Mini Project 2

Name: Tejas Ravi Rao (txr171830)

Section – 1

Q1(a)

After reading the roadrace.csv file from specified path, the first step was to obtain data of the “Maine“ variable which identifies whether a runner is from Maine or from somewhere else (using Maine and Away)

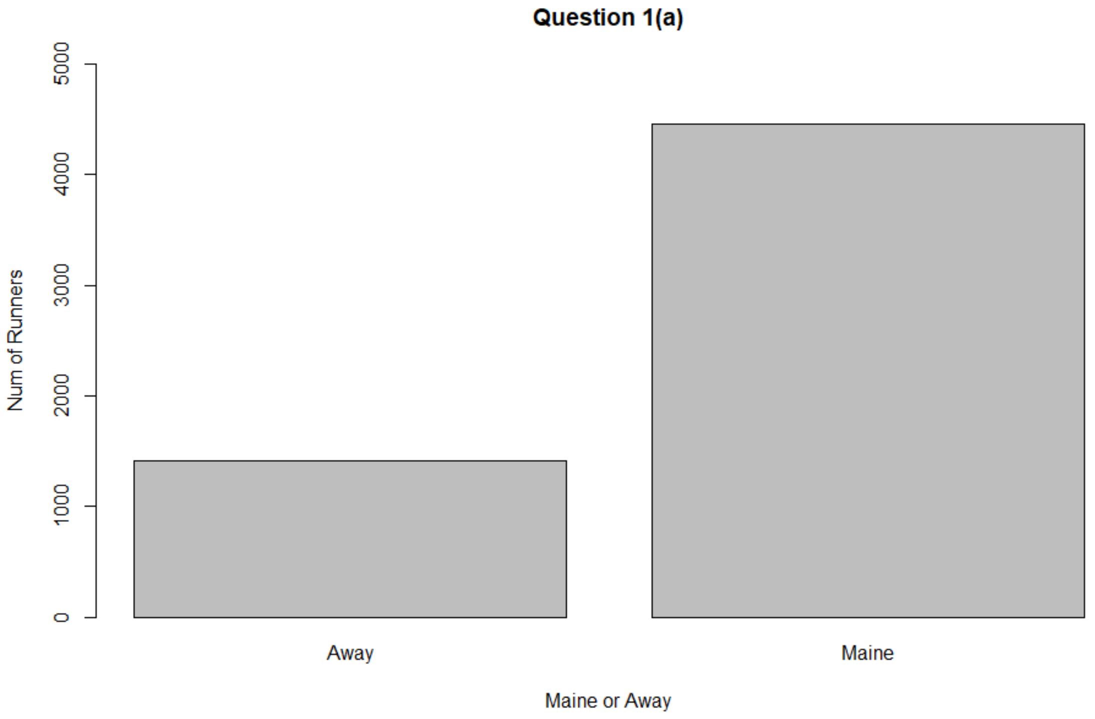
```
# Read csv file from path
raceData = read.csv("D:/Users/rao29/Documents/Sem 5/Stats for Data Science/MP2/roadrace.csv", na.strings = "*")
```

```
# Store Maine column from csv into MaineVar
MaineVar = raceData$Maine
```

We then need to plot graph of the variable Maine. For this the *barplot* function is used. I have set the main, xlab, ylab and ylim parameters in the *barplot* function.

```
# Use barplot to show bar graph of Maine variable
barplot(table(MaineVar), main = "Question 1(a)", xlab = "Maine or Away", ylab = "Num of Runners", ylim = c(0,5000))
```

The following bar graph is shown



From the bar graph we can observe that the number of runners from Maine are about three times the number of runners not from Maine(Away). We can get the exact numbers by obtaining the summary of Maine variable.

```
# Summary of Maine or Away runners
summary(MaineVar)
```

We get the following output:

```
Away Maine
1417  4458
```

Q1(b)

We need runner’s data for the Maine group and Away group separately. For this I have used the subset() function to store Maine runner’s and Away runner’s data into *Mgroup* and *Agroup* respectively.

```
# Using subset() function
# to get data for Maine group and Away group separately
Mgroup = subset(raceData, raceData$Maine == "Maine")
Agroup = subset(raceData, raceData$Maine == "Away")
```

