Susheela Polepalli

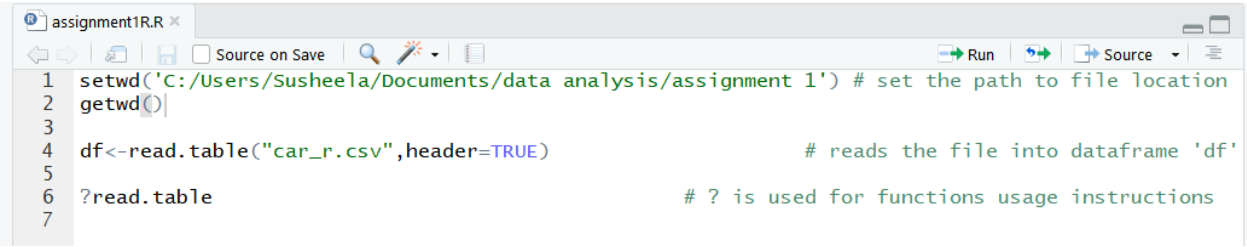
M10727836

BANA7038 Homework 1

1. Import the CSV file 'car_r.csv' using the function "read.table()" or "read.csv()". Where to find the instruction on how to use the functions?

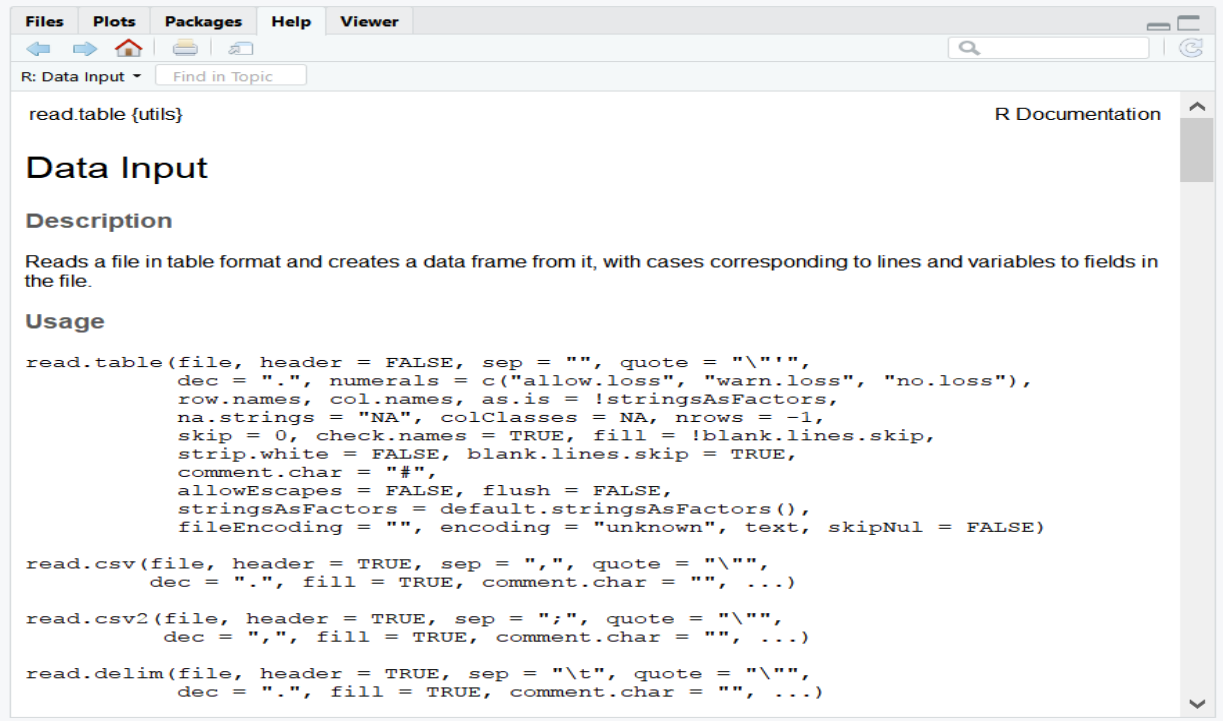
Solution:

We can use the 'read.table()' function to import csv file "car_r.csv"



? is used to find the instruction on how to use functions.

Below shows the output of the command '?read.table'



2. How many variables in the data set? What are their names?

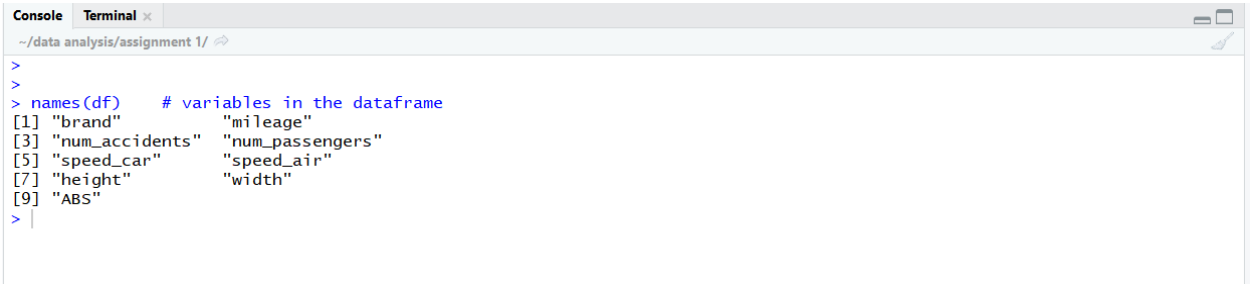
Solution:

There are 9 variables in the dataset.

