R Lab Assignment

Diksha

2024-09-18

R Markdown

```
# Question 1
# Variables x, y, z are defined

x <- 5
y <- 4*x
z <- y / 2

# Computing current result:
result <- z + y * x - z + x
# Print current result
print(paste0("Current result: ", result))
```

```
## [1] "Current result: 105"
```

```
# Solution : Add parentheses here:
result <- (z + y) * (x - (z + x))
# Print expected result
print(paste0("Expected result: ", result))
```

```
## [1] "Expected result: -300"
```

```
# Question 2

# Load library ggplot2
library(ggplot2)

# Load data mpg
data(mpg)

# Fuel efficiency column
fuel_eff <- mpg$cty

# Unique values of fuel efficiency column
fuel_eff_unq <- unique(fuel_eff)

# Sort in ascending the unique values of fuel efficiency column
cty_ascending_order <- sort(fuel_eff_unq, decreasing = FALSE)

# Third lowest fuel efficiency
third_lowest_cty <- cty_ascending_order[3]

# Third highest fuel efficiency
```

```r
# Find the index of the value first
index_last_3 = length(cty_ascending_order) - 3

# Then extract the value
third_highest_cty <- cty_ascending_order[index_last_3]

# Extract the rows with third highest and third lowest fuel efficiency

# Only cars with third lowest fuel efficiency
rows_third_lowest_cty <- mpg[mpg$cty == third_lowest_cty,]

# Only cars with third highest fuel efficiency
rows_third_highest_cty <- mpg[mpg$cty == third_highest_cty,]

# Both cars with third highest and third lowest fuel efficiency
rows_third_lowest_highest_cty <- mpg[mpg$cty == third_lowest_cty | mpg$cty == third_highest_cty,]


# Cars without third lowest fuel efficiency
rows_third_lowest_cty_wo <- mpg[!(mpg$cty == third_lowest_cty),]

# Cars without third highest fuel efficiency
rows_third_highest_cty_wo <- mpg[!(mpg$cty == third_highest_cty),]

# Without both cars with third highest and third lowest fuel efficiency
rows_third_lowest_highest_cty_wo <- mpg[!(mpg$cty == third_lowest_cty | mpg$cty == third_highest_cty),]

# Solution : Save the resulting dataset in mpg_new
mpg_new <- rows_third_lowest_highest_cty_wo <- mpg[!(mpg$cty == third_lowest_cty | mpg$cty == third_high
```

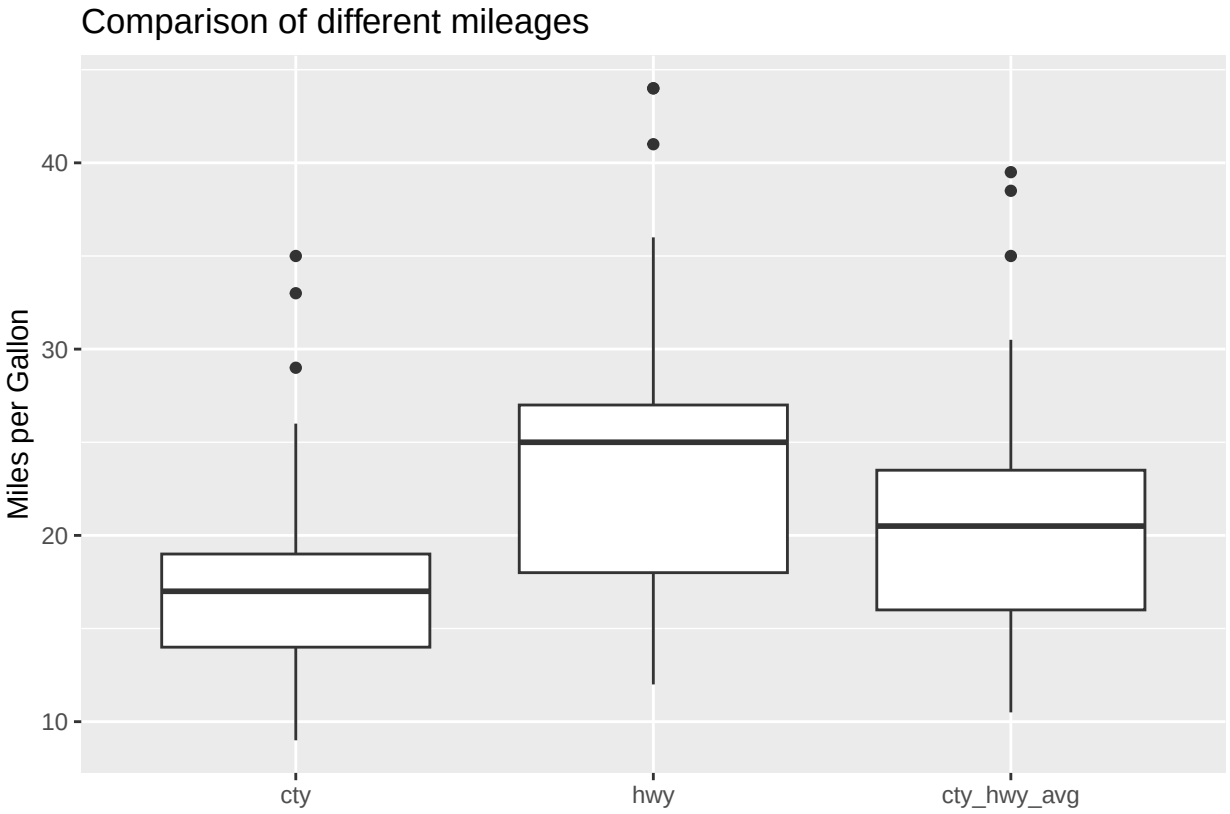
```r
# Question 3

# Average mileage which is the average value of city and highway
mpg_new$cty_hwy_avg <- (mpg_new$hwy + mpg_new$cty)/2

# Package required to reshape the data
library(reshape2)

# Melt the data as required for boxplot
mileage_box <- melt(mpg_new, measure.vars = c("cty", "hwy", "cty_hwy_avg"),
                    variable.name = "Type", value.name = "Mileage")

# Boxplots comparing city, highway, and average mileage
ggplot(mileage_box, aes(x = Type, y = Mileage)) +
  geom_boxplot() +
  ggtitle("Comparison of different mileages") +
  ylab("Miles per Gallon") +
  xlab("")
```




```
# Question 4

# Creating average
avg_miles_manu <- function(manu)
  {
    manu_data <- mpg_new[mpg_new$manufacturer == manu, ]
    mean(manu_data$cty_hwy_avg)
  }

audi_avg_manu <- avg_miles_manu("audi")
print(paste("Average mileage of cars produced by manufacturer audi:", audi_avg_manu))
```

```
## [1] "Average mileage of cars produced by manufacturer audi: 22.0277777777778"
```

