Assignment No -1

By Nikita Pai & Abhilash Hemaraj

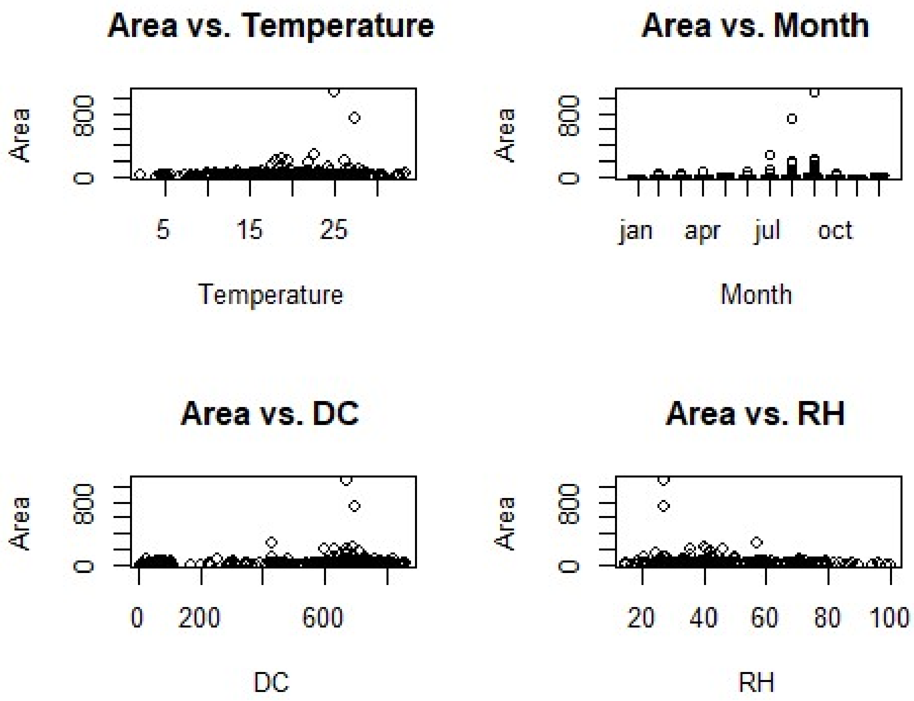
Group No -12

Course: IE7275 34489 Data Mining in Engineering SEC 02 Spring 2020

```
# import the Dataset
forestfires.df <- read.csv("C:/Users/abhil/Downloads/forestfires(1).csv", str
ingsAsFactors=FALSE)
```

#Question 1

```
# a. Plot area vs.temp, area vs. month, area vs. DC, area vs. RH for January
through December combined in one graph
#Scatterplots

par(mfrow=c(2,2))
plot(x = forestfires.df$temp, y = forestfires.df$area,
     main = "Area vs. Temperature",
     xlab = "Temperature",
     ylab = "Area")

plot(x = factor(forestfires.df$month, levels = c("jan", "feb", "mar", "apr",
"may",
                                                 "jun", "jul", "aug", "sep",
"oct",
                                                 "nov", "dec")),
     y = forestfires.df$area,
     main = "Area vs. Month",
     xlab = "Month",
     ylab = "Area")

plot(x = forestfires.df$DC, y = forestfires.df$area,
     main = "Area vs. DC",
     xlab = "DC",
     ylab = "Area")

plot(x = forestfires.df$RH, y = forestfires.df$area,
     main = "Area vs. RH",
     xlab = "RH",
     ylab = "Area")
```