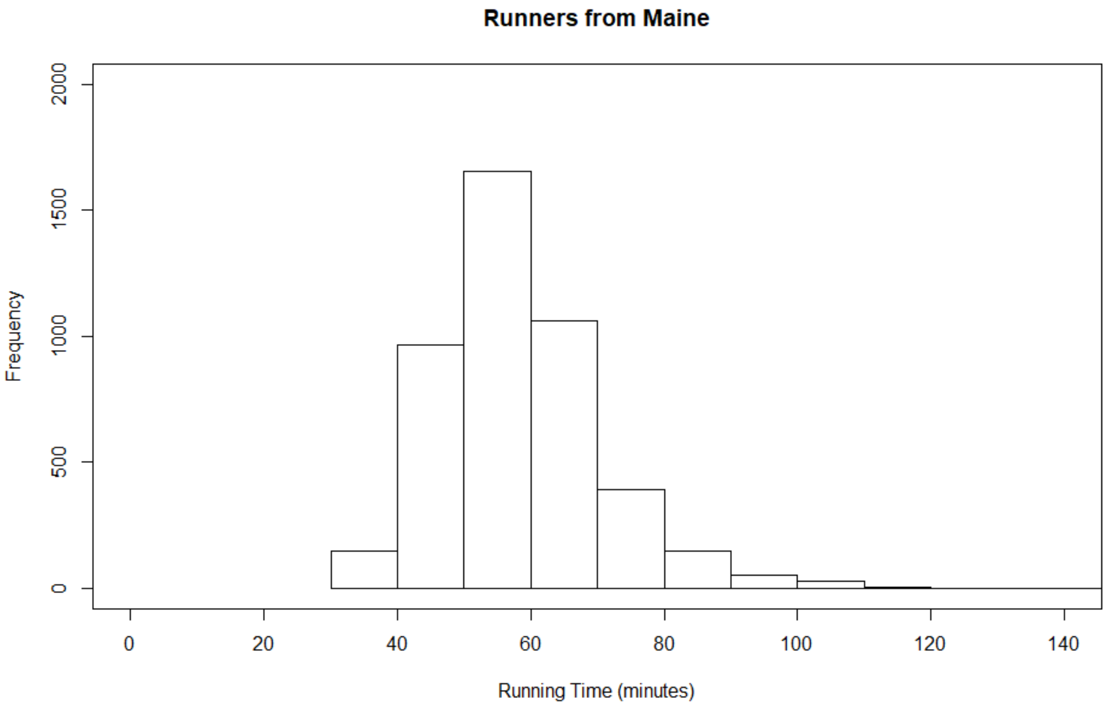
Obtain their respective time data in minutes.

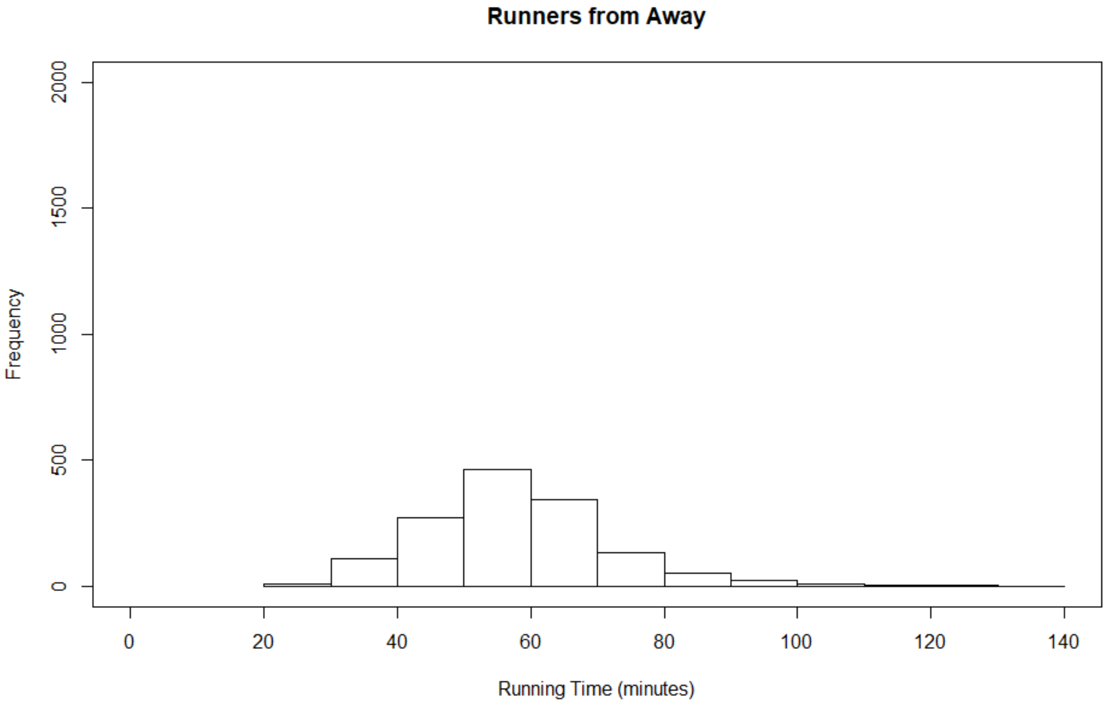
```
# For each of the groups
# get their respective times(in minutes)
Mgroup.time = Mgroup$Time..minutes.
Agroup.time = Agroup$Time..minutes.
```

