



AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)
FACULTY OF SCIENCE & TECHNOLOGY

INTRODUCTION TO DATA
SCIENCE [A]

MID PROJECT REPORT ON
Heart disease classification Dataset

Submitted By

Sl No.	Name	ID
1	AYON GHOSH	21-44703-1
2	SANUAR HOSSAIN	21-45298-2

Submitted To

TOHEDUL ISLAM
Assistant Professor,
Department of Computer Science,
Faculty of Science and Technology,
American International University-Bangladesh
(AIUB)

2023-2024, Fall
Date of Submission: November , 2023

Table of content :

SI No.	Task	Page No
1	About Dataset	03
2	Data Annotation	03-04
3	Finding Missing values	04-06
4	Handel Missing values	07-13
5	Finding Outliers values (Boxplot)	14-18
6	Handel Outliers values	19-22
7	Barplot	23-27
8	Histrogram	27-29
9	Standerd Deviation	30

About Dataset

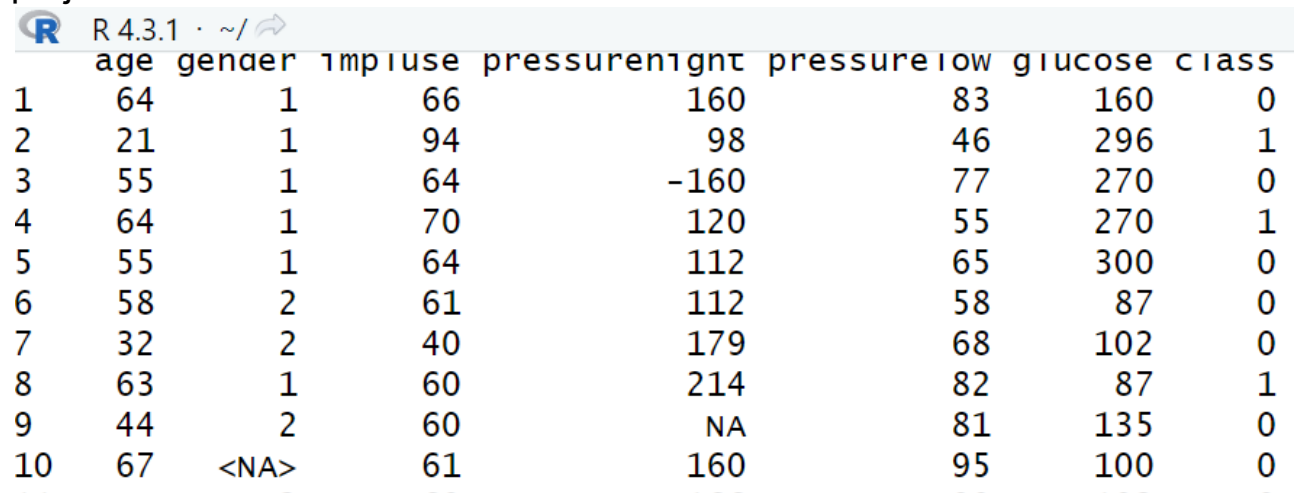
Cardiovascular illnesses (CVDs) are the major cause of death worldwide. CVDs include coronary heart disease, cerebrovascular disease, rheumatic heart disease, and other heart and blood vessel problems. According to the World Health Organization, 17.9 million people die each year. Heart attacks and strokes account for more than four out of every five CVD deaths, with one-third of these deaths occurring before the age of 70. A comprehensive database for factors that contribute to a heart attack has been constructed.

The main purpose here is to collect characteristics of Heart Attack or factors that contribute to it.

The size of the dataset is 1319 samples, which have nine fields, where eight fields are for input fields and one field for an output field. Age, gender(0 for Female, 1 for Male) ,heart rate (impulse), systolic BP (pressurehight), diastolic BP (pressurelow), blood sugar(glucose), CK-MB (kcm), and Test-Troponin (troponin) are representing the input fields, while the output field pertains to the presence of heart attack (class), which is divided into two categories (negative and positive); negative refers to the absence of a heart attack, while positive refers to the presence of a heart attack.