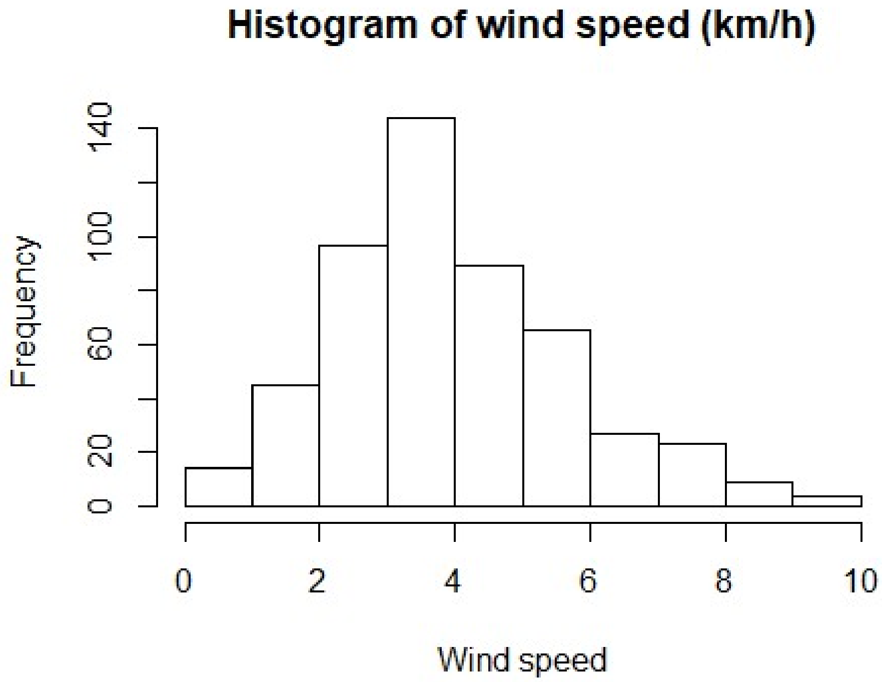
Area vs. Temperature

Area vs. Month

Area vs. DC

Area vs. RH

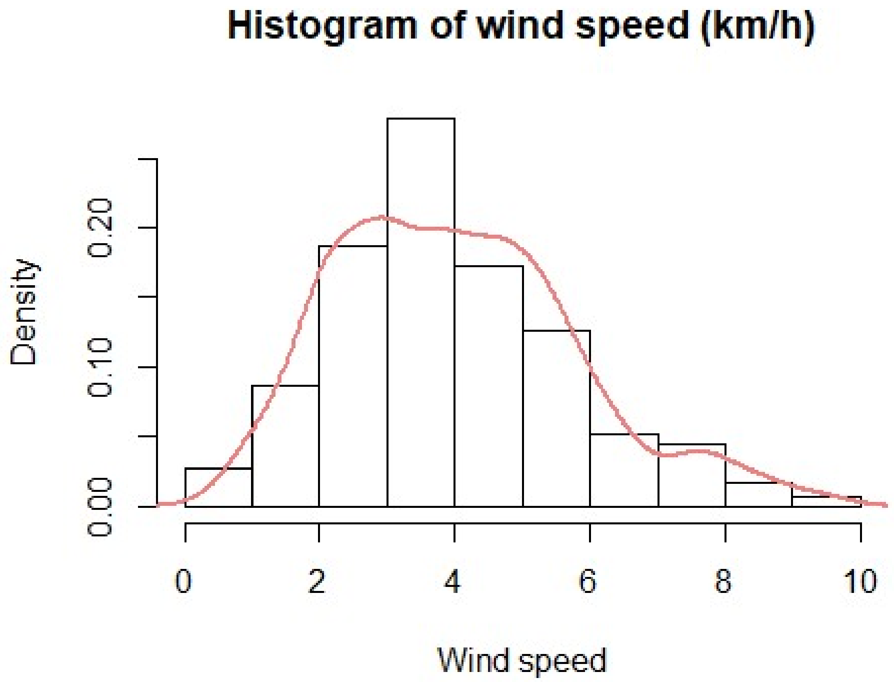
```r
# b. Plot the histogram of wind speed (km/h).

windspeed <- hist(forestfires.df$wind, main = "Histogram of wind speed (km/h)
",
     xlab = "Wind speed")
```



```
# c. Compute the summery statistics (min, 1Q, mean, median, 3Q, max,) of part
b.
summary(forestfires.df$wind)
```

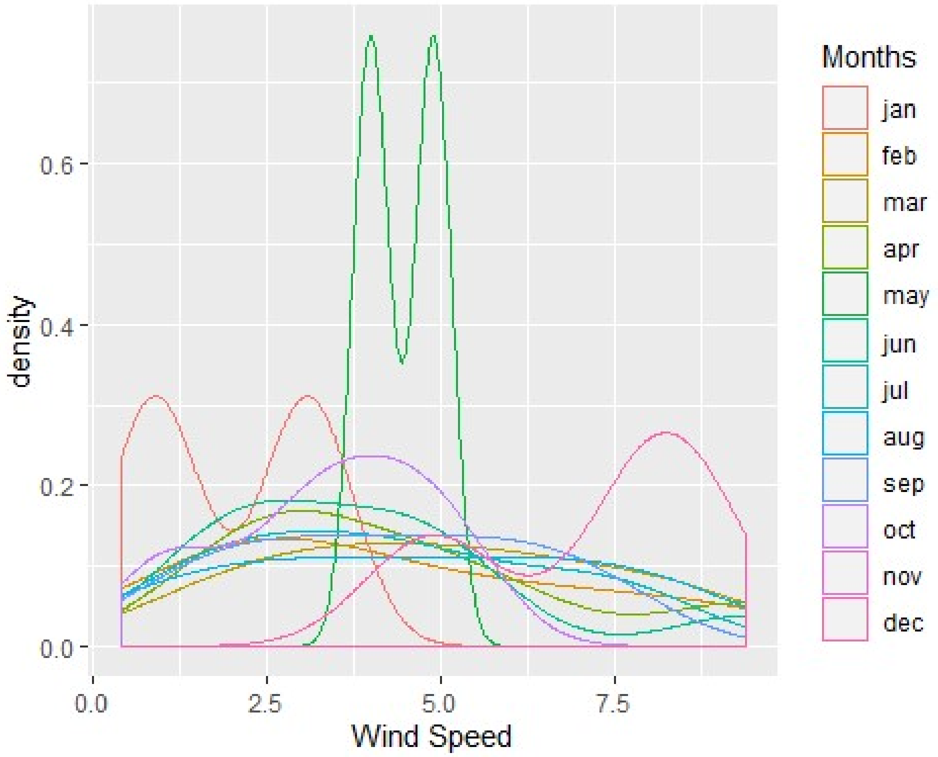
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.400   2.700   4.000   4.018   4.900   9.400
```

```
# d. Add a density line to the histogram in part b.
hist(forestfires.df$wind, main = "Histogram of wind speed (km/h)",
     prob = TRUE,
     xlab = "Wind speed")
lines(density(forestfires.df$wind), # density plot
 lwd = 2, # thickness of line
 col = "lightcoral")
```



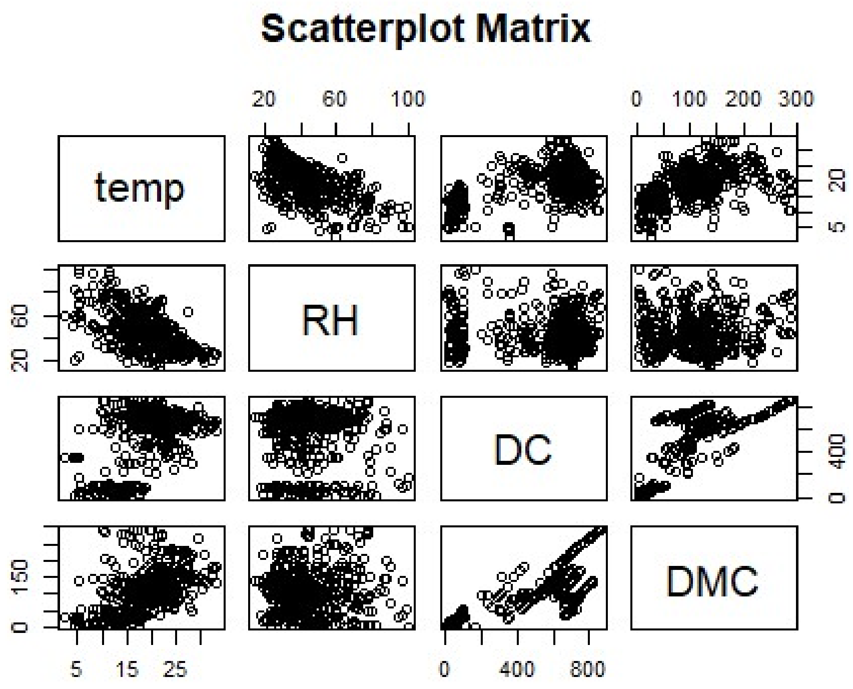
```
# e. Plot the wind speed density function of all months in one plot. Use
# different colors for different months in the graph to interpret your result
clearly.
# [Hint: use ggplot + geom_density or qplot(geom=density)]
library(dplyr)
```