Using the hist() function to plot the histogram of the runner’s times of both groups. Both histograms are on the same scale. Main, xlim, ylim and xlab parameters have been set for both plots.

```
# Histograms of the runner's times (in minutes)
# for both groups
hist(Mgroup.time, main = "Runners from Maine", xlim = c(0,140), ylim = c(0,2000), xlab = "Running Time (minutes)" )
box()
hist(Agroup.time, main = "Runners from Away", xlim = c(0,140), ylim = c(0,2000), xlab = "Running Time (minutes)" )
box()
```

The following two histogram plots were obtained below.





As we observe the two histograms, we can say that both have similar distribution in terms of their respective running times. We also find that both the distributions are right skewed. The reason being, due to the presence of outliers in the data. There is difference in the maximum and minimum running

times when compared between the Maine group and Away group. Also, from both the histograms we can say that majority of the runners from both groups have running times between 50 to 60 minutes.
As we compare the statistics of both groups, we find a lot of similarities between their respective mean, median, Q1 and Q3 values.
As required the mean, standard deviation, median, range, interquartile range values for both groups have been computed.
Based on the values of Q1 and Q3 quartiles, I have also computed the upper and lower bounds that can be used to detect outliers in Maine and Away groups respectively.
The following code and corresponding outputs are shown for each metric.

```
# Summary for Maine group
# Use () to show the stat directly
(Mgroup.time.stats = summary(Mgroup.time))

Min. 1st Qu.  Median    Mean   3rd Qu.  Max.
 30.57   50.00   57.03   58.20   64.24  152.17
```

```
# Mean - Maine group
(Mgroup.time.mean = mean(Mgroup.time))
[1] 58.19514
```

```
# Standard Deviation - Maine group
(Mgroup.time.sd = sd(Mgroup.time))
[1] 12.18511
```

```
# range - Maine group
(Mgroup.time.range = range(Mgroup.time))
[1]  30.567 152.167
```

```
# median - Maine group
(Mgroup.time.median = median(Mgroup.time))
[1] 57.0335
```

```
# interquartile range - Maine group
(Mgroup.time.iqr = IQR(Mgroup.time))
[1] 14.24775
```

```
# obtain Q1 and Q3 values for Maine group
(Mgroup.time.Q1 = Mgroup.time.stats[2])
(Mgroup.time.Q3 = Mgroup.time.stats[5])
```

```
1st Qu.              3rd Qu.
49.9955              64.24325
```

```
# lower and upper bounds to detect outliers for Maine group
(Mgroup.time.lower = max((Mgroup.time.Q1 - (1.5*Mgroup.time.iqr)),min(Mgroup.time)))
[1] 30.567
(Mgroup.time.upper = min((Mgroup.time.Q3 + (1.5*Mgroup.time.iqr)),max(Mgroup.time)))
[1] 85.61487
```

Similarly metrics for Away group were computed. Observe and compare summary data of both groups.

```
# Summary for Away group
# Use () to show the stat directly
(Agroup.time.stats = summary(Agroup.time))

Min. 1st Qu.  Median    Mean  3rd Qu.    Max.
  27.78   49.15   56.92   57.82   64.83  133.71

# Mean - Away group
(Agroup.time.mean = mean(Agroup.time))
[1] 57.82181

# Standard Deviation - Away group
(Agroup.time.sd = sd(Agroup.time))
[1] 13.83538

# range - Away group
(Agroup.time.range = range(Agroup.time))
[1]  27.782 133.710

# median - Away group
(Agroup.time.median = median(Agroup.time))
[1] 56.92

# interquartile range - Away group
(Agroup.time.iqr = IQR(Agroup.time))
[1] 15.674

# obtain Q1 and Q3 values for Away group
(Agroup.time.Q1 = Agroup.time.stats[2])
(Agroup.time.Q3 = Agroup.time.stats[5])
```

```
1st Qu.            3rd Qu.
49.153             64.827
```

