RWorksheet_Pineda-4a

Epiphany Louise O. Pineda

2025-12-05

1.) Use the dataset mpg

A data frame with 234 rows and 11 variables: - 'describe{ - 'item{manufacturer}{manufacturer name} - 'item{model}{model name} - 'item{displ}{engine displacement, in litres} - 'item{year}{year of manufacture} - 'item{cyl}{number of cylinders} - 'item{trans}{type of transmission} - 'item{drv}{the type of drive train, where f = front-wheel drive, r = rear wheel drive, 4 = 4wd} - 'item{cty}{city miles per gallon} - 'item{hwy}{highway miles per gallon} - 'item{fl}{fuel type} - 'item{class}{"type" of car} - } -"mpg"

a. Show your solutions on how to import a csv file into the environment.

```
library(dplyr)
library(ggplot2)
data(mpg)

write.csv(mpg, "mpg.csv", row.names = FALSE)

mpgdata <- read.csv("mpg.csv", header = TRUE, stringsAsFactors = FALSE)
str(mpgdata)
```

b. Which variables from mpg dataset are categorical?

These are the categorical variables in the mpg dataset: manufacturer, model, trans, drv, fl, class, year

c. Which are continuous variables?

These are the continuous variables: displ, cty, hwy

2.) Which manufacturer has the most models in this data set? Which model has the most variations? Show your answer.

```
modelvars <- mpgdata %>%
  group_by(model) %>%
  summarise(totalvars = n()) %>%
  arrange(desc(totalvars))

modelvars
```

a. Group the manufacturers and find the unique models. Show your codes and result.

```
manumodels <- mpgdata %>%
  group_by(manufacturer) %>%
  summarise(totalmodels = n_distinct(model)) %>%
  arrange(desc(totalmodels))

manumodels
```

b. Graph the result by using plot() and ggplot(). Write the codes and its result.

1. Using base R plot()

```
plot(as.factor(manumodels$manufacturer),
     manumodels$totalmodels,
     las = 2,
     main = "Number of Unique Models per Manufacturer",
     xlab = "Manufacturer",
     ylab = "Number of Models")
```

2. Using ggplot()

```
ggplot(manumodels, aes(x = reorder(manufacturer, -totalmodels), y = totalmodels)) +
  geom_bar(stat = "identity") +
  theme_minimal() +
  labs(title = "Unique Models per Manufacturer",
       x = "Manufacturer",
       y = "Number of Unique Models") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

2.) Same dataset will be used. You are going to show the relationship of the model and the manufacturer.

a. What does ggplot(mpg, aes(model, manufacturer)) + geom_point() show?

Each point represents a row (a car entry) in the dataset. x-axis: car model y-axis: manufacturer If a manufacturer has multiple rows of the same model, points stack vertically.

b. For you, is it useful? If not, how could you modify the data to make it more informative?

Not very useful in its current form because of overlapping points.

Better alternatives:

```
ggplot(mpgdata, aes(model, manufacturer)) +
  geom_count() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Count of Each Model per Manufacturer",
       x = "Model",
       y = "Manufacturer")
```

3.) Plot the model and the year using ggplot(). Use only the top 20 observations. Write the codes and its results.

```
top20data <- mpgdata[1:20, ]

ggplot(top20data, aes(x = model, y = factor(year))) +
  geom_point(color = "blue", size = 3) +
  labs(title = "Model vs Year (Top 20 Observations)",
       x = "Model",
       y = "Year") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

4.) Using the pipe (%>%), group the model and get the number of cars per model. Show codes and its result

```
modelcount <- mpgdata %>%
  group_by(model) %>%
  summarise(carnum = n()) %>%
  arrange(desc(carnum))
```

```
modelcount
```

a. Plot using geom_bar() using the top 20 observations only. The graphs should have a title, labels and colors. Show code and results.

```
top20models <- modelcount[1:20, ]

ggplot(top20models, aes(x = reorder(model, -carnum), y = carnum, fill = model)) +
  geom_bar(stat = "identity") +
  labs(title = "Top 20 Models by Number of Cars",
       x = "Model",
       y = "Number of Cars") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_brewer(palette = "Set3")
```

b. Plot using the geom_bar() + coord_flip() just like what is shown below. Show codes and its result.

```
ggplot(top20models, aes(x = reorder(model, carnum), y = carnum, fill = model)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(title = "Top 20 Models by Number of Cars",
       x = "Model",
       y = "Number of Cars") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
```

5.) Plot the relationship between cyl - number of cylinders and displ - engine displacement using geom_point with aesthetic color - = engine displacement. Title should be "Relationship between No. of Cylinders and Engine Displacement".

a. How would you describe its relationship? Show the codes and its result.

