# Visualisation Assignment

Akash C R, 2020111004

2023-02-05

## STATISTICAL DECEPTION :

```
library("xlsx")

# Reading sheet = "statistical deception" from the excel
qn1 = read.xlsx(file = "Visualisation_Activity.xlsx", sheetIndex = 1)
```

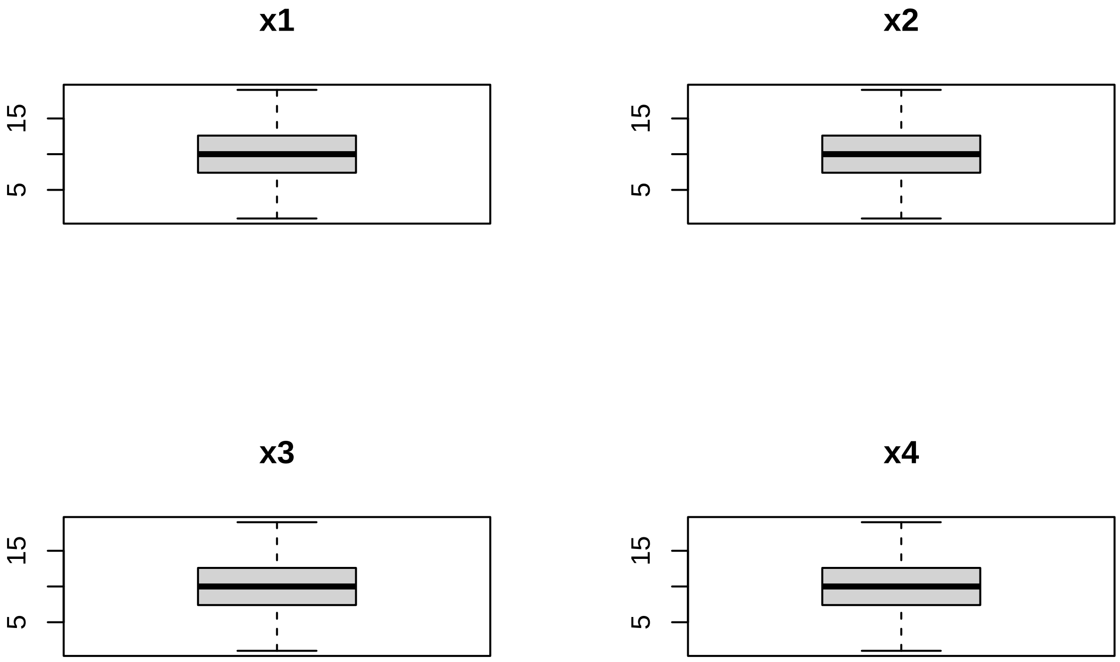
**Now Lets try plotting the data using different plots.**

**I tried out plotting it using histograms, line graph and box plots**

**And the conclusion for the same is as follows.**

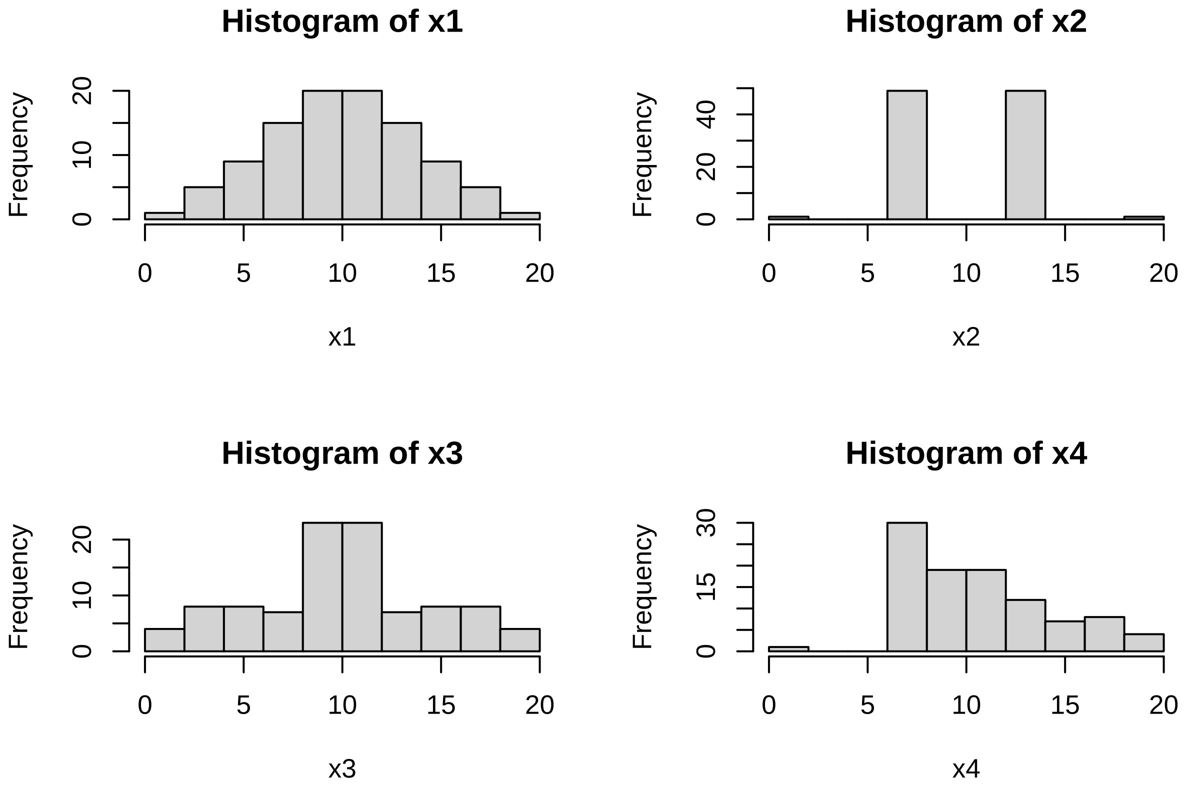
**The box plot for x1, x2, x3, x4 is really misleading.**

```
# Plotting box-plots in 2x2 layout
par(mfrow = c(2,2))
with(qn1, boxplot(x1, main="x1"))
with(qn1, boxplot(x2, main="x2"))
with(qn1, boxplot(x3, main="x3"))
with(qn1, boxplot(x4, main="x4"))
```