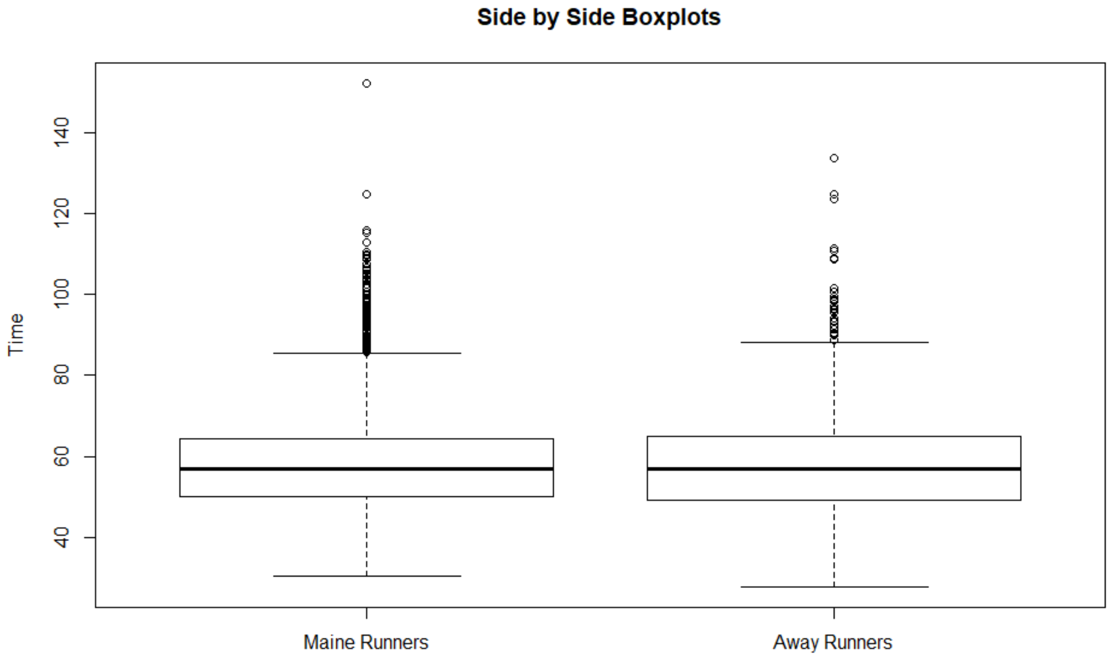
```
# lower and upper bounds to detect outliers for Away group
(Agroup.time.lower = max((Agroup.time.Q1 - (1.5*Agroup.time.iqr)),min(Agroup.time)))
[1] 27.782
(Agroup.time.upper = min((Agroup.time.Q3 + (1.5*Agroup.time.iqr)),max(Agroup.time)))
[1] 88.338
```

As both the distributions are right skewed, we can consider median to be an apt estimator. From the above metric observations, the median for both distributions is approximately 57 mins. I have also computed the respective upper and lower bounds for Maine and Away group respectively.

Using the upper and lower bounds for 1.5IQR rule we can compute the outliers in Maine Data. Here, the outliers must occur below 30.57 mins and above 85.6148 mins. As the min value for Maine group data is 30.57, we observe that the outliers exist only beyond 85.6148 mins up until the max value which is 152.17 mins. This is the reason that the outliers are causing the distribution to be right skewed.

Similarly, using the upper and lower bounds for 1.5IQR rule we can compute the outliers in Away Data. Here the outliers must occur below 27.782 mins and above 88.338 mins. As the min value for Away group data is 27.78, we observe that the outliers exist only beyond 88.338 mins up until the max value which is 133.71 mins. This is the reason that the outliers are causing the distribution to be right skewed.

Q1(c)



We use the same running times of both the groups to plot side by side boxplots. I made use of the boxplot() function to plot the same. The names, main and ylab parameters were set in the function.

```
# side by side box plots
var = c("Maine Runners", "Away Runners")
boxplot(Mgroup.time, Agroup.time, names = var, main = "Side by Side Boxplots", ylab = "Time")
```

The following plot is shown above. As mentioned earlier in Q1(b), We have estimated the median, the Q1 and Q3 quartile values for both Maine and Away Data. The median for both is approximately 57. We have also estimated the upper and lower bounds that are used in 1.5IQR rule to detect outliers in the data.

Therefore, in the boxplots shown, we observe outliers beyond 85.6148 mins in Maine runners Data. Similarly, we observe outliers beyond 88.338 mins in Away runners Data.

