Divyansh Tiwari (CSD - 2020111002)

2023-01-23

Statistical Deception

The following is the code for the question - Statistical Deception

```
library(readxl)
library(ggplot2)
library(tidyr)
library(vioplot)
```

```
## Loading required package: sm
```

```
## Package 'sm', version 2.2-5.7: type help(sm) for summary information
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

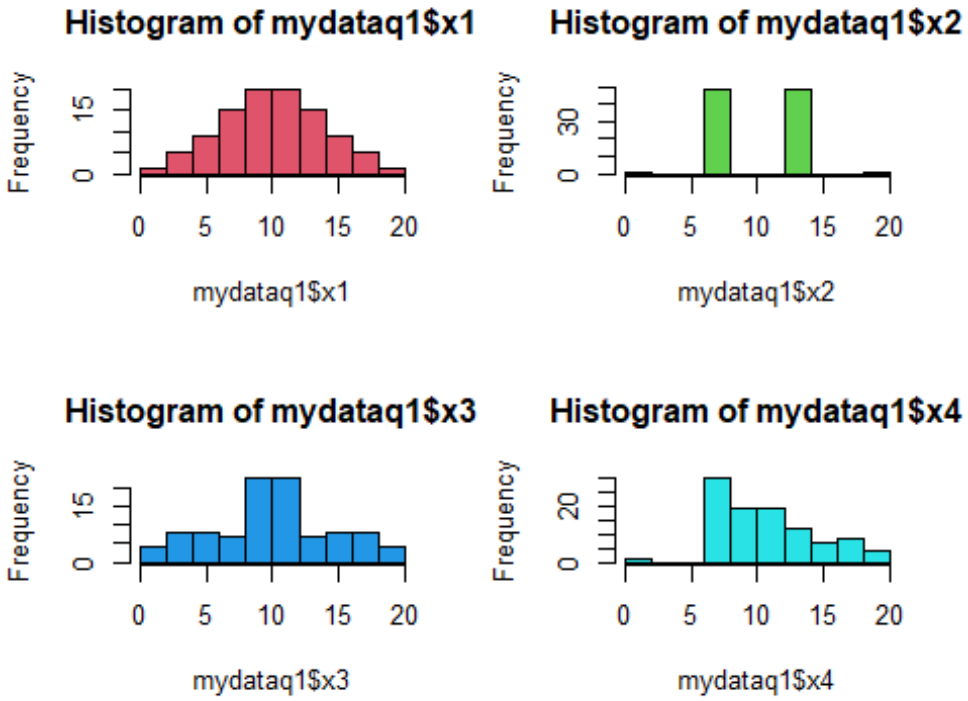
```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
mydataq1 <- read_excel("./Visualisation_Activity.xlsx", 1)


print(mydataq1)
```

```
## # A tibble: 100 × 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  3.17  7.19  1.78  7.40
##  5  3.58  7.21  2.04  7.40
##  6  3.93  7.23  2.31  7.40
##  7  4.23  7.24  2.57  7.40
##  8  4.51  7.26  2.83  7.40
##  9  4.76  7.27  3.09  7.40
## 10  4.99  7.28  3.35  7.40
## # … with 90 more rows
```

The following are the histogram plots

```
par(mfrow =c(2,2))
hist(mydataq1$x1, col=2)
hist(mydataq1$x2, col=3)
hist(mydataq1$x3, col=4)
hist(mydataq1$x4, col=5)
```

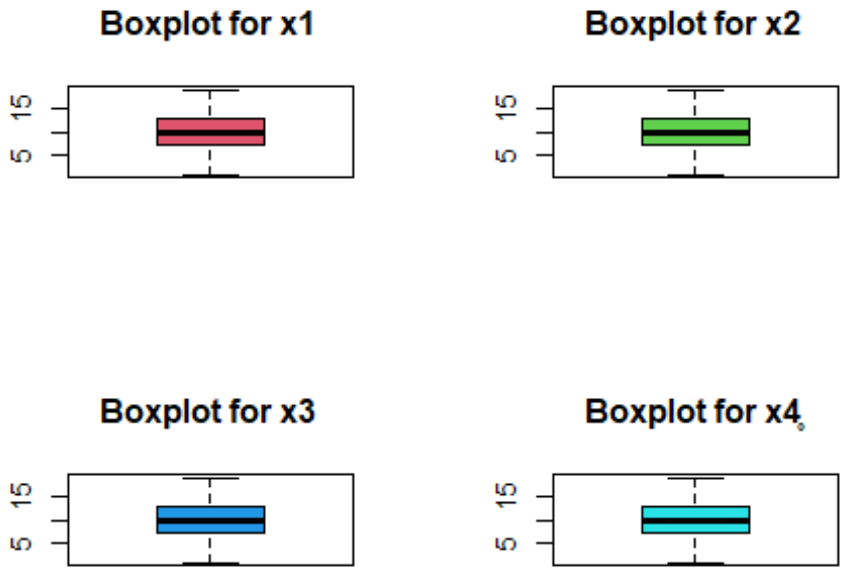


From the histogram plots, we can see for ourselves that the data distribution is varied and it shares information with us that the mean of the data is approximately equal, however the data distribution is quite varied.

Now, we shall take a look at the box plots.

The following are the box plots.