**Seeing the above 4 bar-plots for x1,x2,x3,x4 it seems that the data is distributed similarly in all the 4 columns. It gives us the idea that the data present in all the columns is same, which in fact is wrong as we can see from the following graphs.**

```r
# plotting histograms in 2x2 layout
par(mfrow = c(2,2))
with(qn1, hist(x1))
with(qn1, hist(x2))
with(qn1, hist(x3))
with(qn1, hist(x4))
```



**As we can see from the above histograms, data distribution is very different in all the four columns, which is contrary to the idea we got from the box plots.**

**So, we can conclude that the Box-Plot is misleading in this particular case while histograms provide the right idea about how the data is distributed.**

---

# PERSONALITY AND MOTION

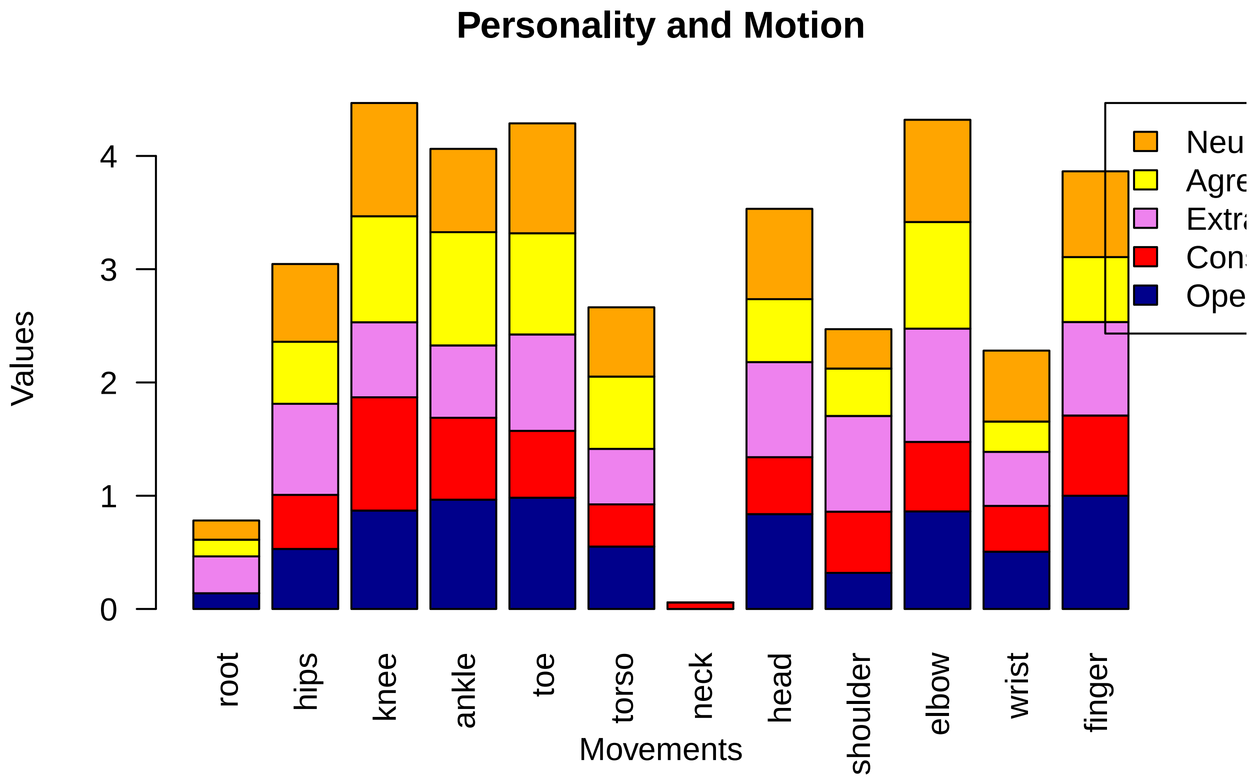
**In this question we have to visualize how each of the 12 joints contribute to predicting each of five personality scores.**

**I am picking stacked bar chart to show the same as one of the plots. And also radar plot for each personality to see which joint movement affects the personality score the more.**

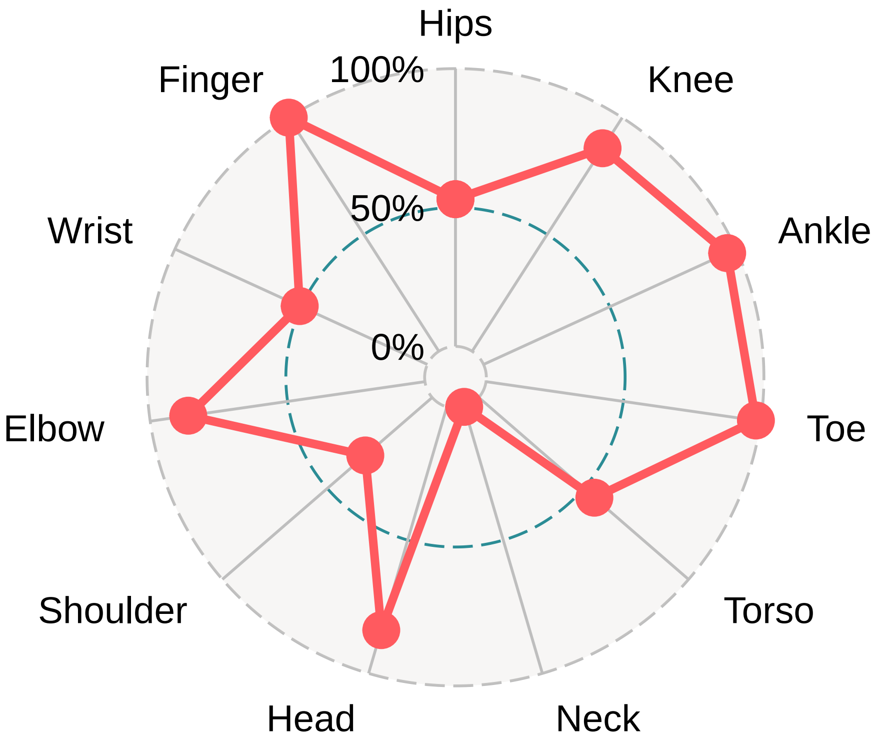
```
library(ggplot2)
library(tidyverse)

#Reading second sheet from excel
qn2 = read.xlsx(file = "Visualisation_Activity.xlsx", sheetIndex = 2)
```

```r
# creating df in the required form for stacked-bar plot
#Movements = rep(qn2$Movements, ncol(qn2) - 1)
#Personality = rep(colnames(qn2)[-1], nrow(qn2))
#Values = character()
#Movements = qn2$Movements
#for(i in 1:nrow(qn2))
#{
#  for(j in 2:ncol(qn2))Values = append(Values, qn2[i,j])
#}


#df = data.frame(Movements, Personality, Values)


#ggplot(df, aes(y = Values, x = Movements, fill = Personality, group = Personality)) +  geom_bar(stat =

#p <- barplot(t(qn2[,-1]), main = "Personality and Movements",
#        xlab = "Movements", ylab = "Values",
#        col = c("darkblue", "red", "violet", "yellow", "orange"),
#        legend.text = colnames(qn2[,-1]),
#        args.legend = list(x = "topright",
#                           inset = c(- 0.30, 0)))
```

```r
mydf = as.table(matrix(
  t(as.matrix(qn2[, -1])),
  nrow = 5,
  dimnames = list(
   colm = c("openness", "conscientiousness", "extaversion", "agreeableness", "neuroticism"),
   rowm = c("root", "hips", "knee", "ankle", "toe", "torso", "neck", "head", "shoulder", "elbow", "wris
  ))
)


barplot(mydf, main="Personality and Motion", xlab = "Movements",
               ylab = "Values",
               col = c("darkblue", "red", "violet", "yellow", "orange"),
               legend.text = colnames(qn2[,-1]),
                args.legend = list(x = "topright", inset = c(- 0.30, 0)),
               las = 2, cex.names = 1)
```