Command used is:

```
8  names(df)     # variables in the dataframe
```

The output console shows the list of variables names in the data set.



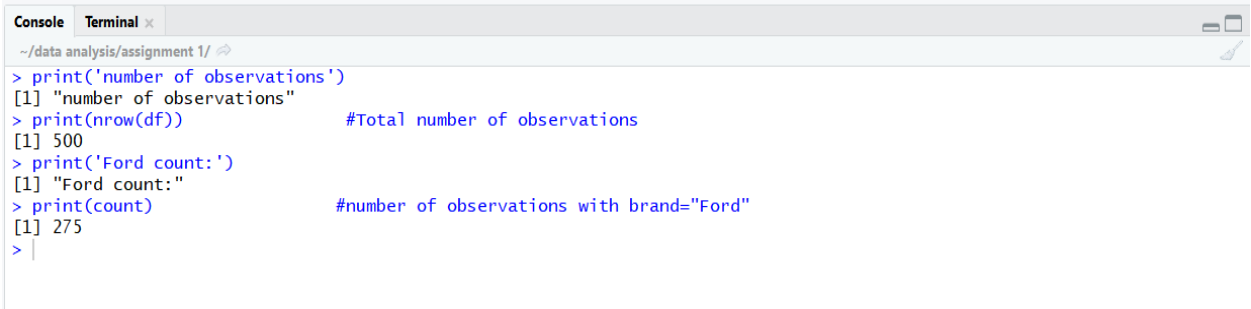
3. How many observations in total? How many observations for Ford?

Solution:

Code for Number of observations and Ford Observations are:

```
9
10 count=0     # Number of observations with brand="ford"
11 for(i in 1:500){
12   if(df[i,1]=='Ford'){
13     count=count+1
14   }
15 }
16 print('number of observations')
17 print(nrow(df))                #Total number of observations
18 print('Ford count:')
19 print(count)                   #number of observations with brand="Ford"
20
```

Output:



Number of observations are 500 and number of Ford observations are 275.

4. Calculate the mean for each of the car parameters (measures). Please also report the corresponding standard deviation.

Solution:

The mean and standard deviation for corresponding car parameters are obtained as follows:

```
21  colMeans(df[2:8],na.rm=TRUE)
22  print("Standard deviations:")
23  i=0
24  for(i in 2:8){
25    print(colnames(df[i]))
26
27    print(c(sd(as.numeric(df[[i]]),na.rm = TRUE)))
28  }
29
```

The output is:

```
Console  Terminal
~/data analysis/assignment 1/
> colMeans(df[2:8],na.rm=TRUE)
       mileage  num_accidents num_passengers      speed_car      speed_air         height          width
  39564.630393       2.154000       6.690000      50.059638       0.245482       5.914164       6.013667
> print("Standard deviations:")
[1] "Standard deviations:"
> for(i in 2:8){
+   print(colnames(df[i]))
+
+   print(c(sd(as.numeric(df[[i]]),na.rm = TRUE)))
+ }
[1] "mileage"
[1] 10819.68
[1] "num_accidents"
[1] 1.423495
[1] "num_passengers"
[1] 3.742983
[1] "speed_car"
[1] 9.77354
[1] "speed_air"
[1] 3.084353
[1] "height"
[1] 1.054882
[1] "width"
[1] 0.4714572
>
```

	Mileage	Num_accidents	Num_passengers	Speed_car	Speed_air	Height	width
Mean	39564.63	2.154	6.69	50.059	0.24	5.912	6.013
Standard deviation	10819.68	1.423495	3.742	9.773	3.0845	1.054882	0.47145

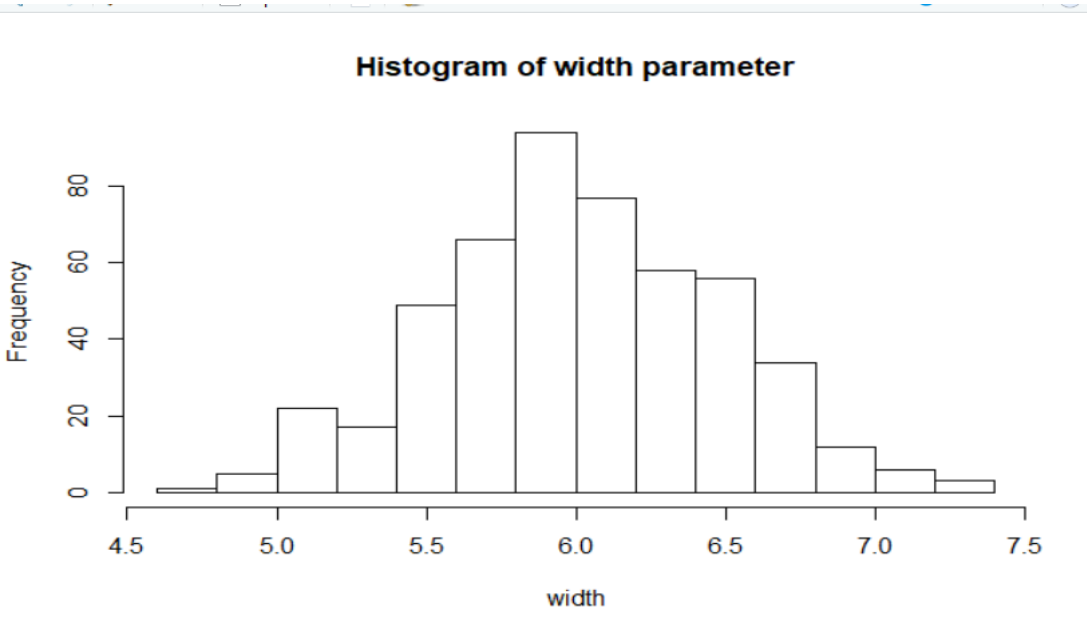
5. Obtain the histogram for each of the car parameters.