```
##
## 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

library(ggplot2)
wind.df <- summarise(group_by(forestfires.df, month, wind))

ggplot(wind.df, aes(x = wind.df$wind, color = factor(wind.df$month, levels =
c("jan", "feb", "mar", "apr", "may",
                         "jun", "jul", "aug", "sep",
"oct",
                         "nov", "dec")))) +
  geom_density(position = "identity") +
  xlab("Wind Speed") +
  labs(col = "Months")

## Warning: Groups with fewer than two data points have been dropped.
```



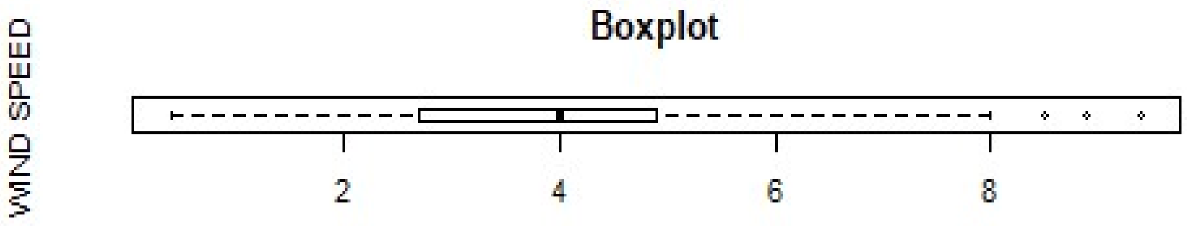
```
# f. Plot the scatter matrix for temp, RH, DC and DMC.
# How would you interpret the result in terms of correlation among these data
?
```

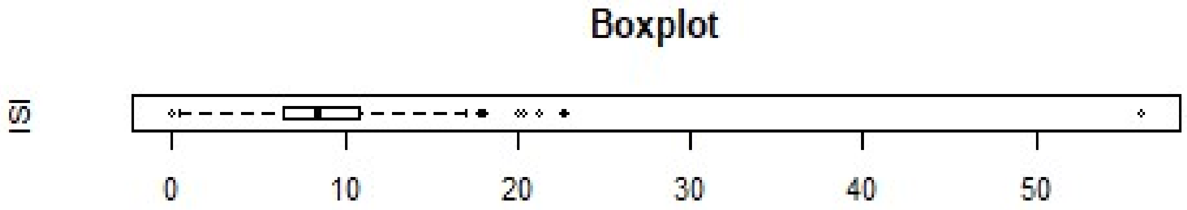
```
temp <- forestfires.df$temp
RH <- forestfires.df$RH
DC <- forestfires.df$DC
DMC <- forestfires.df$DMC
pairs(~temp+RH+DC+DMC, data=forestfires.df,
     main="Scatterplot Matrix")
```

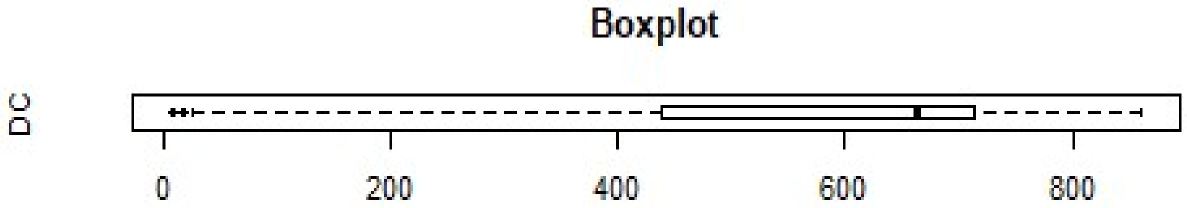


The Variables DC and DMC follow a linearly increasing relationship; whereas variables temp and Rh have an inverse relation between each other. The rest of variable combination do not exhibit a very apparent pattern.