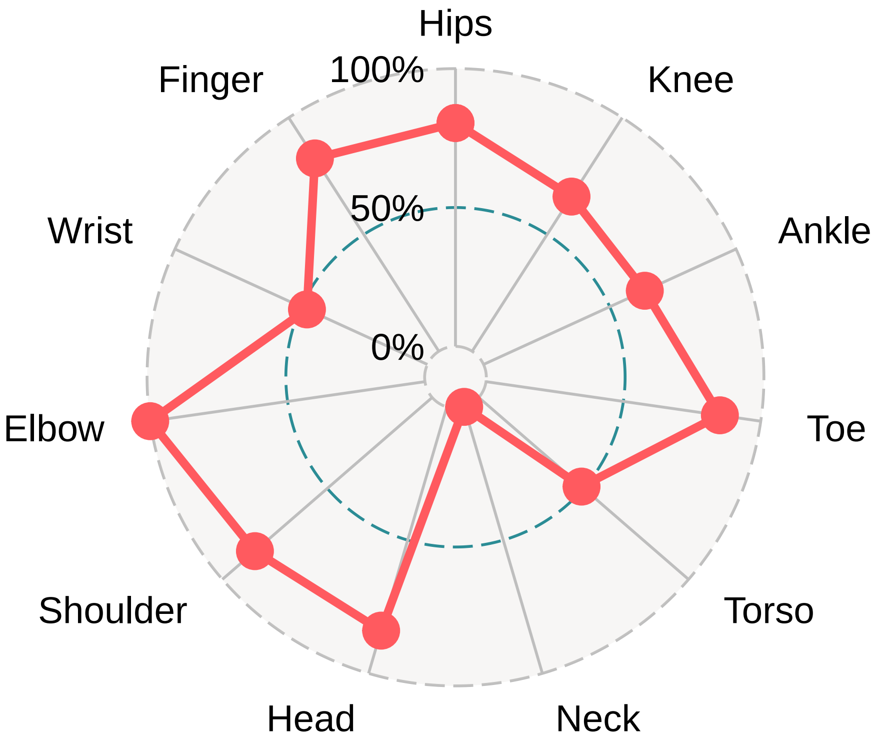
```
library(ggradar)
# remotes::install_github("ricardo-bion/ggradar")

open<-t(qn2$Openness)
temp <- c("Root", "Hips", "Knee", "Ankle", "Toe", "Torso", "Neck", "Head", "Shoulder", "Elbow", "Wrist"
colnames(open) <- temp

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



```
#ggradar(plt)
```

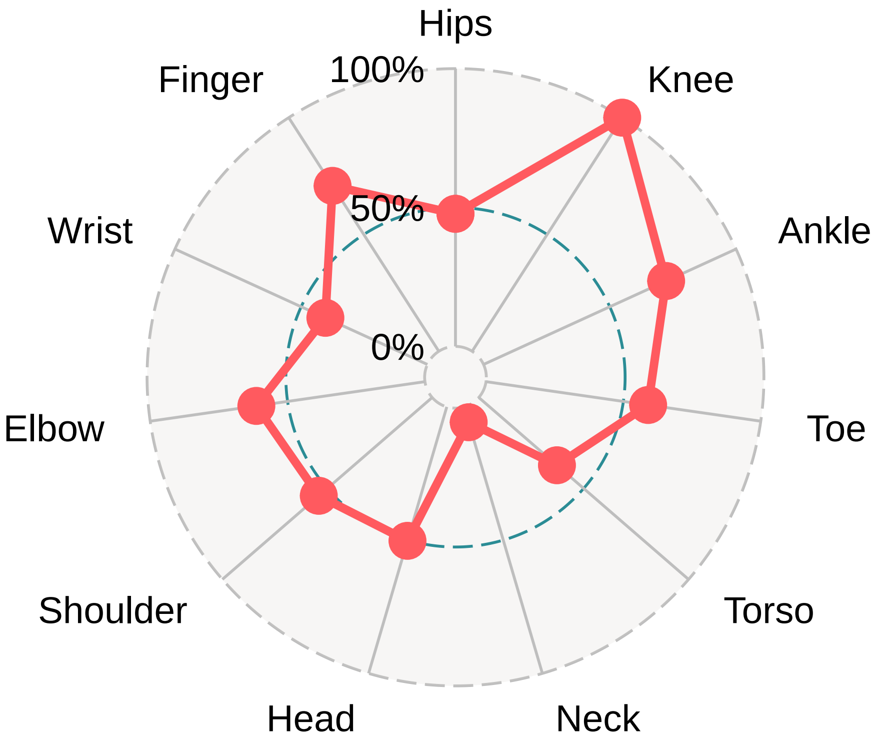
**### Openness vs Movements.**

```
extra<-t(qn2$Extraversion)
temp <- c("Root", "Hips", "Knee", "Ankle", "Toe", "Torso", "Neck", "Head", "Shoulder", "Elbow", "Wrist"
colnames(extra) <- temp