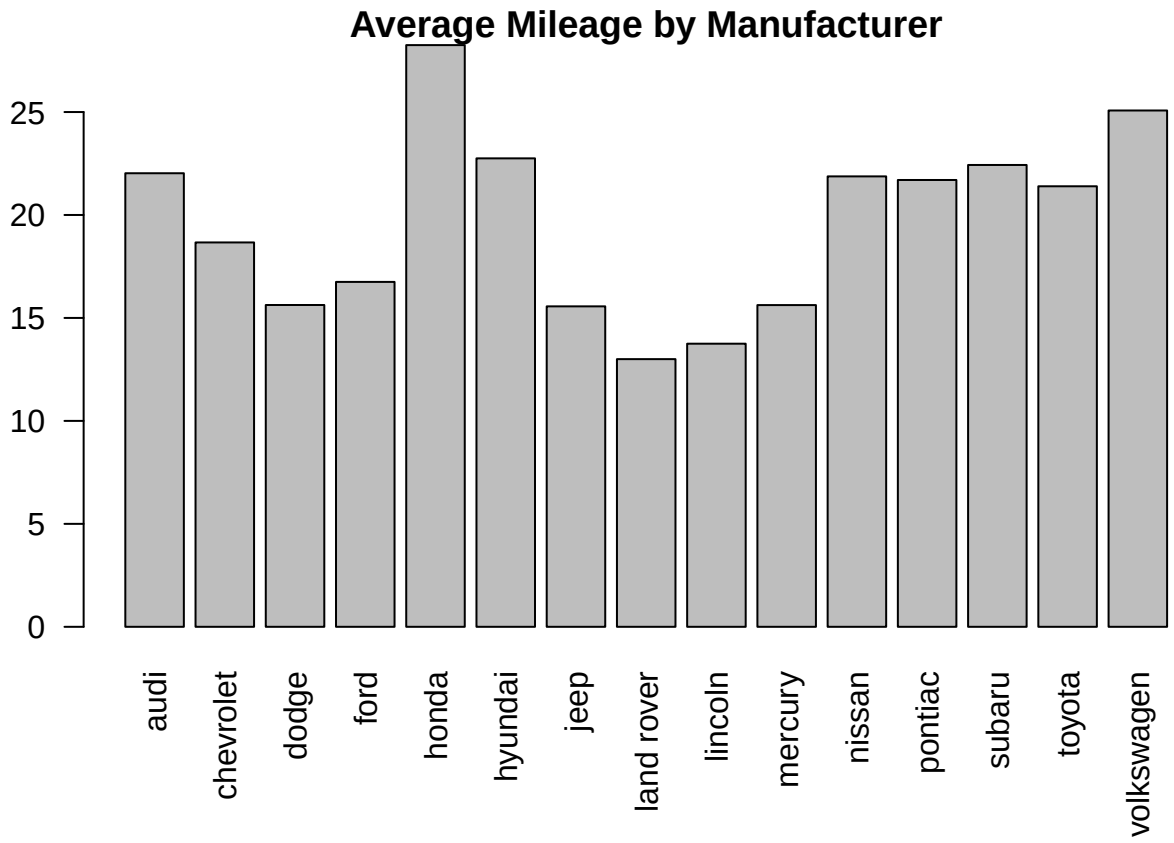
```
manus <- unique(mpg_new$manufacturer)

average_mileage_df <- data.frame(manufacturer = character(),
                         avg_mil = numeric(),
                         stringsAsFactors = FALSE)

# Looping over all the manufacturers and store the average mileage of each manufacturer
for (manu in manus) {
  avg_mileage <- avg_miles_manu(manu)
  average_mileage_df <- rbind(average_mileage_df,
                              data.frame(manufacturer = manu, avg_mil = avg_mileage))
}
```

