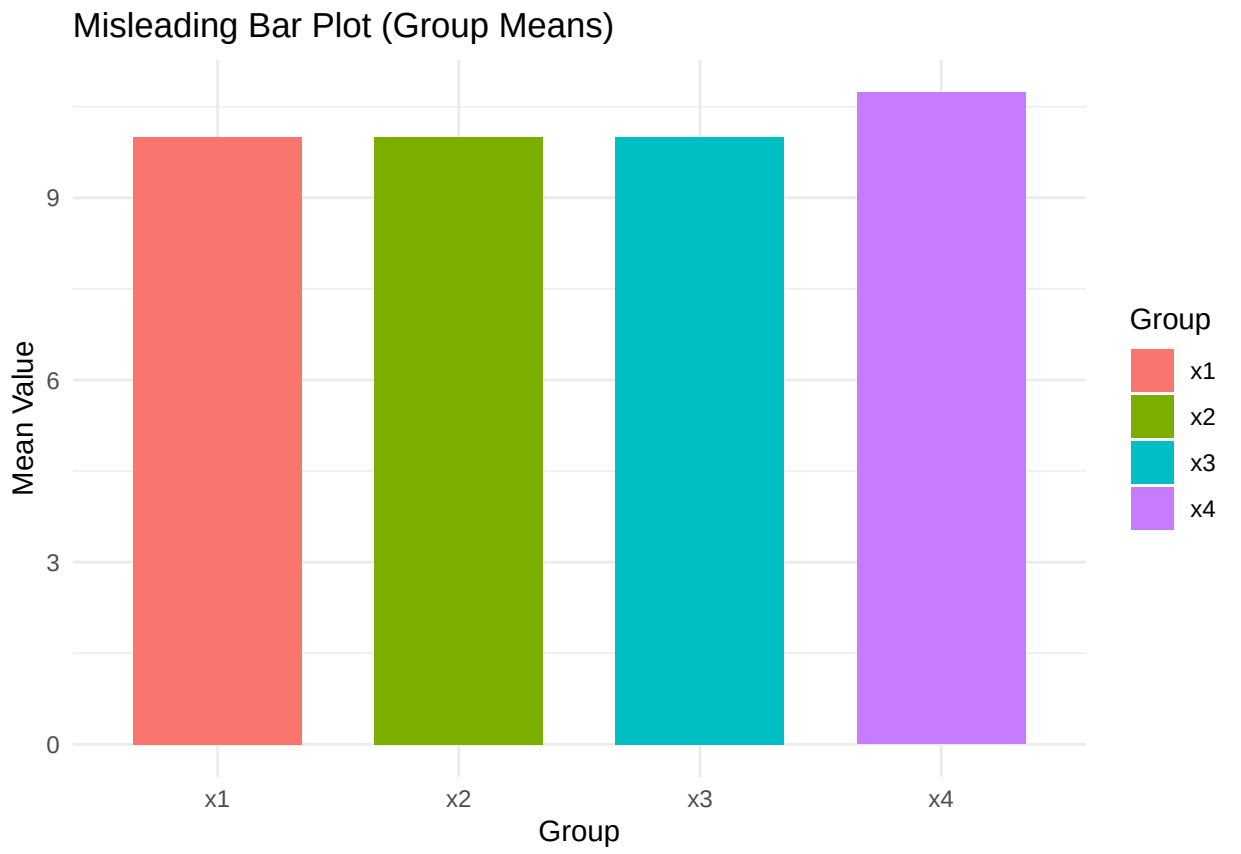
Data Visualisation Assignment

Shubham Raut(2024201019)