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



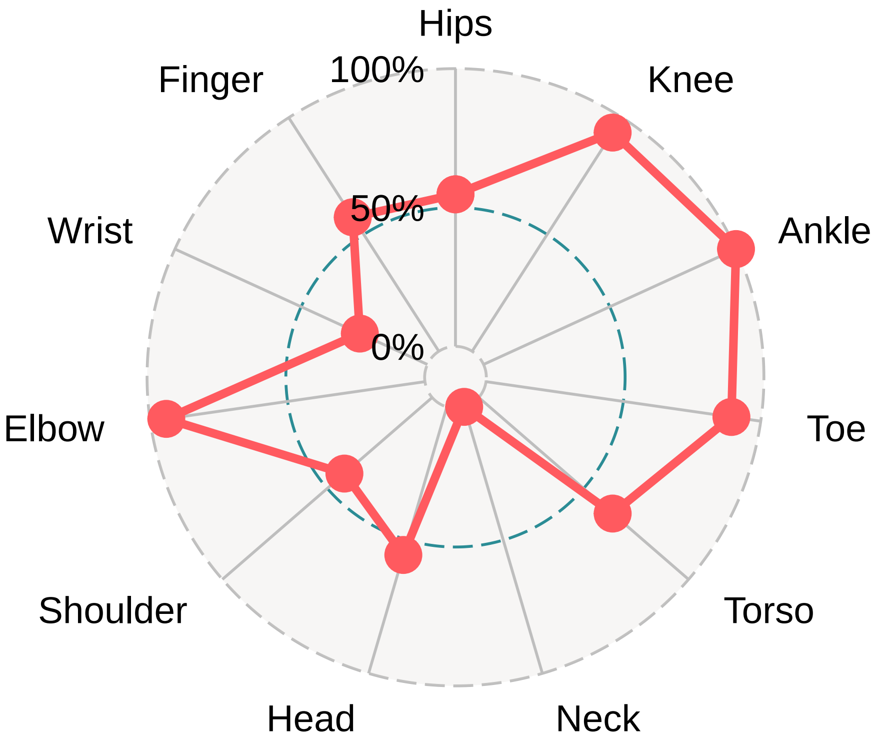
**### Extraversion VS Movements**

```
consci<-t(qn2$Conscientiousness)
temp <- c("Root", "Hips", "Knee", "Ankle", "Toe", "Torso", "Neck", "Head", "Shoulder", "Elbow", "Wrist"
colnames(consci) <- temp

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



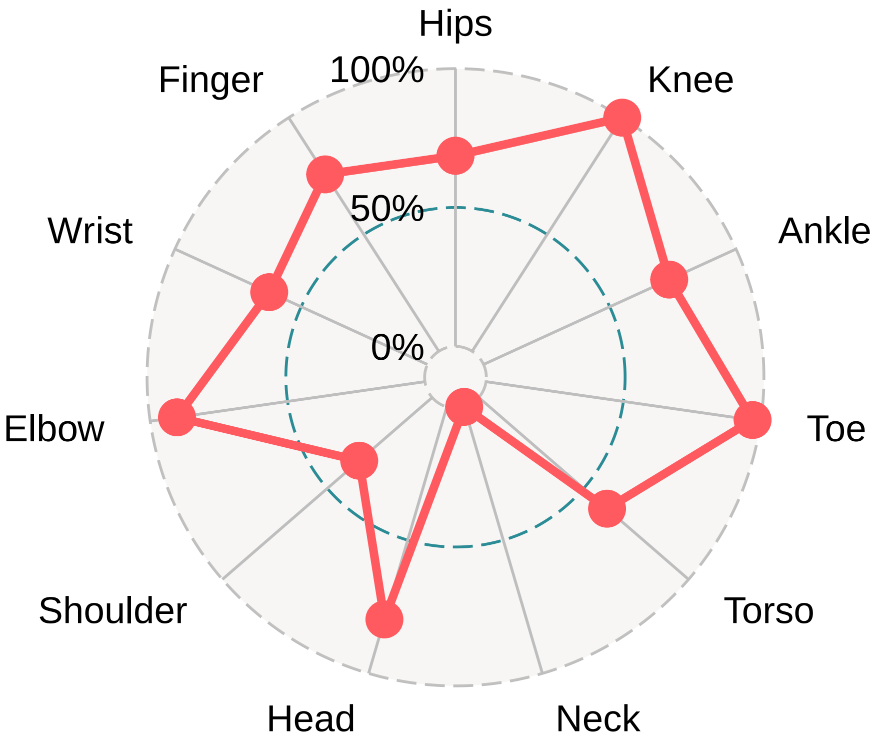
**### Conscientiousness VS Movements**

```
agree<-t(qn2$Agreeableness)
temp <- c("Root", "Hips", "Knee", "Ankle", "Toe", "Torso", "Neck", "Head", "Shoulder", "Elbow", "Wrist"
colnames(agree) <- temp

