2020102005 - Srujana Vanka - Visualisation Activity

2024-01-22

```
knitr::opts_chunk$set(echo = TRUE, warning=FALSE, message=FALSE)
```

```
library(readxl)
library(ggplot2)
library(magrittr)
library(dplyr)
library(plotrix)
```

1 Statistical Deception

```
data1 <- read_excel("./2024_Assignment1_BRSM.xlsx",
    sheet = "Statistical Deception")
df <- data1
head(df)
```

```
## # A tibble: 6 × 4
##      x1    x2    x3    x4
##   <dbl> <dbl> <dbl> <dbl>
## 1  1     1     1     1
## 2  2.02  7.10  1.26  7.40
## 3  2.68  7.16  1.52  7.40
## 4  3.18  7.19  1.78  7.40
## 5  3.59  7.21  2.04  7.40
## 6  3.93  7.23  2.31  7.40
```

We load data from the specified data collection into our data frame object. Before displaying the data, it would be beneficial to obtain a measure of central trends (mean, median, mode) for our data in order to determine which visualization approaches would work best for it, hence utilizing the summary function.

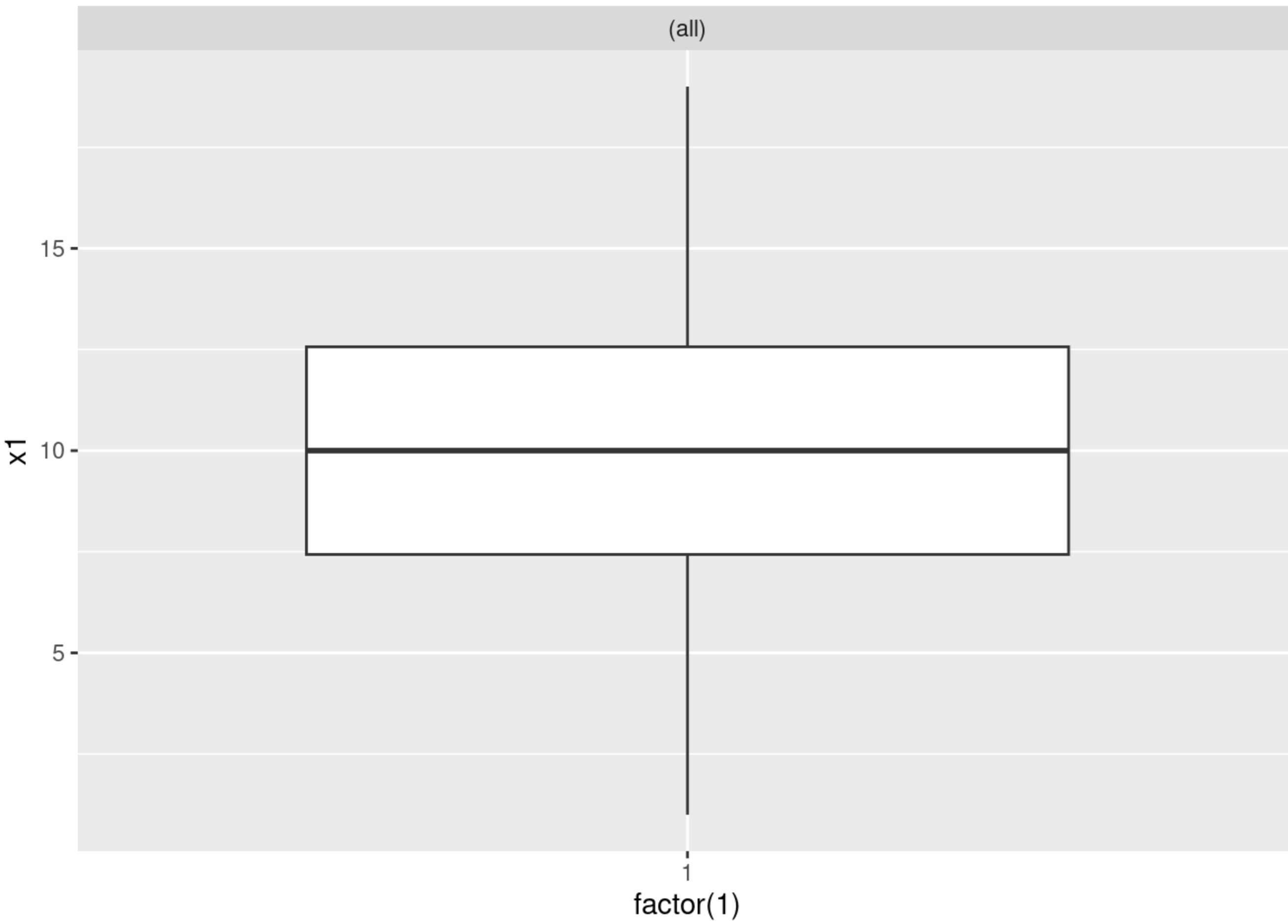
```
summary(df)
```

```
##        x1               x2               x3               x4
##  Min.   : 1.000   Min.   : 1.000   Min.   : 1.000   Min.   : 1.000
##  1st Qu.: 7.433   1st Qu.: 7.405   1st Qu.: 7.465   1st Qu.: 7.403
##  Median :10.000   Median :10.000   Median :10.000   Median :10.000
##  Mean   :10.000   Mean   :10.000   Mean   :10.000   Mean   :10.736
##  3rd Qu.:12.567   3rd Qu.:12.595   3rd Qu.:12.535   3rd Qu.:12.597
##  Max.   :19.000   Max.   :19.000   Max.   :19.000   Max.   :19.000
```

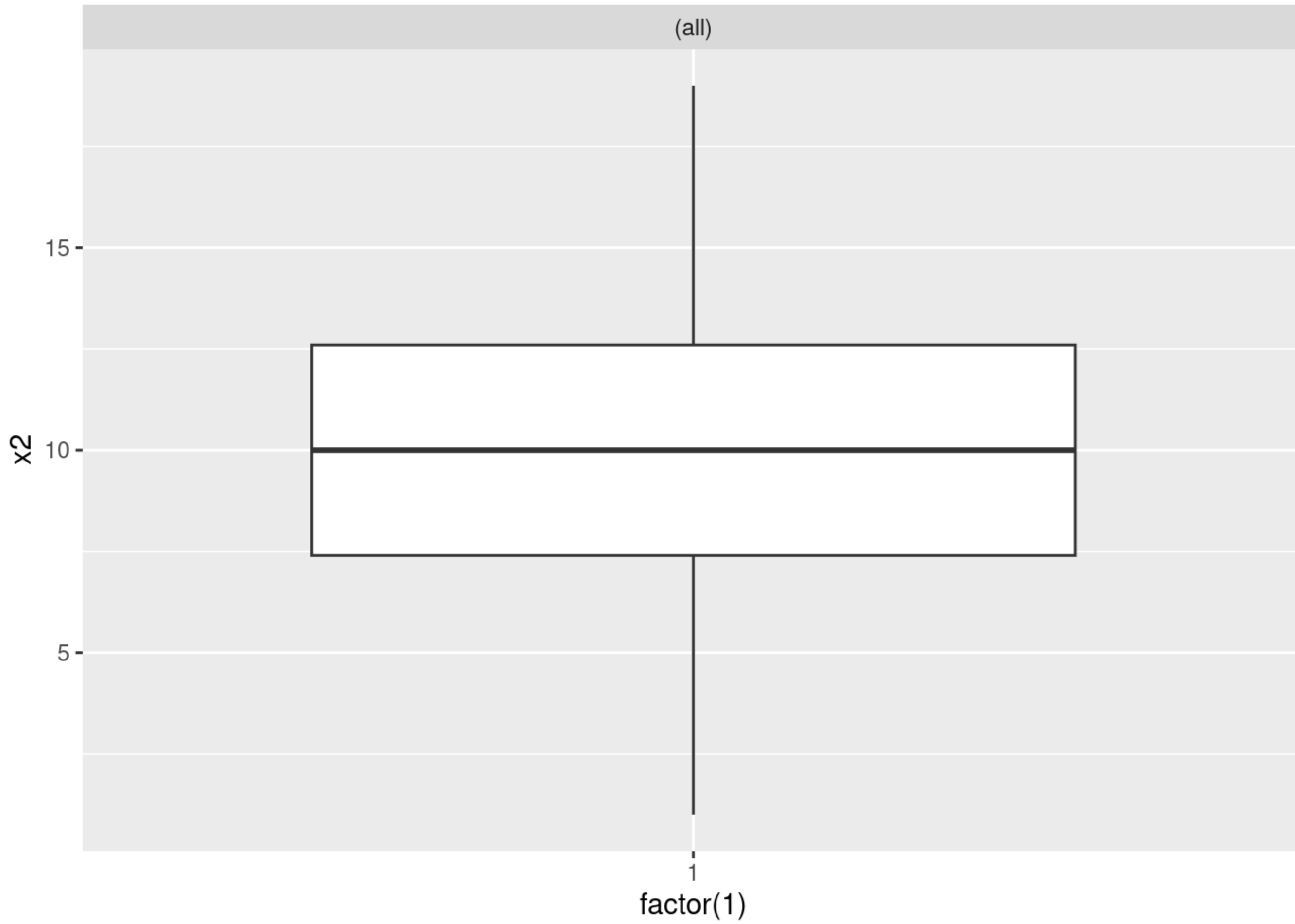
Here, we observe that the median, and the quantile data values are almost the same for all the data objects x1,x2,x3,x4.

Plotting the Data - Box plots

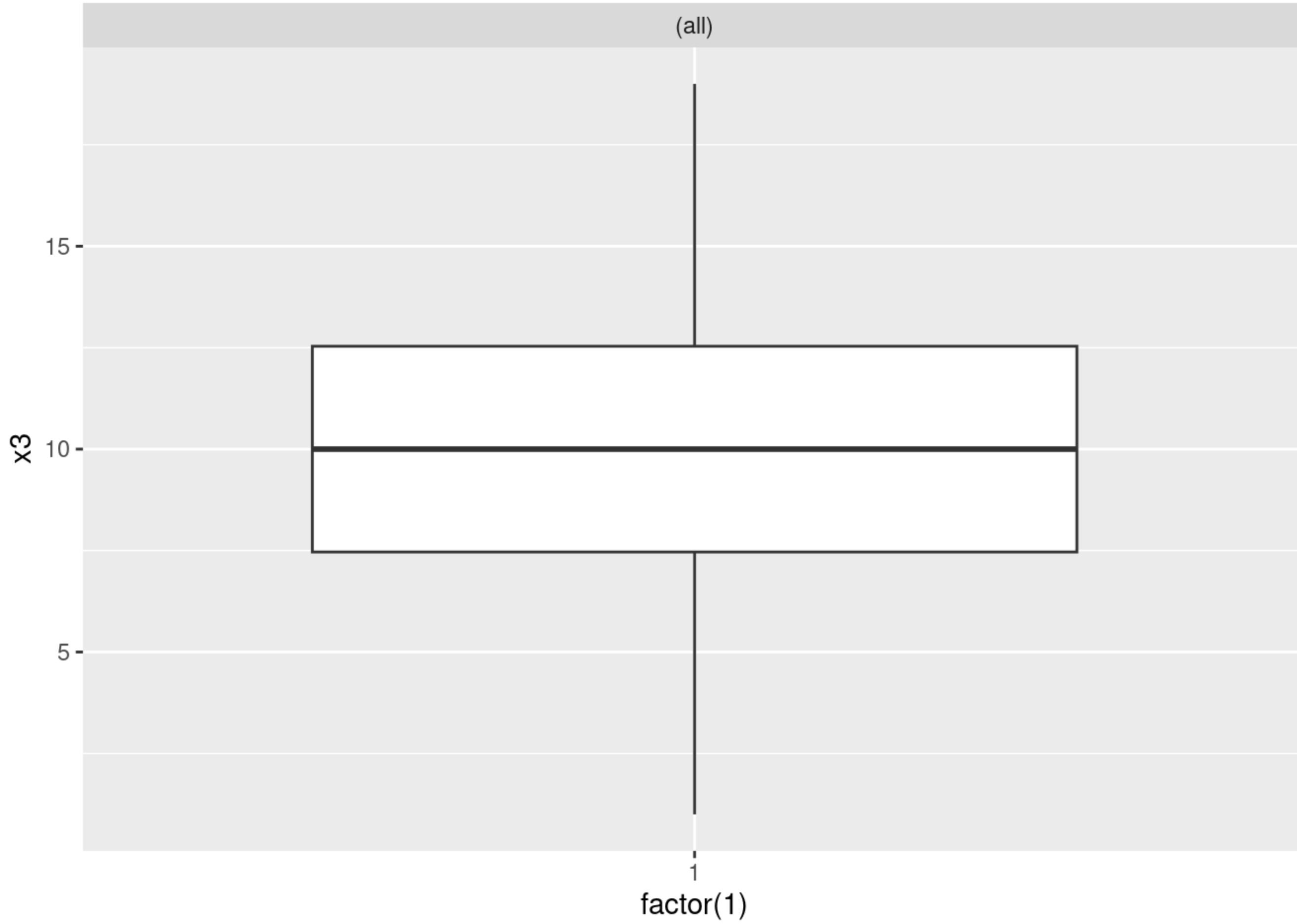
```
df <- data1
ggplot(df, aes(x=factor(1), y=x1)) +
  geom_boxplot() +
  facet_wrap(~., scales="free_y")
```