```
par(mfrow =c(2,2))
boxplot(mydataq1$x1, col=2, main="Boxplot for x1")
boxplot(mydataq1$x2, col=3, main="Boxplot for x2")
boxplot(mydataq1$x3, col=4, main="Boxplot for x3")
boxplot(mydataq1$x4, col=5, main="Boxplot for x4")
```

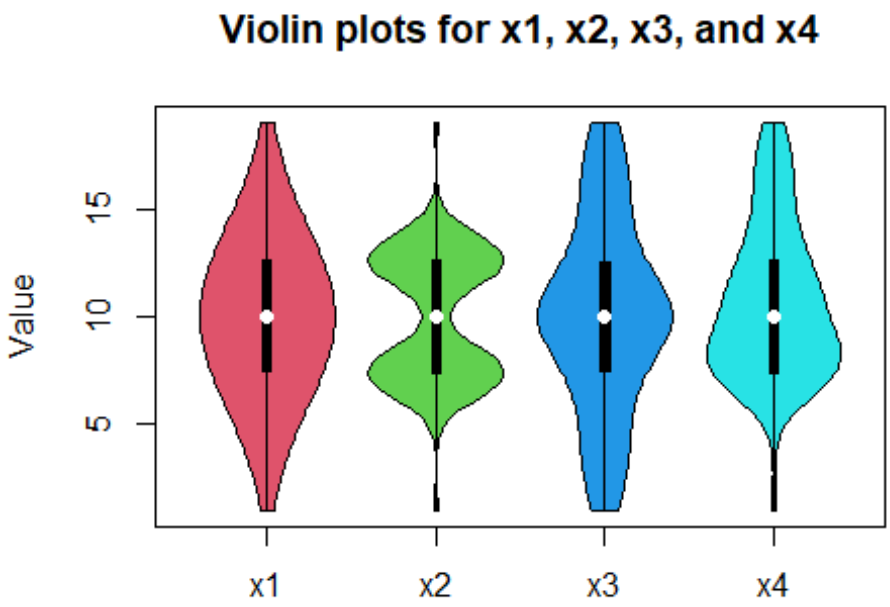


Through the box plots, we can see for ourselves that no information about the data can be gained except that the box plots are exactly the same which suggests that the data distribution is same which is misleading.

Now, we shall plot the violin plot which supposedly will be the bets plot for this data.

The following is the violin plot for the data.

```
vioplot(mydataq1, ylab="Value",  col=2:5, main="Violin plots for x1, x2, x3,
and x4")
```



From the violin plot given above, we can clearly gain a lot of information about the data. We can observe that the plot shows that the data distribution is different for each of the columns of the data and the median for the is the same as well. The difference between the data is in the why that it is being distributed. For x1, the data is distributed in a normal-like fashion, where as the distribution of other columns is different. One more thing to note is that the data distribution pattern is similar in the 1st and 3rd quartiles for the first 3 columns of the data.

Since, the violin plot helps us gain so much information for this data set, it is one of the best visualisations that can be done for this data.

Also, the worst visualisation for this data that is misleading is the boxplot as discussed above.

Personality and Motion

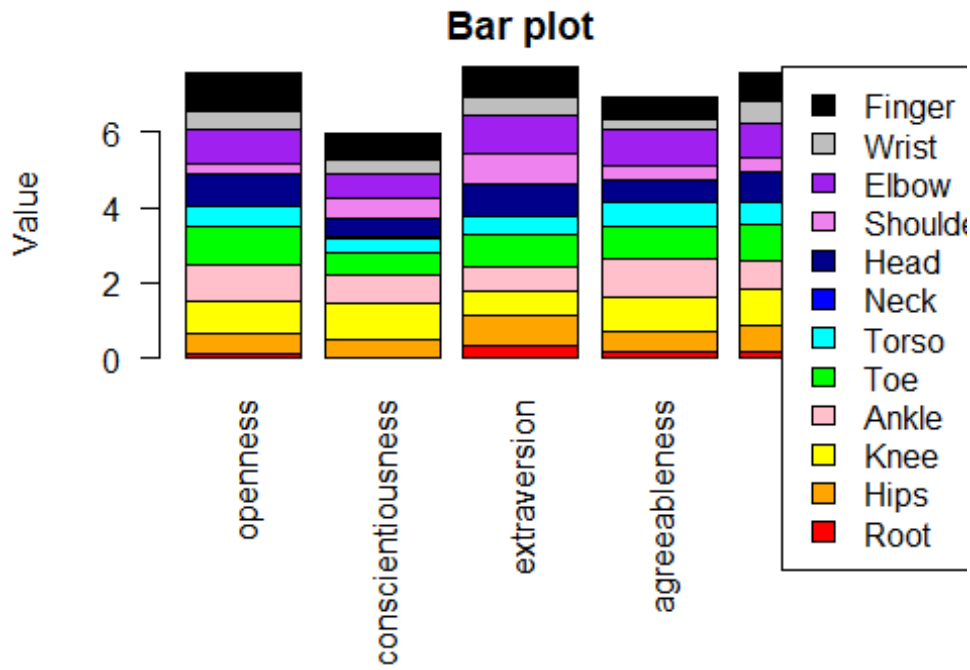
The following is the code for the question - Personality and Motion

```
library(fmsb)

mydataq2 <- read_excel("./Visualisation_Activity.xlsx", 2)

print(mydataq2)
```

```
## # A tibble: 12 × 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
##  7 Neck         0                 0.0576       0             0           0
##  8 Head         0.838             0.503        0.840         0.556
0.798
##  9 Shoulder     0.319             0.541        0.845         0.418
0.348
## 10 Elbow        0.861             0.614        1             0.941
0.902
## 11 Wrist        0.506             0.404        0.477         0.268
0.627
## 12 Finger       1                 0.708        0.826         0.574
0.757
```

Given below is the stacked bar plot to analyze which Joint is important to assess the personality traits.

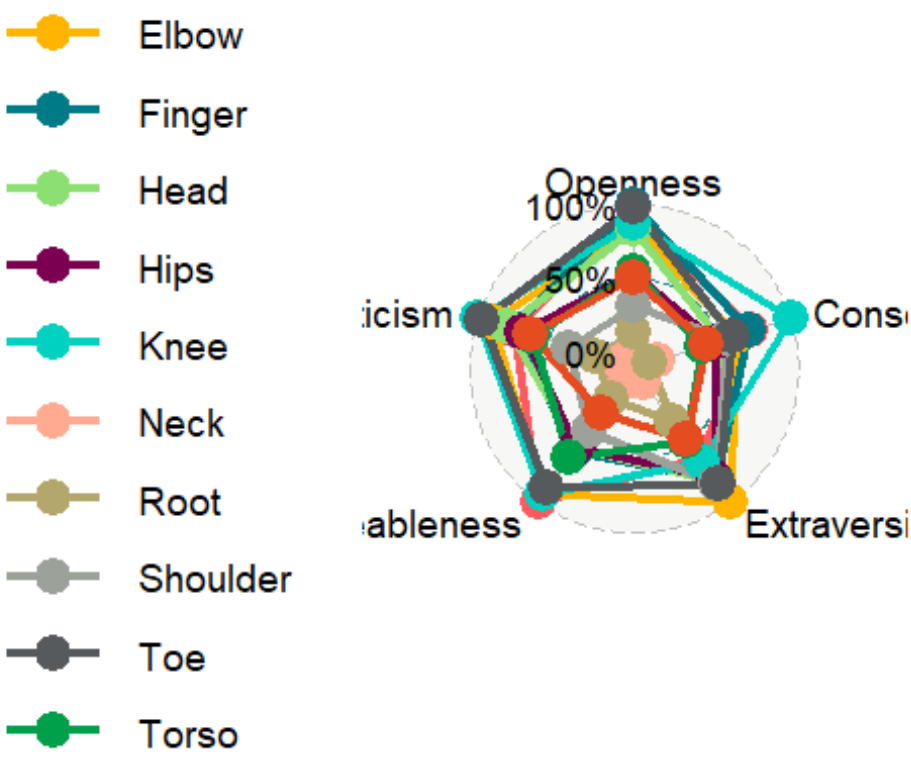
```
openness <- matrix(mydataq2$Openness)
conscientiousness <- matrix(mydataq2$Conscientiousness)
extraversion <- matrix(mydataq2$Extraversion)
agreeableness <- matrix(mydataq2$Agreeableness)
neuroticism <- matrix(mydataq2$Neuroticism)