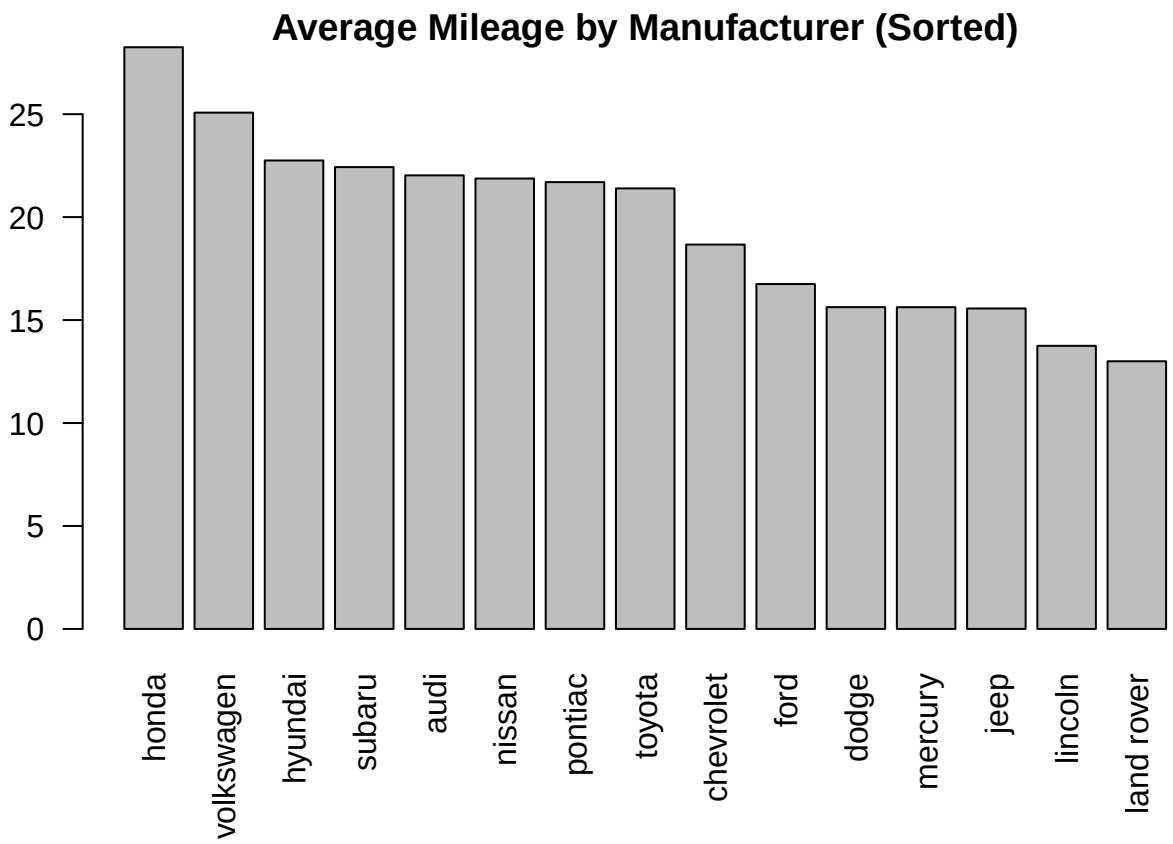
```r
# Adjust margins to accommodate long x-axis labels
par(mar = c(6, 2, 1, 1) + 0.1)

# Plot the results in a barplot
barplot(average_mileage_df$avg_mil,
        names.arg = average_mileage_df$manufacturer,
        las = 2,
        xlab = "",
        ylab = "Average Mileage",
        main = "Average Mileage by Manufacturer")
```




```r
# Sorting the manufacturers by average mileage in descending order
average_mileage_df_sorted <- average_mileage_df[order(-average_mileage_df$avg_mil), ]

# Plotting the sorted barplot
barplot(average_mileage_df_sorted$avg_mil,
        names.arg = average_mileage_df_sorted$manufacturer,
        las = 2,
        xlab = "",
        ylab = "Average Mileage",
        main = "Average Mileage by Manufacturer (Sorted)")
```


Average Mileage by Manufacturer (Sorted)
25
20
15
10
5
0
honda
volkswagen
hyundai
subaru
audi
nissan
pontiac
toyota
chevrolet
ford
dodge
mercury
jeep
lincoln
land rover