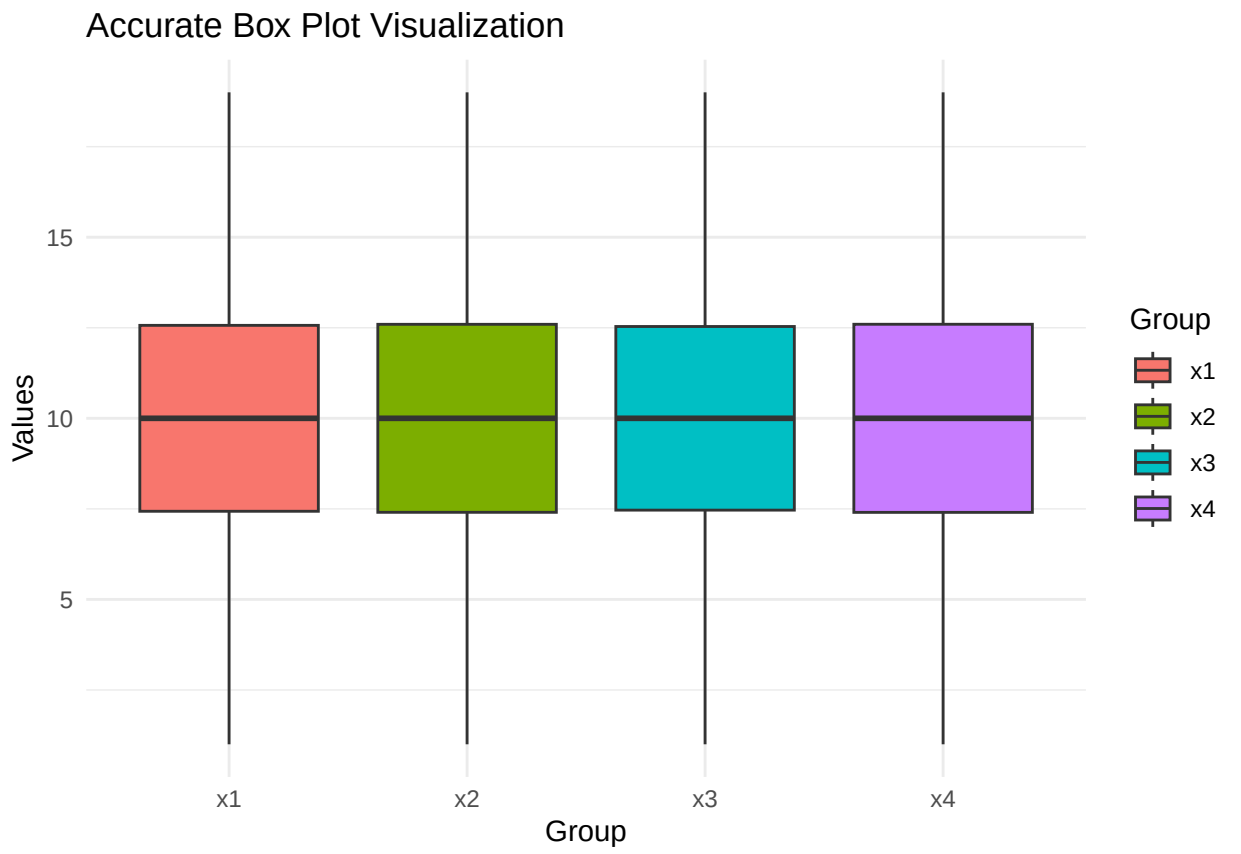
```r
# Question 1:Statistical Deception
# Load required libraries
library(readxl)
library(ggplot2)

# Load the Excel file and read the Statistical Deception sheet
file_path <- "~/Downloads/BRSM_Visualisation_Assignment.xlsx"
deception_data <- read_excel(file_path, sheet = "Statistical Deception")

# Reshape data for ggplot visualization
library(reshape2)
deception_data_melt <- melt(deception_data, variable.name = "Group", value.name = "Value")
```

```
## No id variables; using all as measure variables
```

```r
# Plot 1: Misleading Bar Plot
ggplot(deception_data_melt, aes(x = Group, y = Value, fill = Group)) +
  geom_bar(stat = "summary", fun = "mean", width = 0.7) +
  labs(title = "Misleading Bar Plot (Group Means)", x = "Group", y = "Mean Value") +
  theme_minimal()
```




```
# Plot 2: Better Box Plot
ggplot(deception_data_melt, aes(x = Group, y = Value, fill = Group)) +
  geom_boxplot() +
  labs(title = "Accurate Box Plot Visualization", x = "Group", y = "Values") +
  theme_minimal()
```

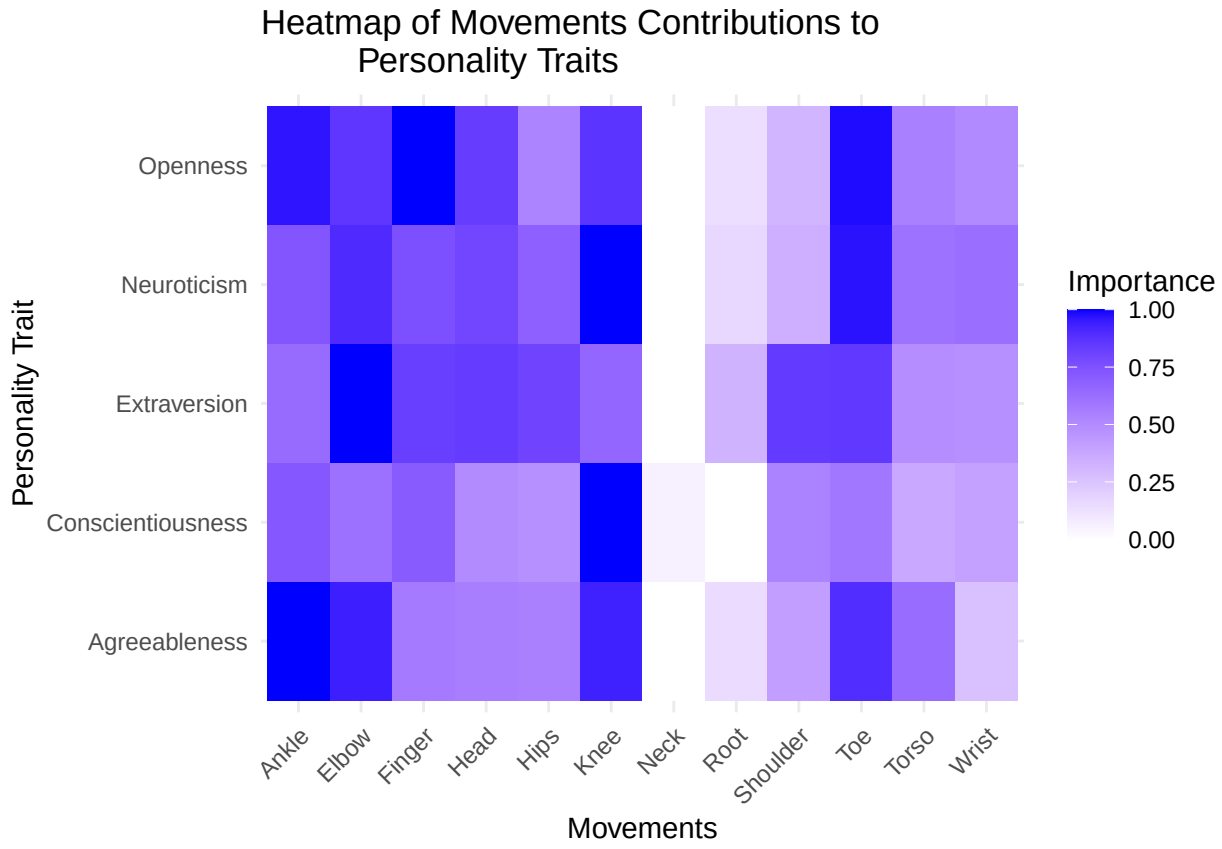



```
#The bar plot shows only the mean values, ignoring the variability and outliers in
# the data, which can give a false impression of group differences.
#The box plot(better) shows the distribution, spread, and outliers, providing a
# clearer and  more accurate representation of the data
# ------------------------------------------------------------------------------
# Ques. 2 (Personality and Motion)
# Load required libraries
library(readxl)
library(ggplot2)
library(tidyr)
```