To confirm, I have implemented the 1.5IQR rule to find the outliers present in Maine and Away data.

```
# outliers - Maine group
(Mgroup.time.outliers = subset(Mgroup, (Mgroup$Time..minutes. < (Mgroup.time.Q1 -
1.5*Mgroup.time.iqr))|(Mgroup$Time..minutes. > (Mgroup.time.Q3 + 1.5*Mgroup.time.iqr))))

# outiers - Away group
(Agroup.time.outliers = subset (Agroup, (Agroup$Time..minutes. < (Agroup.time.Q1 -
1.5*Agroup.time.iqr)) | (Agroup$Time..minutes. > (Agroup.time.Q3 +1.5*Agroup.time.iqr))))
```

Therefore, from the boxplot we can observe that the distributions are right skewed. These are due to the presence of outliers in both data. Also, the maximum running time is from the Maine runners' group. As we compare the boxplot and the statistics of the data, we find that the summary calculations made are correct and hold correct for the boxplot.

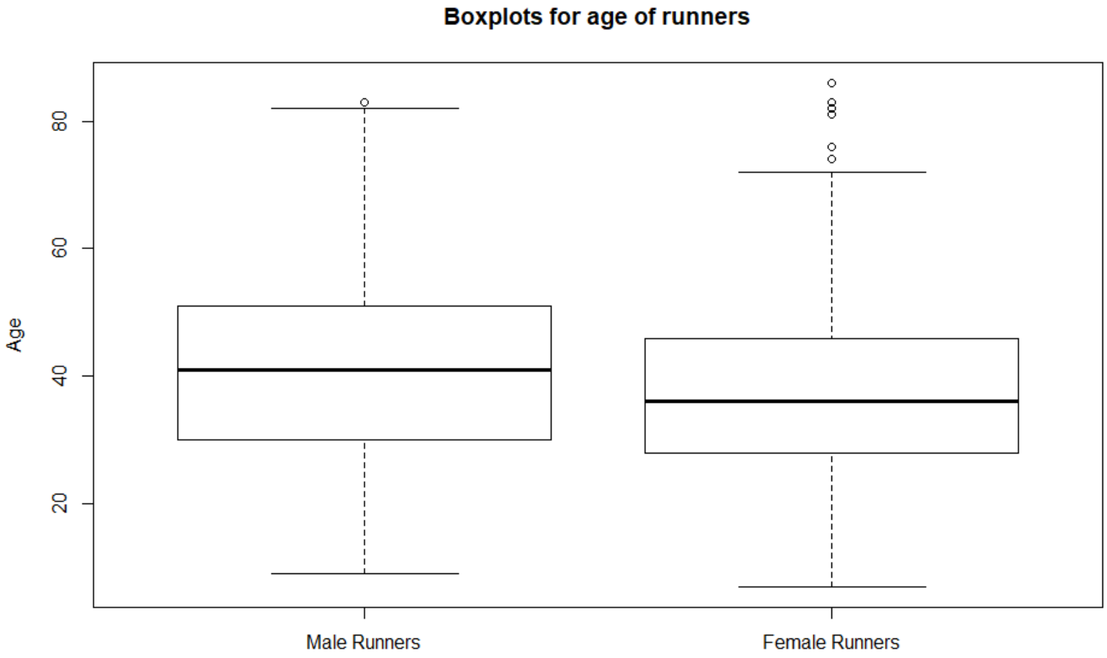
Q1(d)

We need the ages for male and female runners separately, hence here again I have used the subset() function.

```
# Obtain Male and Female Data
genMale = subset(raceData, raceData$Sex == "M")
genFemale = subset(raceData, raceData$Sex == "F")
```

We use the stored data to plot side by side box plots based on the ages of the male and female runners. The names, main and ylab parameters were set. The plot is as shown below.

```
# create side by side boxplots
var2 = c("Male Runners", "Female Runners")
boxplot(genMale$Age, genFemale$Age, names = var2, main = "Boxplots for age of runners", ylab = "Age")
```



As required, the relevant summary statistics for both the ages of male and female runners have been computed. Observe and compare the boxplot with the statistics. The following code and corresponding output for each metric is shown below.

```
# Summary for male age
(genMale.age.stats = summary(genMale$Age))

Min. 1st Qu.  Median    Mean  3rd Qu.    Max.
9.00   30.00  41.00     40.45   51.00     83.00

# stats for Male Age
# mean - Male Age
(genMale.age.mean = mean(genMale$Age))
[1] 40.4468

# standard deviation - Male Age
(genMale.age.sd = sd(genMale$Age))
[1] 13.99289
```

```
# range - Male Age
(genMale.age.range = range(genMale$Age))
[1]  9 83

# median - Male Age
(genMale.age.median = median(genMale$Age))
[1] 41

# interquartile range - Male Age
(genMale.age.iqr = IQR(genMale$Age))
[1] 21

# Q1 and Q3 values - Male Age
(genMale.age.Q1 = genMale.age.stats[2])
(genMale.age.Q3 = genMale.age.stats[5])

1st Qu.                 3rd Qu.
30                      51
```

