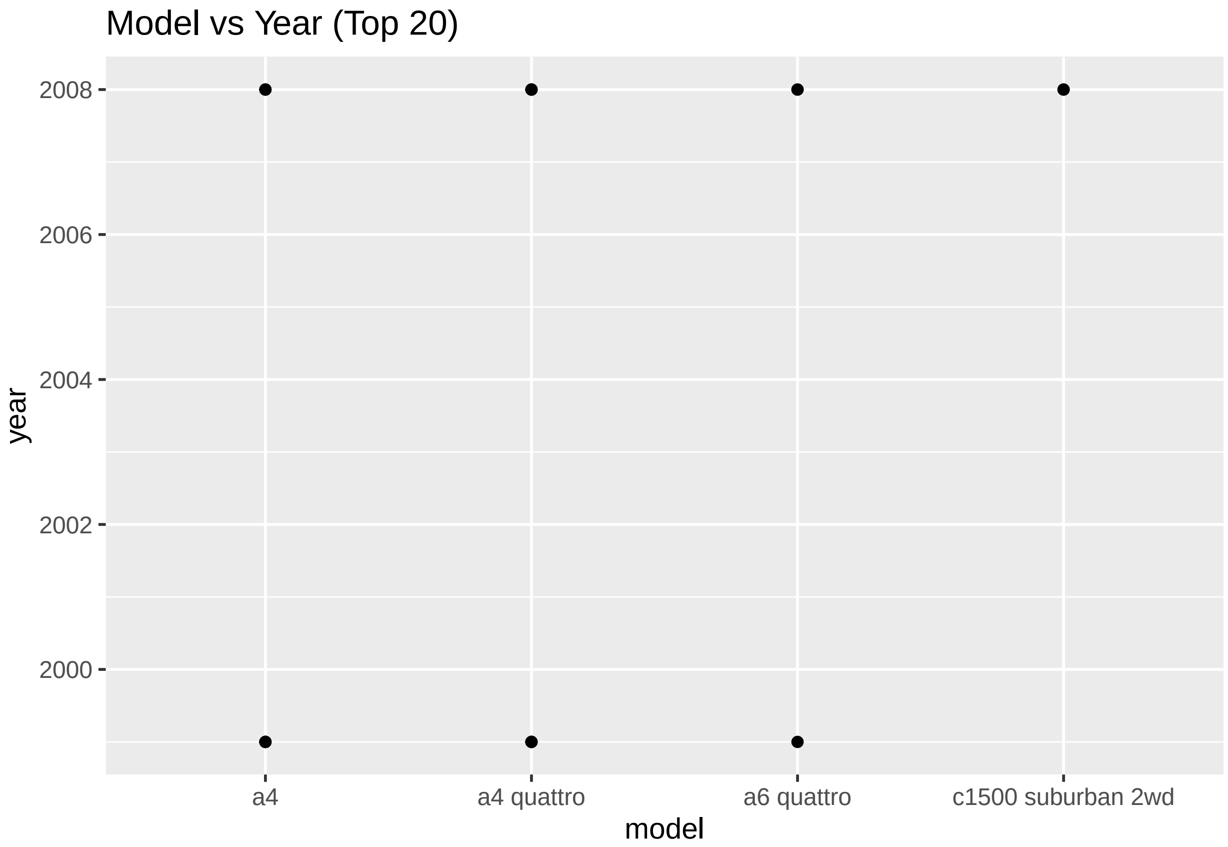
```
# ggplot2 version
ggplot(manufacturer_unique, aes(x = reorder(manufacturer, -unique_models), y = unique_models)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(title = "Unique Models per Manufacturer", x = "Manufacturer", y = "Count") #[cite: 20]
```



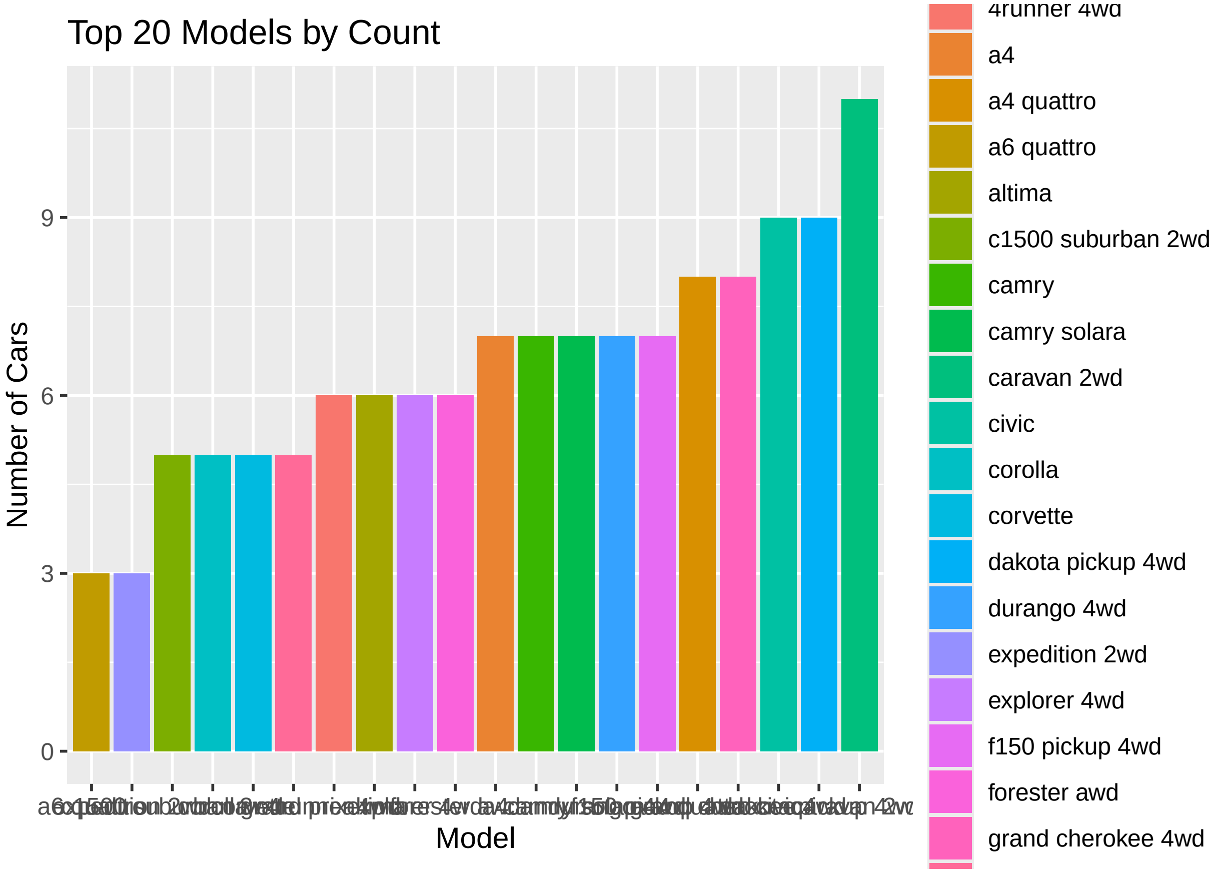
```
# 2. Relationship of model and manufacturer [cite: 21]
# a. This shows a scatter plot of every model vs its manufacturer [cite: 22]
ggplot(mpg, aes(model, manufacturer)) + geom_point() #[cite: 22]
```



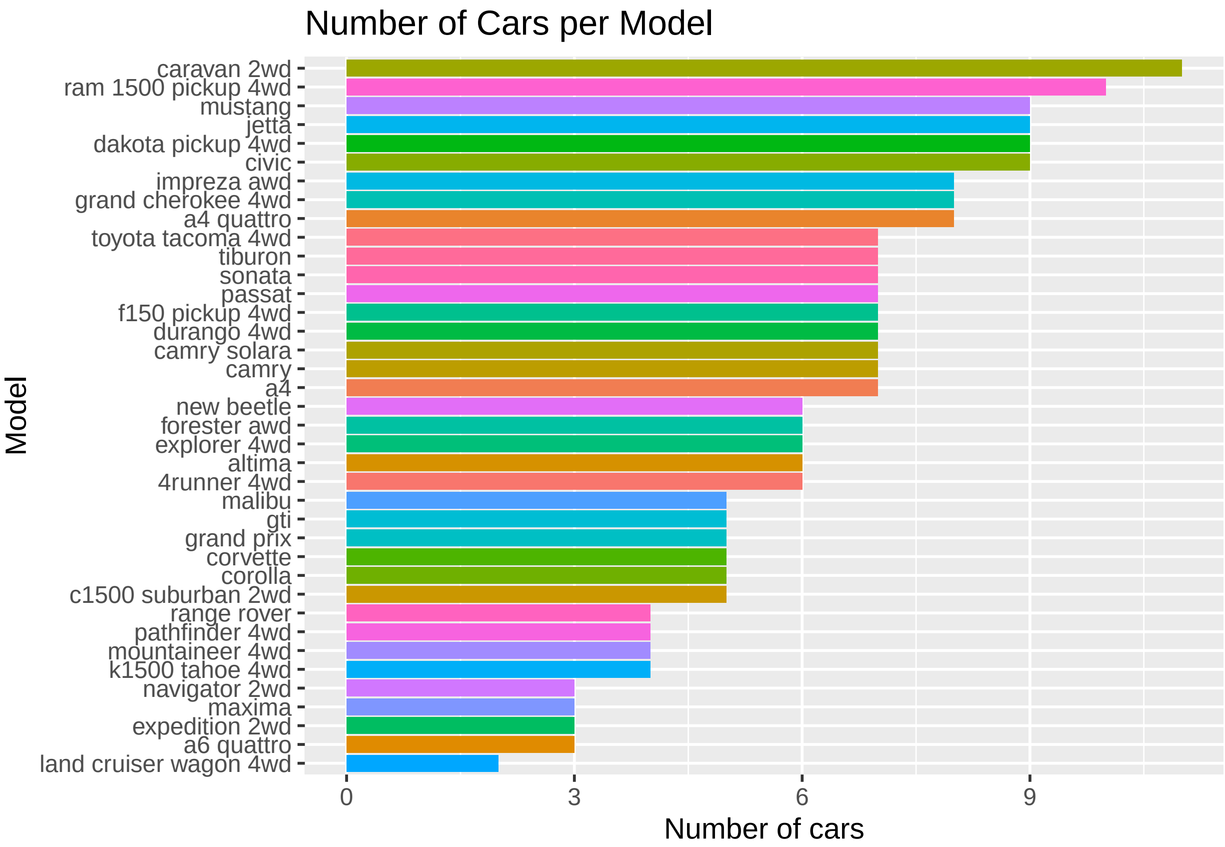
```