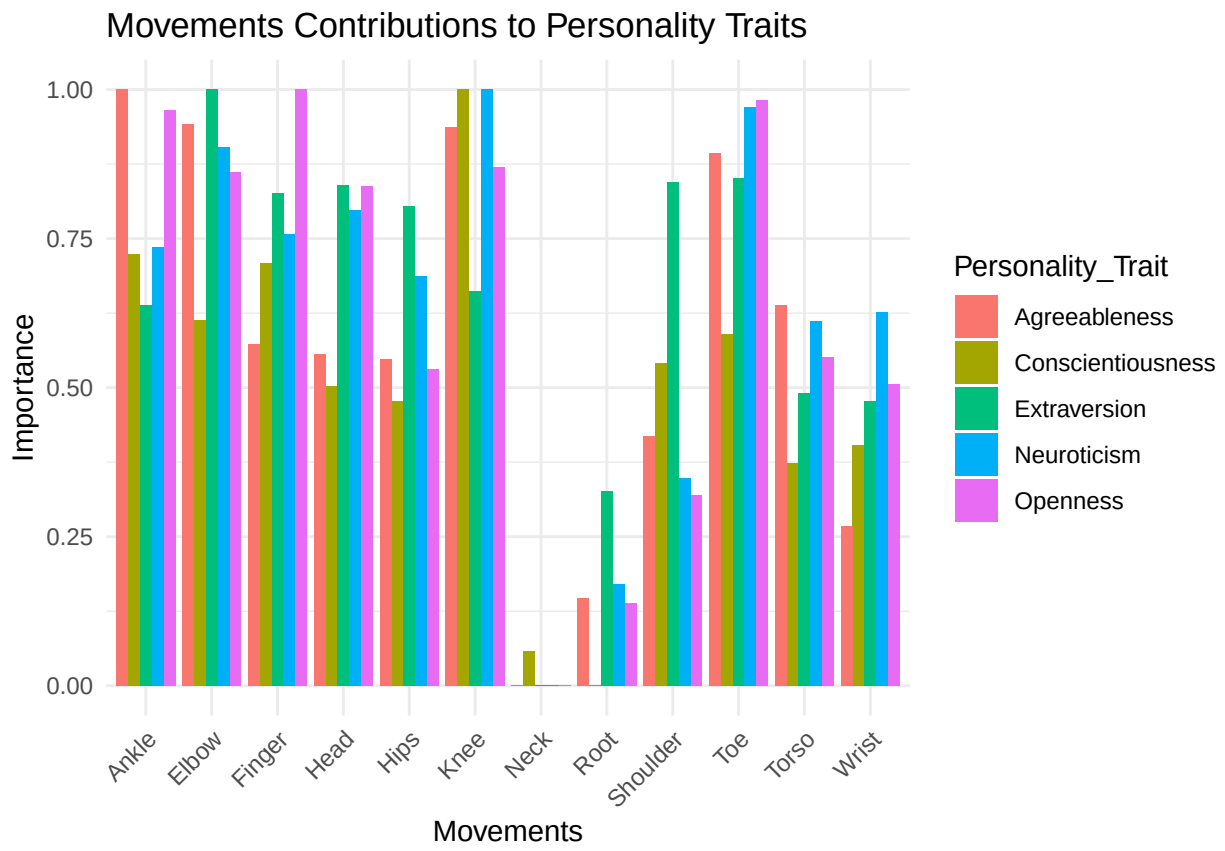
```
##
## Attaching package: 'tidyr'

## The following object is masked from 'package:reshape2':
##
##     smiths
```

```
# Load the specific sheet for Movement Personality Results
data <- read_excel("~/Downloads/BRSM_Visualisation_Assignment.xlsx", sheet =
 "Movement Personality Results")

# Reshape using pivot_longer
long_data <- data %>%
  pivot_longer(cols = -Movements, names_to = "Personality_Trait", values_to = "Importance")
```

```r
# Plot the heatmap
heatmap_plot <- ggplot(long_data, aes(x = Movements, y = Personality_Trait, fill = Importance)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "blue") +
  labs(title = "Heatmap of Movements Contributions to
          Personality Traits",
       x = "Movements",
       y = "Personality Trait",
       fill = "Importance") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

print(heatmap_plot)
```




```r
# Grouped bar plot as an alternative
bar_plot <- ggplot(long_data, aes(x = Movements, y = Importance, fill = Personality_Trait)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Movements Contributions to Personality Traits",
       x = "Movements",
       y = "Importance") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

print(bar_plot)
```