Similarly, the summary and statistics for Female Age have been recorded.

```
# summary for female age
(genFemale.age.stats = summary(genFemale$Age))

Min. 1st Qu.  Median    Mean   3rd Qu.  Max.
7.00   28.00   36.00       37.24   46.00   86.00

# stats for female age
# mean - Female age
(genFemale.age.mean = mean(genFemale$Age))
[1] 37.23653

# standard deviation - Female Age
(genFemale.age.sd = sd(genFemale$Age))
[1] 12.26925

# range - Female Age
(genFemale.age.range = range(genFemale$Age))
[1]  7 86
```

```
# median - Female Age
(genFemale.age.median = median(genFemale$Age))
[1] 36

# interquartile range - Female Age
(genFemale.age.iqr = IQR(genFemale$Age))
[1] 18

# Q1 and Q3 values for Female Age
(genFemale.age.Q1 = genFemale.age.stats[2])
(genFemale.age.Q3 = genFemale.age.stats[5])

1st Qu.                 3rd Qu.
28                      46
```

As we compare the side by side boxplots with the statistics, we observe that there a couple of outliers among the female runners, whereas there is only one outlier among the male runners. The median age of male runners is more than the median age of female runners. The oldest runner recorded is a female. Due, to more outliers we may also conclude that a greater number of older women participated in the marathon as compared to men.

Q2)

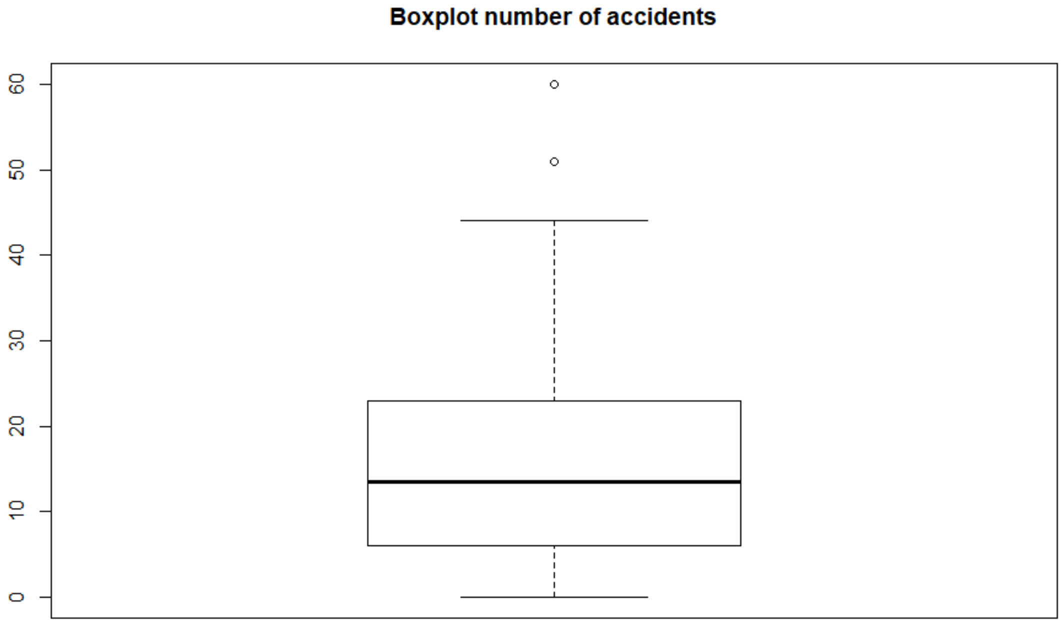
After reading motorcycle.csv from specified path, the first step was to obtain the summary on the motorcycle accidents data. The following stats were obtained as shown below.

```
# Question 2
# read csv file
acc = read.csv("D:/Users/rao29/Documents/Sem 5/Stats for Data Science/MP2/motorcycle.csv")

# summary of accident data
summary(acc)

      County   Fatal.Motorcycle.Accidents
 ABBEVILLE: 1       Min.   : 0.00
 AIKEN    : 1       1st Qu.: 6.00
 ALLENDALE: 1       Median :13.50
 ANDERSON : 1       Mean   :17.02
 BAMBERG  : 1       3rd Qu.:23.00
 BARNWELL : 1       Max.   :60.00
 (Other)  :42
```