here <- data.frame(openness, conscientiousness, extraversion, agreeableness,
neuroticism)

par(mar = c(10, 4, 2, 2) + 0.2)

barplot(as.matrix(here) , main = "Bar plot", ylab="Value", col=c("Red",
"Orange", "Yellow", "Pink", "Green", "Cyan", "Blue", "Darkblue", "Violet",
"Purple", "Grey", "Black"),  xpd = TRUE, legend = c("Root", "Hips", "Knee",
"Ankle", "Toe", "Torso", "Neck", "Head", "Shoulder", "Elbow", "Wrist",
"Finger"), args.legend = list(x = "topright", inset = c(- 0.17, 0)), pch =
15, beside=FALSE, las=2)
```



The graph given above shows us the ratio of the importance of the joint while assessing the personality traits. However, the ratios are not visile quite well and hence this graph does not provide appropriate visualisation.

Now, we will be looking at the radar plot for the same.

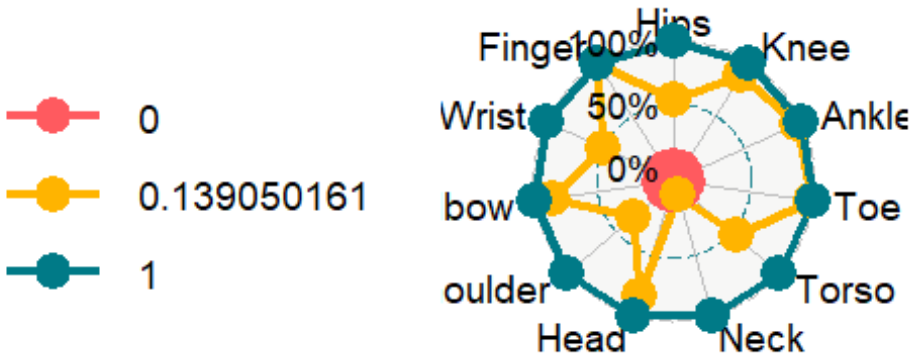
```
library(ggradar)
ggradar(mydataq2)
```



The above graph is a radar plot and could very well depict the influence of the Joint Importance values for personality traits. However, this graph is quite clumsy and hence we will try to clean it a bit by dividing the plot on the basis of each personality trait.

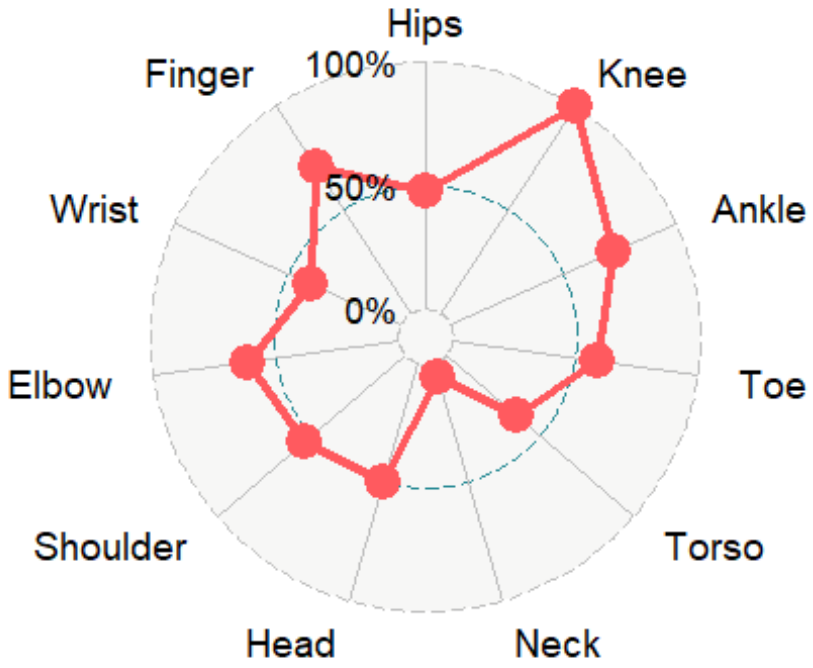
The following plot is for personality trait - Openness

```
data1 <- data.frame(rbind(rep(1, 12), rep(0, 12),
t(matrix(mydataq2$Openness))))
colnames(data1) <- c("Root", "Hips", "Knee", "Ankle", "Toe", "Torso", "Neck",
"Head", "Shoulder", "Elbow", "Wrist", "Finger")

ggradar(data1)
```



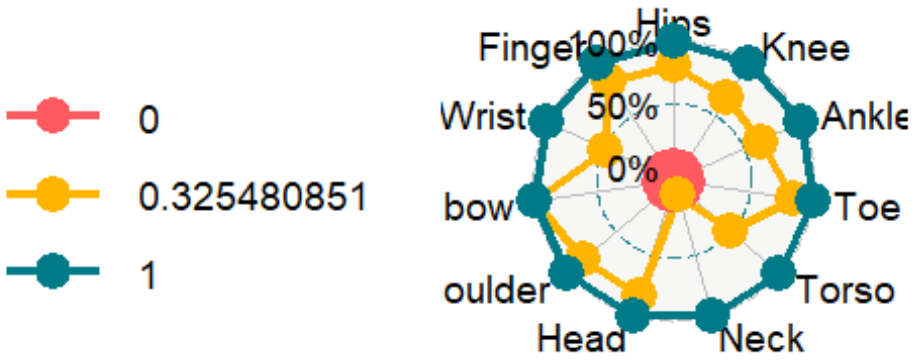
The following plot is for personality trait - Conscientiousness