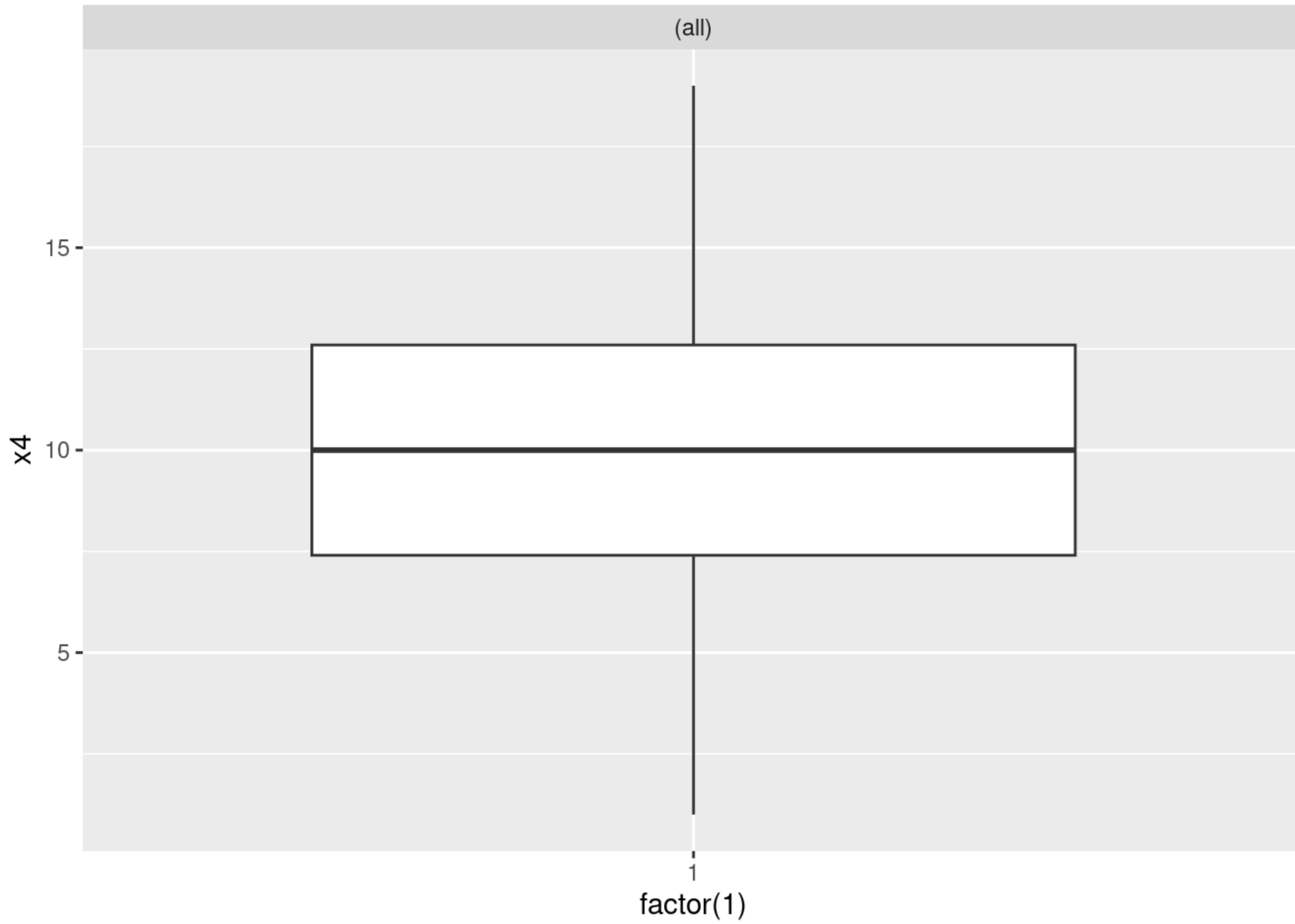

```
ggplot(df, aes(x=factor(1), y=x2)) +
geom_boxplot() +
facet_wrap(~., scales="free_y")
```




```
ggplot(df, aes(x=factor(1), y=x3)) +
geom_boxplot() +
facet_wrap(~., scales="free_y")
```




```
ggplot(df, aes(x=factor(1), y=x4)) +
geom_boxplot() +
facet_wrap(~., scales="free_y")
```



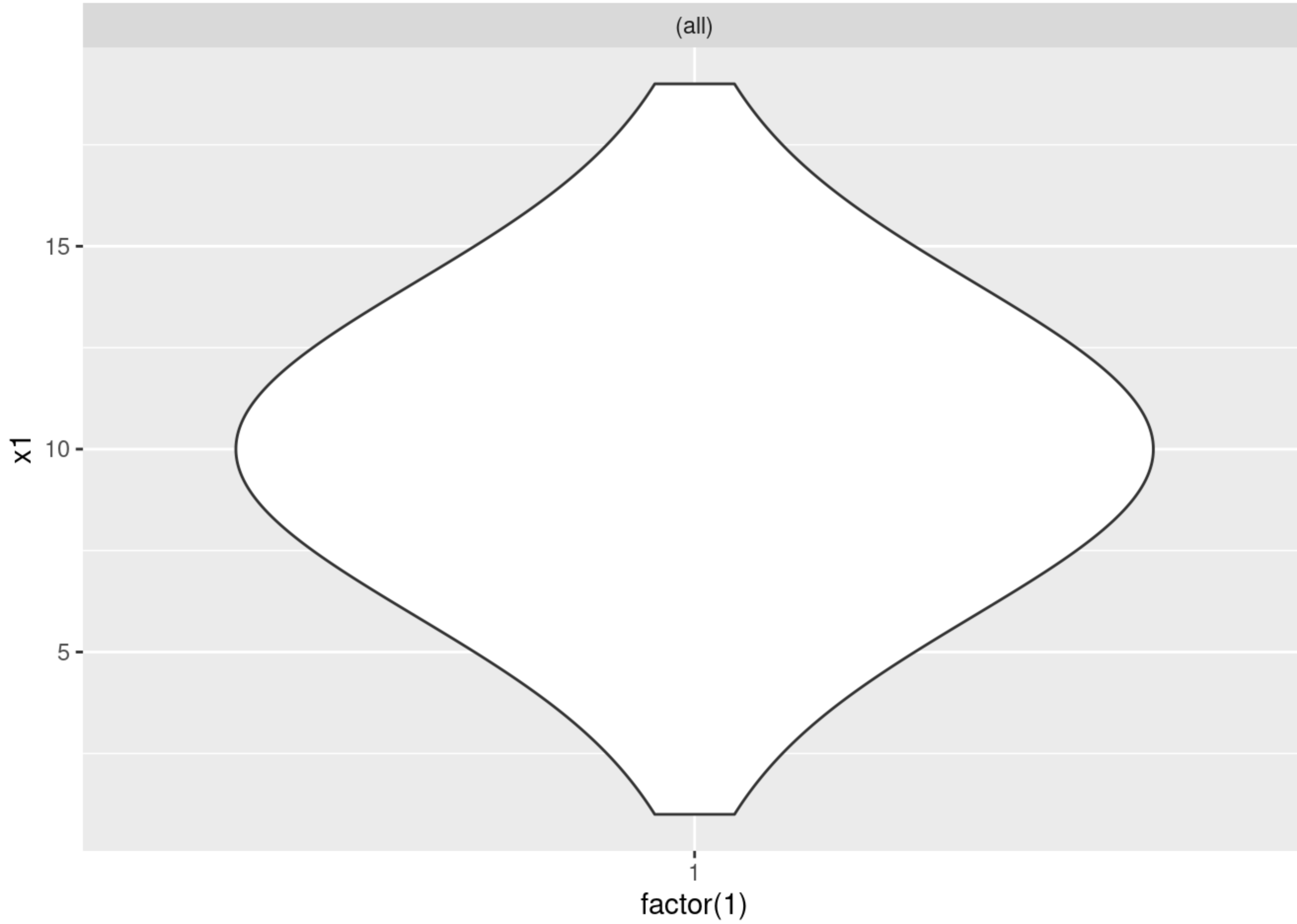
Observations and Inferences

Box plot gives the same plot for all x1,x2,x3,x4, so it is misleading. Boxplots can be misleading if the data has outliers, as outliers can significantly affect the appearance of the plot, giving a false representation of the distribution of the data.

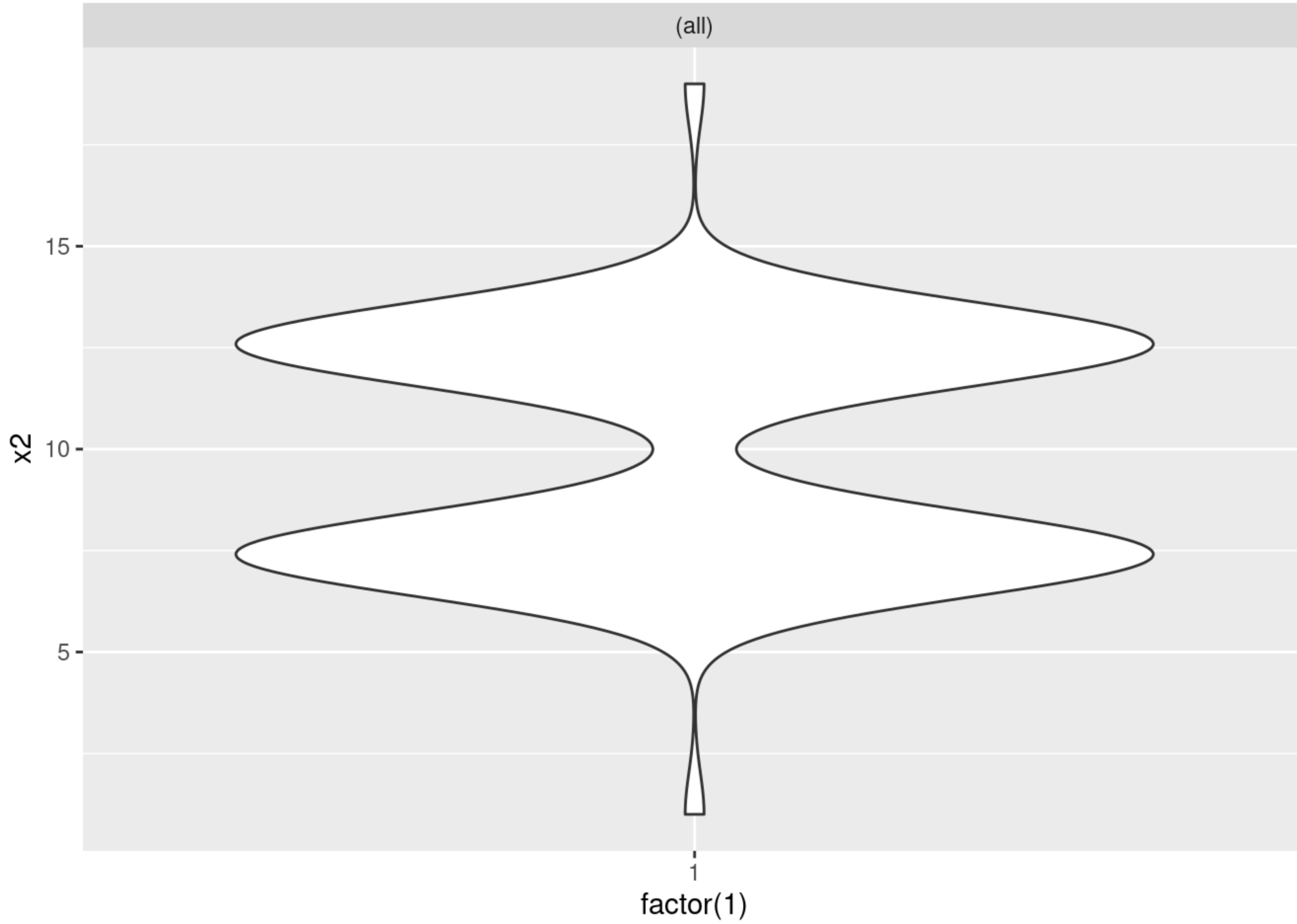
Additionally, the boxplot only shows summary statistics, such as the median and quartiles, and does not show the full distribution of the data, which could also contribute to its misleading nature.