The total number of accidents were attained. This will be used to plot boxplot and draw observations from the plot. The main parameter was set to describe the boxplot.

```
# total number of accidents
acc.total = acc$Fatal.Motorcycle.Accidents
```

```
# boxplot of number of accidents
boxplot(acc.total, main = "Boxplot number of accidents")
```



Observing the boxplot, we find that there are 2 counties considered to be outliers. As required, statistics on total number of accidents have been obtained. Compare and observe with boxplot to draw conclusions on data.

```
# statistics on number of accidents
(acc.total.stats = summary(acc.total))

Min.  1st Qu.  Median  Mean   3rd Qu.  Max.
0.00    6.00    13.50   17.02   23.00  60.00
```

```
# mean - number of accidents
(acc.total.mean = mean(acc.total))
[1] 17.02083
```

```
# standard deviation - number of accidents
(acc.total.sd = sd(acc.total))
[1] 13.81256
```

```
# median - number of accidents
(acc.total.median = median(acc.total))
[1] 13.5

# Q1 and Q3 values
(acc.total.Q1 = acc.total.stats[2])
(acc.total.Q3 = acc.total.stats[5])

1st Qu.                    3rd Qu.
  6                          23

# interquartile range - number of accidents
(acc.total.iqr = IQR(acc.total))
[1] 17

# obtain the outliers
(acc.total.outliers = subset(acc, ((acc.total < (acc.total.Q1 - 1.5*acc.total.iqr))|(acc.total > (acc.total.Q3 +
1.5*acc.total.iqr)))))

    County          Fatal.Motorcycle.Accidents
23 GREENVILLE                   51
26     HORRY                    60
```

It is observed that the counties of Greenville and Horry are the outliers. We also observe that the distribution is right skewed. This is due to the presence of outliers. The mean observed is 17 and median is 13.50. The interquartile range is obtained which is used in 1.5IQR rule to detect the outlier counties. We also observe that given data is insufficient to say why these two counties have the highest number of motorcycle accidents when compared to others.