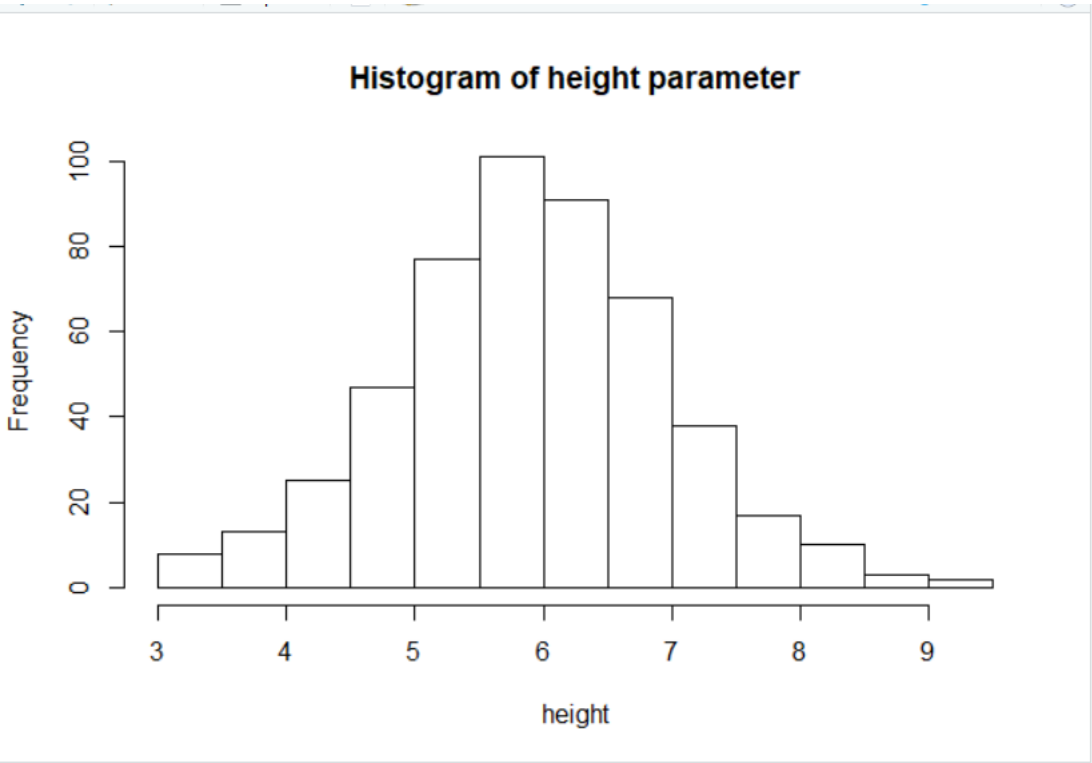
Solution:

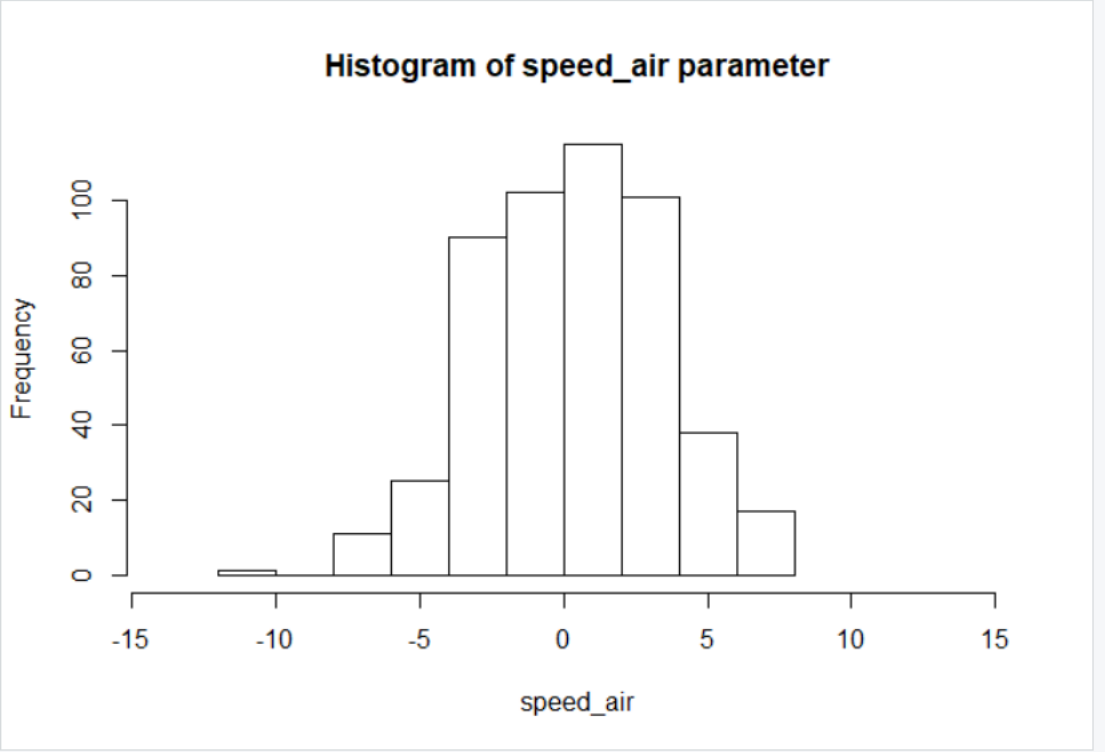
The histogram for each of the car parameters are obtained as follows.