2. Violin graphs

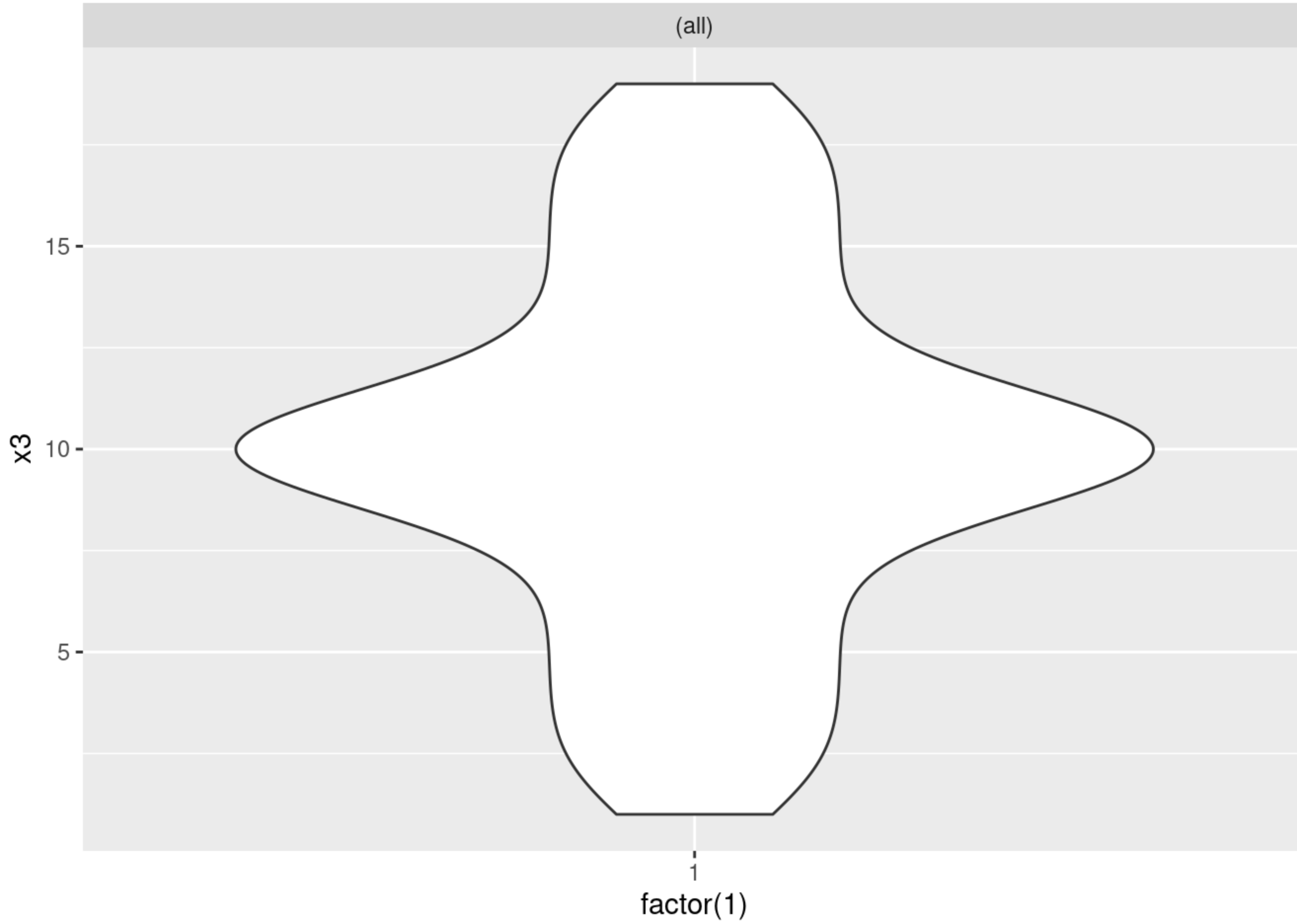
```
ggplot(df, aes(x=factor(1), y=x1)) +
geom_violin() +
facet_wrap(~., scales="free_y")
```



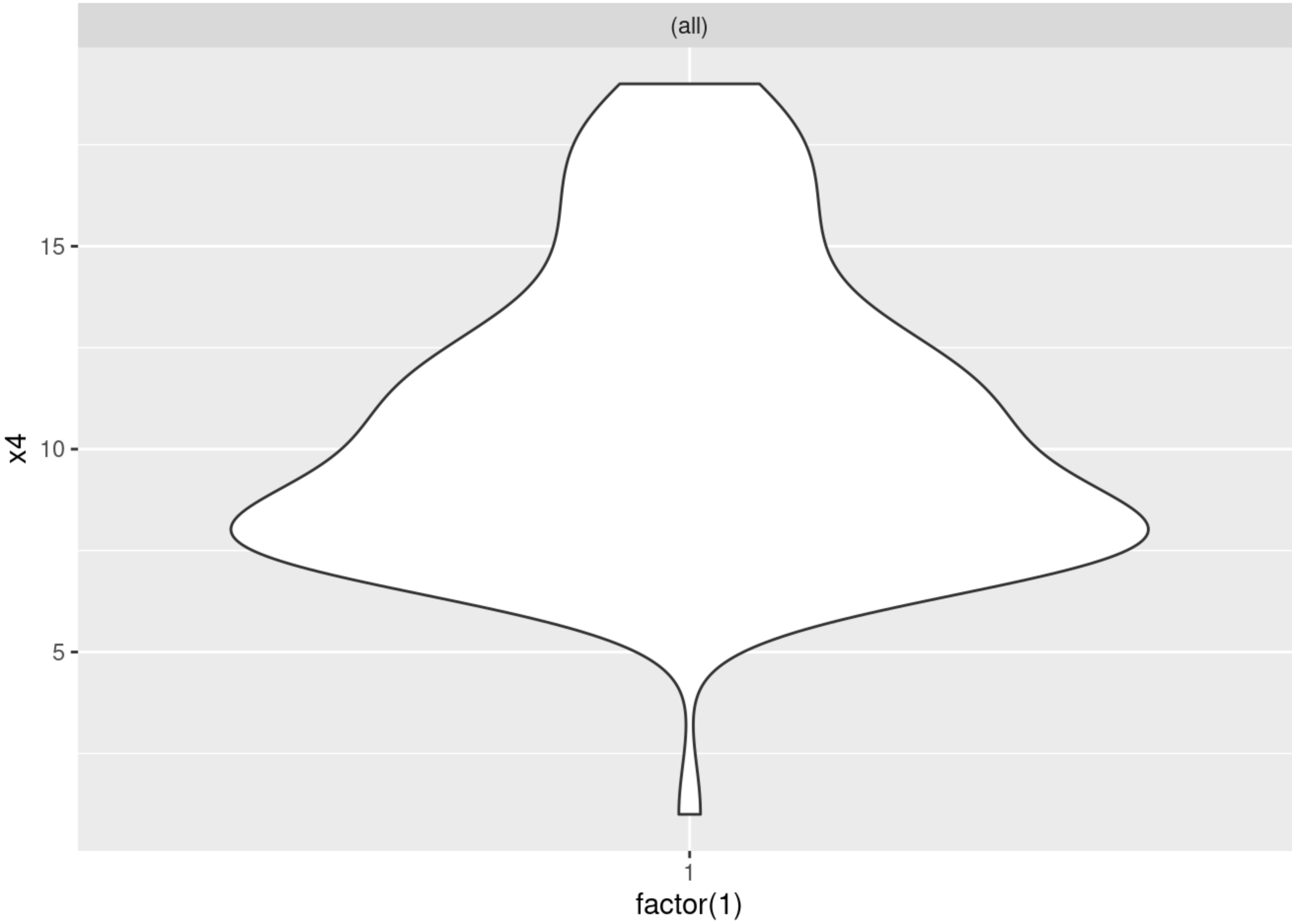

```
ggplot(df, aes(x=factor(1), y=x2)) +
geom_violin() +
facet_wrap(~., scales="free_y")
```




```
ggplot(df, aes(x=factor(1), y=x3)) +
geom_violin() +
facet_wrap(~., scales="free_y")
```




```
ggplot(df, aes(x=factor(1), y=x4)) +
geom_violin() +
facet_wrap(~., scales="free_y")
```



Observations and Inferences

Violin plots are considered to be a better visualization method compared to boxplots in cases where the data has multiple modes or is not symmetrical, as they provide a more complete representation of the distribution of the data.

Personality and Motion

```
data2 <- read_excel("./2024_Assignment1_BRSM.xlsx",
    sheet = "Movement Personality Results")
df <- data2
head(df)
```

```
## # A tibble: 6 × 6
##   Movements Openness Conscientiousness Extraversion Agreeableness Neuroticism
##   <chr>        <dbl>             <dbl>        <dbl>         <dbl>       <dbl>
## 1 Root         0.139             0            0.325         0.147       0.169
## 2 Hips         0.530             0.477        0.804         0.548       0.686
## 3 Knee         0.869             1            0.662         0.936       1
## 4 Ankle        0.965             0.723        0.639         1           0.735
## 5 Toe          0.982             0.590        0.851         0.893       0.970
## 6 Torso        0.551             0.373        0.490         0.638       0.612
```

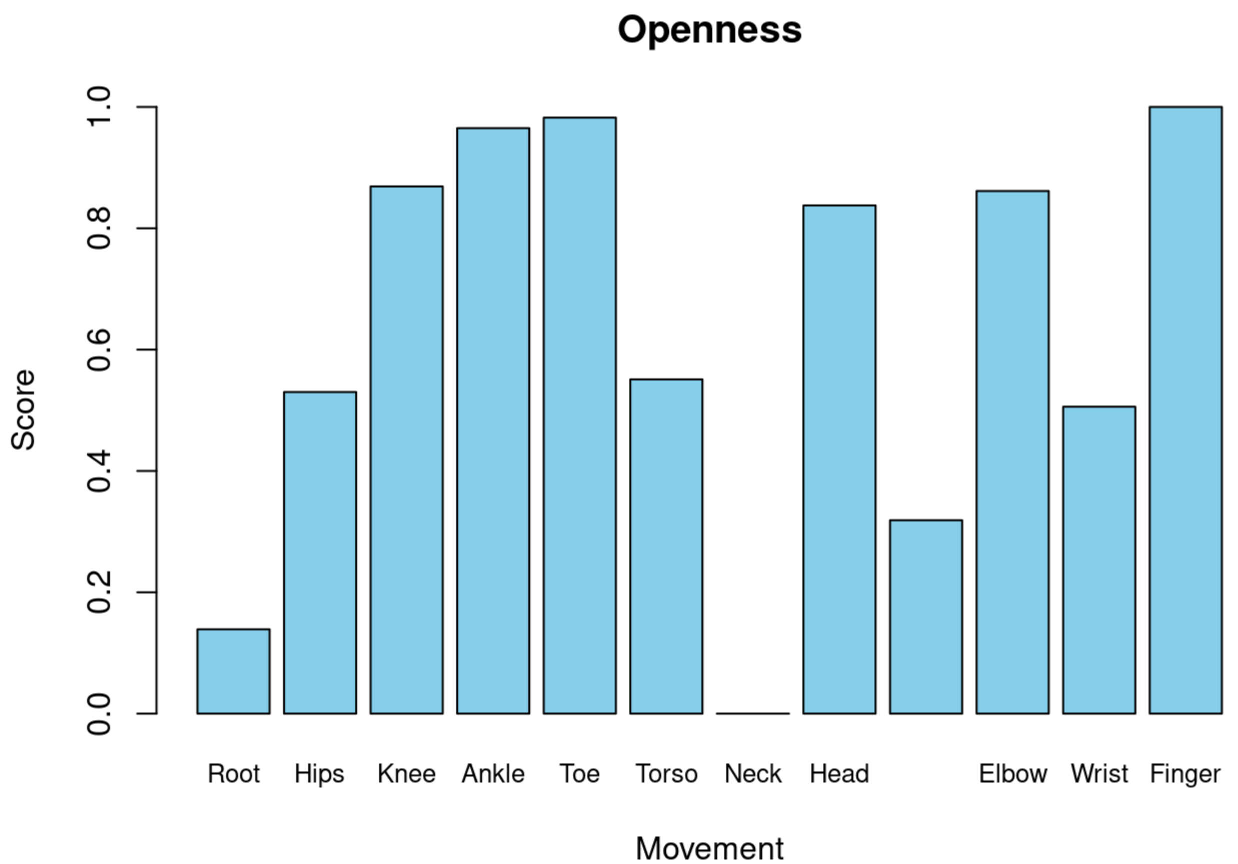
```
summary(df)
```

```
##   Movements            Openness      Conscientiousness  Extraversion
##  Length:12          Min.   :0.0000   Min.   :0.0000    Min.   :0.0000
##  Class :character   1st Qu.:0.4591   1st Qu.:0.3963    1st Qu.:0.4867
##  Mode  :character   Median :0.6943   Median :0.5216    Median :0.7332
##                     Mean   :0.6300   Mean   :0.4992    Mean   :0.6466
##                     3rd Qu.:0.8930   3rd Qu.:0.6373    3rd Qu.:0.8409
##                     Max.   :1.0000   Max.   :1.0000    Max.   :1.0000
##  Agreeableness     Neuroticism
##  Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.3807   1st Qu.:0.5456
##  Median :0.5646   Median :0.7109
##  Mean   :0.5766   Mean   :0.6337
##  3rd Qu.:0.9040   3rd Qu.:0.8238
##  Max.   :1.0000   Max.   :1.0000
```