# --- 3. Top 20 Observations Plot --- [cite: 24]
top_20 <- head(mpg, 20) #[cite: 24]
ggplot(top_20, aes(x = model, y = year)) +
  geom_point() +
  labs(title = "Model vs Year (Top 20)") #[cite: 24, 25]
```



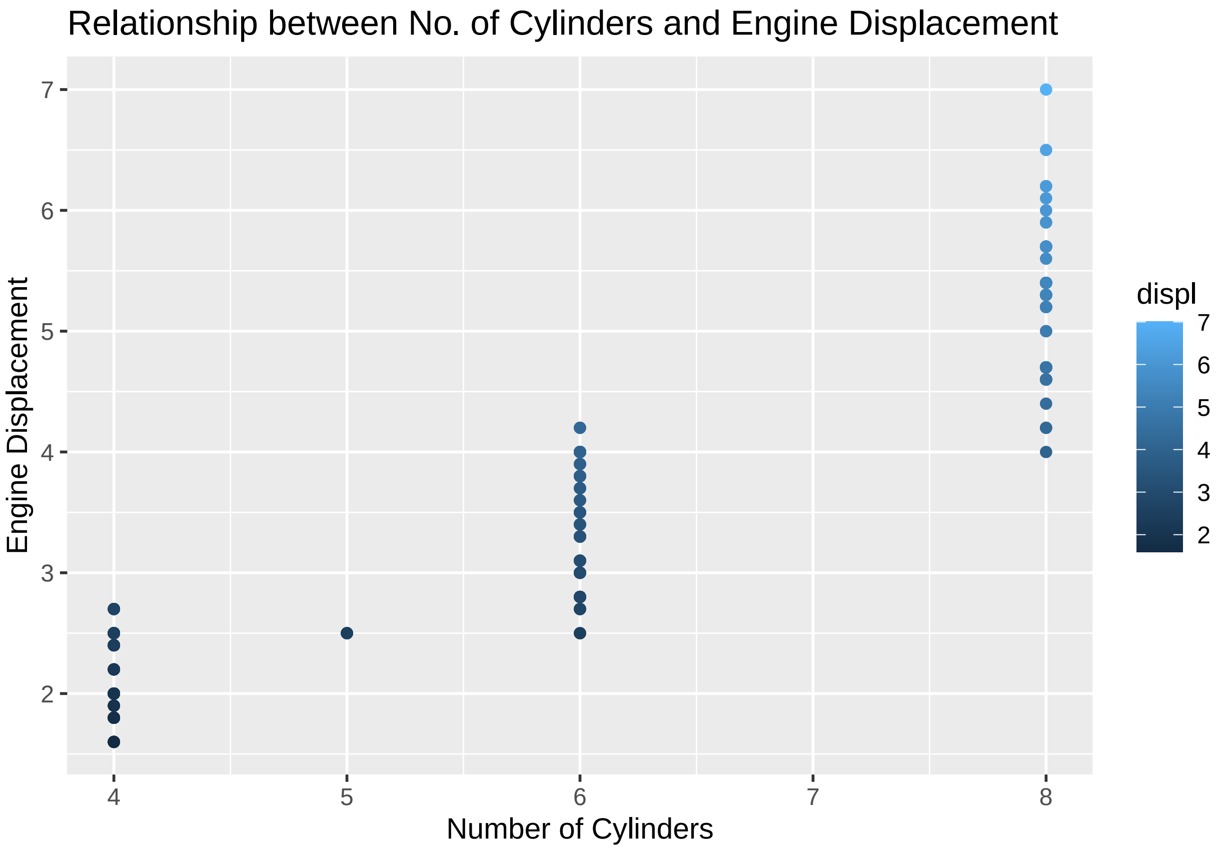
```r
# --- 4. Number of Cars per Model --- [cite: 26]
# Group model and get number of cars using pipe [cite: 26]
model_counts <- mpg %>%
  group_by(model) %>%
  summarise(count = n()) #[cite: 26]

# a. Plot top 20 observations [cite: 28]
ggplot(head(model_counts, 20), aes(x = reorder(model, count), y = count, fill = model)) +
  geom_bar(stat = "identity") +