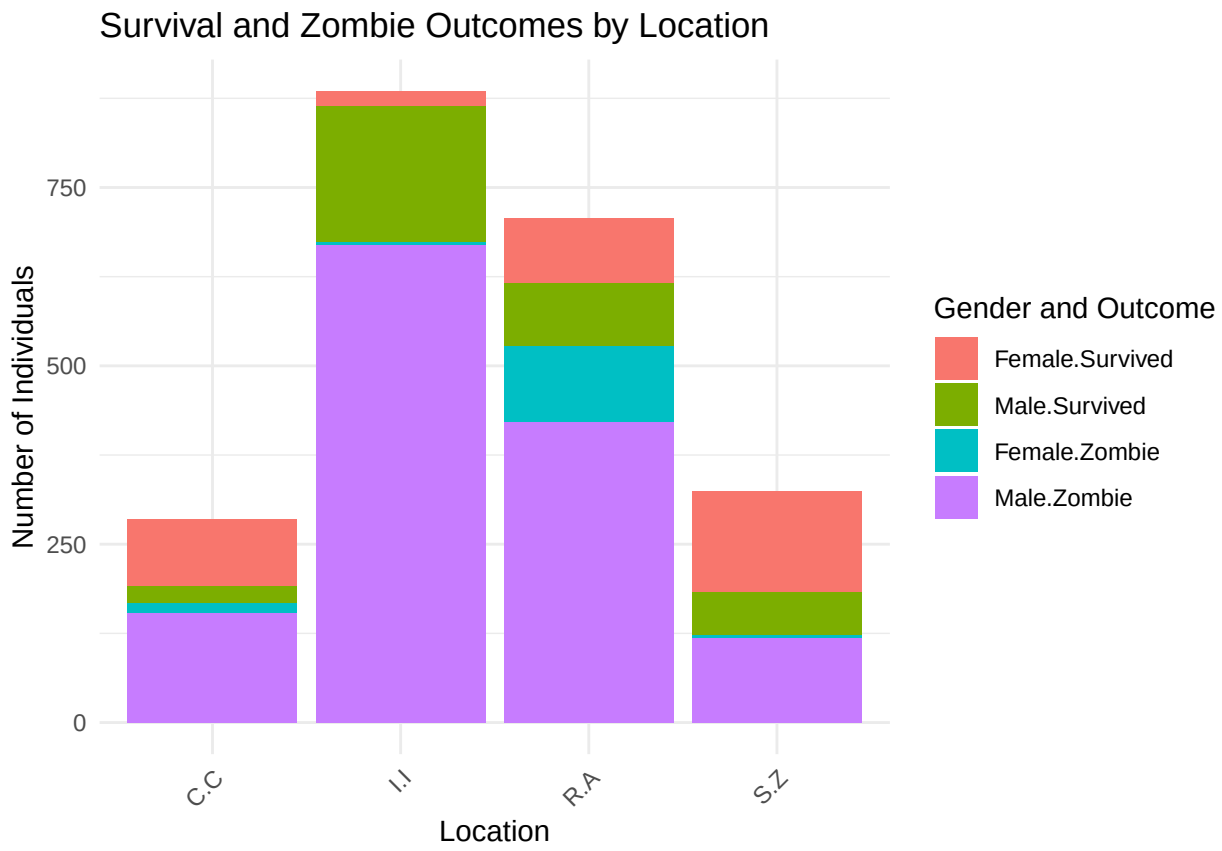



```
#Note: HeatMap is preferred over barplot as the heatmap visualization is better for
# spotting patterns and differences due to its use of a continuous color scale.

#-------------------------------------------------------------------------------

# Question 3: (Data Plotting Adventure)
# 3.1 : Subtask 1 (Last of Us)
# 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
```