```
print(mydataq2$Conscientiousness)

##  [1] 0.00000000 0.47739811 1.00000000 0.72320515 0.59016029 0.37263471
##  [7] 0.05760356 0.50263242 0.54054434 0.61371081 0.40417715 0.70801986

data2 <- data.frame(rbind(rep(1, 12), rep(0, 12),
t(matrix(mydataq2$Conscientiousness))))
colnames(data2) <- c("Root", "Hips", "Knee", "Ankle", "Toe", "Torso", "Neck",
"Head", "Shoulder", "Elbow", "Wrist", "Finger")
# print(data2)
# ggradar(data2)

data2 <-t (mydataq2$Conscientiousness)
colnames(data2) <- c("Root", "Hips", "Knee", "Ankle", "Toe", "Torso", "Neck",
"Head", "Shoulder", "Elbow", "Wrist", "Finger")

plt <- data2 %>% ggradar()
plt
```



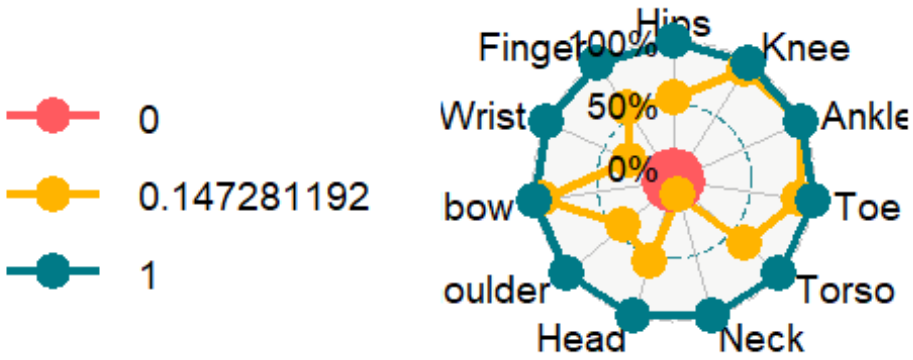
The following plot is for personality trait - Extraversion

```
print(mydataq2$Extraversion)

##  [1] 0.3254809 0.8042766 0.6621905 0.6387358 0.8512893 0.4899196 0.0000000
##  [8] 0.8395976 0.8449116 1.0000000 0.4770724 0.8255081

data3 <- data.frame(rbind(rep(1, 12), rep(0, 12),
t(matrix(mydataq2$Extraversion))))
colnames(data3) <- c("Root", "Hips", "Knee", "Ankle", "Toe", "Torso", "Neck",
"Head", "Shoulder", "Elbow", "Wrist", "Finger")

ggradar(data3)
```



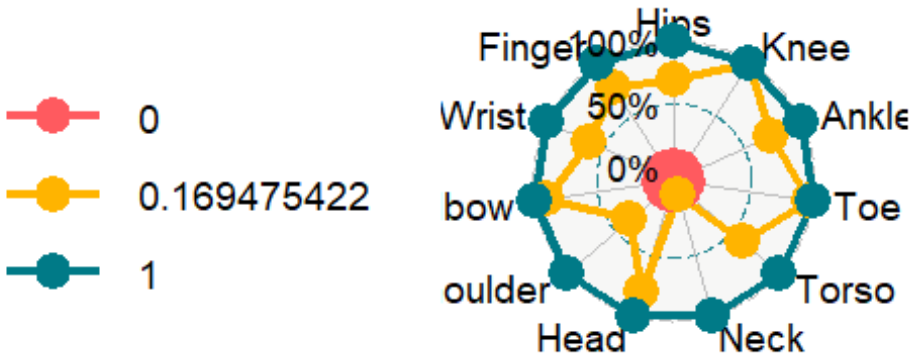
The following plot is for personality trait - Agreeableness

```
data4 <- data.frame(rbind(rep(1, 12), rep(0, 12),
t(matrix(mydataq2$Agreeableness))))
colnames(data4) <- c("Root", "Hips", "Knee", "Ankle", "Toe", "Torso", "Neck",
"Head", "Shoulder", "Elbow", "Wrist", "Finger")

ggradar(data4)
```



The following plot is for personality trait - Neuroticism

```
data5 <- data.frame(rbind(rep(1, 12), rep(0, 12),
t(matrix(mydataq2$Neuroticism))))
colnames(data5) <- c("Root", "Hips", "Knee", "Ankle", "Toe", "Torso", "Neck",
"Head", "Shoulder", "Elbow", "Wrist", "Finger")