Bar plots

Bar graph for Openness variable

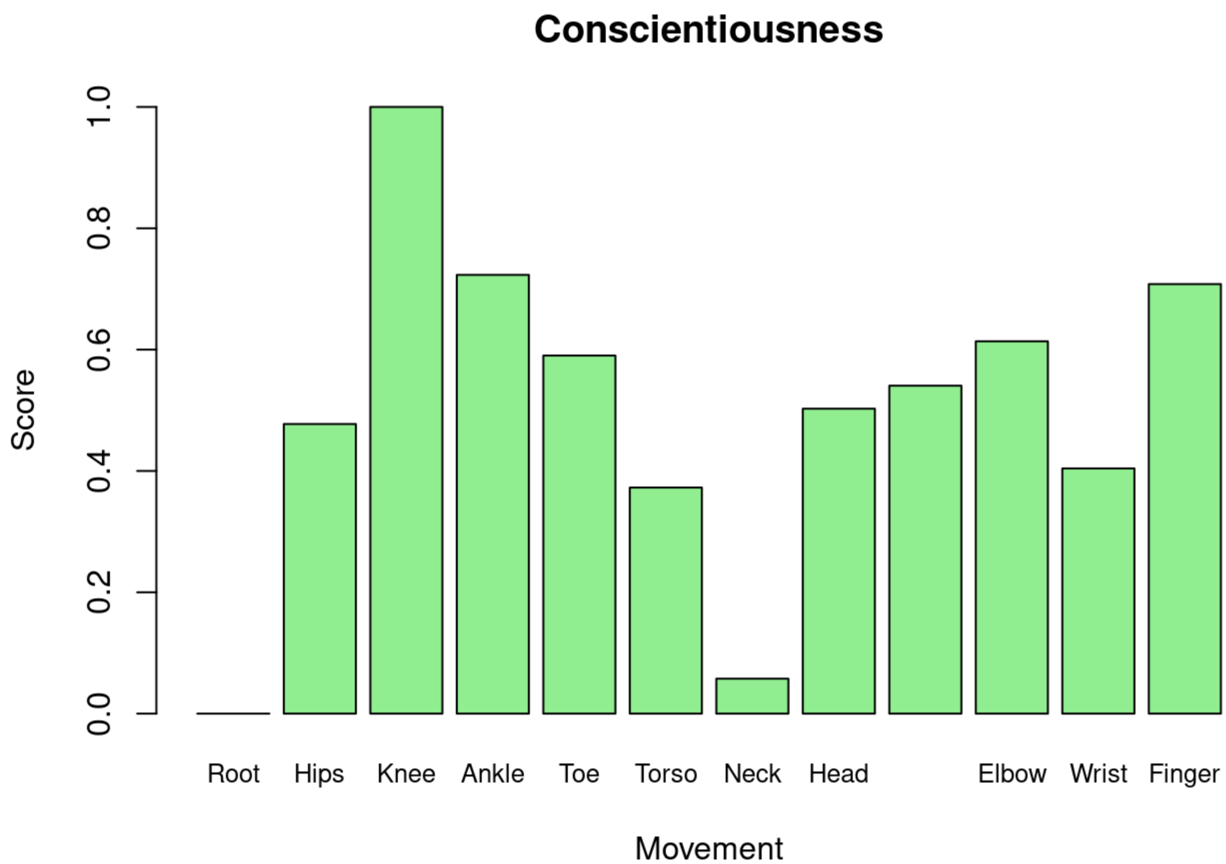
```
movements <- data2
barplot(data2$Openness,
        main = "Openness",
        xlab = "Movement",
        ylab = "Score",
        names.arg = data2$Movements,
        col = "skyblue",
        ylim = c(0, max(data2$Openness)),
        cex.names = 0.8
)
```



####

Bar graph for Conscientiousness variable

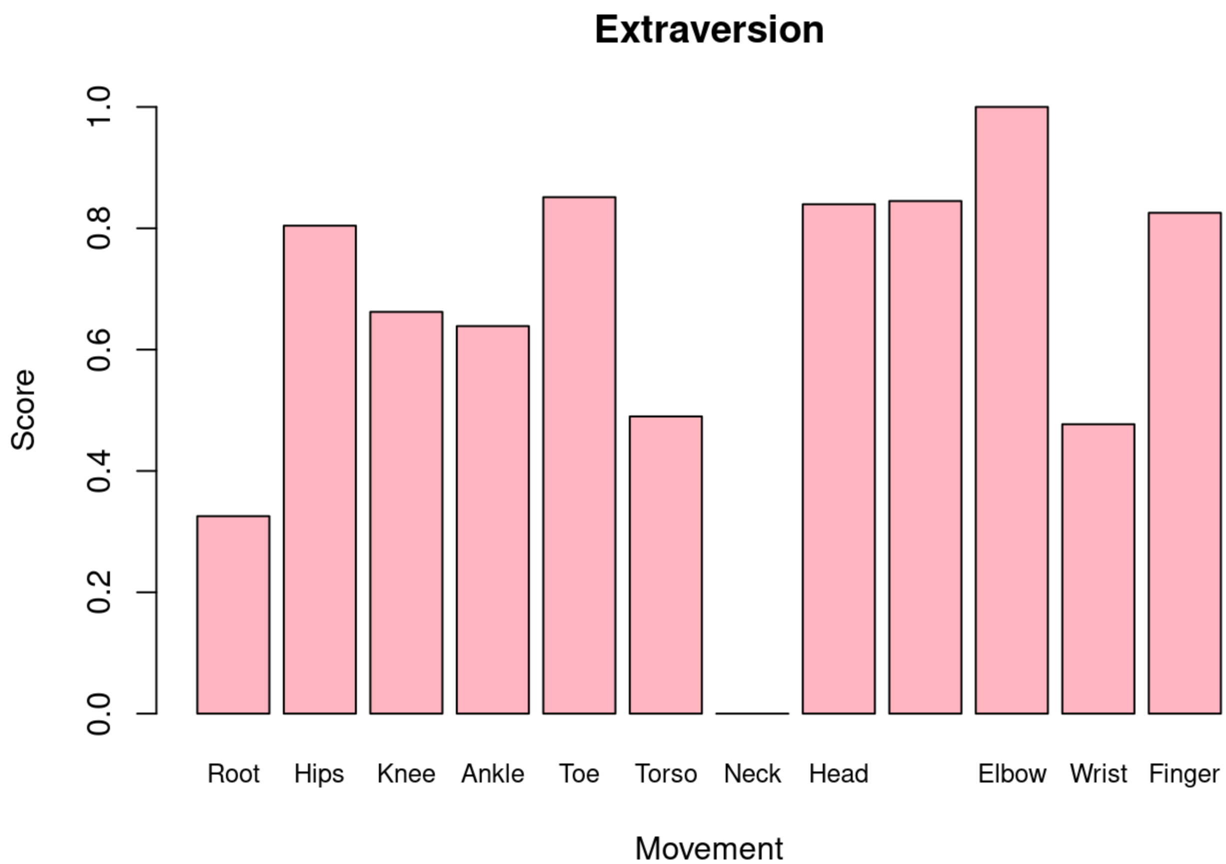
```
barplot(data2$Conscientiousness,
        main = "Conscientiousness",
        xlab = "Movement",
        ylab = "Score",
        names.arg = data2$Movements,
        col = "lightgreen",
        ylim = c(0, max(data2$Conscientiousness)),
        cex.names = 0.8
)
```



####

Bar graph for Extraversion variable

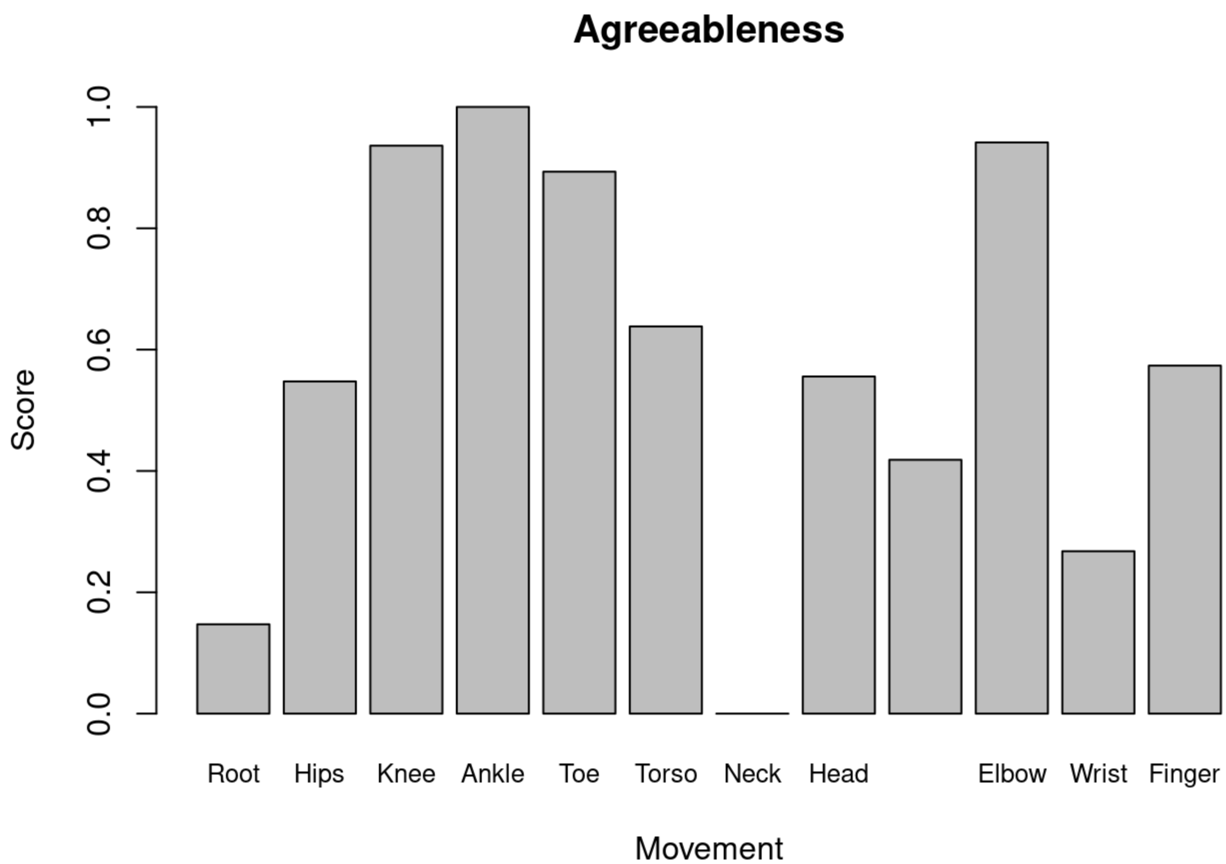
```
barplot(data2$Extraversion,
        main = "Extraversion",
        xlab = "Movement",
        ylab = "Score",
        names.arg = data2$Movements,
        col = "lightpink",
        ylim = c(0, max(data2$Extraversion)),
        cex.names = 0.8
)
```



####

Bar graph for Agreeableness variable

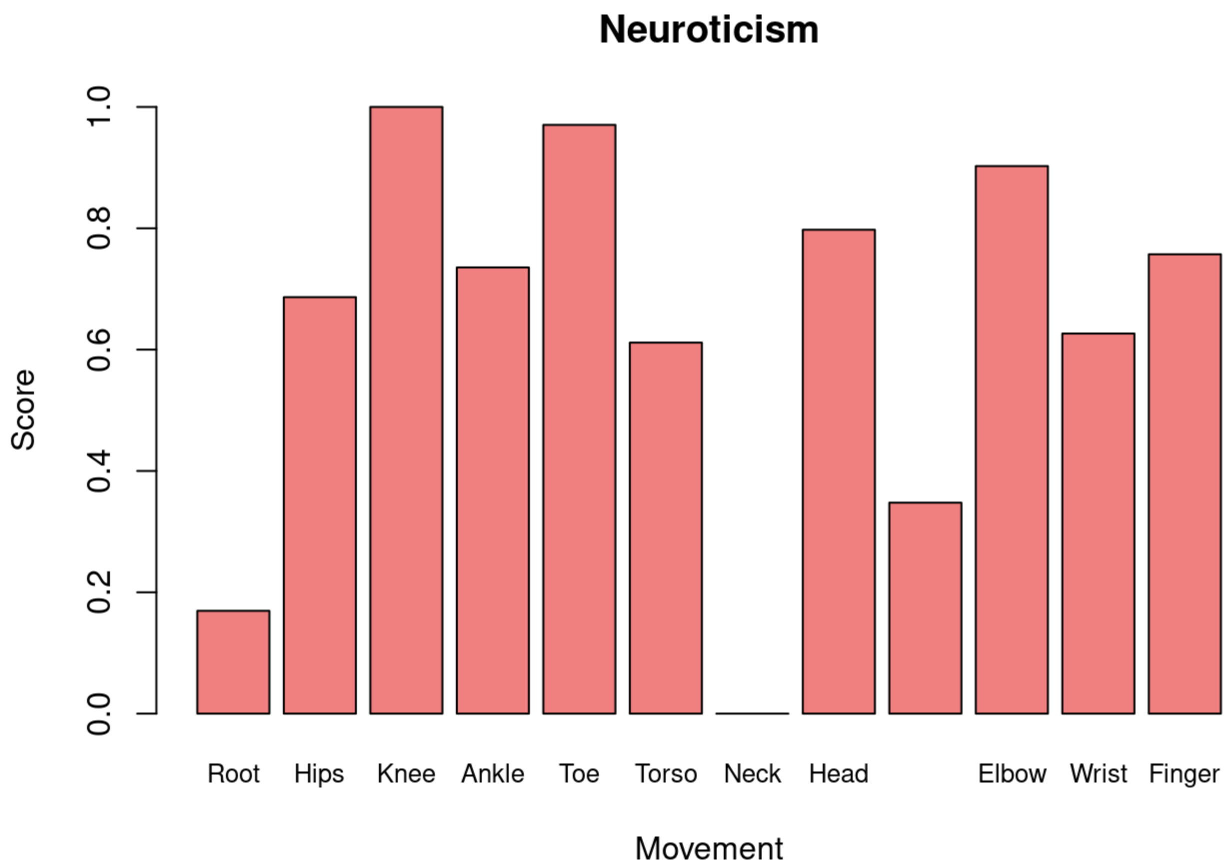
```
barplot(data2$Agreeableness,
        main = "Agreeableness",
        xlab = "Movement",
        ylab = "Score",
        names.arg = data2$Movements,
        col = "gray",
        ylim = c(0, max(data2$Agreeableness)),
        cex.names = 0.8
)
```