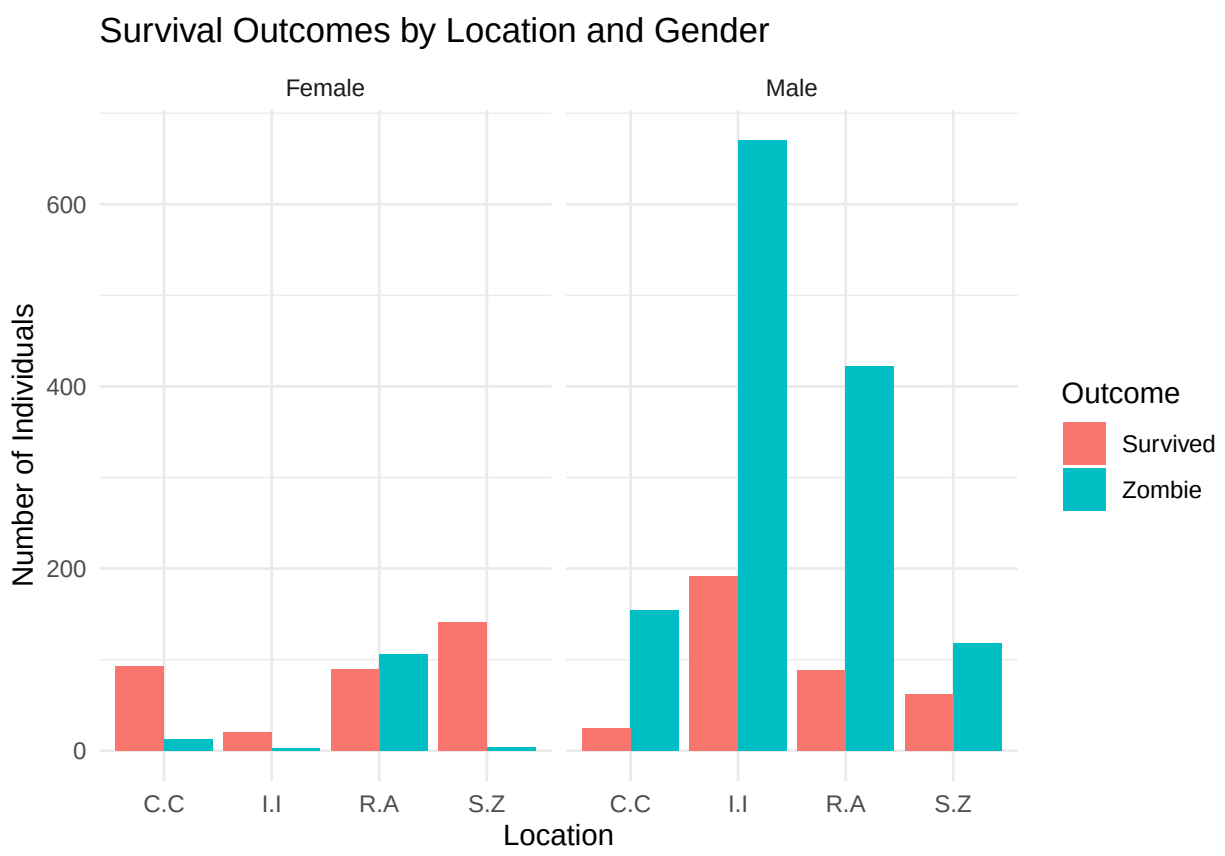
```
# Create the data as a data frame
```

```r
# Safe Zone(S.Z), Contaminated City(C.C), Rural Area(R.A),Isolated Island(I.I)
data <- data.frame(
  Location = rep(c("S.Z", "C.C", "R.A", "I.I"), each = 4),
  Gender = rep(c("Male", "Male", "Female", "Female"), times = 4),
  Outcome = rep(c("Zombie", "Survived"), times = 8),
  Count = c(118, 62, 4, 141,   # Safe Zone
            154, 25, 13, 93,  # Contaminated City
            422, 88, 106, 90, # Rural Area
            670, 192, 3, 20)  # Isolated Island
)

# Plot survival outcomes by location
ggplot(data, aes(x = Location, y = Count, fill = interaction(Gender, Outcome))) +
  geom_bar(stat = "identity", position = "stack") +
  labs(title = "Survival and Zombie Outcomes by Location",
       x = "Location",
       y = "Number of Individuals",
       fill = "Gender and Outcome") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



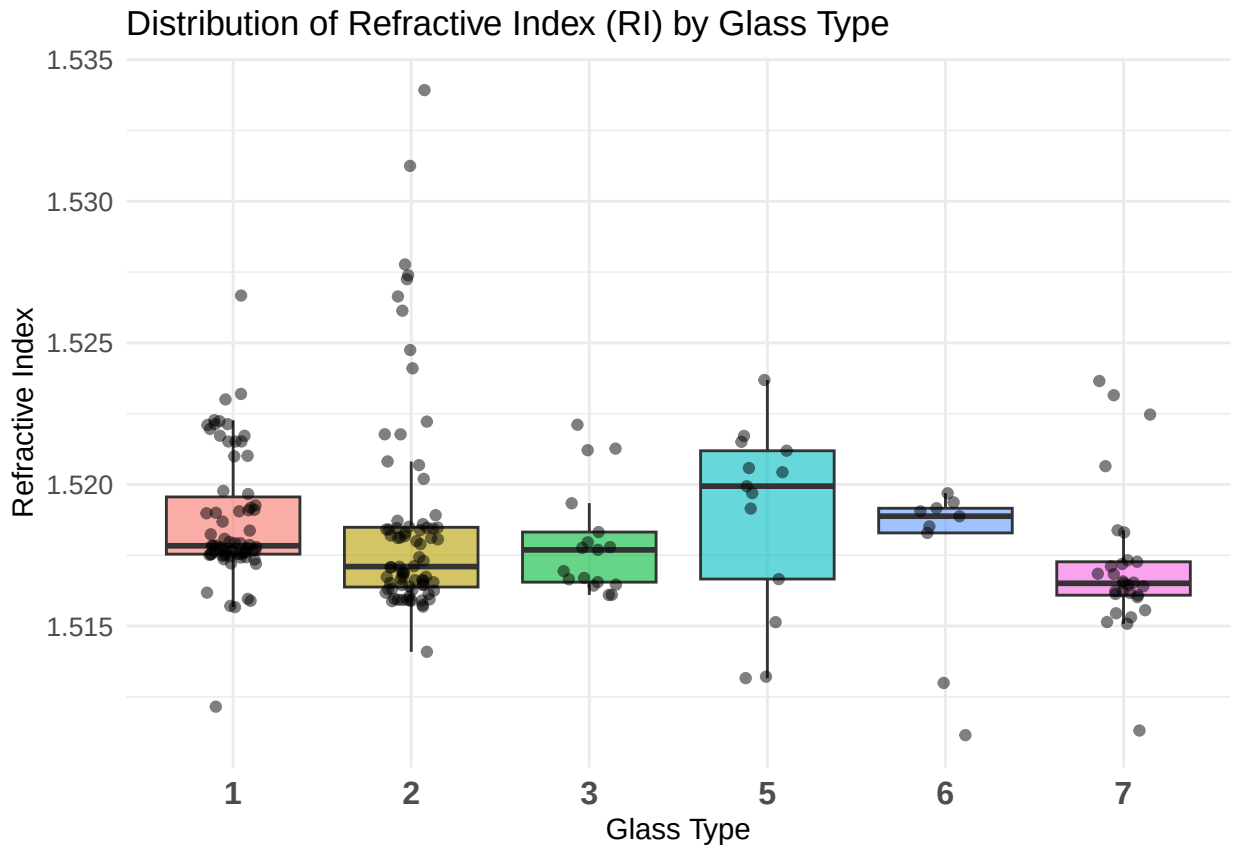