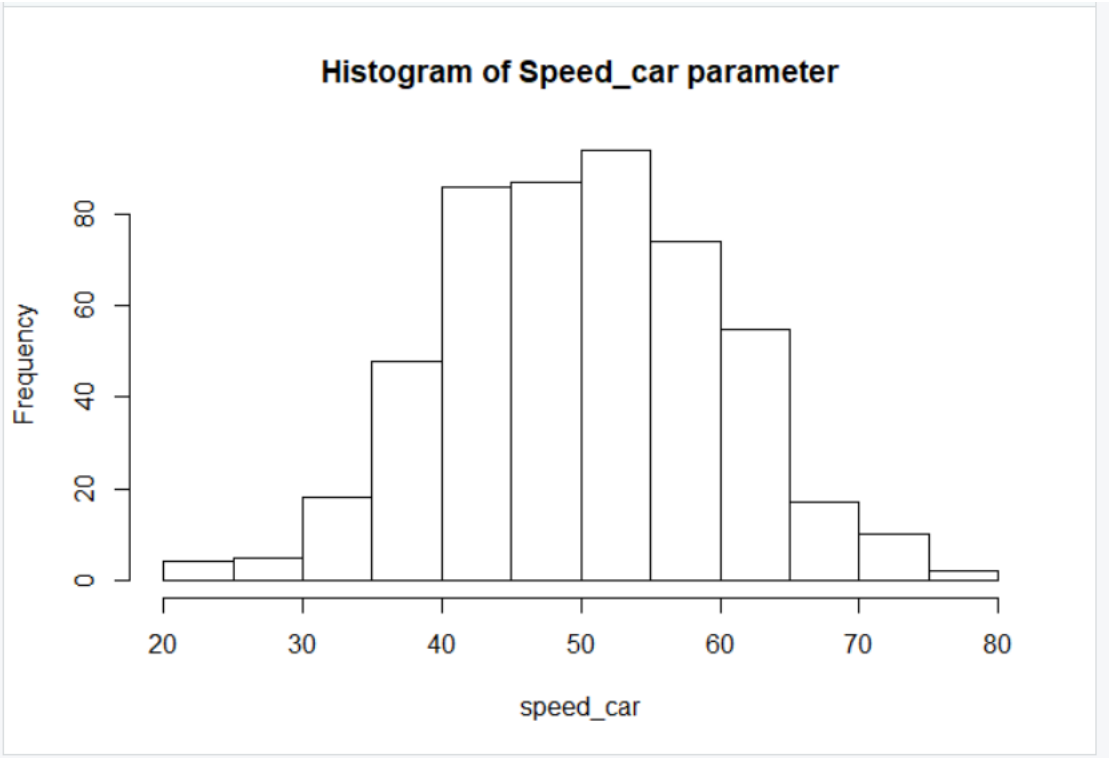
```
32
33 #histogram
34 hist(as.numeric(df$brand),xlab="Brand",main="Histogram of Brand parameter")
35 hist(as.numeric(df$mileage),xlab="mileage",main="Histogram of mileage parameter")
36 hist(as.numeric(df$num_accidents),xlab="num_accidents",main="Histogram of Num_accidents parameter")
37 hist(as.numeric(df$num_passengers),xlab="num_passengers",xlim=c(0,20),main="Histogram of Num_passengers parameter")
38 hist(as.numeric(df$speed_car),xlab="speed_car",main="Histogram of Speed_car parameter")
39 hist(as.numeric(df$speed_air),xlab="speed_air",xlim=c(-14,15),main="Histogram of speed_air parameter")
40 hist(as.numeric(df$height),xlab="height",main="Histogram of height parameter")
41 hist(as.numeric(df$width),xlab="width",main="Histogram of width parameter")
42 hist(as.numeric(df$ABS),xlab="ABS",main="Histogram of ABS Parameter")
```