Data Annotation :

```
projectdata$gender<- factor(projectdata$gender, levels=c("male","female"),
labels= c(1,2))
projectdata
projectdata$class<- factor(projectdata$class, levels=c("positive","negative"),
labels= c(1,0))
projectdata
```



R 4.3.1 · ~/

	age	gender	impulse	pressurehight	pressurelow	glucose	class
1	64	1	66	160	83	160	0
2	21	1	94	98	46	296	1
3	55	1	64	-160	77	270	0
4	64	1	70	120	55	270	1
5	55	1	64	112	65	300	0
6	58	2	61	112	58	87	0
7	32	2	40	179	68	102	0
8	63	1	60	214	82	87	1
9	44	2	60	NA	81	135	0
10	67	<NA>	61	160	95	100	0

Here ,we have perform data annotation for columns gender and class. converts catagorical data into numerical data.In gerger column we replace male by 1 and female by 2 and class column 0 for negative and 1 for positive.

Finding Missing Values :

```
which(is.na(projectdata$age))
```

```
which(is.na(projectdata$gender))
```

```
which(is.na(projectdata$impulse))
```

```
which(is.na(projectdata$pressurehight))
```

```
which(is.na(projectdata$pressurelow))
```

```
which(is.na(projectdata$glucose))
```

```
which(is.na(projectdata$class))
```

```
sum(is.na(projectdata))
```

```
is.na(projectdata)
```

```
colSums(is.na(projectdata))
```

Here , we have apply some different functions to find missing value.

```
> which(is.na(projectdata$age))  
[1]  11  25  37  74 122  
~ |
```

This function return the index of missing value of age column.

```
> which(is.na(projectdata$gender))  
[1] 10 39 46  
> |
```

Return the index of missing value of gender column.

```
> which(is.na(projectdata$impulse))
integer(0)
```

Return integer (0) so ,Impluse column has no missing value .

```
> which(is.na(projectdata$pressurehigh))
[1]  9 16
```

Return the index of missing value of pressurehigh column.

```
> which(is.na(projectdata$pressurelow))
integer(0)
```

Return integer (0) so ,pressurelow column has no missing value .

```
> which(is.na(projectdata$glucose))
integer(0)
```

Return integer (0) so ,glucose column has no missing value .

```
> which(is.na(projectdata$class))
integer(0)
```

Return integer (0) so ,class column has no missing value .

which(is.na()) function return missing value index of a colum .

Missing value finding using **is.na()** function

```
> is.na(projectdata)
      age gender impluse pressurehight pressurelow glucose class
[1,] FALSE  FALSE    FALSE             FALSE        FALSE  FALSE  FALSE
[2,] FALSE  FALSE    FALSE             FALSE        FALSE  FALSE  FALSE
[3,] FALSE  FALSE    FALSE             FALSE        FALSE  FALSE  FALSE
[4,] FALSE  FALSE    FALSE             FALSE        FALSE  FALSE  FALSE
[5,] FALSE  FALSE    FALSE             FALSE        FALSE  FALSE  FALSE
[6,] FALSE  FALSE    FALSE             FALSE        FALSE  FALSE  FALSE
[7,] FALSE  FALSE    FALSE             FALSE        FALSE  FALSE  FALSE
[8,] FALSE  FALSE    FALSE             FALSE        FALSE  FALSE  FALSE
[9,] FALSE  FALSE    FALSE             TRUE         FALSE  FALSE  FALSE
[10,] FALSE  TRUE     FALSE             FALSE        FALSE  FALSE  FALSE
```

Here, `is.na()` function check that is there any index has missing value or not if index has missing value its return TRUE for the specific index and FALSE for those index has no missing value. It represent boolean value for missing value.

Missing value finding using **colSum()** function

```
colsums(is.na(projectdata))
      age      gender      impluse pressurehight pressurelow      glucose
      5          3          0          2          0          0
class
0
```

Here, this function return the total number of missing value of a specific column.

Missing value finding using **sum()** function

```
integer(0)
> sum(is.na(projectdata))
[1] 10
```

Here, this function return the total number (10) of missing value of a dataset.