```r
# Grouped bar plot
ggplot(data, aes(x = Location, y = Count, fill = Outcome)) +
  geom_bar(stat = "identity", position = "dodge") +
  facet_wrap(~ Gender) +
```

```
  labs(title = "Survival Outcomes by Location and Gender",
       x = "Location",
       y = "Number of Individuals",
       fill = "Outcome") +
  theme_minimal()
```

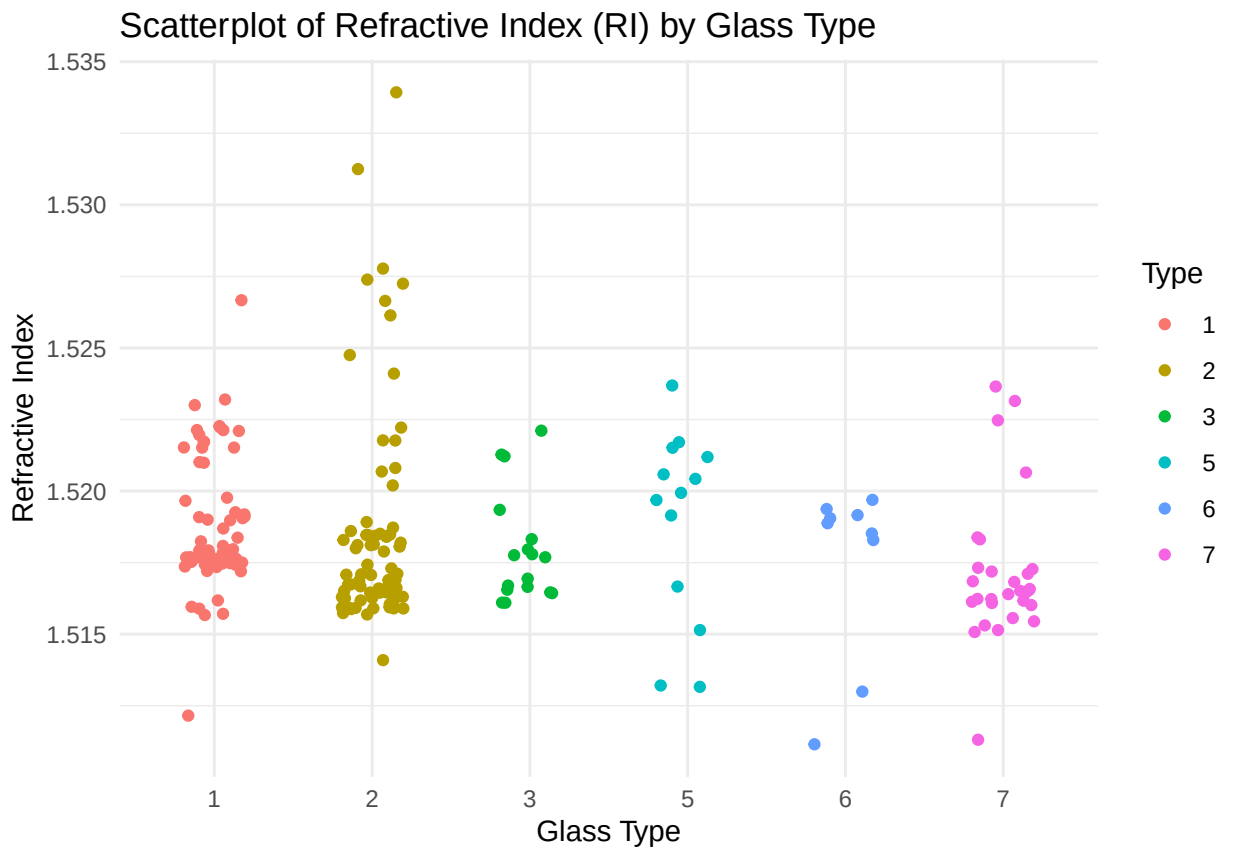



```
# 3.2 (Glass Glimpse)
# Load necessary libraries
library(ggplot2)
library(readxl)  # For reading Excel files

glass_data <- read_excel("~/Downloads/BRSM_Visualisation_Assignment.xlsx",
                         sheet = "Glass Glimpse")

# Convert 'Type' to a factor for better plotting
glass_data$Type <- as.factor(glass_data$Type)

# Plot: Boxplot with jitter for better visualization
ggplot(glass_data, aes(x = Type, y = RI, fill = Type)) +
  geom_boxplot(outlier.shape = NA, alpha = 0.6) +  # Semi-transparent boxplot
  geom_jitter(width = 0.15, alpha = 0.5, color = "black") + # Jitter points to avoid overlap
  scale_y_continuous(breaks = seq(floor(min(glass_data$RI, na.rm = TRUE)),
                                  ceiling(max(glass_data$RI, na.rm = TRUE)),
                                  by = 0.005)) + # Adjust y-axis labels
  theme_minimal() +
```