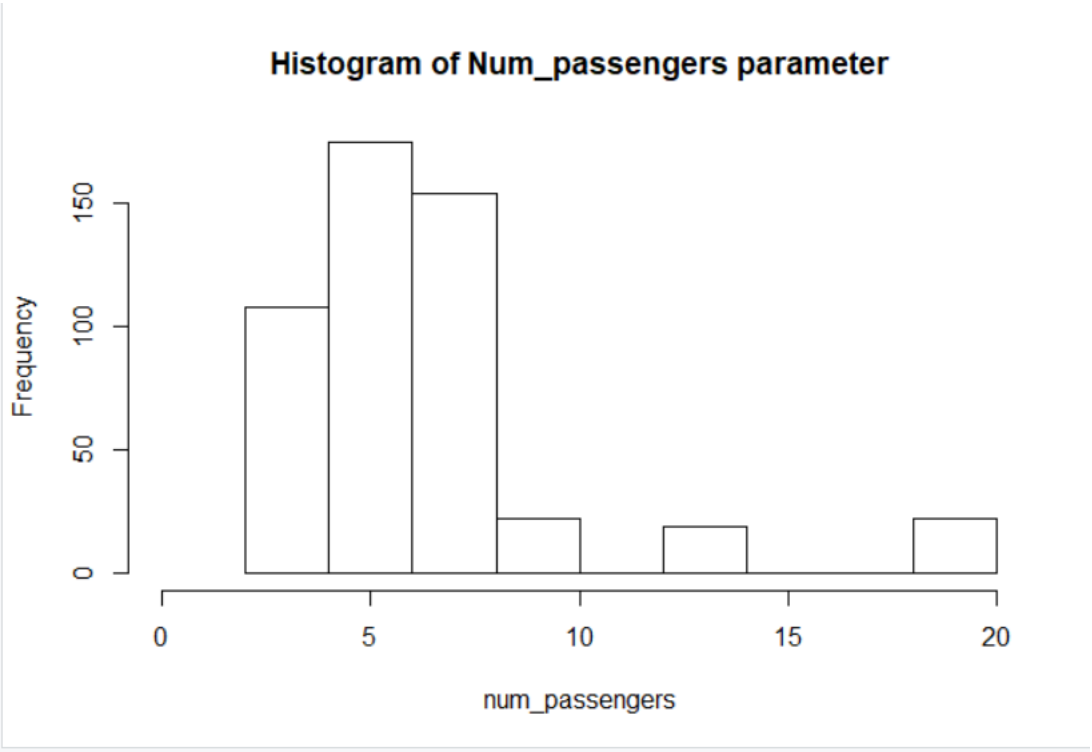
The histograms are

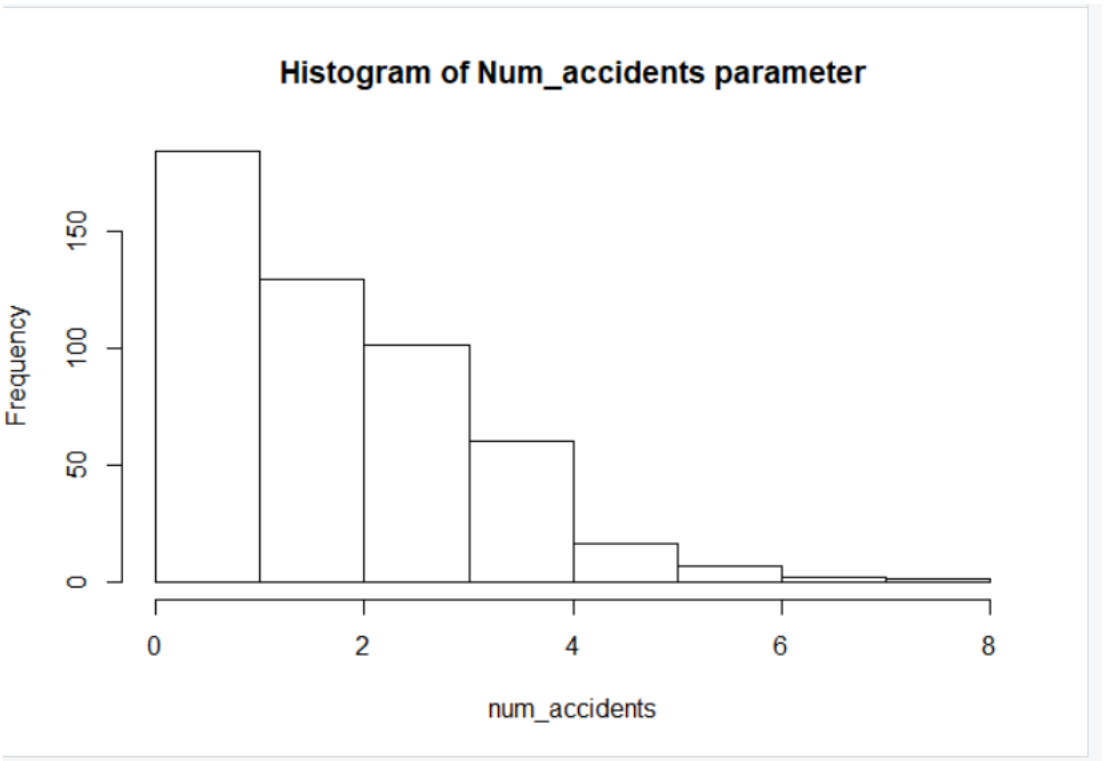


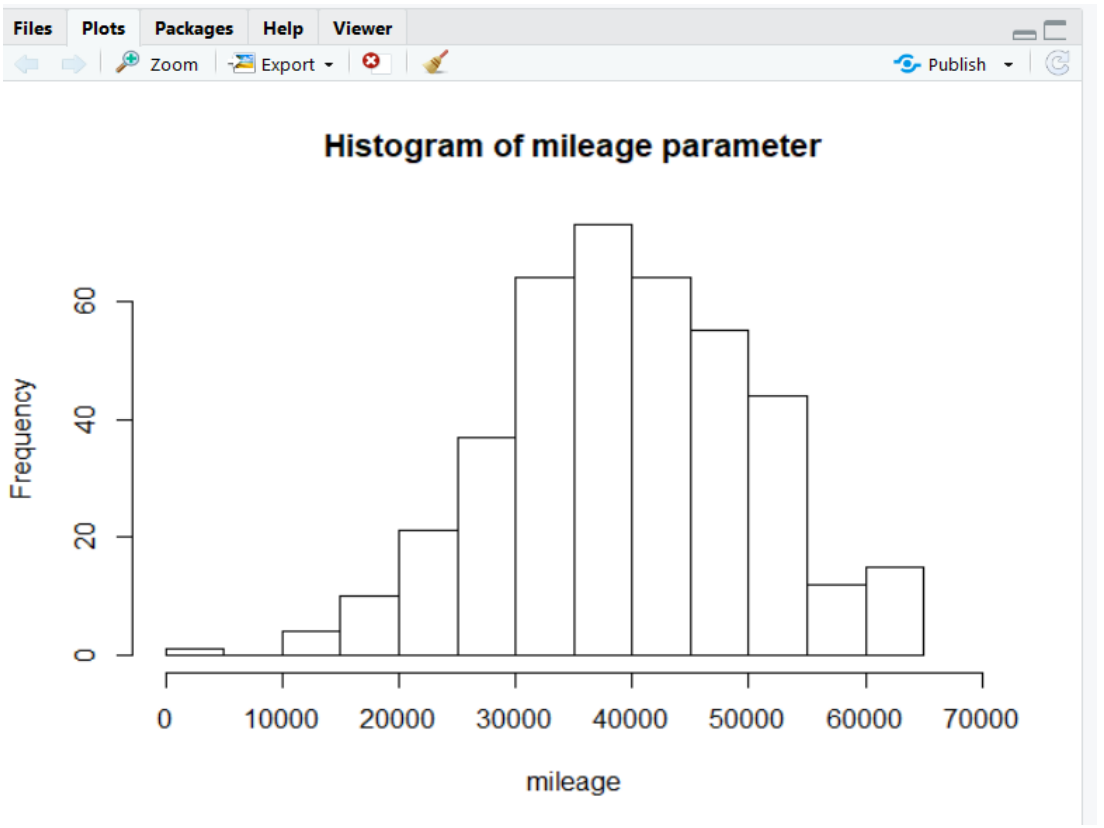

Histogram of height parameter