####

Bar graph for Neuroticism variable

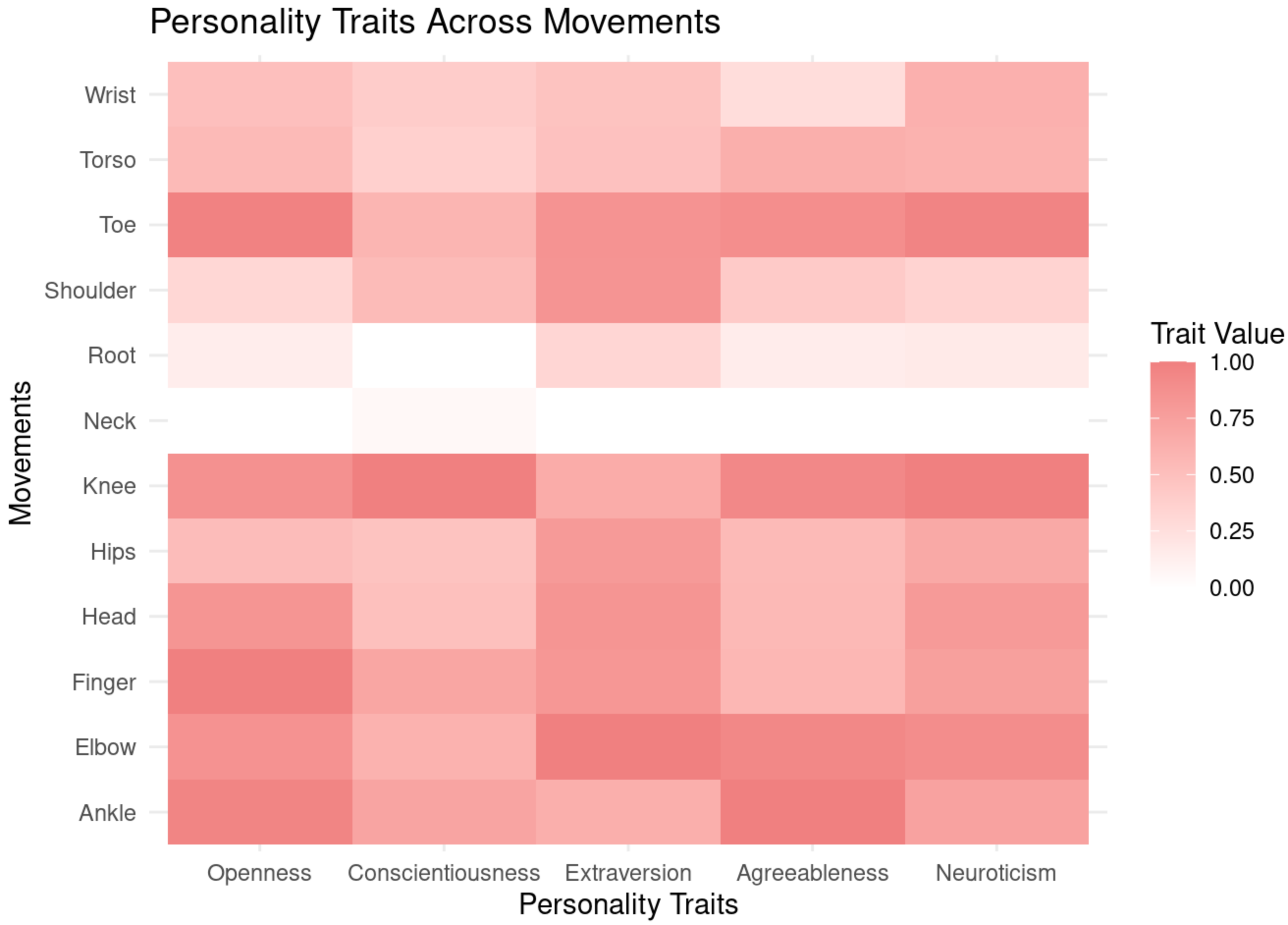
```
barplot(data2$Neuroticism,
        main = "Neuroticism",
        xlab = "Movement",
        ylab = "Score",
        names.arg = data2$Movements,
        col = "lightcoral",
        ylim = c(0, max(data2$Neuroticism)),
        cex.names = 0.8
)
```



###

2. Heat map

```
library(reshape2)
df_melted <- melt(df, id.vars = "Movements")

# Plot heatmap
ggplot(df_melted, aes(x = variable, y = Movements, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "lightcoral") +
  labs(title = "Personality Traits Across Movements",
       x = "Personality Traits",
       y = "Movements",
       fill = "Trait Value") +
  theme_minimal()
```



Observations and Inferences

The dataset explores the connection between personality traits and various joint movements, assigning importance values to each joint. The visualizations include a bar plot and a heatmap.

The heatmap proves to be a more effective visualization, showcasing correlations between different joint movements and personality traits. With a color gradient indicating the strength of these correlations, it becomes easier to discern patterns. For instance, higher ankle and elbow movements might strongly correlate with agreeableness, shown by a darker shade on the heatmap.

Contrastingly, the bar plot, representing five bars for each joint attribute, becomes cluttered and challenging to interpret. Its lack of clarity makes it less reader-friendly, especially when compared to the heatmap. The heatmap's color gradient offers a more intuitive way to gauge the relative importance of each joint, making it a superior visualization technique for this dataset.

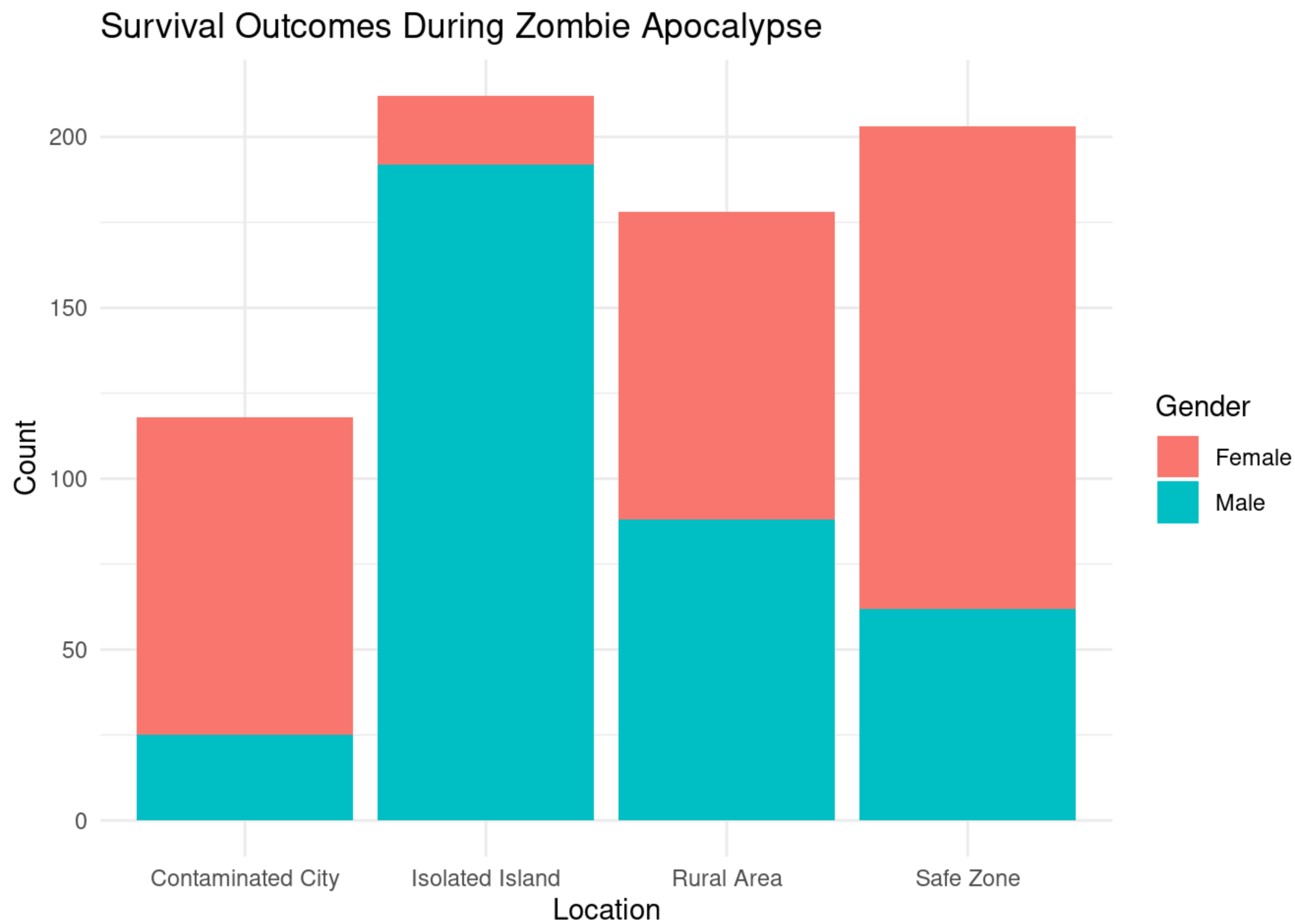
3 Data Plotting Adventure

3.1 Subtask 1: The Last of Us

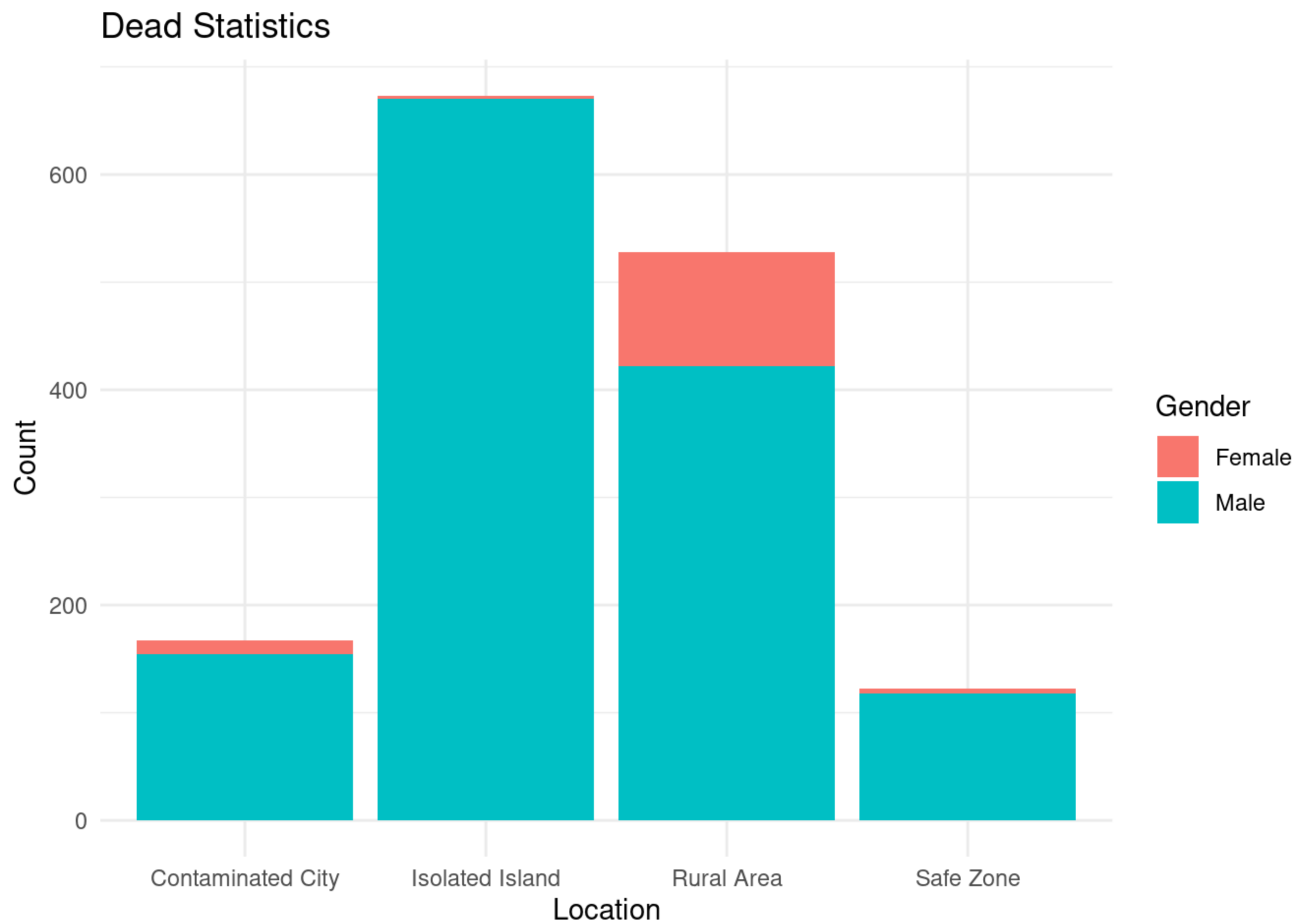
```
# Creating the dataset based on the information
data3 <- data.frame(
  Location = rep(c("Safe Zone", "Contaminated City", "Rural Area", "Isolated Island"), e
ach = 2),
  Gender = rep(c("Male", "Female"), times = 4),
  TurnedIntoZombies = c(118, 4, 154, 13, 422, 106, 670, 3),
  Survived = c(62, 141, 25, 93, 88, 90, 192, 20)
)

# Display the data
data3
```

```
##            Location Gender TurnedIntoZombies Survived
## 1         Safe Zone   Male               118       62
## 2         Safe Zone Female                 4      141
## 3 Contaminated City   Male               154       25
## 4 Contaminated City Female                13       93
## 5        Rural Area   Male               422       88
## 6        Rural Area Female               106       90
## 7   Isolated Island   Male               670      192
## 8   Isolated Island Female                 3       20
```