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



### Agreeableness VS Movements

```
neuro<-t(qn2$Neuroticism)
temp <- c("Root", "Hips", "Knee", "Ankle", "Toe", "Torso", "Neck", "Head", "Shoulder", "Elbow", "Wrist"
colnames(neuro) <- temp

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



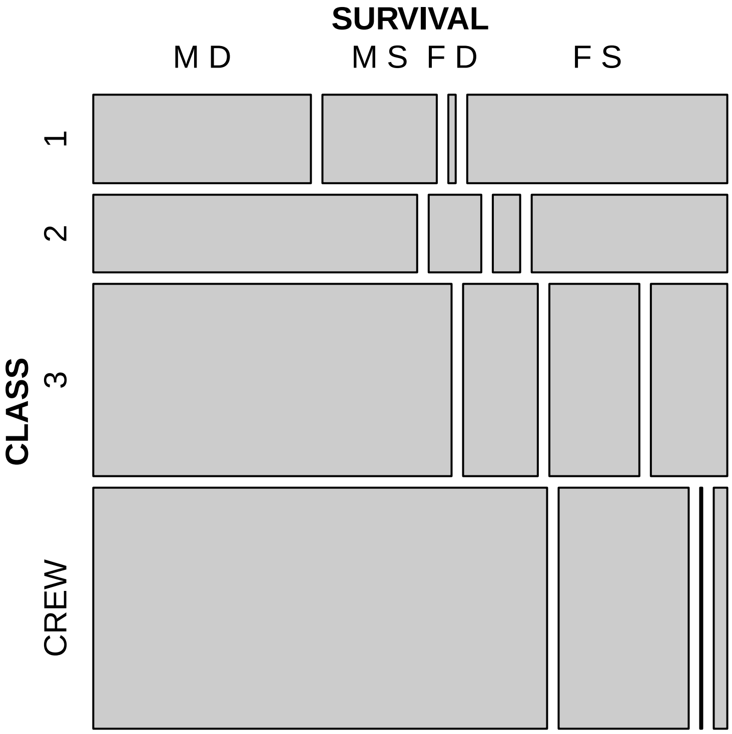
### Neuroticism VS Movements

## DATA PLOTTING ADVENTURE

### Subtask 1 : Sinking Ship

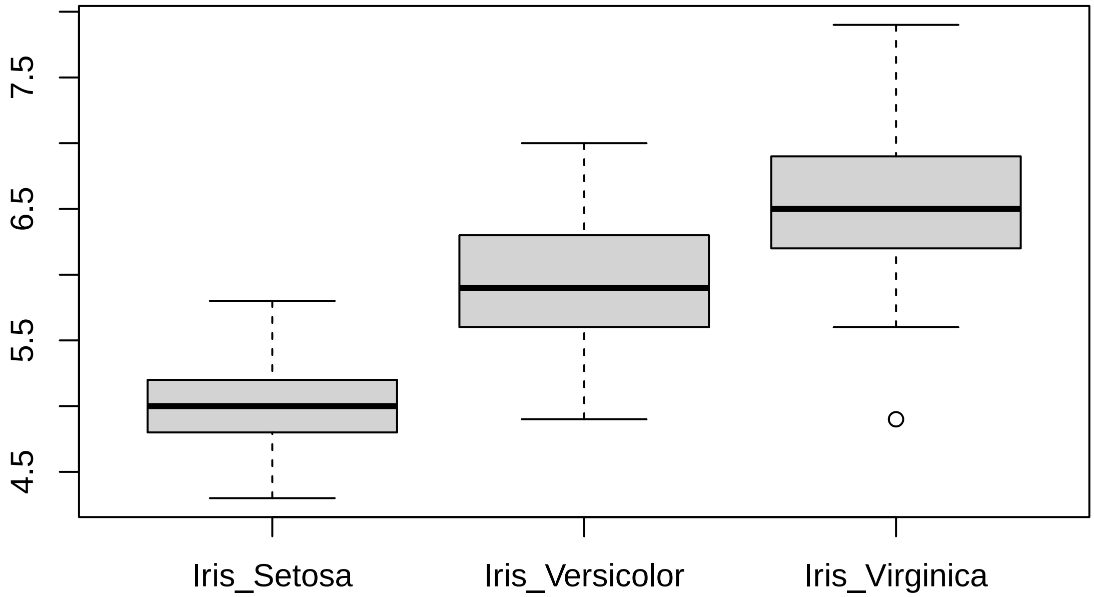
```
library(vcd)


count = matrix(c(118, 62, 4, 141,
          154, 25, 13, 93,
          422, 88, 106, 90,
          670, 192, 3, 20))

data <- as.table(
  matrix(
    count, nrow = 4, byrow = TRUE,
    dimnames = list(
      CLASS = c("1", "2", "3", "CREW"),
      SURVIVAL = c("M D", "M S", "F D", "F S")
    )
  )
)
```

```
mosaic(data)
```



### Subtask 2 : Petal Prediction

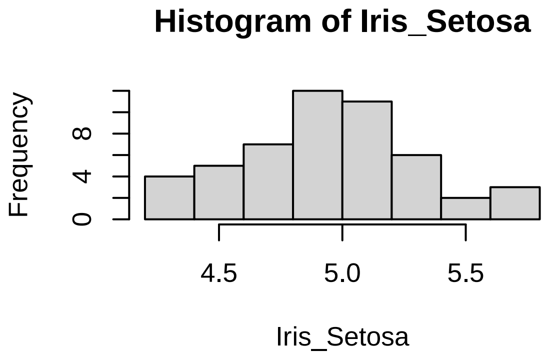
```
qn4_2 = read.xlsx(file = "Visualisation_Activity.xlsx", sheetIndex = 3)

Iris_Setosa = qn4_2[1:50,]$SepalLengthCm
Iris_Versicolor = qn4_2[51:100,]$SepalLengthCm
Iris_Virginica = qn4_2[101:150, ]$SepalLengthCm

data <- data.frame(Iris_Setosa, Iris_Versicolor, Iris_Virginica)

boxplot(data)
```

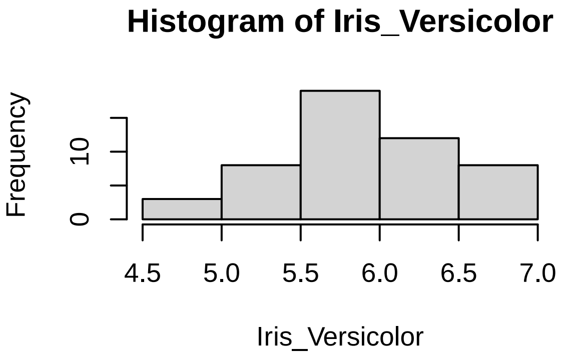


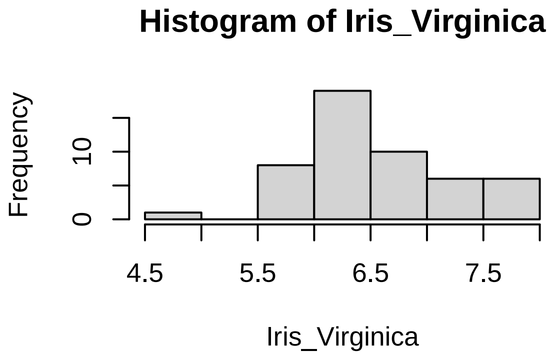
**As we can see from the boxplots of different species, it is clear that Petal_length order is in Iris_setosa < Iris_versicolor < Iris_virginica.**

**And from the below given histograms we can see the actual distribution of the Petal_length of the respective species of the flowers.**