ggradar(data5)
```



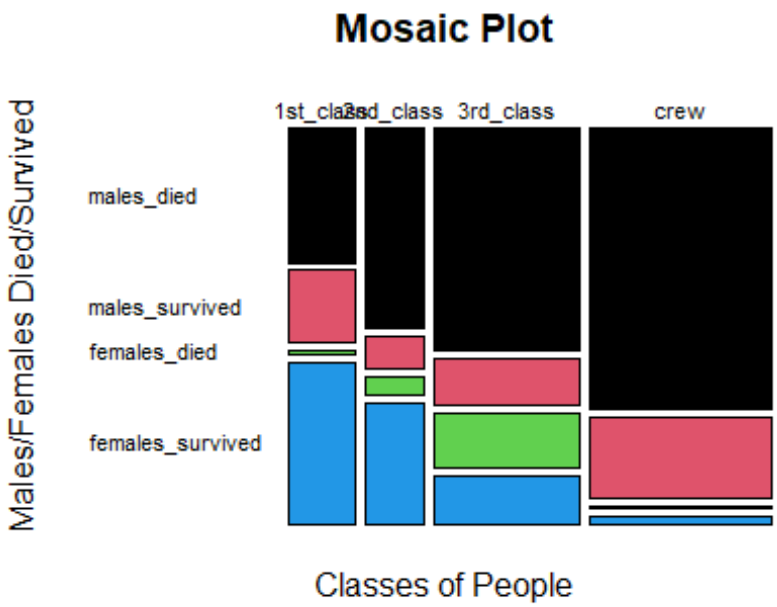
From the radar plots given above, we can clearly see the ratio of the importance of the joint in predicting the personality traits and hence this is a better visualisation technique for this question

Data Plotting Adventure

The following is the plot for the subtask 4.1 (Sinking Ship)

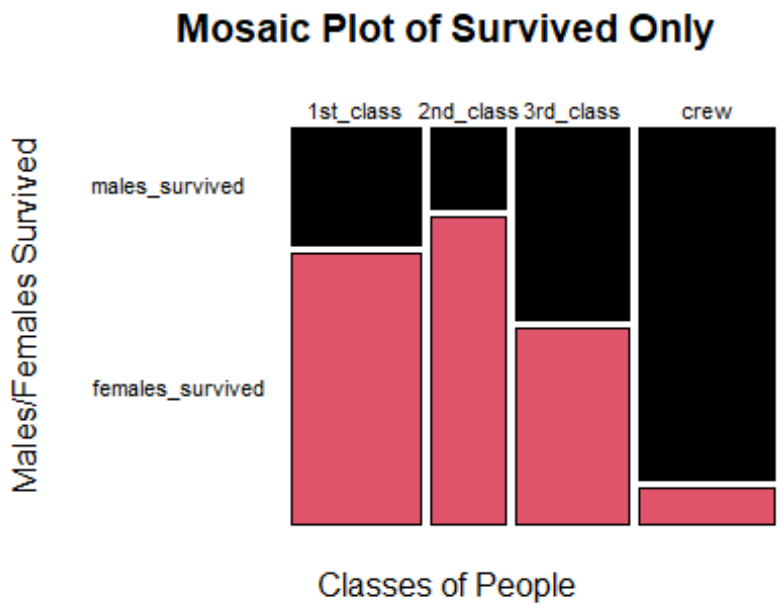
```
data = matrix(c(118, 62, 4, 141, 154, 25, 13, 93, 422, 88, 106, 90, 670, 192,
3, 20), ncol = 4, byrow = TRUE)

rownames(data) <- c('1st_class', '2nd_class', '3rd_class', 'crew')
colnames(data) <- c('males_died', 'males_survived', 'females_died',
'females_survived')

print(data)

##           males_died males_survived females_died females_survived
## 1st_class        118             62            4               141
## 2nd_class        154             25           13                93
## 3rd_class        422             88          106                90
## crew             670            192            3                20

windows.options(width=20, height=20, reset=FALSE)
```

```
mosaicplot(data, color=1:4, las=1, legend=TRUE, xlab="Classes of People",
ylab="Males/Females Died/Survived", main = "Mosaic Plot")

## Warning: In mosaicplot.default(data, color = 1:4, las = 1, legend = TRUE,
##     xlab = "Classes of People", ylab = "Males/Females Died/Survived",
##     main = "Mosaic Plot") :
##  extra argument 'legend' will be disregarded
```




```
data = matrix(c(62, 141, 25, 93, 88, 90,192, 20), ncol = 2, byrow = TRUE)

rownames(data) <- c('1st_class', '2nd_class', '3rd_class', 'crew')
colnames(data) <- c('males_survived', 'females_survived')

print(data)

##           males_survived females_survived
## 1st_class             62              141
## 2nd_class             25               93
## 3rd_class             88               90
## crew                 192               20

windows.options(width=20, height=20, reset=FALSE)

mosaicplot(data, color=1:4, las=1, legend=TRUE, xlab="Classes of People",
ylab="Males/Females Survived", main = "Mosaic Plot of Survived Only")
```

```
## Warning: In mosaicplot.default(data, color = 1:4, las = 1, legend = TRUE,
##     xlab = "Classes of People", ylab = "Males/Females Survived",
##     main = "Mosaic Plot of Survived Only") :
##  extra argument 'legend' will be disregarded
```



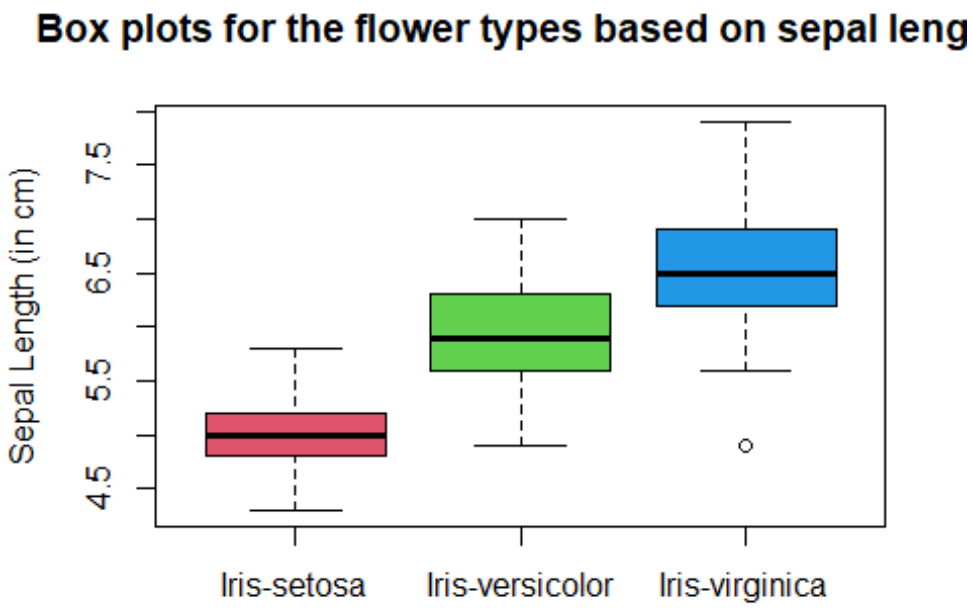
The following is the subtask 2: Petal Prediction

```
mydataq3 <- read_excel("./Visualisation_Activity.xlsx", 3)
print(mydataq3)
```

```
## # A tibble: 150 × 6
##       Id SepalLengthCm SepalWidthCm PetalLengthCm PetalWidthCm Species
##    <dbl>         <dbl>        <dbl>         <dbl>        <dbl> <chr>
##  1     1           5.1          3.5           1.4          0.2 Iris-setosa
##  2     2           4.9          3             1.4          0.2 Iris-setosa
##  3     3           4.7          3.2           1.3          0.2 Iris-setosa
##  4     4           4.6          3.1           1.5          0.2 Iris-setosa
##  5     5           5            3.6           1.4          0.2 Iris-setosa
##  6     6           5.4          3.9           1.7          0.4 Iris-setosa
##  7     7           4.6          3.4           1.4          0.3 Iris-setosa
##  8     8           5            3.4           1.5          0.2 Iris-setosa
##  9     9           4.4          2.9           1.4          0.2 Iris-setosa
## 10    10           4.9          3.1           1.5          0.1 Iris-setosa
## # … with 140 more rows
```