Positive relationship: As the number of cylinders increases, engine displacement also tends to increase. Cars with 4 cylinders generally have smaller engines (lower displ). Cars with 6 or 8 cylinders have larger engines (higher displ). The points may form distinct clusters around 4, 6, and 8 cylinders.

```
ggplot(mpgdata, aes(x = cyl, y = displ, color = displ)) +
  geom_point(size = 3) +
  labs(title = "Relationship between No. of Cylinders and Engine Displacement",
       x = "Number of Cylinders",
       y = "Engine Displacement (litres)",
       color = "Displacement") +
  theme_minimal()
```

6.)Plot the relationship between displ (engine displacement) and hwy(highway miles per gallon). Mapped it with a continuous variable you have identified in #1-c. What is its result? Why it produced such output?

```
ggplot(mpgdata, aes(x = displ, y = hwy, color = cty)) +
  geom_point(size = 3) +
  labs(title = "Relationship between Engine Displacement and Highway MPG",
       x = "Engine Displacement (litres)",
       y = "Highway Miles per Gallon",
       color = "City MPG") +
  theme_minimal()
```

6. Import the traffic.csv onto your R environment.

```
library(readr)

traffic <- data.frame(
  Date = c("2025-12-01","2025-12-01","2025-12-01","2025-12-01",
           "2025-12-01","2025-12-01","2025-12-01","2025-12-01",
           "2025-12-02","2025-12-02","2025-12-02","2025-12-02"),
  Time = c("07:00","08:00","09:00","10:00",
           "07:00","08:00","09:00","10:00",
           "07:00","08:00","09:00","10:00"),
  Junction1 = c(34,40,38,42,37,39,45,43,41,47,46,44),
  Junction2 = c(21,25,23,27,22,24,28,26,25,29,30,27)
)

write.csv(traffic, "traffic.csv", row.names = FALSE)
trafficdata <- read.csv("traffic.csv")
```

a. How many numbers of observation does it have? What are the variables of the traffic dataset? Show your answer.

Observation [1] 12 4 [1] “Date” “Time” “Junction1” “Junction2”

b. Subset the traffic dataset into junctions. What is the R codes and its output?

```
junctiondata <- trafficdata[, c("Junction1", "Junction2")]

junctiondata
```

c. Plot each junction in a using geom_line(). Show your solution and output.

```
library(tidyr)

trafficlong <- trafficdata %>%
  pivot_longer(cols = starts_with("Junction"),
               names_to = "Junction",
               values_to = "TrafficFlow")

ggplot(trafficlong, aes(x = Time, y = TrafficFlow, color = Junction)) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  labs(title = "Traffic Flow at Each Junction",
       x = "Time",
       y = "Traffic Volume") +
  theme_minimal()
```

7.) From alexa_file.xlsx, import it to your environment

```
library(readxl)

alexa <- read_excel("RWorksheets#4 - #4c/RWorksheet#4b/alexa_file.xlsx")
```

a. How many observations does alexa_file have? What about the number of columns? Show your solution and answer.

```
numobs <- nrow(alexafile)
```

```
numcols <- ncol(alexafile)

numobs
numcols
```

b. Group the variations and get the total of each variations. Use dplyr package. Show solution and answer.

```
varsummary <- alexafile %>%
  group_by(variation) %>%
  summarise(total = n())

varsummary
```

c. Plot the variations using the ggplot() function. What did you observe? Complete the details of the graph. Show solution and answer.

```
ggplot(varsummary, aes(x = variation, y = total, fill = variation)) +
  geom_bar(stat = "identity") +
  labs(
    title = "Total Count of Each Alexa Variation",
    x = "Variation",
    y = "Total Count"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

d. Plot a geom_line() with the date and the number of verified reviews. Complete the details of the graphs. Show your answer and solution.

```
alexafile$date <- as.Date(alexafile$date, format="%Y-%m-%d")
alexafile$verifiedreviews <- as.numeric(gsub(",", "", alexafile$verified_reviews))

verifsummary <- alexafile %>%
  group_by(date) %>%
  summarise(totalverified = sum(verifiedreviews, na.rm = TRUE))

ggplot(verifsummary, aes(x = date, y = totalverified)) +
  geom_line(color = "blue", linewidth = 1) +
  labs(
    title = "Number of Verified Reviews Over Time",
    x = "Date",
    y = "Total Verified Reviews"
  ) +
  theme_minimal()
```

e. Get the relationship of variations and ratings. Which variations got the highest rating? Plot a graph to show its relationship. Show your solution and answer.

```
ratingsummary <- alexafile %>%
  group_by(variation) %>%
  summarise(avgrating = mean(rating, na.rm = TRUE)) %>%
  arrange(desc(avgrating))

ratingsummary
```

```r
ggplot(ratingsummary, aes(x = variation, y = avgrating, fill = variation)) +
  geom_bar(stat = "identity") +
  labs(
    title = "Average Rating by Alexa Variation",
    x = "Variation",
    y = "Average Rating"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```