RWorksheet#4C_Sapan

Leorenze Marc Sapan

2025-12-18

```r
# 1.)

library(dplyr)
library(ggplot2)

data(mpg)

# 1a.)
write.csv(mpg, "mpg.csv", row.names = FALSE)

# Import CSV file
mpg_data <- read.csv("mpg.csv", header = TRUE, stringsAsFactors = FALSE)
str(mpg_data)


# 1b.)

# Categorical variables:
# manufacturer, model, trans, drv, fl, class, year


# 1c.)

# Continuous variables:
# displ, cty, hwy



# 2.)

model_variations <- mpg_data %>%
  group_by(model) %>%
  summarise(total_variations = n()) %>%
  arrange(desc(total_variations))

model_variations


# 2a.)

manufacturer_models <- mpg_data %>%
  group_by(manufacturer) %>%
```

```r
  summarise(total_models = n_distinct(model)) %>%
  arrange(desc(total_models))

manufacturer_models


# 2b.)

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

ggplot(manufacturer_models,
       aes(x = reorder(manufacturer, -total_models),
           y = total_models)) +
  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.)


# 2a.)
#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.


# 2b.)
#Not very useful in its current form because of overlapping points.

#Better alternatives:
ggplot(mpg_data, 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.)

top20_observations <- mpg_data[1:20, ]
```

```r
ggplot(top20_observations,
       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.)

model_car_count <- mpg_data %>%
  group_by(model) %>%
  summarise(number_of_cars = n()) %>%
  arrange(desc(number_of_cars))

model_car_count


# 4a.)

top20_models <- model_car_count[1:20, ]

ggplot(top20_models,
       aes(x = reorder(model, -number_of_cars),
           y = number_of_cars,
           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")


# 4b.) HORIZONTAL BAR PLOT

ggplot(top20_models,
       aes(x = reorder(model, number_of_cars),
           y = number_of_cars,
           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")
```

```r
# 5.)

# 5a.)
#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 increas
#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(mpg_data,
       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.)

ggplot(mpg_data,
       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.)

library(readr)
library(tidyr)

traffic_data <- 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_data, "traffic.csv", row.names = FALSE)
traffic_data <- read.csv("traffic.csv")


# 6a.)
```

```r
#Observation
#[1] 12 4
#[1] "Date" "Time" "Junction1" "Junction2"


# 6b.)

junction_data <- traffic_data[, c("Junction1", "Junction2")]
junction_data


# 6c.)
traffic_long <- traffic_data %>%
  pivot_longer(cols = starts_with("Junction"),
               names_to = "Junction",
               values_to = "Traffic_Volume")

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




# 7.) ALEXA DATASET

library(readxl)

alexa_data <- read_excel("alexa_file.xlsx")


# 7a.)
num_observations <- nrow(alexa_data)
num_columns <- ncol(alexa_data)

num_observations
num_columns


# 7b.)

variation_summary <- alexa_data %>%
  group_by(variation) %>%
  summarise(total_count = n())

variation_summary
```

```r
# 7c.)

ggplot(variation_summary,
       aes(x = variation, y = total_count, 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))


# 7d.)

alexa_data$date <- as.Date(alexa_data$date)
alexa_data$verified_reviews_numeric <-
  as.numeric(gsub(",", "", alexa_data$verified_reviews))

verified_summary <- alexa_data %>%
  group_by(date) %>%
  summarise(total_verified_reviews =
              sum(verified_reviews_numeric, na.rm = TRUE))

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


# 7e.) VARIATION VS RATING

rating_summary <- alexa_data %>%
  group_by(variation) %>%
  summarise(average_rating = mean(rating, na.rm = TRUE)) %>%
  arrange(desc(average_rating))

rating_summary

ggplot(rating_summary,
       aes(x = variation, y = average_rating, 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))
```