```
par(mfrow = c(2,2))
hist(Iris_Setosa)
hist(Iris_Versicolor)
hist(Iris_Virginica)
```



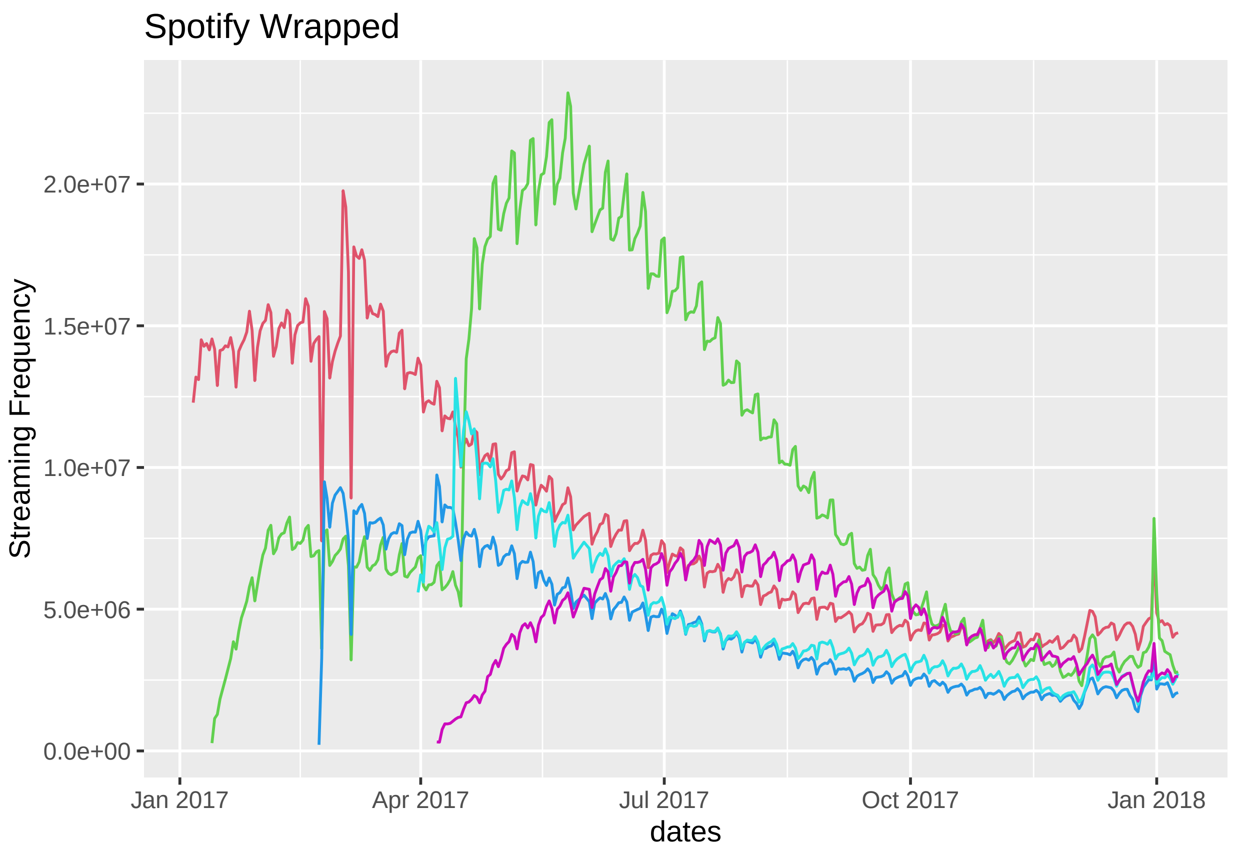




---

### Subtask 3: Spotify Wrapped

```
qn4 <- read.xlsx("Visualisation_Activity.xlsx", sheetIndex =  4)

songs = qn4[, 2:6]
dates <- as.Date(qn4$Date)

plt <- ggplot(qn4, 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) + ggtitle("Spotify Wrapped") + ylab("Streaming Frequency")