```
subset1 <- (subset(mydataq3, Species=="Iris-setosa", select="SepalLengthCm"))
print(subset1)
```

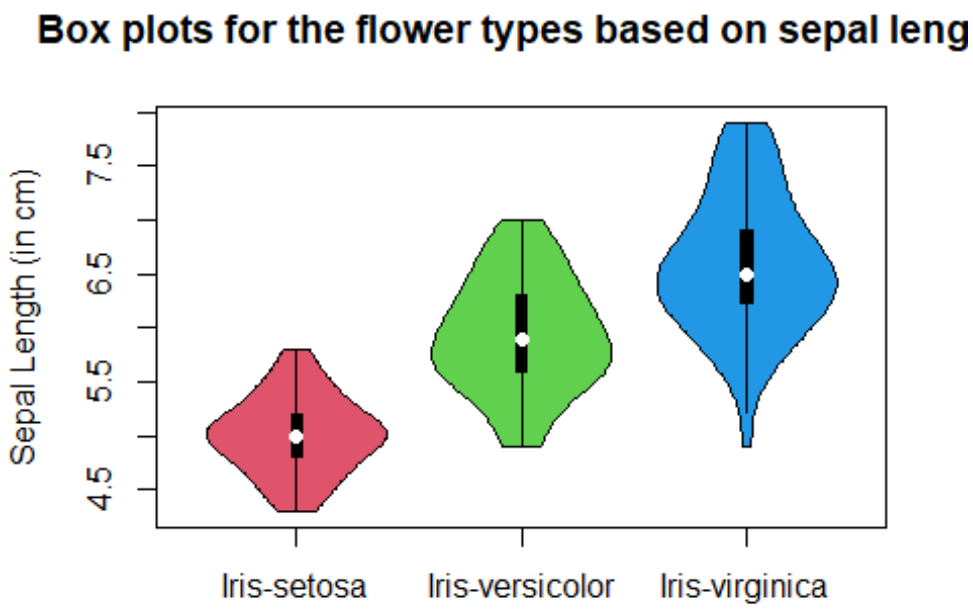
```
## # A tibble: 50 × 1
##    SepalLengthCm
##            <dbl>
##  1           5.1
##  2           4.9
##  3           4.7
##  4           4.6
##  5           5
##  6           5.4
##  7           4.6
##  8           5
##  9           4.4
## 10           4.9
## # … with 40 more rows

colnames(subset1) <- "Iris-setosa"


subset2 <- (subset(mydataq3, Species=="Iris-versicolor",
select="SepalLengthCm"))
colnames(subset2) <- "Iris-versicolor"
subset3 <- (subset(mydataq3, Species=="Iris-virginica",
select="SepalLengthCm"))
colnames(subset3) <- "Iris-virginica"
boxplot(c(subset1, subset2, subset3), ylab="Sepal Length (in cm)", main="Box
plots for the flower types based on sepal length", col=2:4)
```



The above plot is highly useful when we intend to understand the relationship between the sepal length and the flower species as it shows the median sepal length for the flower species and the range of the sepal length along with the quartiles depicting the spread of the data.

We can also look at the violin plot for the same as shown below for understanding the relationship between the sepal length in cm and flower species.

```
vioplot(c(subset1, subset2, subset3), ylab="Sepal Length (in cm)", main="Box
plots for the flower types based on sepal length", col=2:4)
```



The following is the code for the subtask 3: Spotify Wrapped

```
mydataq4 <- read_excel("./Visualisation_Activity.xlsx", 4)

songs_df = mydataq4[, 2:6]
dates <- as.Date(mydataq4$Date)

plt <- ggplot(mydataq4, aes(dates)) +
  geom_line(aes(y = Shape.of.You), col=2) +
  geom_line(aes(y = Despacito), col=3) +
  geom_line(aes(y = Something.Just.Like.This), col=4) +
  geom_line(aes(y = HUMBLE.), col=5) +
  geom_line(aes(y = Unforgettable), col=6)

plt + ggtitle("Spotify Wrapped") + xlab("Dates: Jan 2017 to Jan 2018") +
ylab("Streaming Frequency") + scale_fill_discrete(labels = c("Shape of You",
```

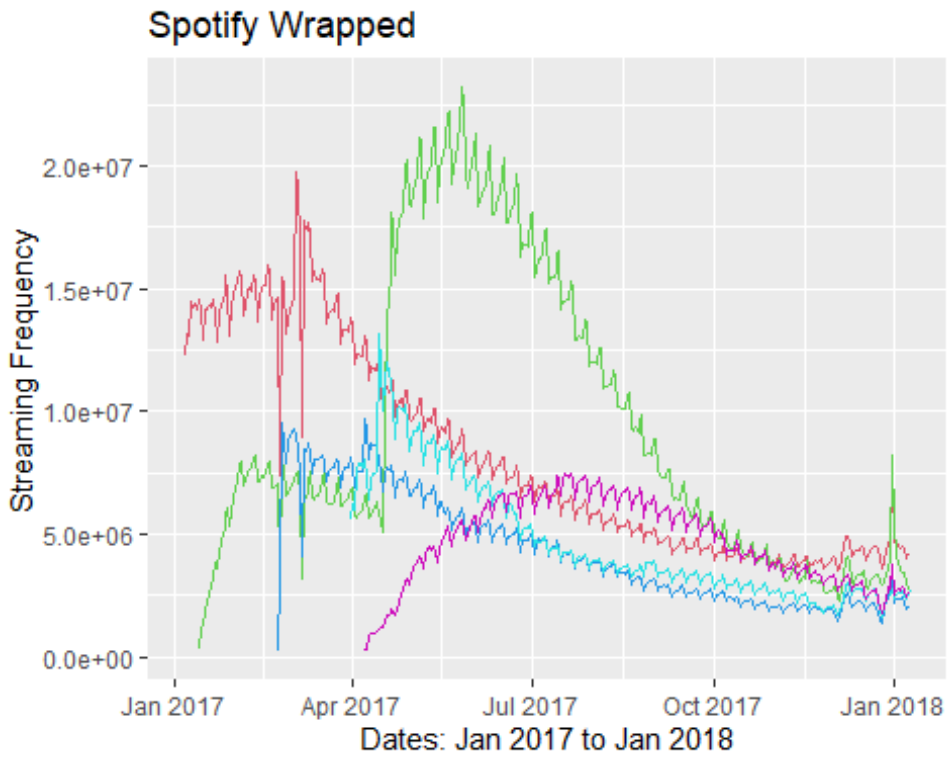
```
"Despacito", "Something Just Like This", "Humble", "Unforgettable")) +
theme(legend.position="bottom")
```