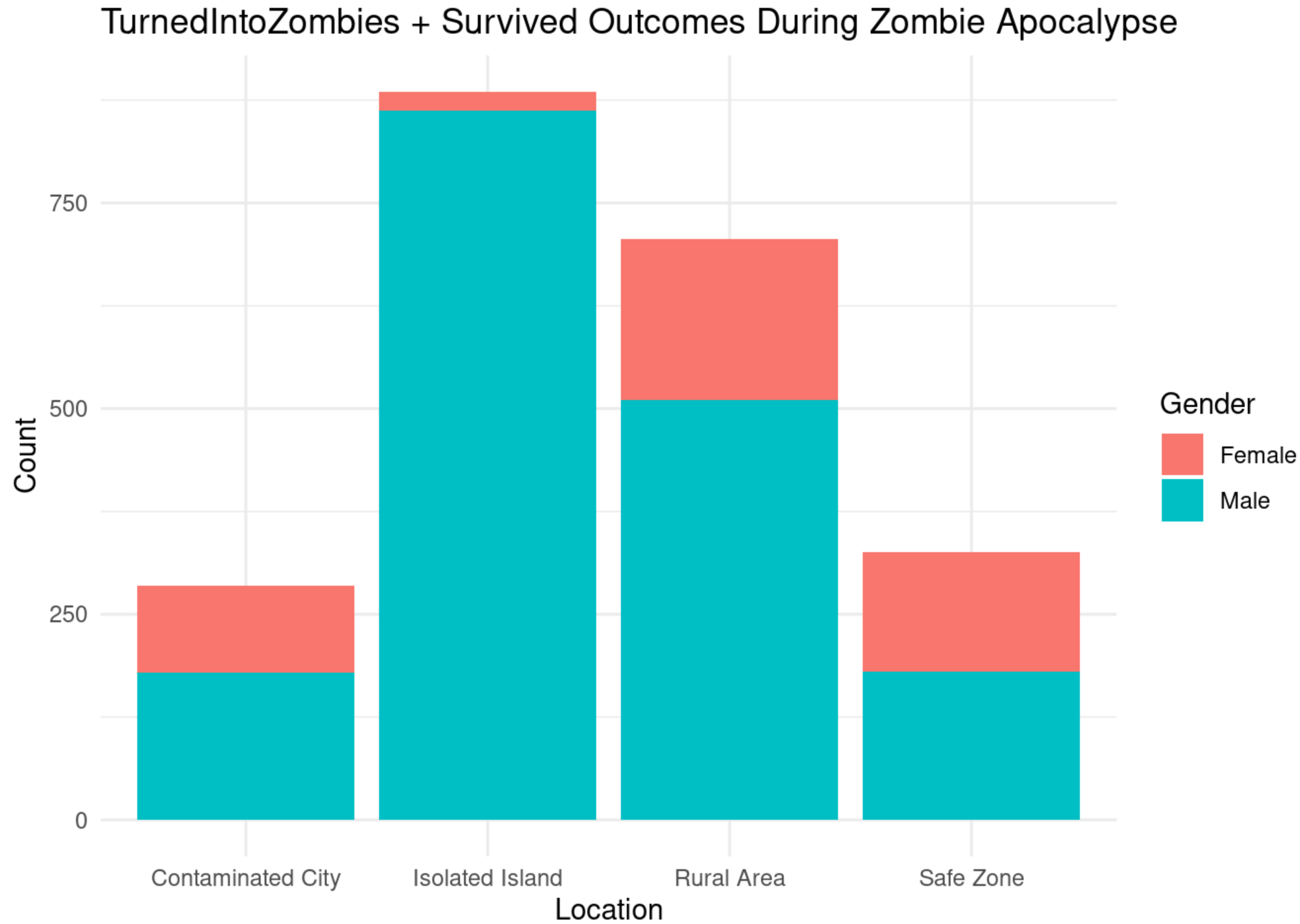
```
ggplot(data3, aes(x = Location, y = Survived, fill = Gender)) +
  geom_bar(stat = "identity", position = "stack") +
  labs(x = "Location", y = "Count", fill = "Gender") +
  ggtitle("Survival Outcomes During Zombie Apocalypse") +
  theme_minimal()
```




```
ggplot(data3, aes(x = Location, y = TurnedIntoZombies, fill = Gender)) +
  geom_bar(stat = "identity") +
  labs(x = "Location", y = "Count", fill = "Gender") +
  ggtitle("Dead Statistics") +
  theme_minimal()
```




```
ggplot(data3, aes(x = Location, y = TurnedIntoZombies+Survived, fill = Gender)) +
  geom_bar(stat = "identity", position = "stack") +
  labs(x = "Location", y = "Count", fill = "Gender") +
  ggtitle("TurnedIntoZombies + Survived Outcomes During Zombie Apocalypse") +
  theme_minimal()
```



Observations and Inferences

The stacked bar plot visually compares survival outcomes across locations during a zombie apocalypse. It highlights the total count of survivors in each location, emphasizing the composition of males and females. Gender disparities, common outcomes, and the varying impact of the apocalypse on survival can be quickly assessed. The plot provides a concise overview of the distribution of survivors in different categories, aiding in the identification of trends and patterns.

The Isolated Island emerges as the zone with the highest number of survivors, particularly among males, indicating a resilient community in the face of the zombie threat. The Safe Zone, despite a considerable number of males turning into zombies, shows a significant number of male survivors. The Contaminated City faces a high overall impact, with a notable number of males turning into zombies, but the survival rate is relatively high in both genders. The Rural Area experiences a substantial number of individuals turning into zombies, especially among males, but the survival rate remains relatively balanced between genders. Females exhibit a higher survival rate in most zones.

3.2 Subtask 2: Glass Glimpse

```
# Loading the dataset
data4 <- read_excel("./2024_Assignment1_BRSM.xlsx",
    sheet = "Glass Glimpse")
head(data4)
```

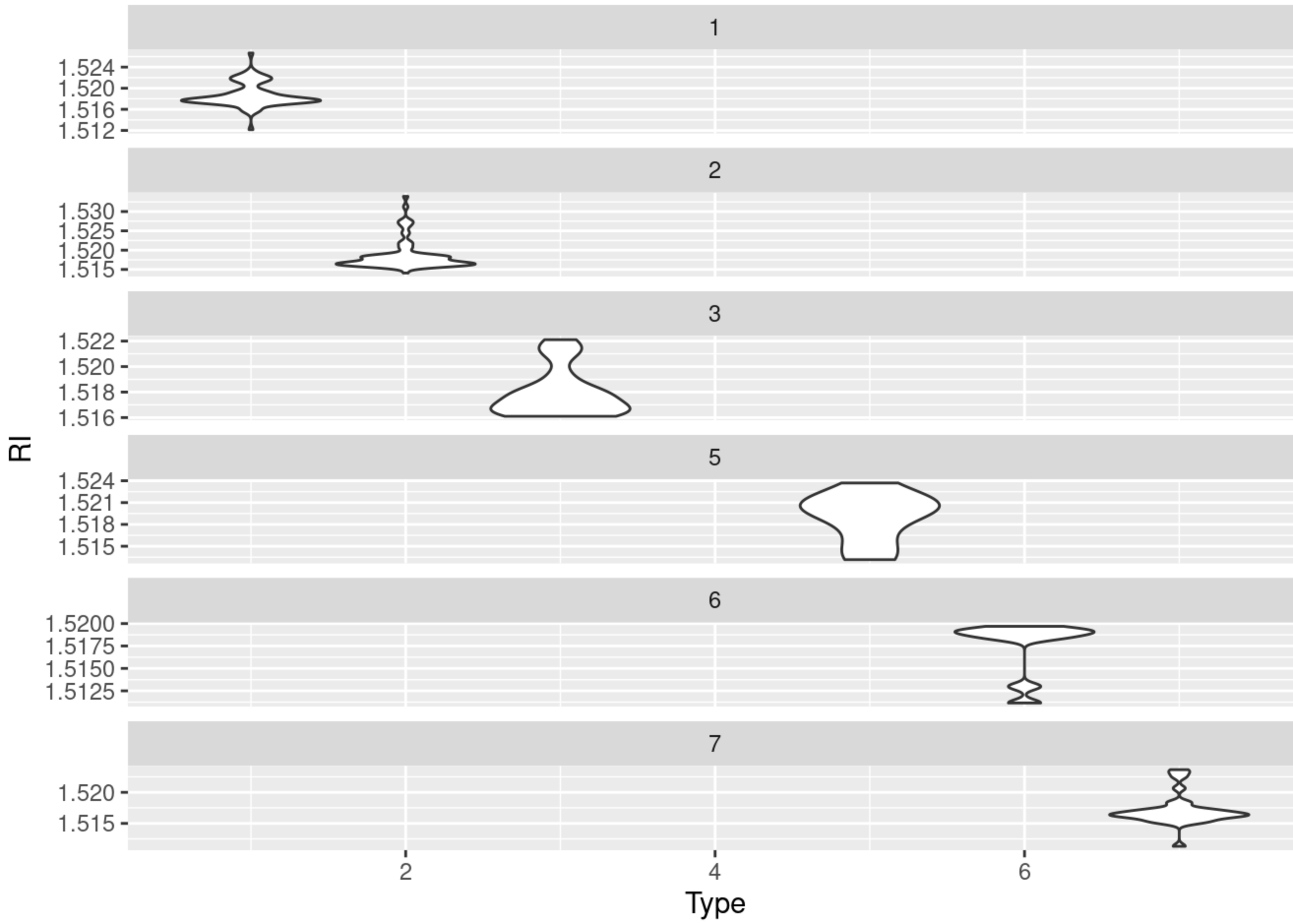
```
## # A tibble: 6 × 10
##      RI    Na    Mg    Al    Si     K    Ca    Ba    Fe  Type
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  1.52  13.6  4.49  1.1   71.8  0.06  8.75     0  0        1
## 2  1.52  13.9  3.6   1.36  72.7  0.48  7.83     0  0        1
## 3  1.52  13.5  3.55  1.54  73.0  0.39  7.78     0  0        1
## 4  1.52  13.2  3.69  1.29  72.6  0.57  8.22     0  0        1
## 5  1.52  13.3  3.62  1.24  73.1  0.55  8.07     0  0        1
## 6  1.52  12.8  3.61  1.62  73.0  0.64  8.07     0  0.26     1
```

```
# extracting RI and Type from the data
df <- dplyr :: select(data4,RI,Type)
summary(df)
```

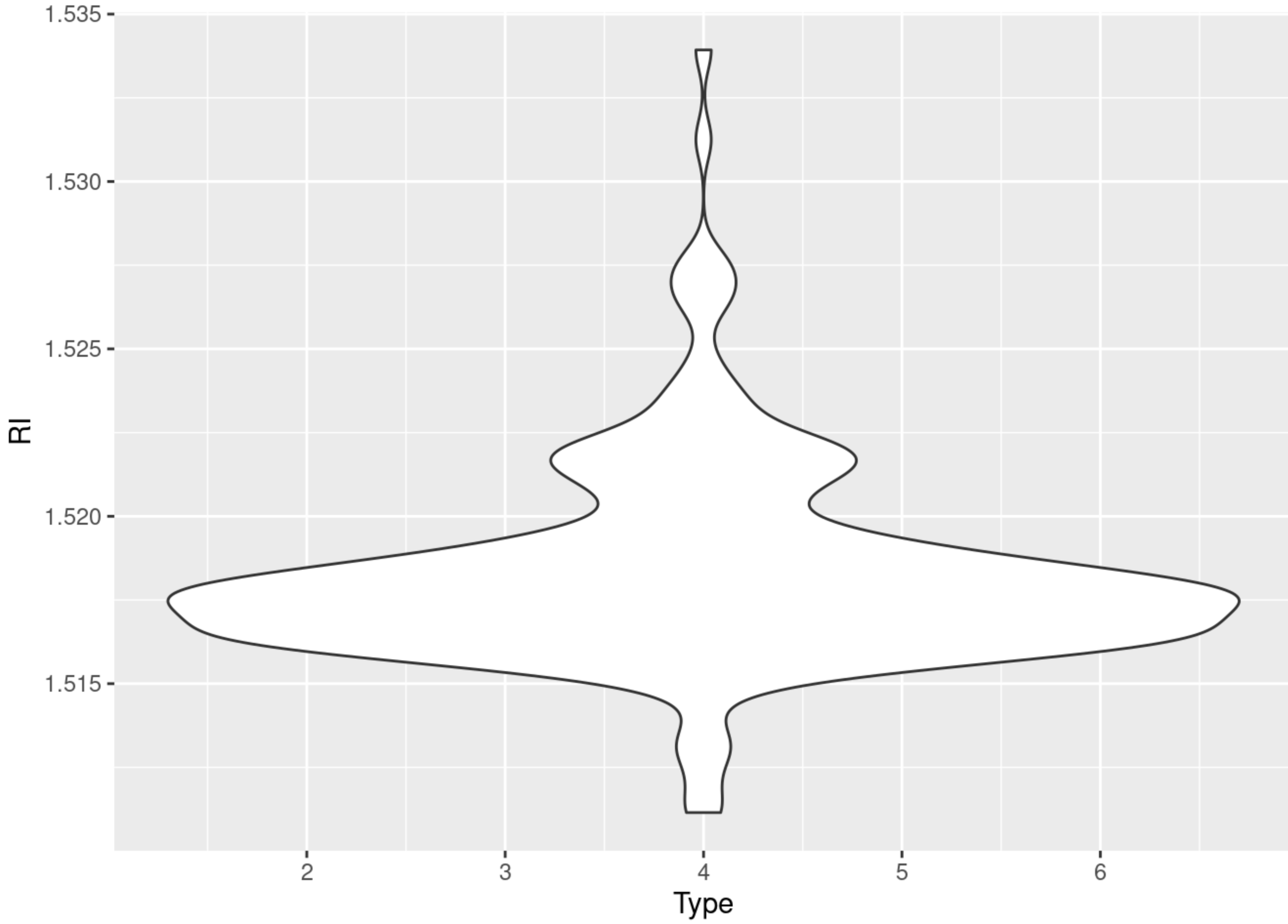
```
##        RI             Type
##  Min.   :1.511   Min.   :1.00
##  1st Qu.:1.517   1st Qu.:1.00
##  Median :1.518   Median :2.00
##  Mean   :1.518   Mean   :2.78
##  3rd Qu.:1.519   3rd Qu.:3.00
##  Max.   :1.534   Max.   :7.00
```

Each violin plot in a single column (stacked)

```
ggplot(df, aes(x=Type, y=RI)) +
  geom_violin() +
  facet_wrap(~Type, scales="free_y", ncol = 1)
```




```
ggplot(df, aes(x=Type, y=RI)) +
  geom_violin() +
  facet_grid(~., scales="free_y")
```



Observations and Inferences

Violin plot of Refractive Index (RI) against Glass Type can provide insights into the distribution and central tendency of RI for different glass types.