plt
```



**Legend :**

**RED = Shape of you**

**GREEN = Despacito**

**BLUE = Something just like this**

**CYAN = Humble**

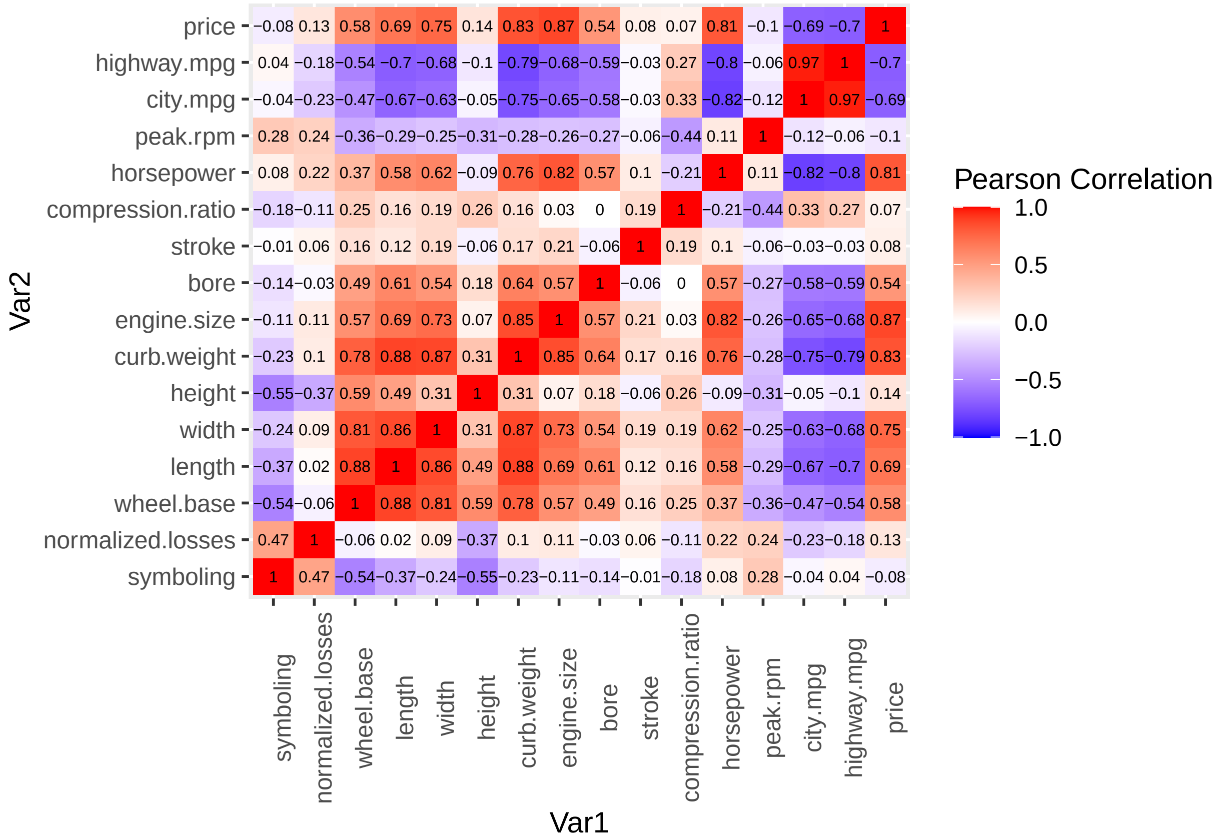
**VIOLET = Unforgettable**

**As we can from the line graph, even though different songs had their different peak times and different peak points, toward the end of 2018 number of plays of all the songs reduced together**

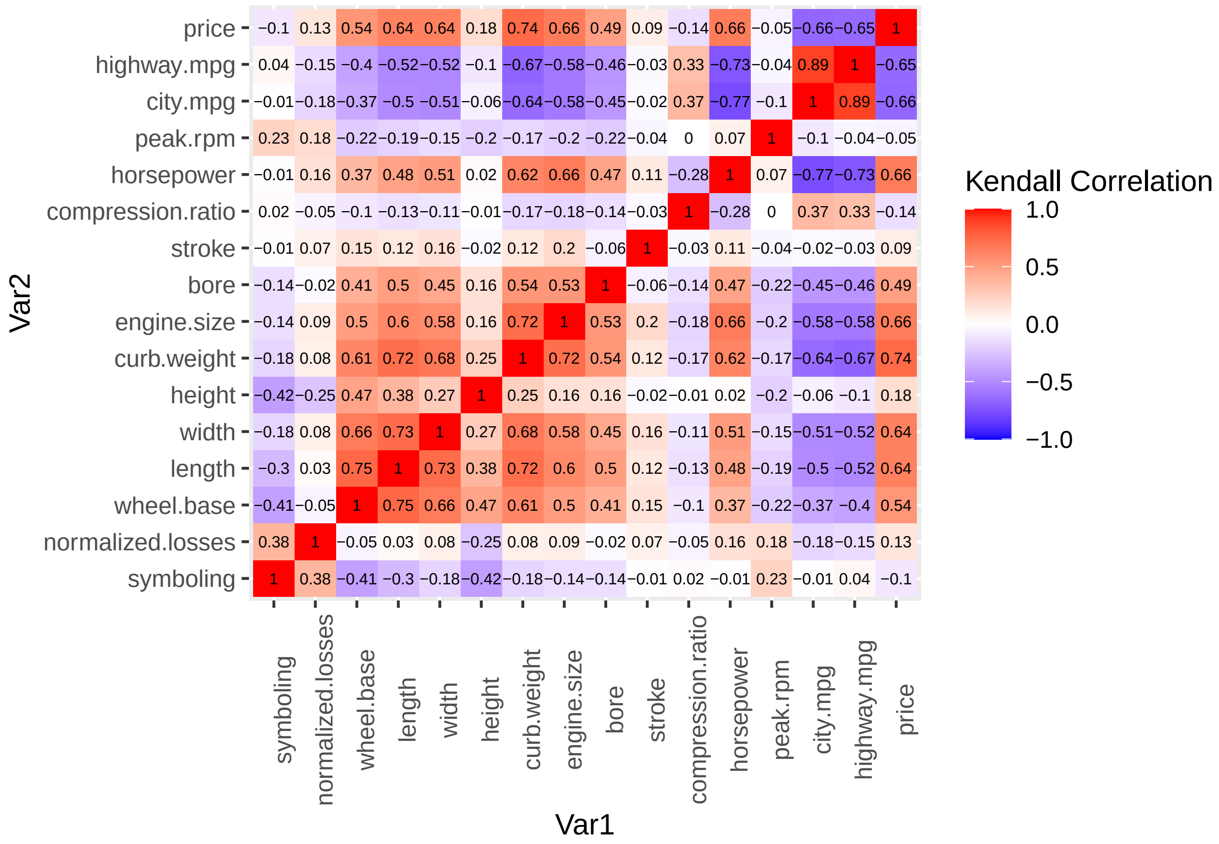
## NEED FOR SPEED

**Plot a heatmap for different types of correlations between the features present in the dataset and write about the inferences you can make from it. Also justify as to why one is better than the other**

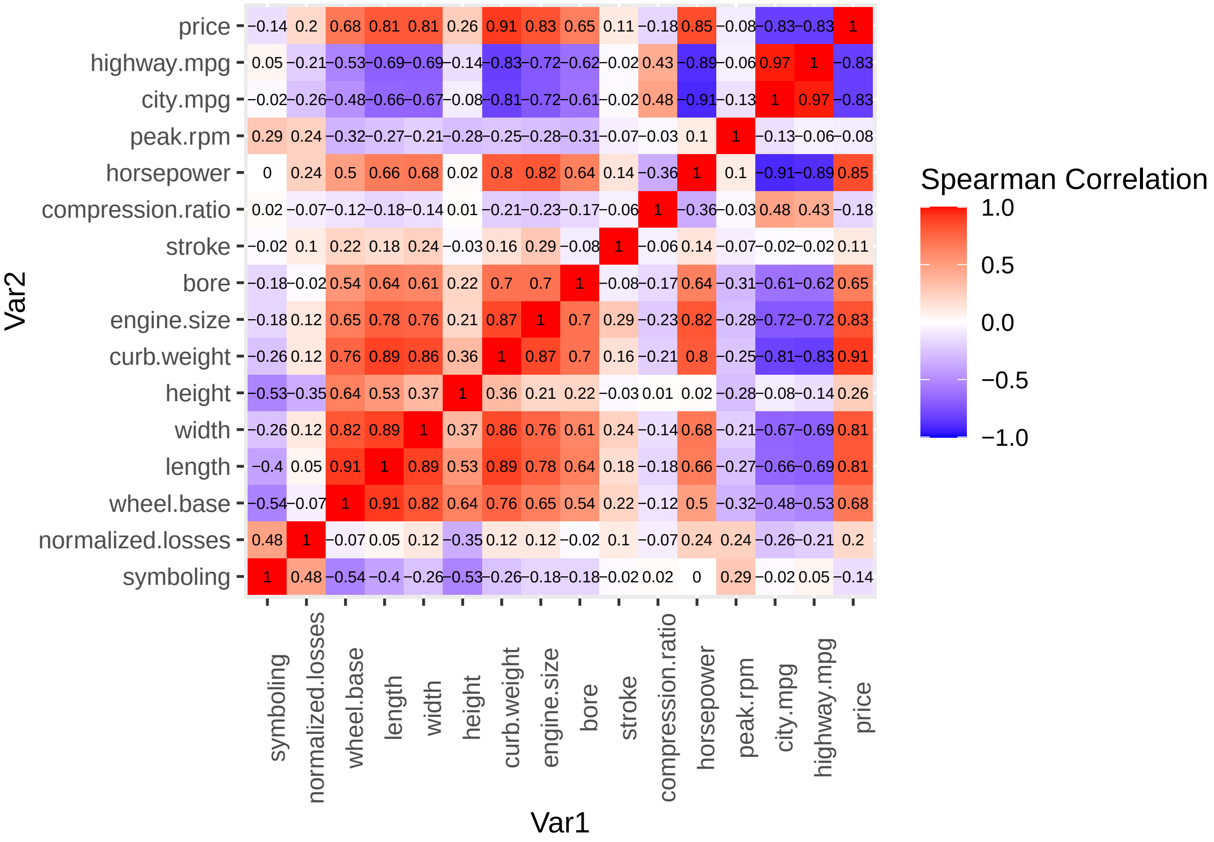
```
#Reading the excel file.
q5 = read.xlsx(file='Visualisation_Activity.xlsx', sheetIndex=5)