```
## Warning: Removed 7 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 47 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 84 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 91 rows containing missing values (`geom_line()`).
```



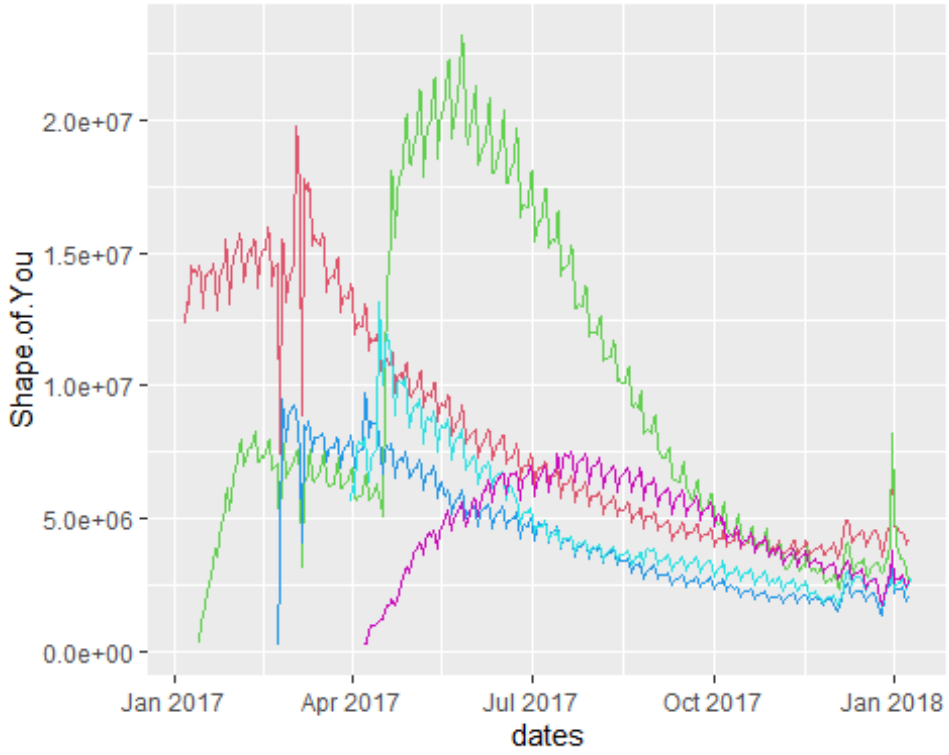

```
plt
```

```
## Warning: Removed 7 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 47 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 84 rows containing missing values (`geom_line()`).
```

```
## Warning: Removed 91 rows containing missing values (`geom_line()`).
```



Please note that the following is the legend for the graph given above: RED = Shape of you, GREEN = Despacito, BLUE = Something just like this, CYAN = Humble, VIOLET = Unforgettable

NEED FOR SPEED: HEATMAP

The following is the task 5 (Need For Speed: Heatmaps)

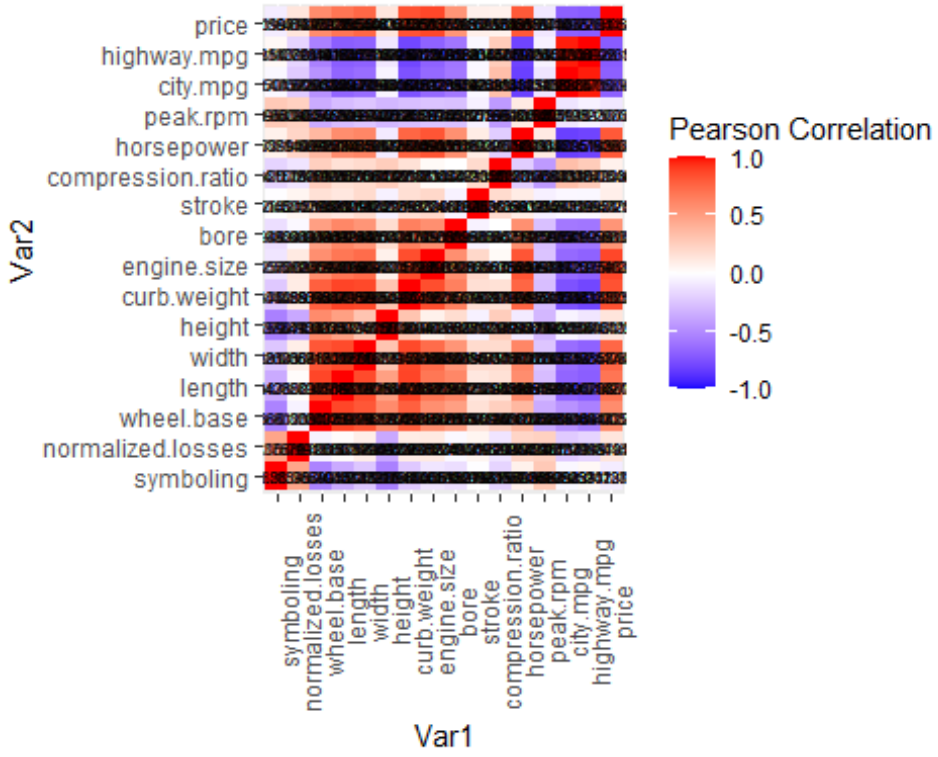
Given below are the steps to perform before plotting the heatmaps

```
library(reshape2)