Notably, there is substantial overlap among the violin plots for each Glass Type, indicating similarities in the RI distributions. The region between RI values 1.515 and 1.520 exhibits a marked change in the thickness of the violins. In this range, the plot is notably thick, suggesting a higher density of data points or a clustering of observations. This thickened section is followed by a significant thinning of the violins, indicating a potential decrease in the density of data points. Overall, the violin plot provides valuable insights into the distributional characteristics of RI across Glass Types, emphasizing potential subgroupings and concentration patterns within the dataset.

3.3 Subtask 3: Night at the Museum

```
# Loading the dataset
data5 <- read_excel("./2024_Assignment1_BRSM.xlsx",
    sheet = "Museum Visitor")
head(data5)
```

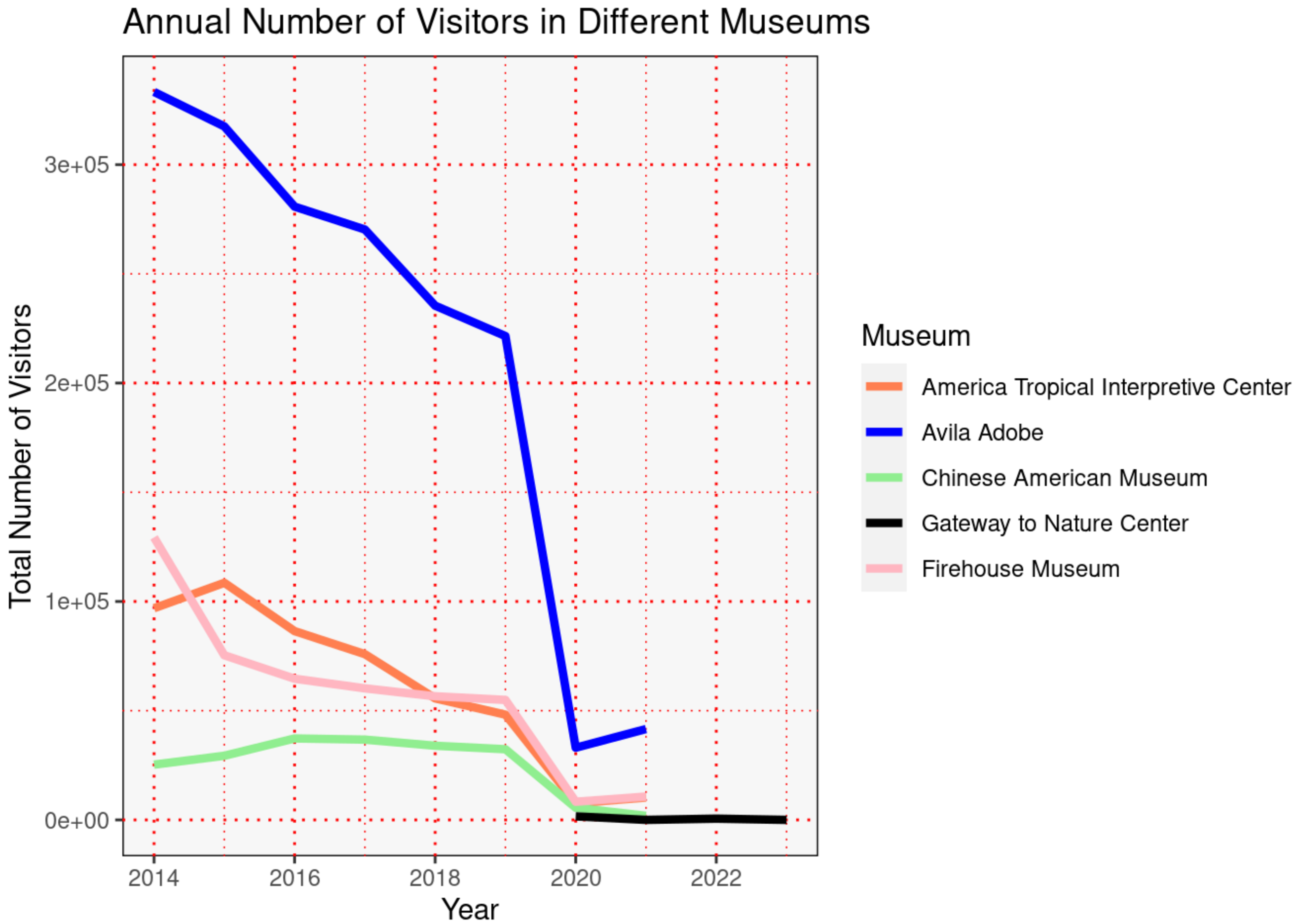
```
## # A tibble: 6 × 6
##   Month    America Tropical Interpretive …¹ `Avila Adobe` Chinese American Mus…²
##   <chr>                               <dbl>         <dbl>                  <dbl>
## 1 Jan 2014                             6602         24778                   1581
## 2 Feb 2014                             5029         18976                   1785
## 3 Mar 2014                             8129         25231                   3229
## 4 Apr 2014                             2824         26989                   2129
## 5 May 2014                            10694         36883                   3676
## 6 Jun 2014                            11036         29487                   2121
## # ℹ abbreviated names: ¹`America Tropical Interpretive Center`,
## #   ²`Chinese American Museum`
## # ℹ 2 more variables: `Gateway to Nature Center` <dbl>,
## #   `Firehouse Museum` <dbl>
```

```
df <- data5
```

```
library(reshape2)
df <- melt(df, id.vars = "Month")
head(df)
```

```
##      Month                             variable value
## 1 Jan 2014 America Tropical Interpretive Center  6602
## 2 Feb 2014 America Tropical Interpretive Center  5029
## 3 Mar 2014 America Tropical Interpretive Center  8129
## 4 Apr 2014 America Tropical Interpretive Center  2824
## 5 May 2014 America Tropical Interpretive Center 10694
## 6 Jun 2014 America Tropical Interpretive Center 11036
```

```r
library(dplyr)
library(lubridate)

# Convert the 'Month' column to a date format
df$Month <- as.Date(paste("01", df$Month), format = "%d %B %Y")

# Extract the year from the 'Month' column
df$Year <- lubridate::year(df$Month)

# Aggregate data by year and museum
df_summarized <- df %>%
  group_by(Year, variable) %>%
  summarize(TotalVisitors = sum(value))

colorLegends <- c("coral", "blue", "lightgreen", "black", "lightpink")
lineThickness <- 1.5

# Plotting the line graph for museums by year
ggplot(df_summarized, aes(x = Year, y = TotalVisitors, color = variable)) +
  geom_line(size = lineThickness) +
  theme(panel.background = element_rect(fill = 'gray97', color = 'black'),
        panel.grid.major = element_line(color = 'red', linetype = 'dotted'),
        panel.grid.minor = element_line(color = 'red', linetype = 'dotted')) +
  labs(title = "Annual Number of Visitors in Different Museums",
       x = "Year",
       y = "Total Number of Visitors",
       color = "Museum") +
  scale_color_manual(values = colorLegends)
```



####

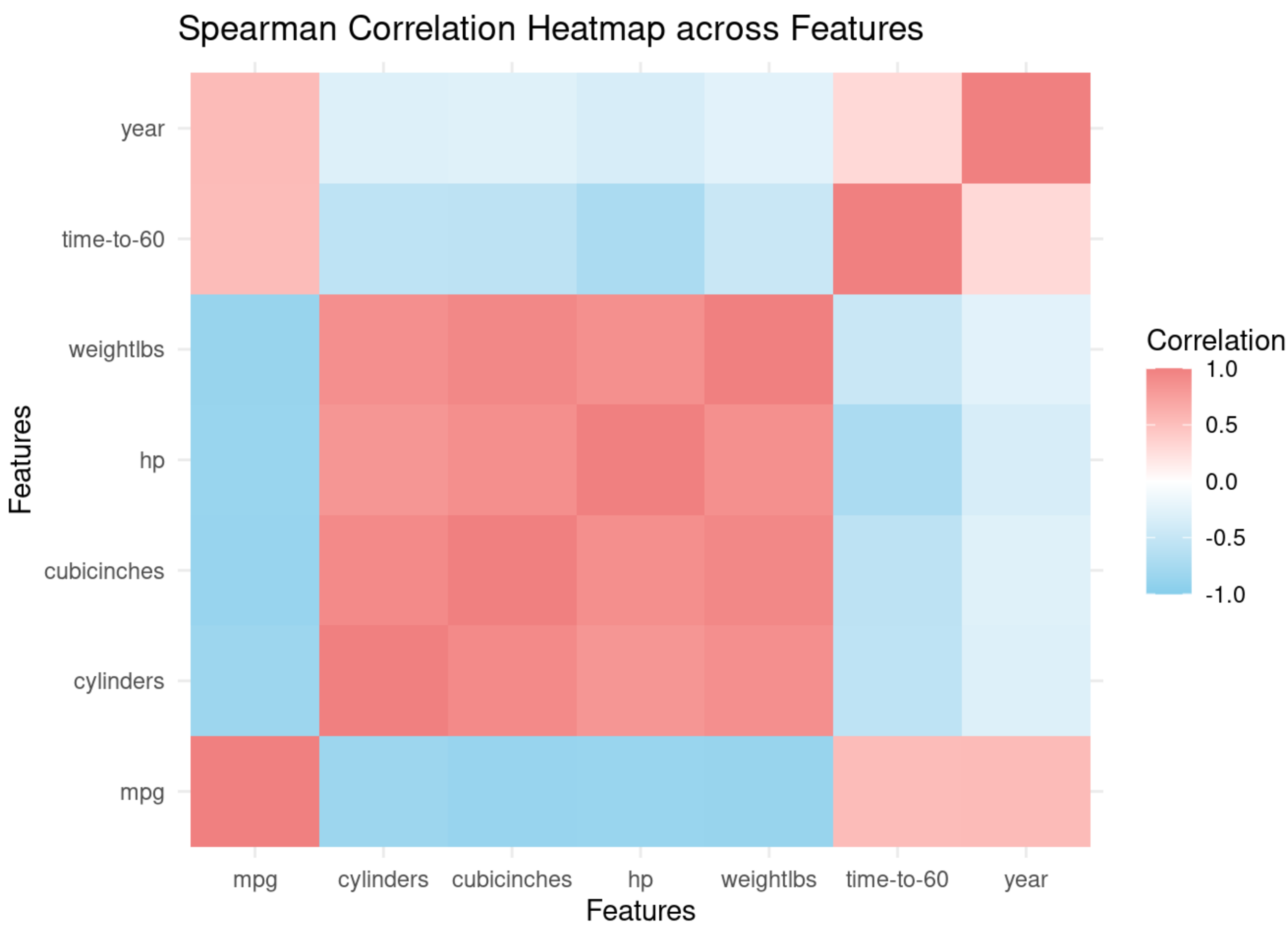
Observations and Inferences A line graph is utilized to demonstrate the evolution of streams over time, with each song represented by a different color. The streams for the museums America Tropical Interpretive Center, Avila Adobe, Chinese American Museum, Gateway to Nature Center and Firehouse Museum experienced a rise and eventual drop in numbers. The data is plotted as individual point connected by lines, which makes it easy to see the overall pattern or trend of the data. Line plots are useful for showing changes in data over time or the relationship between two continuous variables. They are also useful for highlighting overall trends, increases and decreases, and the relative magnitude of changes in the data.