```
# g. Create boxplot for wind, ISI and DC. Are there any anomalies/outliers? I
nterpret your result.
par(mfrow = c(3, 1))
boxplot(forestfires.df$wind, horizontal = TRUE,
        main="Boxplot",
        ylab="WIND SPEED")
boxplot(forestfires.df$ISI, horizontal = TRUE,
        main="Boxplot",
        ylab="ISI")
boxplot(forestfires.df$DC, horizontal = TRUE,
        main="Boxplot",
        ylab="DC")
```

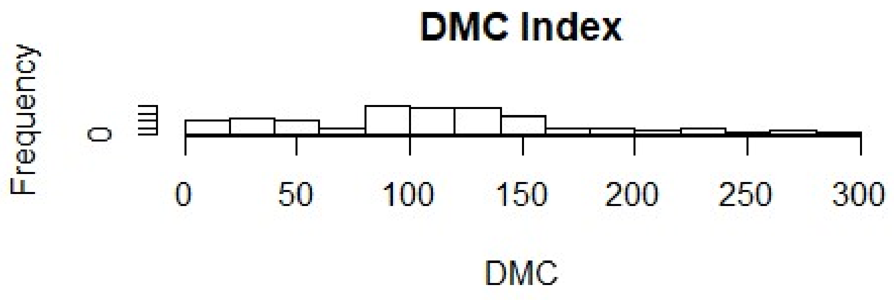


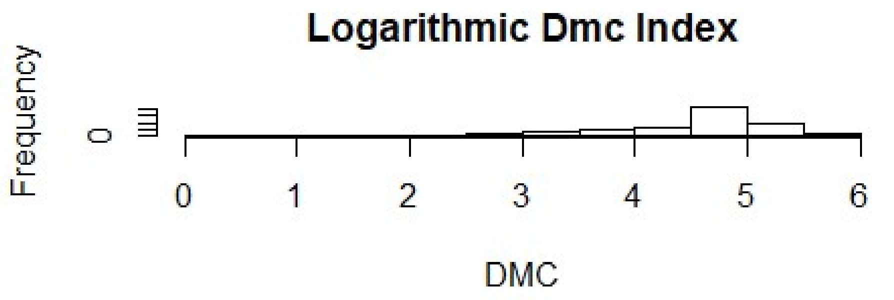




Yes, the presence of outliers is quiet evident in all the three boxplots, also the boxplots span different range of values hence there is a differrence in median.

```r
# h. Create the histogram of DMC. Create the histogram of log of DMC. Compare
the result and explain your answer.
par(mfrow = c(2,1))
DMC <- hist(forestfires.df$DMC, main = "DMC Index", xlab = "DMC")
logdmc <- log(forestfires.df$DMC)
hist(logdmc, main = "Logarithmic Dmc Index", xlab = "DMC" )
```



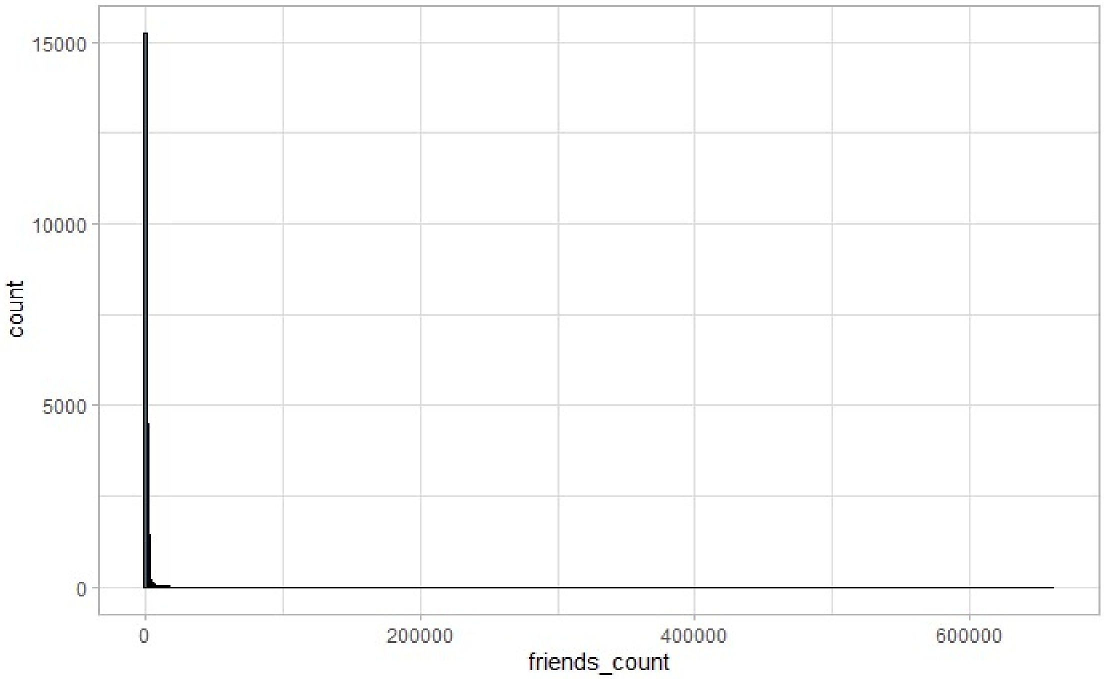


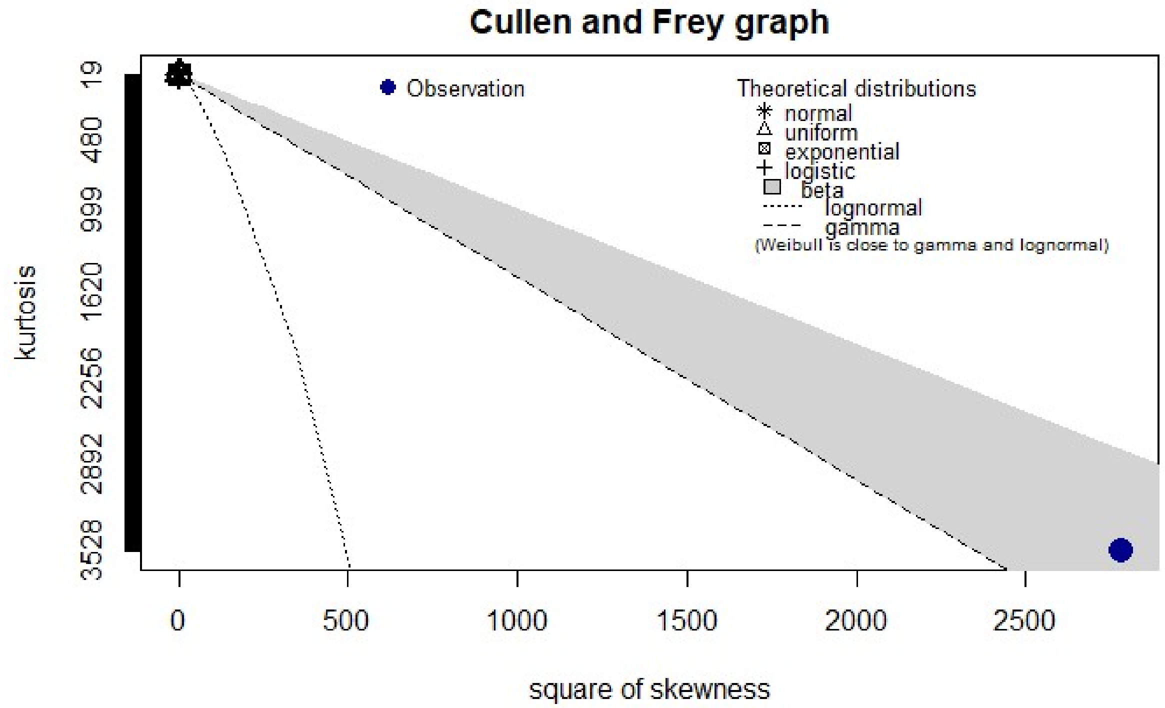
The histogram of DMC with normal scaling doesn't follow a certain characteristic. But, when converted into a log index the distribution looks right skewed.

#Question 2