```
  theme(
    legend.position = "none",
    axis.text.y = element_text(size = 10),
    axis.text.x = element_text(size = 12, face = "bold")
  ) +
  labs(
    title = "Distribution of Refractive Index (RI) by Glass Type",
    x = "Glass Type",
    y = "Refractive Index"
  )
```



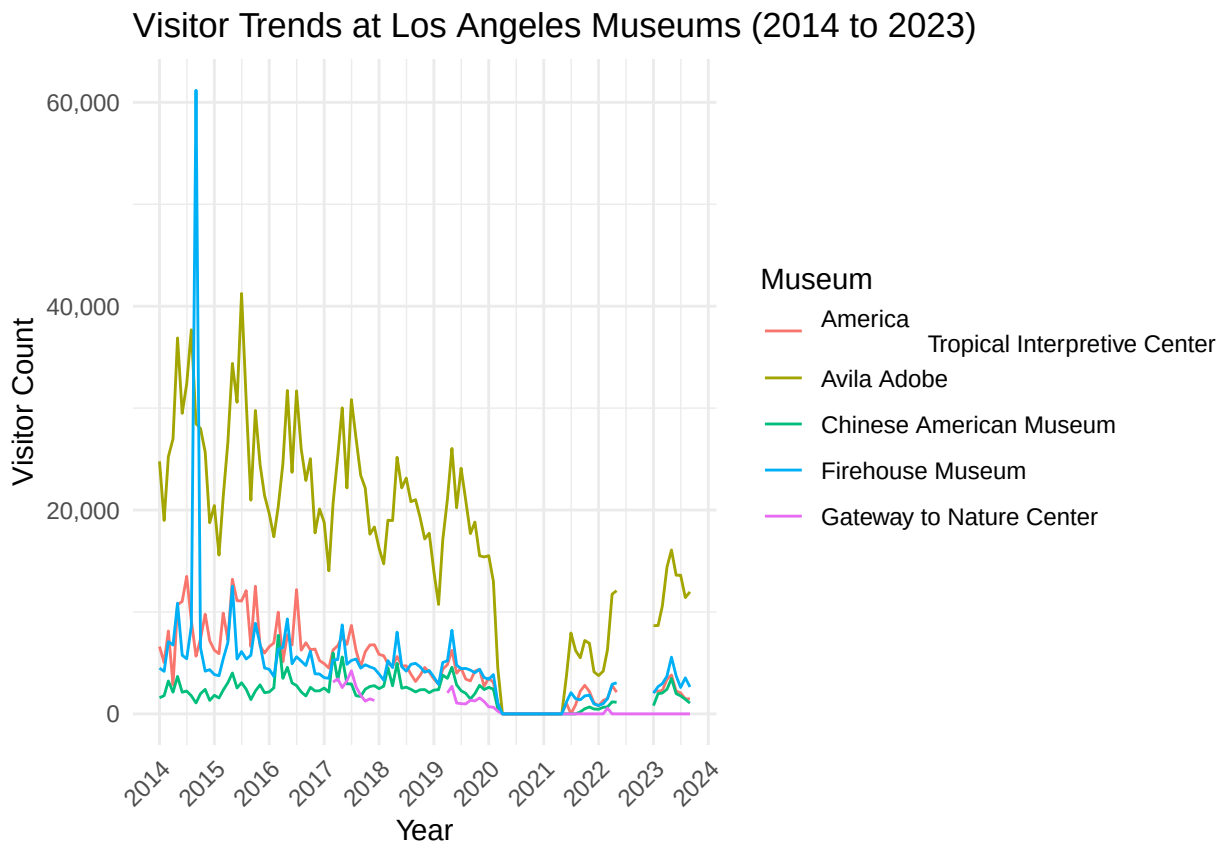

```
# Plot 2: Scatterplot to show the relationship between RI and Type
ggplot(glass_data, aes(x = Type, y = RI, color = Type)) +
  geom_jitter(width = 0.2) +
  labs(
    title = "Scatterplot of Refractive Index (RI) by Glass Type",
    x = "Glass Type",
    y = "Refractive Index"
  ) +
  theme_minimal()
```




```
# Inferences
# 1. The boxplot shows the distribution, median, and spread of RI values for each
# glass type.
# 2. The scatterplot highlights individual data points and their spread, which can
# indicate overlaps or outliers.

#-------------------------------------------------------------------------------

# 3.3 (Night at the Museum)
# Load required libraries
library(ggplot2)
library(readxl)
library(scales)

museum_data <- read_excel("~/Downloads/BRSM_Visualisation_Assignment.xlsx", sheet = "Museum Visitor")

# Convert Month column to Date format
museum_data$Month <- as.Date(paste0("01-", museum_data$Month), format = "%d-%b %Y")

# Plotting the time-series trends for visitor counts
ggplot(museum_data, aes(x = Month)) +
  geom_line(aes(y = `America Tropical Interpretive Center`, color = "America
                Tropical Interpretive Center")) +
  geom_line(aes(y = `Avila Adobe`, color = "Avila Adobe")) +
  geom_line(aes(y = `Chinese American Museum`, color = "Chinese American Museum")) +
  geom_line(aes(y = `Gateway to Nature Center`, color = "Gateway to Nature Center")) +
```