q5 = subset(q5, select = -c(make, fuel.type, aspiration, num.of.doors, body.style, drive.wheels, engine


library(reshape2)
#calculate correlation coefficients, rounded to 2 decimal places
cor_q51 <- round(cor(q5, method = "pearson"), 2)
cor_q52 <- round(cor(q5, method = "kendall"), 2)
cor_q53 <- round(cor(q5, method = "spearman"), 2)

#melt the data frame , basically reducing the size of correlation matrix.
melted_cor1 <- melt(cor_q51)
melted_cor2 <- melt(cor_q52)
melted_cor3 <- melt(cor_q53)

library(ggplot2)
#create correlation heatmap
ggplot(data = melted_cor1, 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))
```



```r
ggplot(data = melted_cor2, 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))
```



```
ggplot(data = melted_cor3, 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))
```



| | symboling | normalized.losses | wheel.base | length | width | height | curb.weight | engine.size | bore | stroke | compression.ratio | horsepower | peak.rpm | city.mpg | highway.mpg | price |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| price | −0.14 | 0.2 | 0.68 | 0.81 | 0.81 | 0.26 | 0.91 | 0.83 | 0.65 | 0.11 | −0.18 | 0.85 | −0.08 | −0.83 | −0.83 | 1 |
| highway.mpg | 0.05 | −0.21 | −0.53 | −0.69 | −0.69 | −0.14 | −0.83 | −0.72 | −0.62 | −0.02 | 0.43 | −0.89 | −0.06 | 0.97 | 1 | −0.83 |
| city.mpg | −0.02 | −0.26 | −0.48 | −0.66 | −0.67 | −0.08 | −0.81 | −0.72 | −0.61 | −0.02 | 0.48 | −0.91 | −0.13 | 1 | 0.97 | −0.83 |
| peak.rpm | 0.29 | 0.24 | −0.32 | −0.27 | −0.21 | −0.28 | −0.25 | −0.28 | −0.31 | −0.07 | −0.03 | 0.1 | 1 | −0.13 | −0.06 | −0.08 |
| horsepower | 0 | 0.24 | 0.5 | 0.66 | 0.68 | 0.02 | 0.8 | 0.82 | 0.64 | 0.14 | −0.36 | 1 | 0.1 | −0.91 | −0.89 | 0.85 |
| compression.ratio | 0.02 | −0.07 | −0.12 | −0.18 | −0.14 | 0.01 | −0.21 | −0.23 | −0.17 | −0.06 | 1 | −0.36 | −0.03 | 0.48 | 0.43 | −0.18 |
| stroke | −0.02 | 0.1 | 0.22 | 0.18 | 0.24 | −0.03 | 0.16 | 0.29 | −0.08 | 1 | −0.06 | 0.14 | −0.07 | −0.02 | −0.02 | 0.11 |
| bore | −0.18 | −0.02 | 0.54 | 0.64 | 0.61 | 0.22 | 0.7 | 0.7 | 1 | −0.08 | −0.17 | 0.64 | −0.31 | −0.61 | −0.62 | 0.65 |
| engine.size | −0.18 | 0.12 | 0.65 | 0.78 | 0.76 | 0.21 | 0.87 | 1 | 0.7 | 0.29 | −0.23 | 0.82 | −0.28 | −0.72 | −0.72 | 0.83 |
| curb.weight | −0.26 | 0.12 | 0.76 | 0.89 | 0.86 | 0.36 | 1 | 0.87 | 0.7 | 0.16 | −0.21 | 0.8 | −0.25 | −0.81 | −0.83 | 0.91 |
| height | −0.53 | −0.35 | 0.64 | 0.53 | 0.37 | 1 | 0.36 | 0.21 | 0.22 | −0.03 | 0.01 | 0.02 | −0.28 | −0.08 | −0.14 | 0.26 |
| width | −0.26 | 0.12 | 0.82 | 0.89 | 1 | 0.37 | 0.86 | 0.76 | 0.61 | 0.24 | −0.14 | 0.68 | −0.21 | −0.67 | −0.69 | 0.81 |
| length | −0.4 | 0.05 | 0.91 | 1 | 0.89 | 0.53 | 0.89 | 0.78 | 0.64 | 0.18 | −0.18 | 0.66 | −0.27 | −0.66 | −0.69 | 0.81 |
| wheel.base | −0.54 | −0.07 | 1 | 0.91 | 0.82 | 0.64 | 0.76 | 0.65 | 0.54 | 0.22 | −0.12 | 0.5 | −0.32 | −0.48 | −0.53 | 0.68 |
| normalized.losses | 0.48 | 1 | −0.07 | 0.05 | 0.12 | −0.35 | 0.12 | 0.12 | −0.02 | 0.1 | −0.07 | 0.24 | 0.24 | −0.26 | −0.21 | 0.2 |
| symboling | 1 | 0.48 | −0.54 | −0.4 | −0.26 | −0.53 | −0.26 | −0.18 | −0.18 | −0.02 | 0.02 | 0 | 0.29 | −0.02 | 0.05 | −0.14 |

**Pearson's correlation coefficient measures the linear association between two continuous variables. It assumes that the relationship between the variables is linear and that the variables are normally distributed.**

**Spearman's rank correlation coefficient measures the monotonic association between two continuous or ordinal variables. It does not assume that the variables are normally distributed or that the relationship is linear. Instead, it looks for a monotonic relationship between the variables, meaning that as one variable increases, the other variable either increases or decreases consistently.**

**Kendall's rank correlation coefficient is a non-parametric measure of the relationship between two ordinal variables or between two continuous variables that have been ranked. It measures the agreement between the rankings of the variables and is appropriate for use with small sample sizes.**

**Since our data is not normally distributed, and neither is it linearly distributed, Spearman's rank correlation coefficient is best in this case.**