Section - 2

R code

Q1)

```
# Read csv file from path
raceData = read.csv("D:/Users/rao29/Documents/Sem 5/Stats for Data Science/MP2/roadrace.csv",
na.strings = "*")

# Store Maine column from csv into MaineVar
MaineVar = raceData$Maine
```

```
# Question 1(a)
# Use barplot to show bar graph of Maine variable
barplot(table(MaineVar), main = "Question 1(a)", xlab = "Maine or Away", ylab = "Num of Runners", ylim = c(0,5000))

# Summary of Maine or Away runners
summary(MaineVar)

# Question 1(b)
# Using subset() function
# to get data for Maine group and Away group separately
Mgroup = subset(raceData, raceData$Maine == "Maine")
Agroup = subset(raceData, raceData$Maine == "Away")

# For each of the groups
# get their respective times(in minutes)
Mgroup.time = Mgroup$Time..minutes.
Agroup.time = Agroup$Time..minutes.

# Histograms of the runner's times (in minutes)
# for both groups
hist(Mgroup.time, main = "Runners from Maine", xlim = c(0,140), ylim = c(0,2000), xlab = "Running Time (minutes)" )
box()
hist(Agroup.time, main = "Runners from Away", xlim = c(0,140), ylim = c(0,2000), xlab = "Running Time (minutes)" )
box()

# Provide Statistics
# Summary for Maine group
# Use () to show the stat directly
(Mgroup.time.stats = summary(Mgroup.time))

# Mean - Maine group
(Mgroup.time.mean = mean(Mgroup.time))

# Standard Deviation - Maine group
(Mgroup.time.sd = sd(Mgroup.time))

# range - Maine group
(Mgroup.time.range = range(Mgroup.time))
```

```
# median - Maine group
(Mgroup.time.median = median(Mgroup.time))

# interquartile range - Maine group
(Mgroup.time.iqr = IQR(Mgroup.time))

# obtain Q1 and Q3 values for Maine group
(Mgroup.time.Q1 = Mgroup.time.stats[2])
(Mgroup.time.Q3 = Mgroup.time.stats[5])

# lower and upper bounds to detect outliers for Maine group
(Mgroup.time.lower = max((Mgroup.time.Q1 - (1.5*Mgroup.time.iqr)),min(Mgroup.time)))
(Mgroup.time.upper = min((Mgroup.time.Q3 + (1.5*Mgroup.time.iqr)),max(Mgroup.time)))

# Summary for Away group
# Use () to show the stat directly
(Agroup.time.stats = summary(Agroup.time))

# Mean - Away group
(Agroup.time.mean = mean(Agroup.time))

# Standard Deviation - Away group
(Agroup.time.sd = sd(Agroup.time))

# range - Away group
(Agroup.time.range = range(Agroup.time))

# median - Away group
(Agroup.time.median = median(Agroup.time))

# interquartile range - Away group
(Agroup.time.iqr = IQR(Agroup.time))

# obtain Q1 and Q3 values for Away group
(Agroup.time.Q1 = Agroup.time.stats[2])
(Agroup.time.Q3 = Agroup.time.stats[5])

# lower and upper bounds to detect outliers for Away group
(Agroup.time.lower = max((Agroup.time.Q1 - (1.5*Agroup.time.iqr)),min(Agroup.time)))
(Agroup.time.upper = min((Agroup.time.Q3 + (1.5*Agroup.time.iqr)),max(Agroup.time)))
```

```
# Question 1(c)
# side by side box plots
var = c("Maine Runners", "Away Runners")
boxplot(Mgroup.time, Agroup.time, names = var, main = "Side by Side Boxplots", ylab = "Time")

# outliers - Maine group
(Mgroup.time.outliers = subset(Mgroup, (Mgroup$Time..minutes. < (Mgroup.time.Q1 -
1.5*Mgroup.time.iqr))|(Mgroup$Time..minutes. > (Mgroup.time.Q3 + 1.5*Mgroup.time.iqr))))

# outliers - Away group
(Agroup.time.outliers = subset (Agroup, (Agroup$Time..minutes. < (Agroup.time.Q1 -
1.5*Agroup.time.iqr)) | (Agroup$Time..minutes. > (Agroup.time.Q3 +1.5*Agroup.time.iqr))))

# Question 1(d)
# Obtain Male and Female Data
genMale = subset(raceData, raceData$Sex == "M")
genFemale = subset(raceData, raceData$Sex == "F")

# create side by side boxplots
var2 = c("Male Runners", "Female Runners")
boxplot(genMale$Age, genFemale$Age, names = var2, main = "Boxplots for age of runners", ylab =
"Age")

# Summary for male age
(genMale.age.stats = summary(genMale$Age))

# stats for Male Age
# mean - Male Age
(genMale.age.mean = mean(genMale$Age))

# standard deviation - Male Age
(genMale.age.sd = sd(genMale$Age))

# range - Male Age
(genMale.age.range = range(genMale$Age))

# median - Male Age
(genMale.age.median = median(genMale$Age))

# interquartile range - Male Age
(genMale.age.iqr = IQR(genMale$Age))
```

```
# Q1 and Q3 values - Male Age
(genMale.age.Q1 = genMale.age.stats[2])
(genMale.age.Q3 = genMale.age.stats[5])

# summary for female age
(genFemale.age.stats = summary(genFemale$Age))

# stats for female age
# mean - Female age
(genFemale.age.mean = mean(genFemale$Age))

# standard deviation - Female Age
(genFemale.age.sd = sd(genFemale$Age))

# range - Female Age
(genFemale.age.range = range(genFemale$Age))

# median - Female Age
(genFemale.age.median = median(genFemale$Age))

# interquartile range - Female Age
(genFemale.age.iqr = IQR(genFemale$Age))

# Q1 and Q3 values for Female Age
(genFemale.age.Q1 = genFemale.age.stats[2])
(genFemale.age.Q3 = genFemale.age.stats[5])
```

Q2)

```
# Question 2
# read csv file
acc = read.csv("D:/Users/rao29/Documents/Sem 5/Stats for Data Science/MP2/motorcycle.csv")

# summary of accident data
summary(acc)

# total number of accidents
acc.total = acc$Fatal.Motorcycle.Accidents

# boxplot of number of accidents
boxplot(acc.total, main = "Boxplot number of accidents")
```

```
# statistics on number of accidents
(acc.total.stats = summary(acc.total))

# mean - number of accidents
(acc.total.mean = mean(acc.total))

# standard deviation - number of accidents
(acc.total.sd = sd(acc.total))

# median - number of accidents
(acc.total.median = median(acc.total))

# Q1 and Q3 values
(acc.total.Q1 = acc.total.stats[2])
(acc.total.Q3 = acc.total.stats[5])

# interquartile range - number of accidents
(acc.total.iqr = IQR(acc.total))

# obtain the outliers
(acc.total.outliers = subset(acc, ((acc.total < (acc.total.Q1 - 1.5*acc.total.iqr))|(acc.total >
(acc.total.Q3 + 1.5*acc.total.iqr)))))
```