Handel Missing Values :

Apply **na.omit()** function for discard row which have missing value.

```
remove<-na.omit(projectdata)
```

remove

```
> remove<- na.omit(projectdata)
>
> remove
```

	age	gender	implyse	pressurehight	pressurelow	glucose	class
1	64	1	66	160	83	160	0
2	21	1	94	98	46	296	1
3	55	1	64	-160	77	270	0
4	64	1	70	120	55	270	1
5	55	1	64	112	65	300	0
6	58	2	61	112	58	87	0
7	32	2	40	179	68	102	0
8	63	1	60	214	82	87	1
12	63	2	60	150	10	198	0
13	64	1	60	199	5	92	1
14	54	2	94	122	67	97	0
15	47	1	76	120	70	319	0
17	86	2	73	114	68	87	1
18	45	2	70	100	68	96	0
19	37	2	72	107	86	274	0
20	45	1	60	109	65	89	1
21	60	1	92	151	78	301	0
22	48	1	135	98	60	100	1
23	52	1	76	109	85	227	1
24	30	1	63	110	68	107	1
26	72	1	64	106	68	111	1
27	42	1	65	150	68	101	0
28	72	2	64	325	60	95	0
29	47	2	66	134	57	279	1

All the missing value rows are discarded .Like 11,25

for age column

Apply **mean** for handel missing values for age column.

```
mean_value <- mean(projectdata$age, na.rm = TRUE)
```

```
projectdata$age[is.na(projectdata$age)] <-mean_value
```

```
projectdata$age <- ceiling(projectdata$age)
```

projectdata

```

> mean_value <- mean(projectdata$age, na.rm = TRUE)
> projectdata$age[is.na(projectdata$age)] <- mean_value
> projectdata$age <- ceiling(projectdata$age)
> projectdata

```

	age	gender	implyse	pressurehigh	pressurelow	glucose	class
1	64	1	66	160	83	160	0
2	21	1	94	98	46	296	1
3	55	1	64	-160	77	270	0
4	64	1	70	120	55	270	1
5	55	1	64	112	65	300	0
6	58	2	61	112	58	87	0
7	32	2	40	179	68	102	0
8	63	1	60	214	82	87	1
9	44	2	60	122	81	135	0
10	67	<NA>	61	160	95	100	0
11	57	2	60	166	90	102	0

Here, we find mean by mean() function and replace missing value by mean value for age column. Our actual mean is 56.13 ceiling function converts it 57 also replace the value of 11 index by 57.

Apply **median** for handel missing values for age column.

```
median_value <- median(projectdata$age, na.rm = TRUE)
```

```
projectdata$age[is.na(projectdata$age)] <- median_value
```

```
projectdata
```

```

> median_value <- median(projectdata$age, na.rm = TRUE)
> projectdata$age[is.na(projectdata$age)] <- median_value
> projectdata

```

	age	gender	implyse	pressurehigh	pressurelow	glucose	class
1	64	1	66	160	83	160	0
2	21	1	94	98	46	296	1
3	55	1	64	-160	77	270	0
4	64	1	70	120	55	270	1
5	55	1	64	112	65	300	0
6	58	2	61	112	58	87	0
7	32	2	40	179	68	102	0
8	63	1	60	214	82	87	1
9	44	2	60	NA	81	135	0
10	67	<NA>	61	160	95	100	0
11	56	2	60	166	90	102	0

Here, the median is 56 of age column all the missing index replaced by 56

Apply **mode** for handel missing values for age column.

```
mode_val<-mlv(projectdata$age,method="mfv")
```

```
mode_val
```

```
projectdata$age[is.na(projectdata$age)]<-mode_val
```

```
projectdata
```

```
> mode_val<- mlv(projectdata$age, method = "mfv")
> mode_val
[1] 45
> projectdata$age[is.na(projectdata$age)] <-mode_val
> projectdata
```

	age	gender	impluse	pressurehigh	pressurelow	glucose	class
1	64	1	66	160	83	160	0
2	21	1	94	98	46	296	1
3	55	1	64	-160	77	270	0
4	64	1	70	120	55	270	1
5	55	1	64	112	65	300	0
6	58	2	61	112	58	87	0
7	32	2	40	179	68	102	0
8	63	1	60	214	82	87	1
9	44	2	60	NA	81	135	0
10	67	<NA>	61	160	95	100	0
11	45	2	60	166	90	102	0

Here,We find mode value (45) of age column by mlv () function and replace missing value index by this mode value. Like index 11 is has replaced by 45.

for gender column:

Apply **mode** for handel missing values for gender column.gender column is categorical so we can not apply **mean** or **median** for this column.

```
md<-mlv(projectdata$gender,method="mfv")
```

```
md
```

```
projectdata$gender[is.na(projectdata$gender)]<-md
```

```
projectdata
```

```
> projectdata$age[is.na(projectdata$gender)] <- md
> md<- mlv(projectdata$gender, method = "mfv")
> md
[1] 1
Levels: 1 2
> projectdata$gender[is.na(projectdata$gender)] <- md
> projectdata
```

	age	gender	impluse	pressurehight	pressurelow	glucose	class
1	64	1	66	160	83	160	0
2	21	1	94	98	46	296	1
3	55	1	64	-160	77	270	0
4	64	1	70	120	55	270	1
5	55	1	64	112	65	300	0
6	58	2	61	112	58	87	0
7	32	2	40	179	68	102	0
8	63	1	60	214	82	87	1
9	44	2	60	NA	81	135	0
10	1	1	61	160	95	100	0

Here, We find the mode value (1) of gender column and replace missing value index by mode value. Like index 10 has replaced by 1 .

Apply **top-down** for handel missing values for gender column.

```
tp<-projectdata$gender<-na.locf(projectdata$gender)
```

```
tp
```

```
projectdata$gender[is.na(projectdata$gender)]<-tp
```

```
projectdata
```

```
> tp<-projectdata$gender <- na.locf(projectdata$gender)
> tp
 [1] 1 1 1 1 1 2 2 1 2 2 2 2 1 2 1 1 2 2 2 1 1 1 1 1 1 1 2 2
[40] 1 1 1 1 1 1 1 1 1 1 2 1 2 2 2 1 1 1 2 1 1 2 1 1 1 1 1
[79] 1 2 1 2 1 2 1 2 1 1 2 1 1 1 2 2 1 1 1 2 1 1 1 2 2 1 2 1
[118] 1 1 1 1 2 1 1 1 1 1 2 2 1 1 1 2 1 1 1 1 2 1 2 2 1 1 2 1 2
Levels: 1 2
> projectdata$gender[is.na(projectdata$gender)] <- tp
> projectdata
  age gender impluse pressurehigh pressurelow glucose class
1   64      1      66          160           83      160      0
2   21      1      94           98           46      296      1
3   55      1      64          -160          77      270      0
4   64      1      70          120          55      270      1
5   55      1      64          112          65      300      0
6   58      2      61          112          58       87      0
7   32      2      40          179          68      102      0
8   63      1      60          214          82       87      1
9   44      2      60           NA          81      135      0
10  67      2      61          160          95      100      0
11  ...      ...      ...      ...      ...      ...      ...
```

We also apply top-down for the gender column which is find by na.locf() function. The top value of missing index(10) is 2 and its replaced by 2 .

for pressurehigh column:

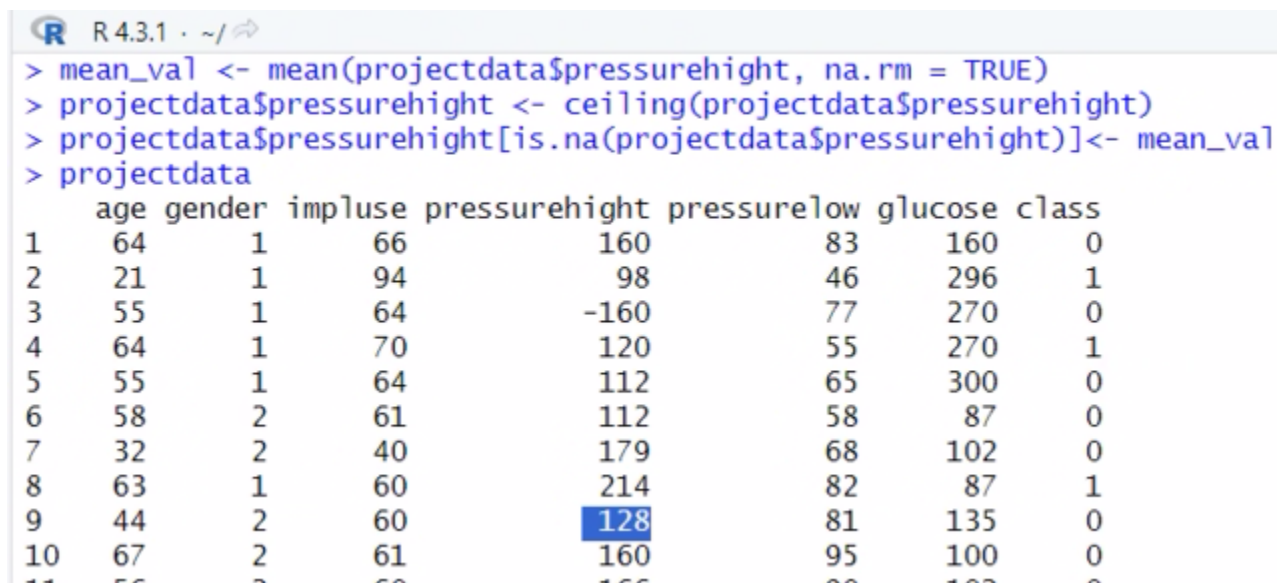
Apply **mean** for handel missing values for pressurehigh column.

```
mean_val <- mean(projectdata$pressurehight, na.rm = TRUE)
```

```
projectdata$pressurehight <- ceiling(projectdata$pressurehight)
```

```
projectdata$pressurehight[is.na(projectdata$pressurehight)]<- mean_val
```

```
projectdata
```