Frequency
0
20
40
60
80
100
3
4
5
6
7
8
9
height

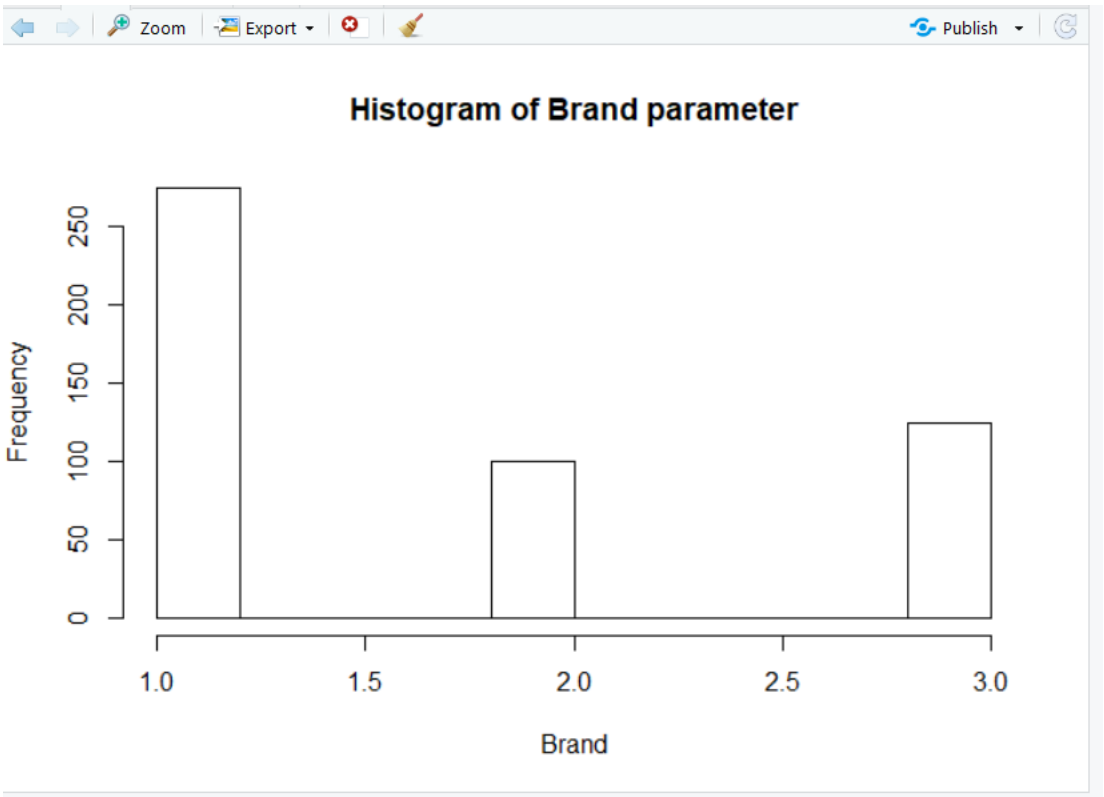

Histogram of speed_air parameter
Frequency
0
20
40
60
80
100
-15
-10
-5
0
5
10
15
speed_air