  labs(title = "Top 20 Models by Count", x = "Model", y = "Number of Cars") #[cite: 28, 29]
```



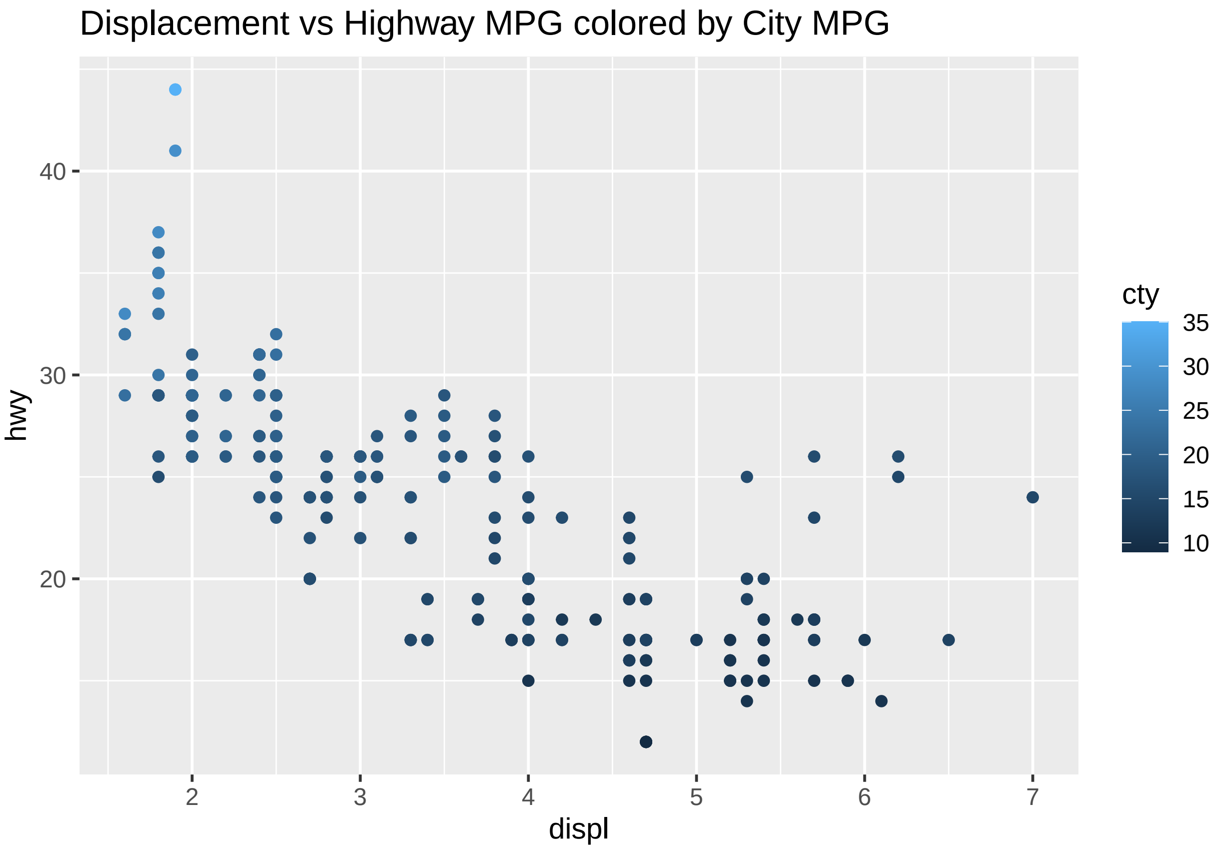
```
# b. Recreate Figure 1: Car Model [cite: 30, 41]
ggplot(model_counts, aes(x = reorder(model, count), y = count, fill = model)) +
  geom_bar(stat = "identity") +
  coord_flip() + #[cite: 30]
  labs(title = "Number of Cars per Model", x = "Model", y = "Number of cars") + #[cite: 39, 40]
  theme(legend.position = "none") #[cite: 30]
```



```
# --- 5. Cylinders vs Displacement --- [cite: 31]
ggplot(mpg, aes(x = cyl, y = displ, color = displ)) + #[cite: 31]