##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##     smiths

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

mydataq5 <- read_excel("./Visualisation_Activity.xlsx", 5)
mydataq5_here <- select_if(mydataq5, is.numeric)

mat1 <- melt(cor(mydataq5_here, method="pearson"))
mat2 <- melt(cor(mydataq5_here, method="spearman"))
mat3 <- melt(cor(mydataq5_here, method="kendall"))
```

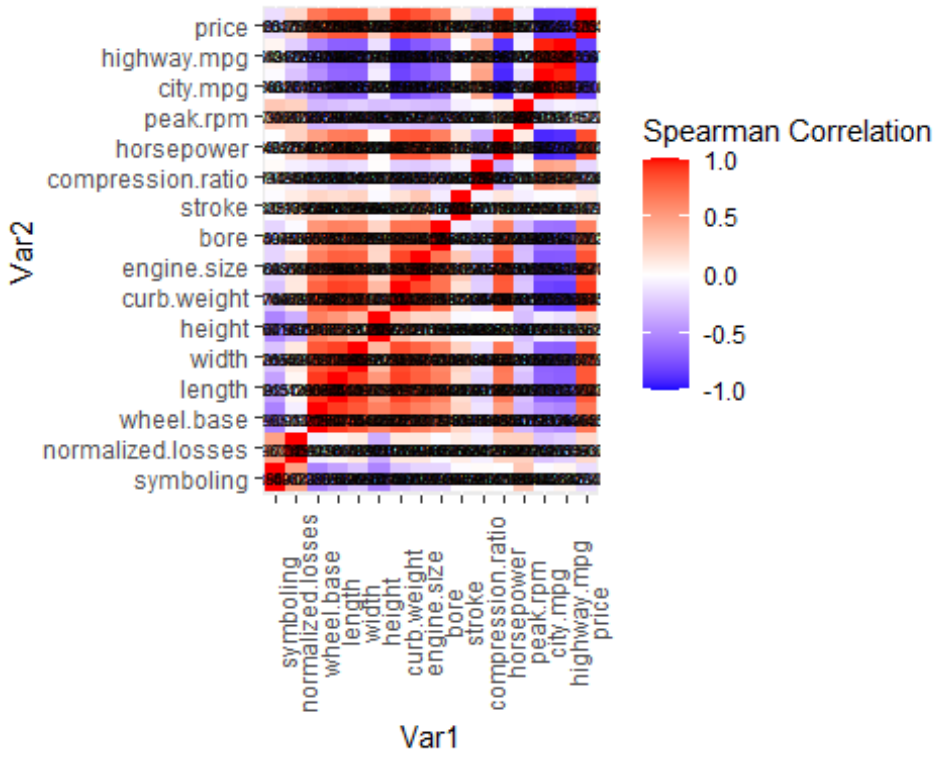
Now, the following is the heat map when the correlation matrix is chosen to be Pearson's.

```
ggplot(data=mat1, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile() +
  geom_text(aes(label = value), size = 2) +
  scale_fill_gradient2(low = "blue", high="red", limit = c(-1, 1),
name="Pearson Correlation") +
  theme(axis.text.x = element_text(angle = 90))
```



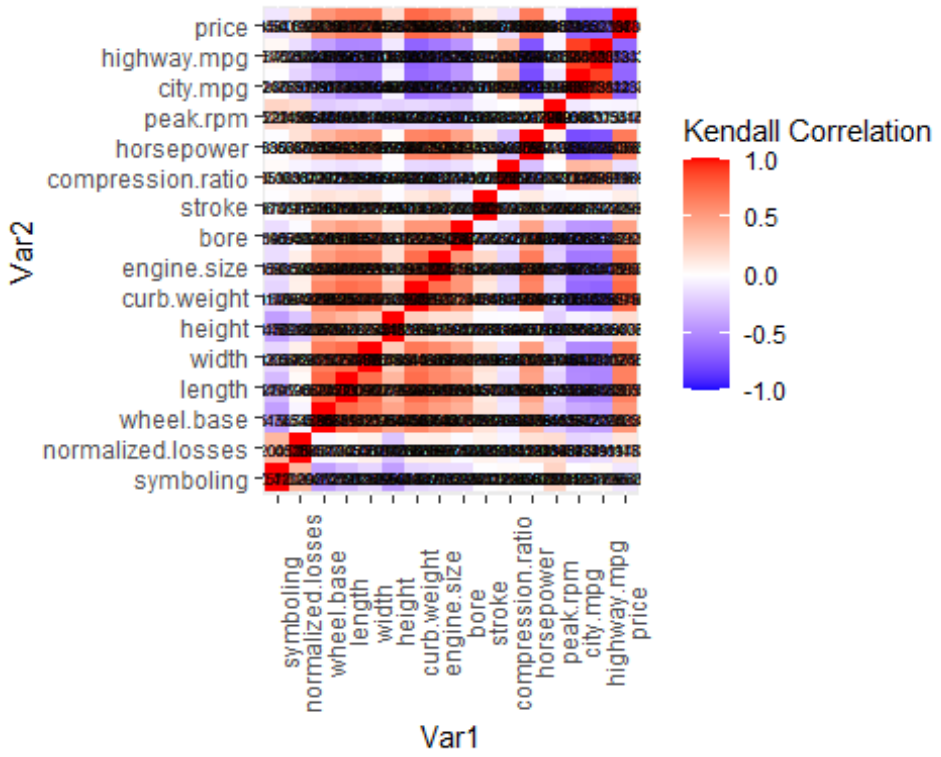
Now, the following is the heat map when the correlation matrix is chosen to be Spearman's.

```
ggplot(data=mat2, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile() +
  geom_text(aes(label = value), size = 2) +
  scale_fill_gradient2(low = "blue", high="red", limit = c(-1, 1),
name="Spearman Correlation") +
  theme(axis.text.x = element_text(angle = 90))
```



Now, the following is the heat map when the correlation matrix is chosen to be Kendall's.

```
ggplot(data=mat3, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile() +
  geom_text(aes(label = value), size = 2) +
  scale_fill_gradient2(low = "blue", high="red", limit = c(-1, 1),
name="Kendall Correlation") +
  theme(axis.text.x = element_text(angle = 90))
```



From the heat maps plotted above, we can see for ourselves that the heat map plot using the Spearman correlation has cells with higher correlational value. Therefore, out of the three matrices, we choose the Spearman Matrix.

We can infer from the heat map that the variables such as curb weight and engine size are highly correlated. Similarly, we can study the heat map to find more such highly correlated variables to get a better knowledge about the data set.