```
  geom_line(aes(y = `Firehouse Museum`, color = "Firehouse Museum")) +
  scale_x_date(date_labels = "%Y", date_breaks = "1 year") +
  scale_y_continuous(labels = comma) +
  labs(title = "Visitor Trends at Los Angeles Museums (2014 to 2023)",
       x = "Year",
       y = "Visitor Count",
       color = "Museum") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
## Warning: Removed 38 rows containing missing values or values outside the scale range
## (`geom_line()`).
```

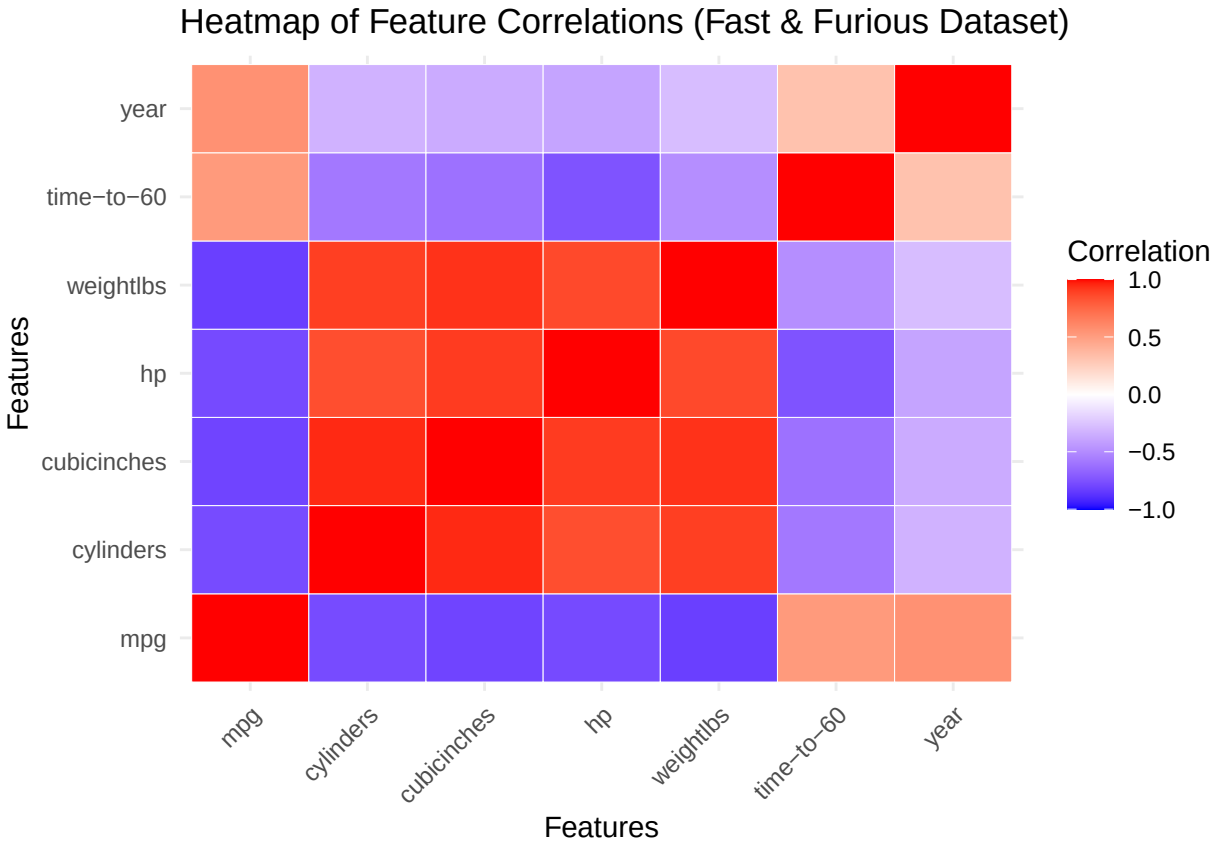



```
#-------------------------------------------------------------------------------

#Question 4: (Fast and Furious)
# Load required libraries
library(readxl)
library(ggplot2)
library(reshape2)
library(RColorBrewer)

fast_furious_data <- read_excel("~/Downloads/BRSM_Visualisation_Assignment.xlsx",
                                sheet = "Fast and Furious")
```

```r
# View the first few rows
head(fast_furious_data)
```

```
## # A tibble: 6 x 8
##     mpg cylinders cubicinches    hp weightlbs `time-to-60`  year brand
##   <dbl>     <dbl>       <dbl> <dbl>     <dbl>        <dbl> <dbl> <chr>
## 1  14           8         350   165      4209           12  1972 US.
## 2  31.9         4          89    71      1925           14  1980 Europe.
## 3  17           8         302   140      3449           11  1971 US.
## 4  15           8         400   150      3761           10  1971 US.
## 5  30.5         4          98    63      2051           17  1978 US.
## 6  23           8         350   125      3900           17  1980 US.
```

```r
# Remove the 'brand' column (non-numeric) for correlation analysis
numeric_data <- fast_furious_data[, c("mpg", "cylinders", "cubicinches", "hp",
                                      "weightlbs", "time-to-60", "year")]

# Compute the correlation matrix
correlation_matrix <- cor(numeric_data, use = "complete.obs")

# Reshape the correlation matrix for ggplot
melted_corr <- melt(correlation_matrix)

# Plotting the heatmap
ggplot(data = melted_corr, aes(Var1, Var2, fill = value)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white",
                       midpoint = 0, limit = c(-1, 1), space = "Lab",
                       name = "Correlation") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(title = "Heatmap of Feature Correlations (Fast & Furious Dataset)",
       x = "Features", y = "Features")
```


Heatmap of Feature Correlations (Fast & Furious Dataset)
Features
year
time−to−60
weightlbs
hp
cubicinches
cylinders
mpg
mpg
cylinders
cubicinches
hp
weightlbs
time−to−60
year
Features
Correlation
1.0
0.5
0.0
−0.5
−1.0