```
# Import the Dataset
M01_quasi_twitter <- read.csv("C:/Users/abhil/Downloads/M01_quasi_twitter(1).
csv", stringsAsFactors=FALSE)

# a. How are the data distributed for friend_count variable?
library(fitdistrplus)

## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

## Loading required package: survival

## Loading required package: npsurv

## Loading required package: lsei

library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine


a <-
ggplot(M01_quasi_twitter, aes(friends_count)) +
  geom_histogram(bins = 500, color = 'Black', fill = 'skyblue4') +
  theme(text=element_text(size=12, family="sans")) +
  theme_light()
a
b <- descdist(M01_quasi_twitter$friends_count)

dist <- scale(M01_quasi_twitter$friends_count, center = TRUE, scale = TRUE)
apply(dist, 2, mean)

## [1] 4.563535e-18

apply(dist, 2, sd)

## [1] 1
```





Using the Culllen and Frey graph we can check the distribution of our Dataset. As observed, our observation falls in the beta distribution spectrum. Normalising the data would involve removing the outliers.

```
# b. Compute the summery statistics (min, 1Q, mean, median, 3Q, max) on frien
d_count.

summary(M01_quasi_twitter$friends_count)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     -84     123     324    1058     849  660549

# c. How is the data quality in friend_count variable? Interpret your answer.
library(dataQualityR)
Friend_Count <- data.frame(M01_quasi_twitter$friends_count)
data(M01_quasi_twitter)

## Warning in data(M01_quasi_twitter): data set 'M01_quasi_twitter' not found

num.file <- paste(tempdir(), "/dq_num.csv", sep= "")
cat.file <- paste(tempdir(), "/dq_cat.csv", sep= "")
checkDataQuality(data = M01_quasi_twitter, out.file.num= num.file, out.file.c
at= cat.file)

## Check for numeric variables completed // Results saved to disk // Time dif
ference of 0.1825109 secs
```

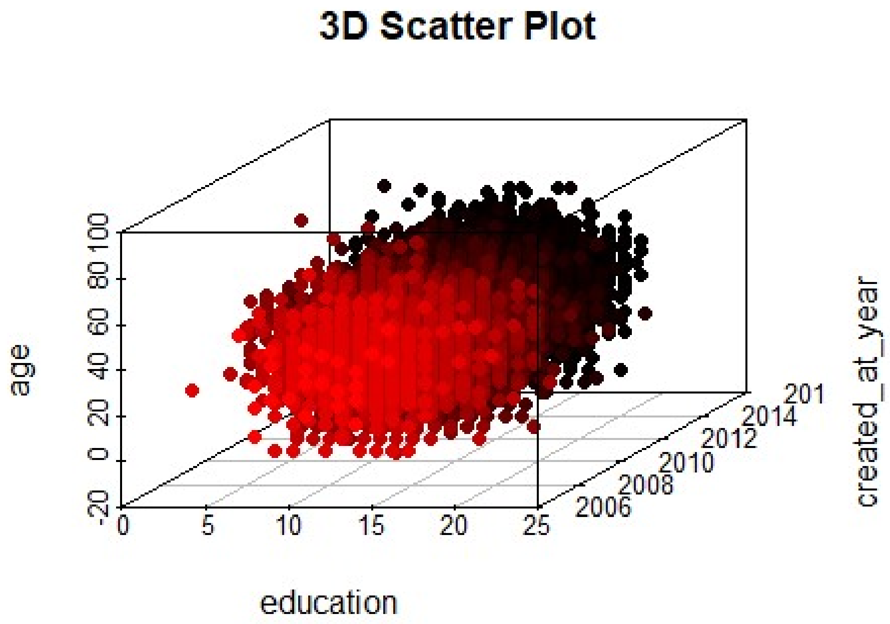
```
## Check for categorical variables completed // Results saved to disk // Time
difference of 0.2812762 secs
```

```
num.file
```

```
## [1] "C:\\Users\\abhil\\AppData\\Local\\Temp\\RtmpiIPaUV/dq_num.csv"
```

Inference: After Checking the Data quality of Friends_count variable, we can see that   the minimum value is -84 which is not possible. There are no missing values in the variable.