  geom_point() +
  labs(title = "Relationship between No. of Cylinders and Engine Displacement", #[cite: 32]
       x = "Number of Cylinders", y = "Engine Displacement")
```

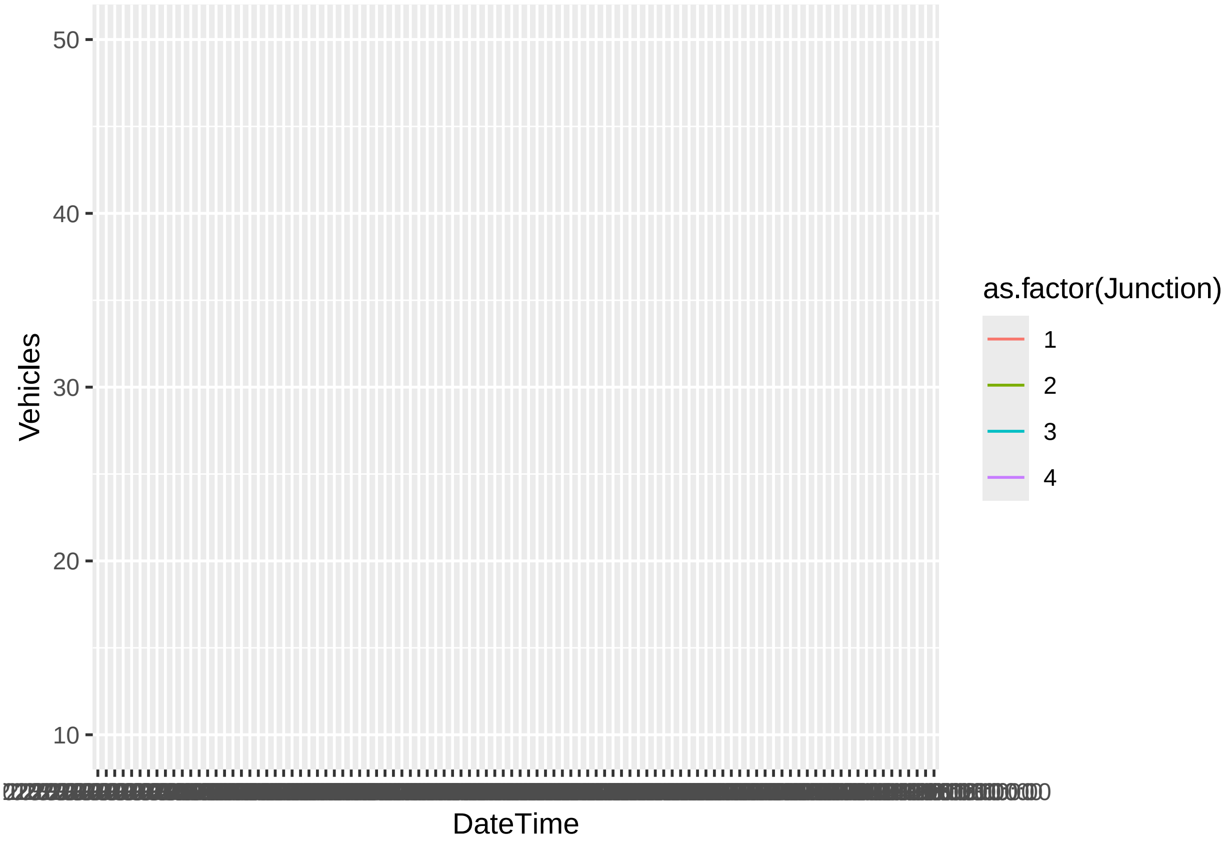


```
# Description: There is a positive correlation; engines with more cylinders tend to have higher displac

# --- 6. Displacement vs Highway MPG --- [cite: 34]
# Mapping with continuous variable 'cty' from #1-c [cite: 35]
ggplot(mpg, aes(x = displ, y = hwy, color = cty)) +
  geom_point() +
  labs(title = "Displacement vs Highway MPG colored by City MPG") #[cite: 34, 35]