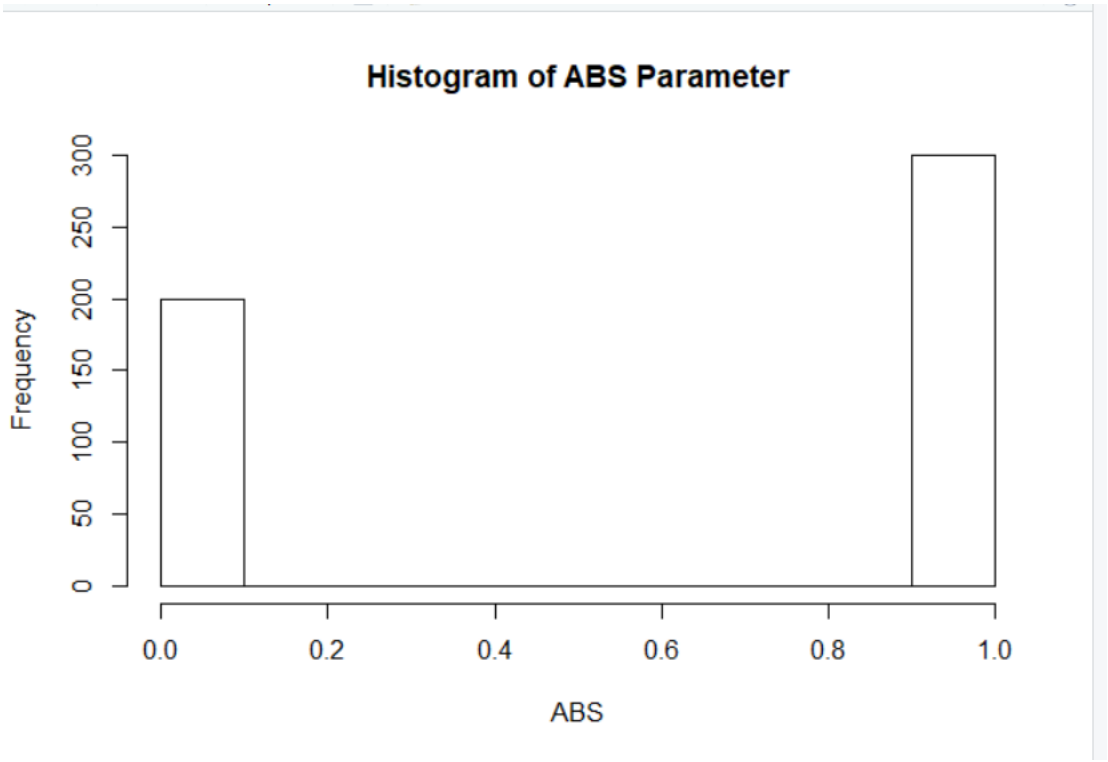

Histogram of Speed_car parameter
Frequency
0
20
40
60
80
20
30
40
50
60
70
80
speed_car


Histogram of Num_passengers parameter
Frequency
0
50
100
150
0
5
10
15
20
num_passengers


Histogram of Num_accidents parameter
Frequency
0
50
100
150
0
2
4
6
8
num_accidents


Files
Plots
Packages
Help
Viewer
Zoom
Export
Publish
Histogram of mileage parameter
Frequency
0
20
40
60
0
10000
20000
30000
40000
50000
60000
70000
mileage


Zoom
Export
Publish
Histogram of Brand parameter
Frequency
0
50
100
150
200
250
1.0
1.5
2.0
2.5
3.0
Brand


Histogram of ABS Parameter
Frequency
0
50
100
150
200
250
300
0.0
0.2
0.4
0.6
0.8
1.0
ABS

6. Is there any missing value in the data set? If yes, which variable? What is the proportion of missing values?

Solution:

Yes. There are missing values in the data set for “mileage” variable.

Code:

```
#Proportion of missing values in Mileage variable
nacount=0
i=0
for (i in 1:nrow(df)){
  if(is.na(as.numeric(df[i,2]))){
    nacount<-nacount+1
  }
}
nacount
print("Proportion of missing values")
print(nacount/nrow(df))
```

Output:

```
> #Proportion of missing values in Mileage variable
> nacount=0
> for (i in 1:nrow(df)){
+   if(is.na(as.numeric(df[i,2]))){
+     nacount=nacount+1
+   }
+ }
> nacount
[1] 100
> print("Proportion of missing values")
[1] "Proportion of missing values"
> print(nacount/nrow(df))
[1] 0.2
>
```

The proportion of missing values in mileage parameter is 0.2.

7. Calculate the relative speed of the car (defined as = speed_car + speed_air, where speed_car is always positive and speed_air can be positive or negative). What is the average relative speed of the car? Convert speed_air to absolute value and calculate the average of the absolute value of speed of the air?

Solution:

The code for relative speed of the car and absolute value of speed_air is ass follows:

```
#relative speed
i=0
relativespeed<-list()
for(i in 1:nrow(df)){
  relativespeed[i]<-df[i,5]+df[i,6]
}

print("Average relative speed:")
mean(as.numeric(relativespeed))

#absolute value

mean(abs(df$speed_air))

```

Output:

```
Console  Terminal
~/data analysis/assignment 1/
> print("Average relative speed:")
[1] "Average relative speed:"
> mean(as.numeric(relativespeed))
[1] 50.30512
> #absolute value
> print("Mean absolute value of speed_air")
[1] "Mean absolute value of speed_air"
> mean(abs(df$speed_air))
[1] 2.508476
>
```

The average relative speed is 50.30512. The mean absolute speed_air is 2.508476.

8. How many cars have mileage less than 40000? How many cars have height less than 5? Please delete those observations (i.e., cars whose mileages are less than 40000 and cars whose heights are less than 5) and delete the observations that contain NAs from the original data set to form a new data set.

Solution:

The following code is used to count the mileage <40000, cars height <5 and to delete the observations from the original dataset to form new data set "updatedDF".

```
#question 8

countmileage=0

for(i in 1:nrow(df)){
  if(df[i,2]<40000 | is.na(df[i,2])){
    countmileage=countmileage+1
  }

}
print("Number of records with Mileage less than 40000 is")
countmileage
i=0
countheight=0
for (i in 1:nrow(df)){
  if(df[i,7]<5){
    countheight=countheight+1
  }
}
print("Number of records with height less than 5 is")
countheight
updatedDF = df[!df$mileage< 40000 & !df$height<5 & !is.na(df$mileage)==TRUE,]
updatedDF
```

Output:

```
> print("Number of records with height less than 5 is")
[1] "Number of records with height less than 5 is"
> countheight
[1] 93
>
```

```
> print("Number of records with Mileage less than 40000 is")
[1] "Number of records with Mileage less than 40000 is"
> countmileage
[1] 310
>
```

Number of records with mileage less than 40000 is 310.Number of records with height less than 5 is 93.The new dataset is named as UpdatedDF which has 158 observations and 9 variables.