```
# d. Produce a 3D scatter plot with highlighting to impression the depth for
variables below on M01_quasi_twitter.csv dataset. created_at_year, education,
age. Put the name of the scatter plot “3D scatter plot”.

library(scatterplot3d)
scatterplot3d(M01_quasi_twitter[,c(18, 4, 20)], pch = 16, angle = 45, main =
"3D Scatter Plot", highlight.3d = TRUE)
```



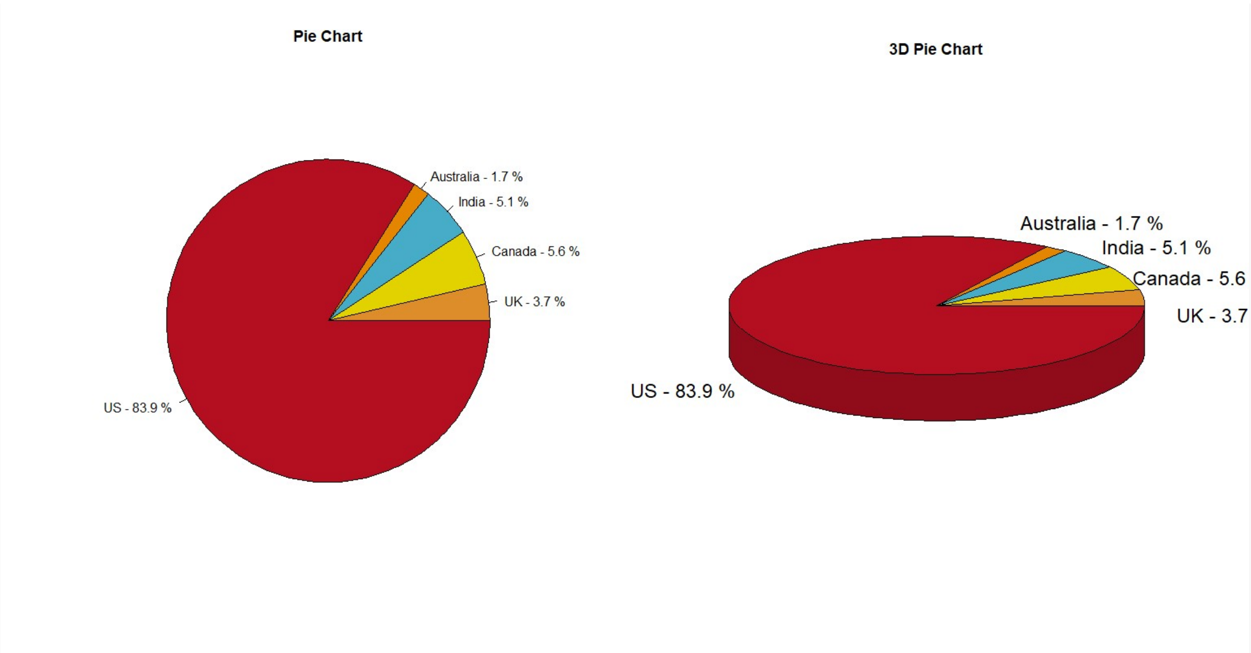
```
# e. Consider 650, 1000, 900, 300 and 14900 tweeter accounts are in UK, Canad
a, India, Australia and US, respectively. Plot the percentage Pie chart inclu
des percentage amount and country name adjacent to it, and also plot 3D pie c
hart for those countries along with the percentage pie chart. Hint: Use C=(1,
2) matrix form to plot the charts together.

library(ggplot2)
library(scales)
library(plotrix)
```

```
##
## Attaching package: 'plotrix'

## The following object is masked from 'package:scales':
##
##     rescale
```

```
library(ggpubr)
```

```
## Loading required package: magrittr
```

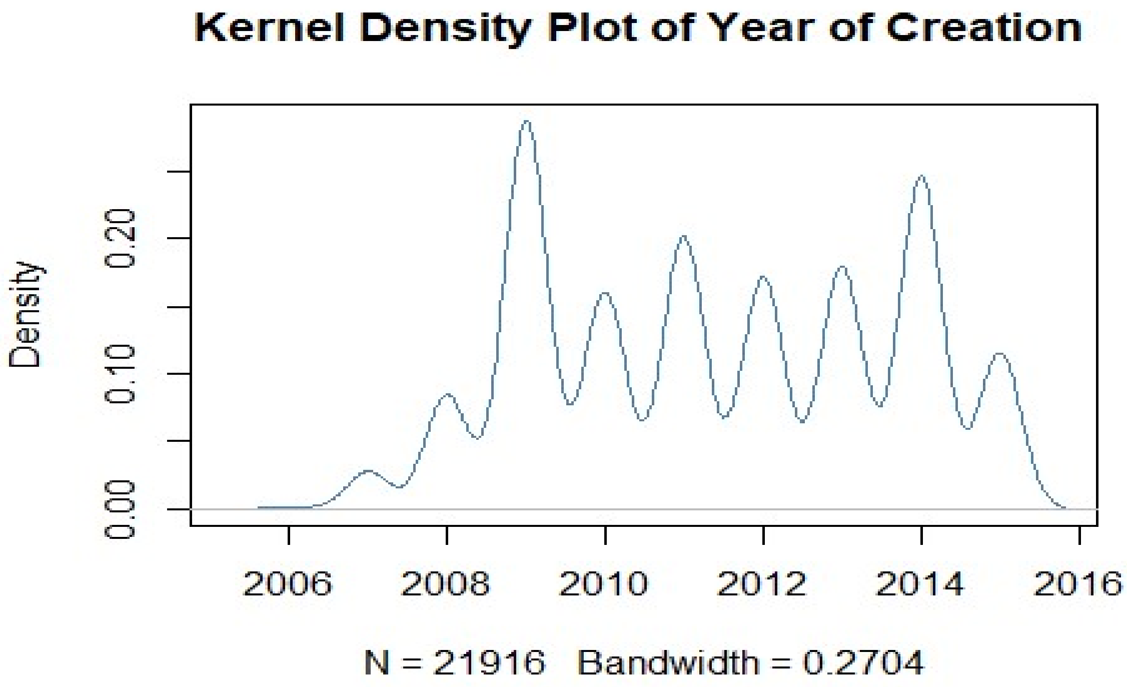
```
library(wesanderson)
countries  <- c("UK", "Canada", "India", "Australia", "US")
Twitter_accounts <- c(650, 1000, 900, 300, 14900)
piepercent<- round(100*Twitter_accounts/sum(Twitter_accounts), 1)
plot.new()
# Plot the chart.
par(mfrow = c(1, 2))
pie(Twitter_accounts, labels = paste(countries, "-",  piepercent, "%"), main
= "Pie Chart", col = wes_palette("FantasticFox1", n = 5, type = c("discrete")
))
#legend("topleft", countries, cex = 0.8, fill = wes_palette("FantasticFox1",
n = 5, type = c("discrete")))

pie3D(piepercent, radius=0.9, labels = paste(countries, "-",  piepercent, "%"
), explode=0,main="3D Pie Chart",col = wes_palette("FantasticFox1", n = 5, ty
pe = c("discrete")))