4 Fast and Furious: Heatmap

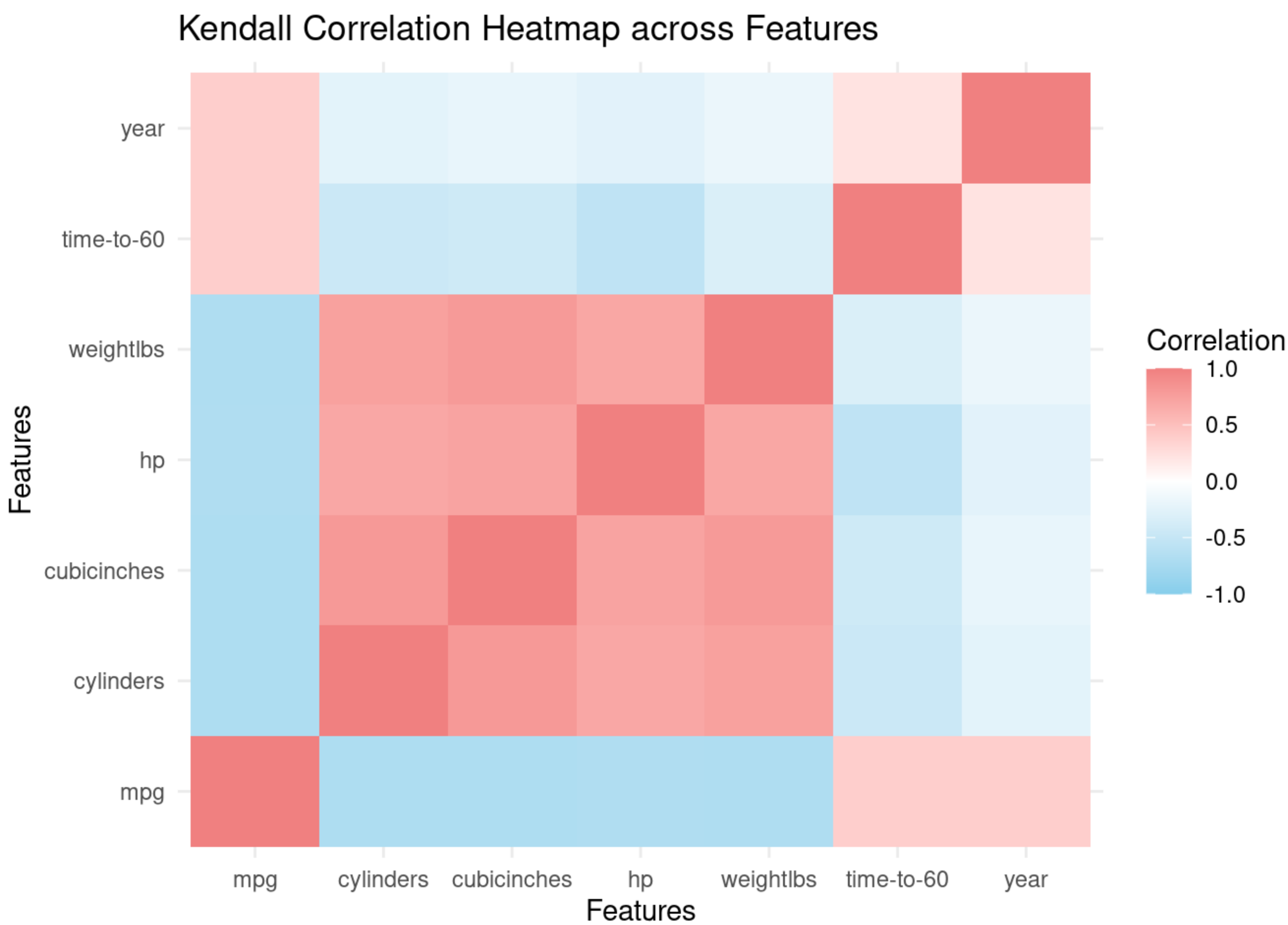
```
# Loading the dataset
data6  <- read_excel("./2024_Assignment1_BRSM.xlsx",
    sheet = "Fast and Furious")
head(data6)
```

```
## # A tibble: 6 × 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.
```

```
# Ensuring all the data is numeric and has no missing values
numericDataFrames <- dplyr::select_if(data6,is.numeric)
df <- data6

numeric_cols <- c("mpg", "cylinders", "cubicinches", "hp", "weightlbs", "time-to-60", "y
ear")

df[, numeric_cols] <- apply(df[, numeric_cols], 2, as.numeric)

df[, numeric_cols] <- lapply(df[, numeric_cols], function(x) ifelse(is.na(x), mean(x, n
a.rm = TRUE), x))

# Create a correlation matrix - Spearman method
corr_matrix_s <- cor(df[, numeric_cols], method = "spearman")

# Plot heatmap using ggplot2
ggplot(data = reshape2::melt(corr_matrix_s), aes(x = Var1, y = Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient2(low = "skyblue", mid = "white", high = "lightcoral", midpoint =
0, limit = c(-1, 1)) +
  theme_minimal() +
  labs(title = "Spearman Correlation Heatmap across Features",
       x = "Features",
       y = "Features",
       fill = "Correlation")
```



```
# Create a correlation matrix - Kendall method
corr_matrix_k <- cor(df[, numeric_cols], method = "kendall")

# Plot heatmap using ggplot2
ggplot(data = reshape2::melt(corr_matrix_k), aes(x = Var1, y = Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient2(low = "skyblue", mid = "white", high = "lightcoral", midpoint =
0, limit = c(-1, 1)) +
  theme_minimal() +
  labs(title = "Kendall Correlation Heatmap across Features",
       x = "Features",
       y = "Features",
       fill = "Correlation")
```



Observations and Inferences

From the heatmap plot, we could infer that there exist a high correlation amongst the following variables (the tiles highlighted in the reddish shade of the color scale) while the tiles highlighted towards the blueish shade show a high negative correlation.

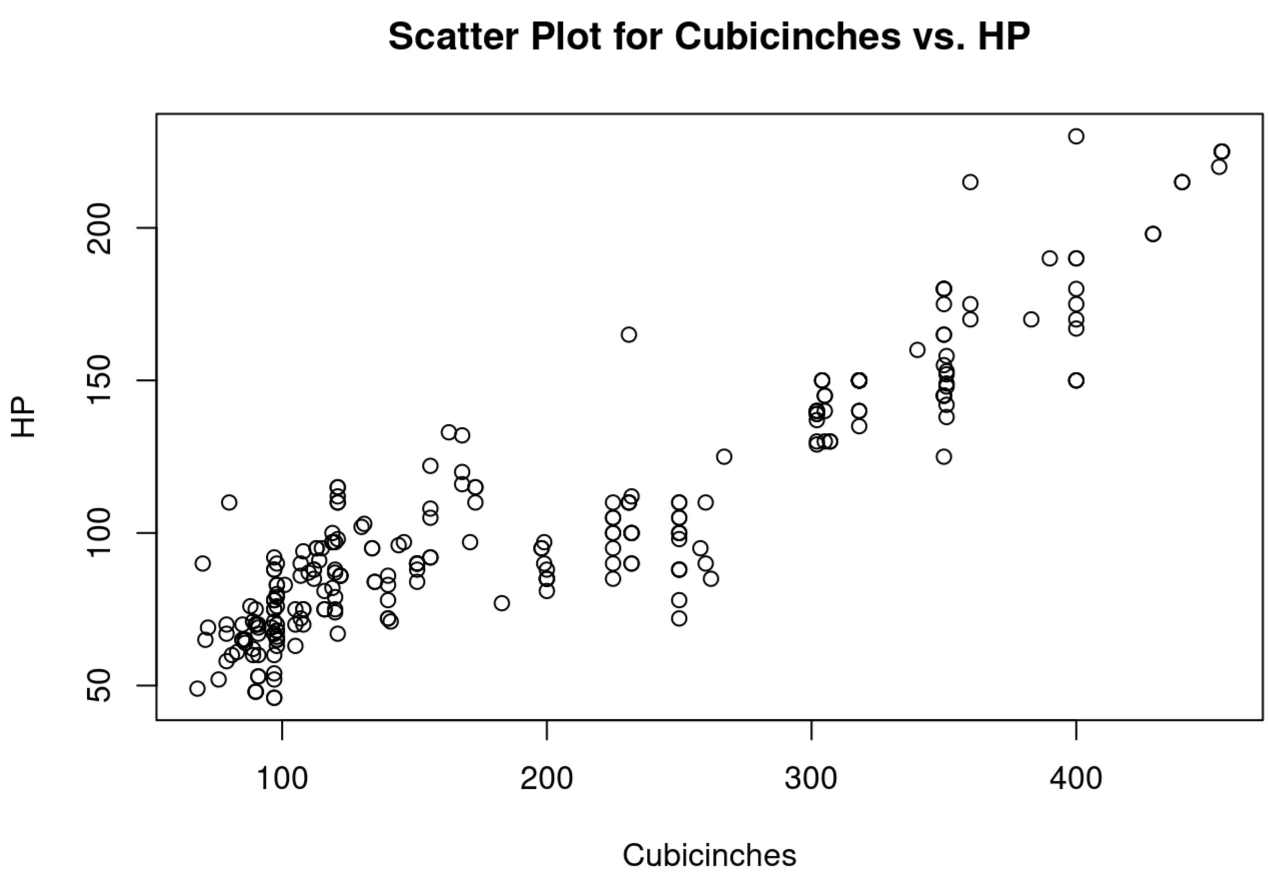
The Kendall and Spearman correlation coefficients are both measures of rank correlation, which means they measure the relationship between variables based on the rank order of the values, rather than the actual values.

The Kendall coefficient measures the number of concordant pairs (pairs where the variables increase or decrease together) minus the number of discordant pairs (pairs where the variables increase or decrease in opposite directions). The Spearman coefficient is based on the difference between the ranks of the values for each variable, and is calculated as the Pearson correlation coefficient for the ranks.

The fact that the Kendall and Spearman coefficients are nearly similar in a heatmap suggests that the relationship between the variables is strong and consistent, regardless of the scale of the variables. This can be useful information for understanding the relationships between variables in the data, and may help to identify patterns or trends that would be difficult to see with other types of data visualizations.

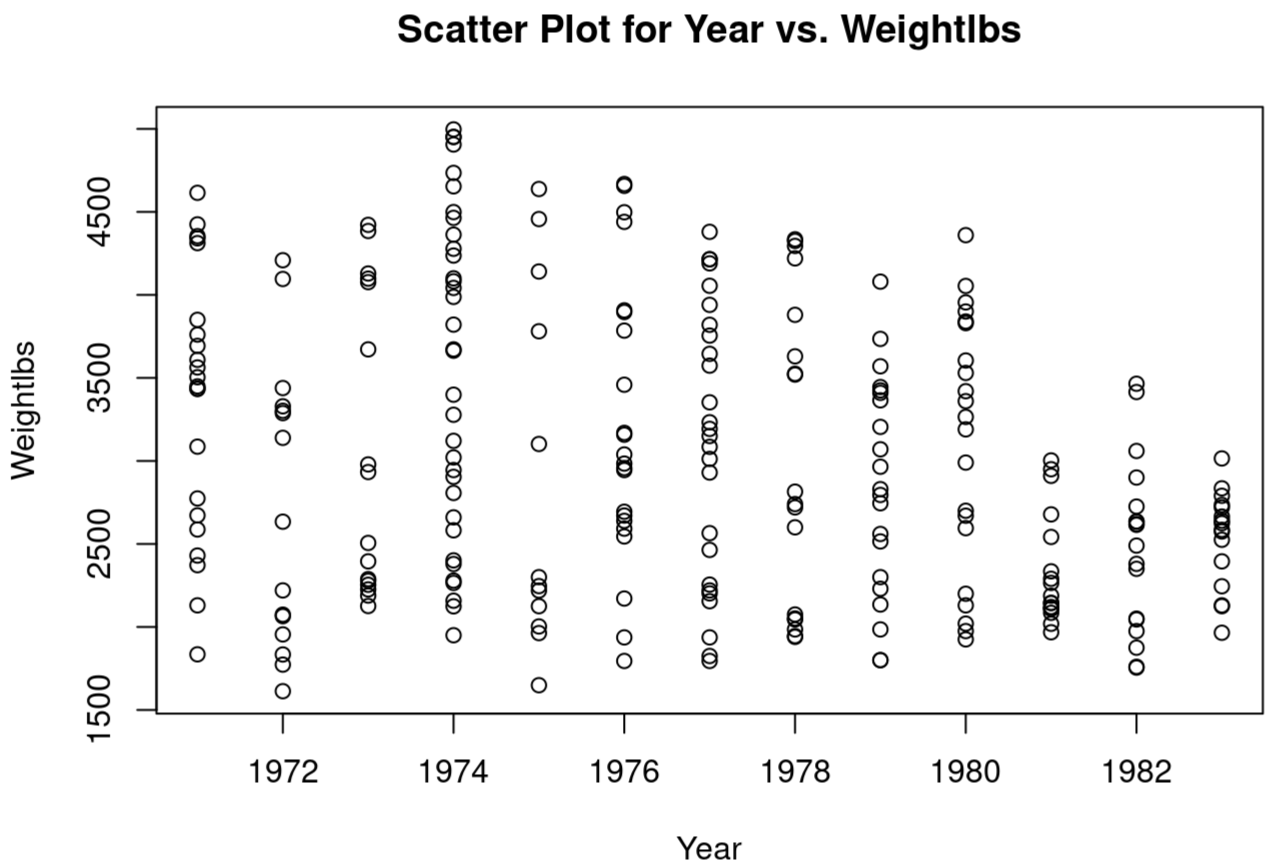
Kendall’s is often better when data doesn’t meet one of the requirements of Pearson’s correlation. Kendall’s is non-parametric meaning that it does not require the two variables to fall into a bell curve. Kendall’s also does not require continuous data. Because it is based on the ranked values of each variable it will work with continuous data, but it can also be used with ordinal data.

```
plot(numericDataFrames$cubicinches, numericDataFrames$hp,
     main = "Scatter Plot for Cubicinches vs. HP",
     xlab = "Cubicinches",
     ylab = "HP")
```



There is quite high correlation among the variables like cubicinches and hp, which is depicted by the scatter plot above.

```
plot(numericDataFrames$year, numericDataFrames$weightlbs,
     main = "Scatter Plot for Year vs. Weightlbs",
     xlab = "Year",
     ylab = "Weightlbs")
```



There is zero to no correlation among the variables like year and weightlbs, which is depicted by the scatter plot above.