```



```r
# --- 6. Traffic Dataset --- [cite: 42]
# Creating dummy traffic.csv for the environment
traffic_dummy <- data.frame(
  DateTime = seq(as.POSIXct("2023-01-01"), by="hour", length.out=100),
  Junction = rep(1:4, each=25),
  Vehicles = sample(10:50, 100, replace=TRUE)
)
write.csv(traffic_dummy, "traffic.csv", row.names = FALSE)

traffic_data <- read.csv("traffic.csv") # [cite: 43]
# b. Subset junctions [cite: 44]
junc_subset <- traffic_data %>% filter(Junction == 1)
# c. Plot each junction [cite: 45]
ggplot(traffic_data, aes(x = DateTime, y = Vehicles, color = as.factor(Junction))) +
  geom_line()
```

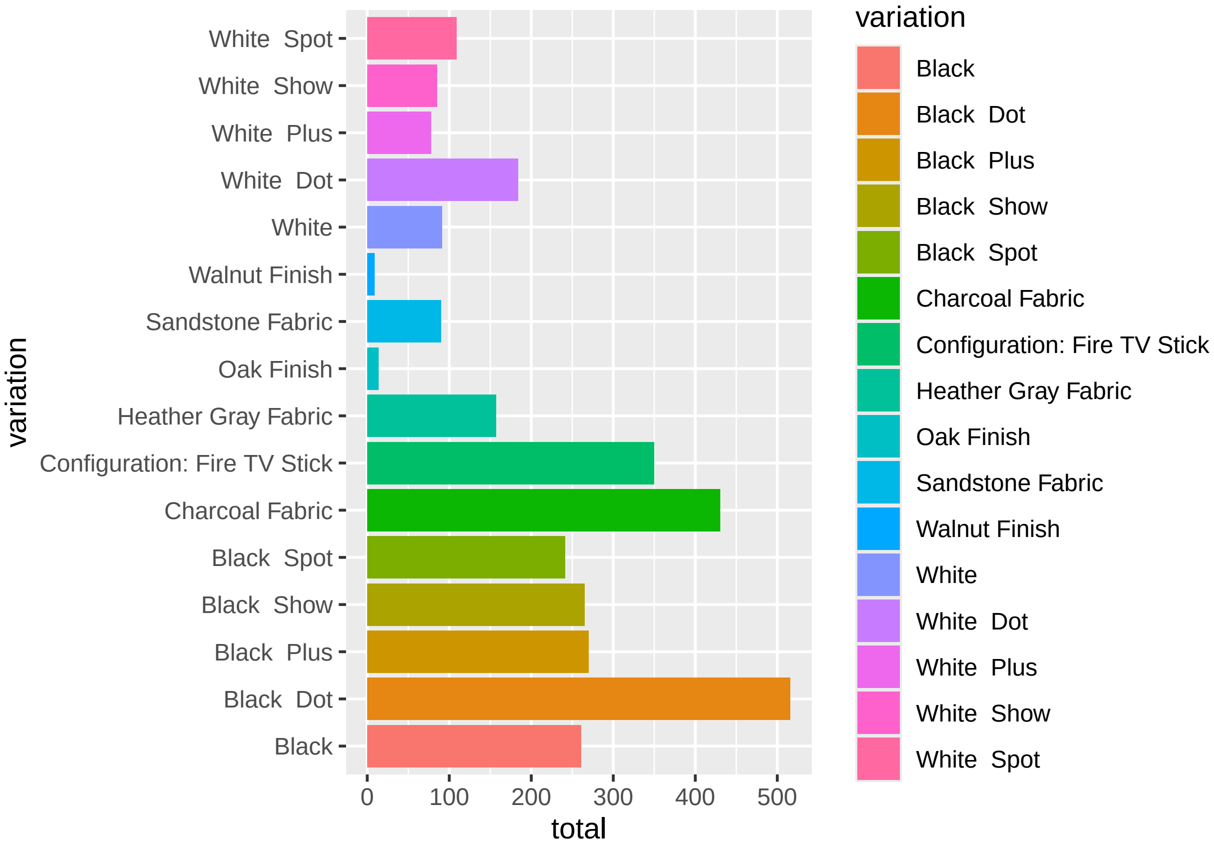
```
## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
```



```r
# --- 7. Alexa Dataset --- [cite: 46]
# a. Observations and columns [cite: 47]
alexa_data <- read_excel("alexa_file.xlsx") #[cite: 46]
dim(alexa_data) #[cite: 47]
```

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

```r
# b. Group variations using dplyr [cite: 48]
alexa_totals <- alexa_data %>% group_by(variation) %>% summarise(total = n()) #[cite: 48]

# c. Plot variations [cite: 49]
ggplot(alexa_totals, aes(x = variation, y = total, fill = variation)) +