#legend("topleft", countries, cex = 0.8, fill = wes_palette("FantasticFox1",
n = 5, type =  c("discrete")))
```



Pie Chart

3D Pie Chart

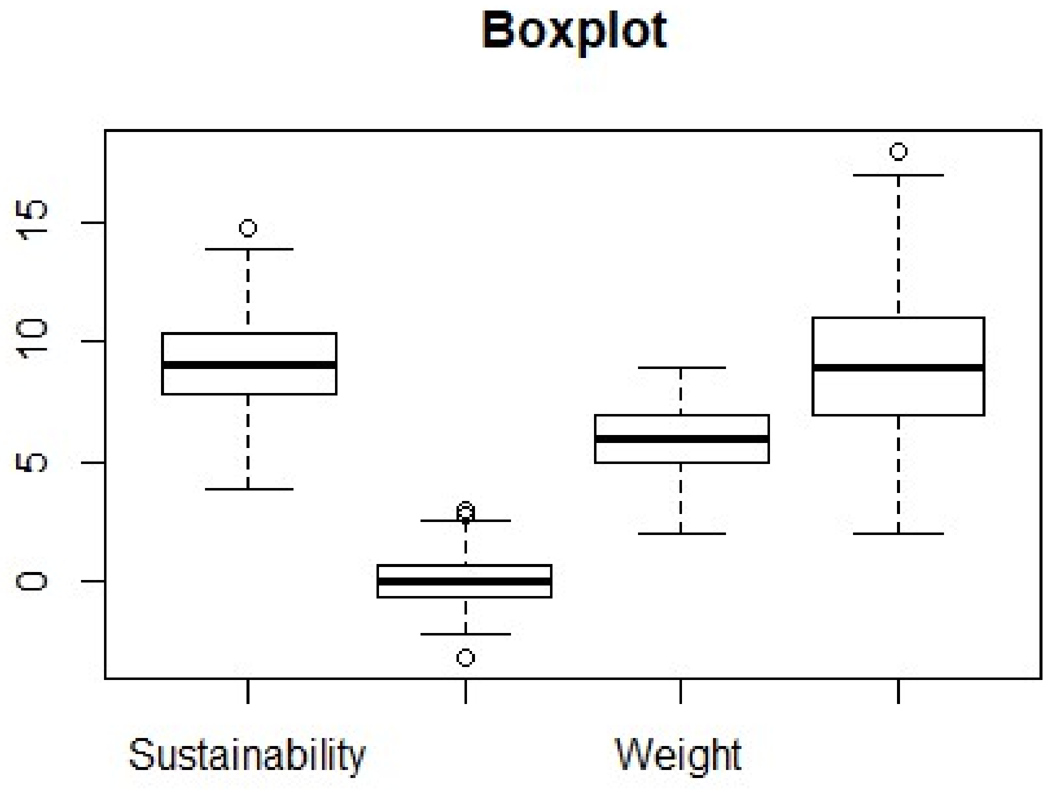
```
# f. Create kernel density plot of created_at_year variable and interpret the
result
plot(density(M01_quasi_twitter$created_at_year), main = "Kernel Density Plot
of Year of Creation ", col = "steelblue")
```



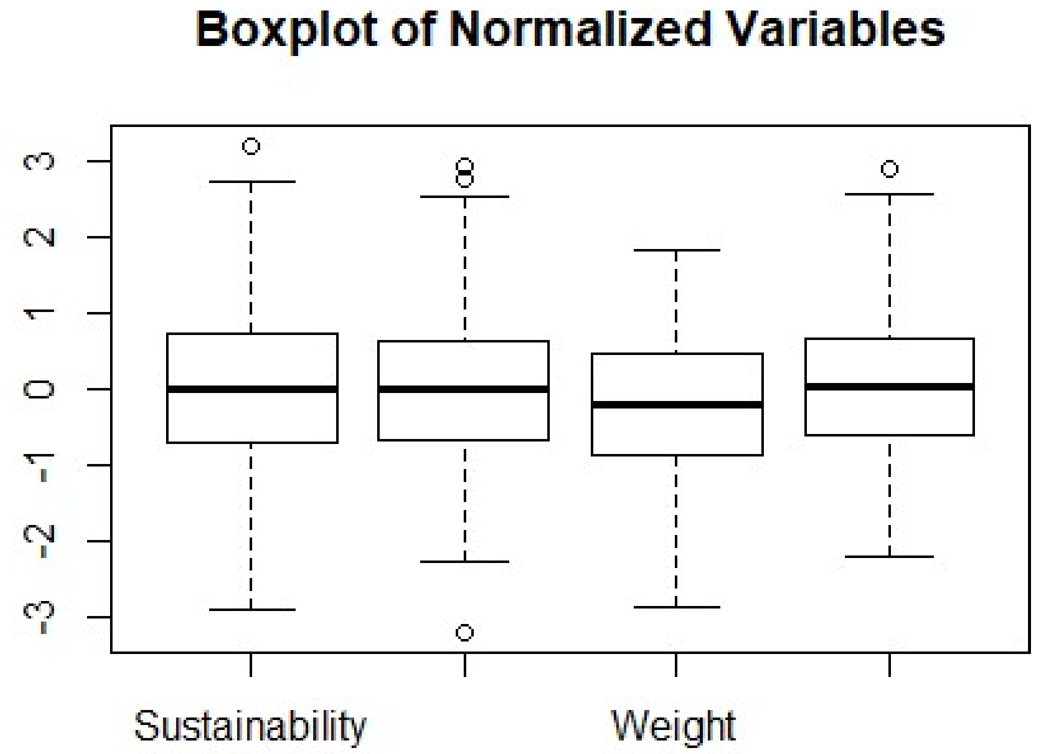
The peaks in the density plot show us where the values are concentrated throughout the time period. The density plot herein is wavy in nature.

#Question 3