9. Divide the new data set (as obtained in Step 8) into three subsets: Ford, GM and Toyota.

Solution:

The code for dividing new data set “updatedDF” into three subsets is:

```
96  #question 9
97  df_Ford<-subset(updatedDF,updatedDF$brand=="Ford")
98  df_GM<-subset(updatedDF,updatedDF$brand=="GM")
99  df_Toyota<-subset(updatedDF,updatedDF$brand=="Toyota")
00
```

The dimensions of the subsets are as follows:

Data	
df	500 obs. of 9 variables
df_Ford	86 obs. of 9 variables
df_GM	38 obs. of 9 variables
df_Toyota	34 obs. of 9 variables
updatedDF	158 obs. of 9 variables

10. Using the new data set (as obtained in Step 8), is there any difference between these three brands (in terms of speed, height, width)? You can compare their means, variances.

Solution:

The means and standard deviations of speed, height and width are computed ans the means of other car parameters such as number of passengers, number of accidents and number of cars are also computed as follows.

```r
#question 10
#speed
mean(df_Ford$speed_car)
mean(df_GM$speed_car)
mean(df_Toyota$speed_car)
sd(as.numeric(df_Ford$speed_car),na.rm = TRUE)
sd(as.numeric(df_GM$speed_car),na.rm = TRUE)
sd(as.numeric(df_Toyota$speed_car),na.rm = TRUE)

dim(df_Ford)
dim(df_GM)
dim(df_Toyota)

#height
mean(df_Ford$height)
mean(df_GM$height)
mean(df_Toyota$height)

sd(as.numeric(df_Ford$height),na.rm = TRUE)
sd(as.numeric(df_GM$height),na.rm = TRUE)
sd(as.numeric(df_Toyota$height),na.rm = TRUE)

#width
mean(df_Ford$width)
mean(df_GM$width)
mean(df_Toyota$width)

sd(as.numeric(df_Ford$width),na.rm = TRUE)
sd(as.numeric(df_GM$width),na.rm = TRUE)
sd(as.numeric(df_Toyota$width),na.rm = TRUE)

#number of accidents
mean(df_Ford$num_accidents)
mean(df_GM$num_accidents)
mean(df_Toyota$num_accidents)

#number of passengers
mean(df_Ford$num_passengers)
mean(df_GM$num_passengers)
mean(df_Toyota$num_passengers)
```

The outputs are as follows:

```
~/data analysis/assignment 1/
> #question 10
> #speed
> mean(df_Ford$speed_car)
[1] 51.43112
> mean(df_GM$speed_car)
[1] 52.23822
> mean(df_Toyota$speed_car)
[1] 52.25184
>
> sd(as.numeric(df_Ford$speed_car),na.rm = TRUE)
[1] 9.666181
> sd(as.numeric(df_GM$speed_car),na.rm = TRUE)
[1] 8.778464
> sd(as.numeric(df_Toyota$speed_car),na.rm = TRUE)
[1] 9.880731
>
> dim(df_Ford)
[1] 86  9
> dim(df_GM)
[1] 38  9
> dim(df_Toyota)
[1] 34  9
>
```

```
> #height
> mean(df_Ford$height)
[1] 6.313565
> mean(df_GM$height)
[1] 6.25896
> mean(df_Toyota$height)
[1] 6.440077
>
> sd(as.numeric(df_Ford$height),na.rm = TRUE)
[1] 0.7370353
> sd(as.numeric(df_GM$height),na.rm = TRUE)
[1] 0.7958463
> sd(as.numeric(df_Toyota$height),na.rm = TRUE)
[1] 0.7955207
>
> #width
> mean(df_Ford$width)
[1] 6.056579
> mean(df_GM$width)
[1] 5.932025
> mean(df_Toyota$width)
[1] 6.048129
>
> sd(as.numeric(df_Ford$width),na.rm = TRUE)
[1] 0.4241807
> sd(as.numeric(df_GM$width),na.rm = TRUE)
[1] 0.4710823
> sd(as.numeric(df_Toyota$width),na.rm = TRUE)
[1] 0.4693645
>
```

Console Terminal

~/data analysis/assignment 1/

```
> #number of accidents
> mean(df_Ford$num_accidents)
[1] 2.383721
> mean(df_GM$num_accidents)
[1] 1.973684
> mean(df_Toyota$num_accidents)
[1] 2.205882
>
> #number of passengers
> mean(df_Ford$num_passengers)
[1] 7.093023
> mean(df_GM$num_passengers)
[1] 6.684211
> mean(df_Toyota$num_passengers)
[1] 6.441176
>
>
```

Summary:

Parameter	Ford	Toyota	GM
Mean speed Mean Standard deviation	51.43112 9.666181	52.25184 9.880731	52.23822 8.778464
Number of cars	86	34	38
Height	6.313565	6.440077	6.25896
Width	6.056579	6.048129	5.932025
Number of accidents	2.383721	2.205882	1.973684
Number of passengers	7.093023	6.4411	6.6842

When comparing the speeds, Toyota is slightly higher than GM followed by Ford.

Number of cars, Ford is having the highest number followed by GM and Toyota.

Height: Toyota > Ford > GM.

Width: Ford > Toyota >GM.

Number of accidents: Ford > Toyota > GM.

Number of passengers: Ford > GM > Toyota.