  geom_bar(stat = "identity") +
  coord_flip() #[cite: 49, 50]
```



```
# d. Date and number of verified reviews [cite: 52]
ggplot(alexa_data, aes(x = date, y = verified_reviews)) +
  geom_line() +
  labs(title = "Verified Reviews Over Time") #[cite: 52, 53]
```

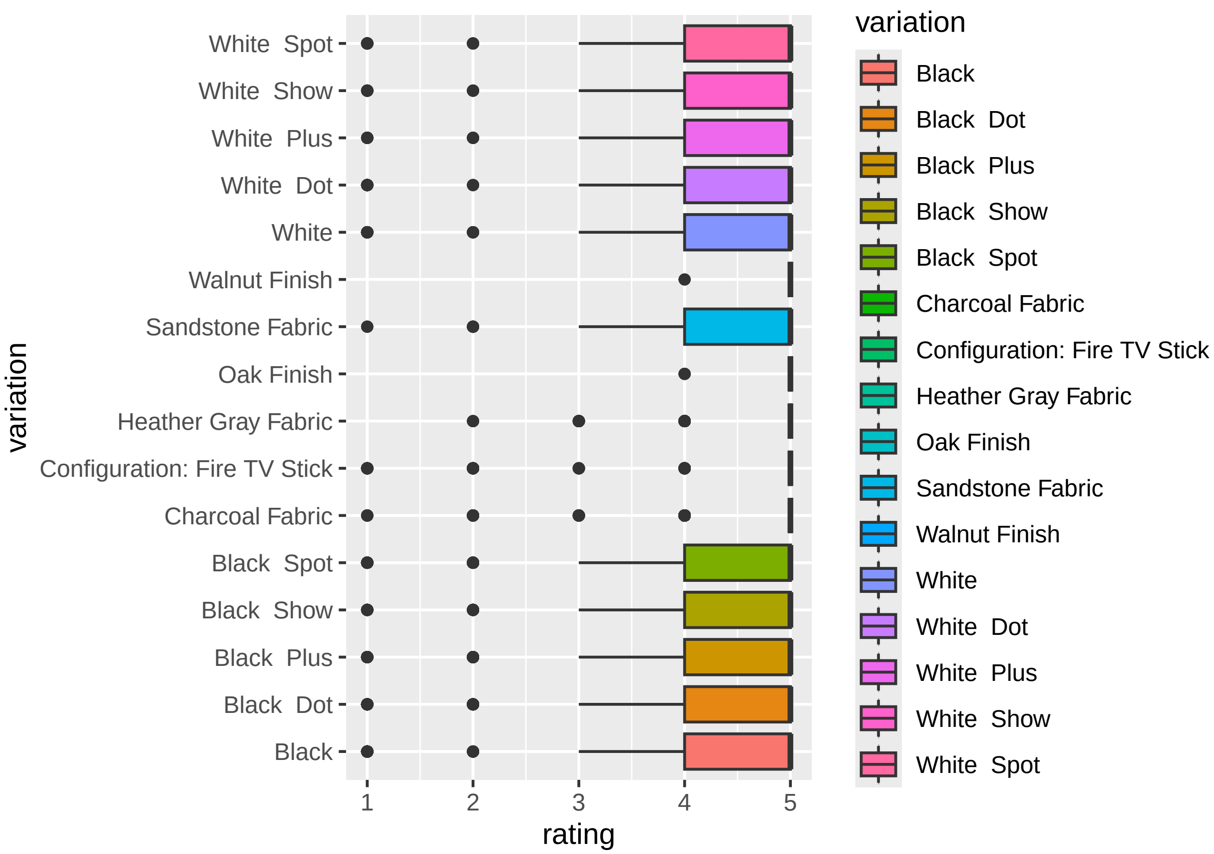
are some serious flaws, particularly if you are the last one to bed or the first to wake. It doesn't seem like the engineer

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

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

npting to add this bulb to my Alexa Echo Plus. Everything I tried, ended in a Discovery Failed message. I tried to set a
igh 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

```
# e. Variations and ratings [cite: 54]
ggplot(alexa_data, aes(x = variation, y = rating, fill = variation)) +
  geom_boxplot() +
  coord_flip() #[cite: 55]
```


variation
White Spot
White Show
White Plus
White Dot
White
Walnut Finish
Sandstone Fabric
Oak Finish
Heather Gray Fabric
Configuration: Fire TV Stick
Charcoal Fabric
Black Spot
Black Show
Black Plus
Black Dot
Black
1
2
3
4
5
rating
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