```
# Import the Dataset
raw_data <- read.csv("C:/Users/abhil/Downloads/raw_data(1).csv", stringsAsFac
tors=FALSE)

# a. Standardize the data and create new dataset with standardized data and n
ame it Ndata
Ndata <- scale(raw_data, center = TRUE, scale = TRUE)
head(Ndata)

##                A          B          C          D
## [1,] -0.46047167 -0.6870000 -0.2019694 -0.2931233
## [2,]  0.82780052 -0.7467798  0.4705888 -0.2931233
## [3,] -0.18769316  0.7693173  0.4705888 -1.2500845
## [4,] -1.41378095  1.5532638 -0.2019694  0.3448509
## [5,]  0.15837732  0.9970078  0.4705888 -0.2931233
## [6,] -0.03285735  0.6893851  0.4705888  0.9828251

apply(Ndata, 2, mean)

##                             A                            B
##  0.0000000000000000159967091  0.0000000000000000006502771
##                             C                            D
##  0.0000000000000000086421006 -0.0000000000000000138819553

apply(Ndata, 2, sd)

## A B C D
## 1 1 1 1

# b. Create the boxplot of all the variables in their original form.
boxplot(raw_data, main="Boxplot", names = c("Sustainability", "Carbon Footpri
nt", "Weight", "Required Power"))
```

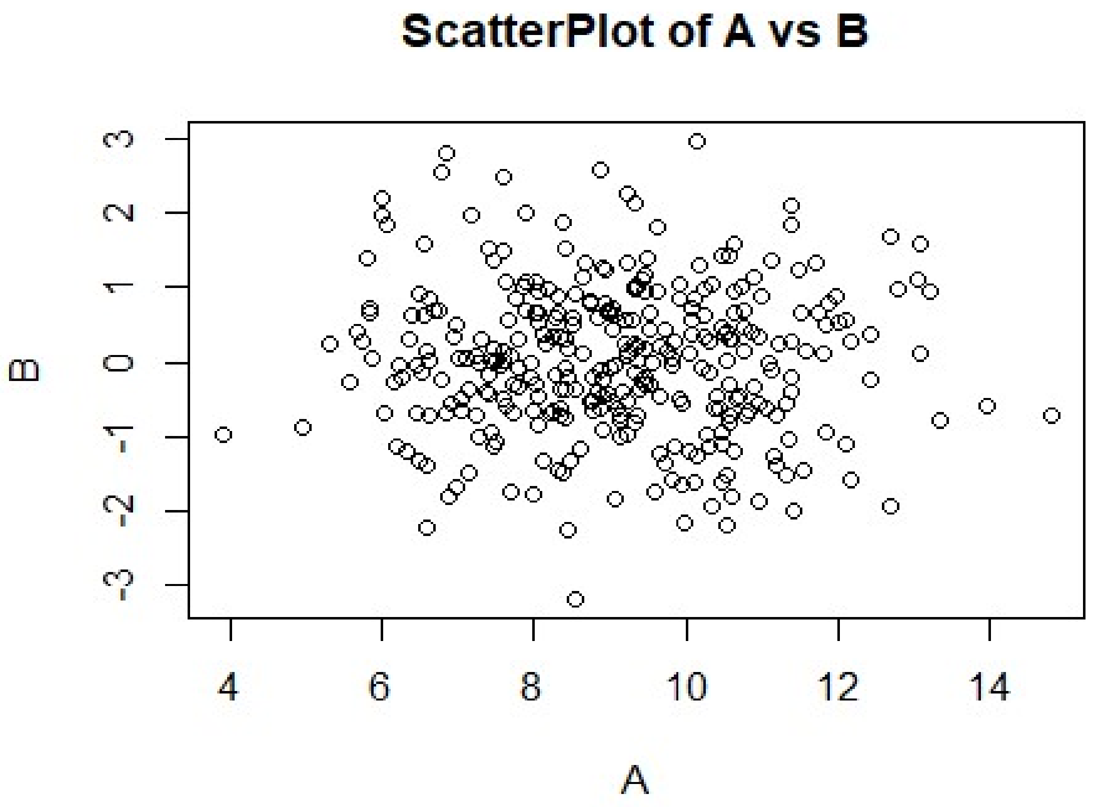


```r
# c. Create boxplot of all the variables in their standardized form.
boxplot(Ndata, main="Boxplot of Normalized Variables", names = c("Sustainability", "Carbon Footprint", "Weight", "Required Power"))
```



d. Compare the result of part b and part c; interpret your answer A. The median of the box plots in the raw data show large variations and there are visible outliers as well. After Standardizing the dataset, we observe that the median of the variables are aligned closely with each other. Standardization did not affect the outliers in the dataset. Standardization affects the Inter Quantile Range; IQR widened in the standard data.

```r
# e. Prepare scatter plot of variables A and B. How are the data correlated in these variables? Interpret your answer.
plot(x = raw_data$A, y = raw_data$B,
     main = "ScatterPlot of A vs B",
     xlab = "A",
     ylab = "B")
```



There is no apparent correlation between variables A and B as we can see the data points are scattered and there is no pattern we can deduce from the same.