The image shows a screenshot of an R console window with the title 'R 4.3.1 ~/' and a blue arrow icon. The console contains four lines of R code: 1. > mean_val <- mean(projectdata\$pressurehight, na.rm = TRUE) 2. > projectdata\$pressurehight <- ceiling(projectdata\$pressurehight) 3. > projectdata\$pressurehight[is.na(projectdata\$pressurehight)]<- mean_val 4. > projectdata Below the code, a data frame is displayed with 10 rows and 8 columns. The columns are labeled: age, gender, impluse, pressurehight, pressurelow, glucose, and class. The data for the first 10 rows is as follows: Row 1: age=64, gender=1, impluse=66, pressurehight=160, pressurelow=83, glucose=160, class=0. Row 2: age=21, gender=1, impluse=94, pressurehight=98, pressurelow=46, glucose=296, class=1. Row 3: age=55, gender=1, impluse=64, pressurehight=-160, pressurelow=77, glucose=270, class=0. Row 4: age=64, gender=1, impluse=70, pressurehight=120, pressurelow=55, glucose=270, class=1. Row 5: age=55, gender=1, impluse=64, pressurehight=112, pressurelow=65, glucose=300, class=0. Row 6: age=58, gender=2, impluse=61, pressurehight=112, pressurelow=58, glucose=87, class=0. Row 7: age=32, gender=2, impluse=40, pressurehight=179, pressurelow=68, glucose=102, class=0. Row 8: age=63, gender=1, impluse=60, pressurehight=214, pressurelow=82, glucose=87, class=1. Row 9: age=44, gender=2, impluse=60, pressurehight=128, pressurelow=81, glucose=135, class=0. Row 10: age=67, gender=2, impluse=61, pressurehight=160, pressurelow=95, glucose=100, class=0. The value 128 in the pressurehight column of row 9 is highlighted with a blue background.

Here,we find mean by mean() function and replace missing value by mean value for age column.Our actual mean is 127.5 ceiling function converts it 128 also replace the value of 9 index by 128.

Apply **median** for handel missing values for pressurehigh column.

```
median_value <- median(projectdata$pressurehight, na.rm = TRUE)
```

```
ceiled_median <- ceiling(median_value)
```

```
projectdata$pressurehight[is.na(projectdata$pressurehight)]<-
```

```
ceiled_median
```

projectdata

```
R 4.3.1 ~/  
> median_value <- median(projectdata$pressurehight, na.rm = TRUE)  
> ceiled_median <- ceiling(median_value)  
> projectdata$pressurehight[is.na(projectdata$pressurehight)]<- ceiled_median  
> projectdata
```

	age	gender	impluse	pressurehight	pressurelow	glucose	class
1	64	1	66	160	83	160	0
2	21	1	94	98	46	296	1
3	55	1	64	-160	77	270	0
4	64	1	70	120	55	270	1
5	55	1	64	112	65	300	0
6	58	2	61	112	58	87	0
7	32	2	40	179	68	102	0
8	63	1	60	214	82	87	1
9	44	2	60	122	81	135	0
10	67	<NA>	61	160	95	100	0

Here,we find median by median() function and replace missing value by median value for pressurehigh column.Our actual median is 121.5 ceiling function converts it 122 also replace the value of 9 index by 122.

Apply **mode** for handel missing values for pressurehigh column.

```
C:\Users\user\AppData\Local\Temp\KmpKXAtw\downloaded_packages  
> library(modeest)  
> mode_v<- mlv(projectdata$pressurehight, method = "mfv")  
> mode_v  
[1] 130 135  
>
```

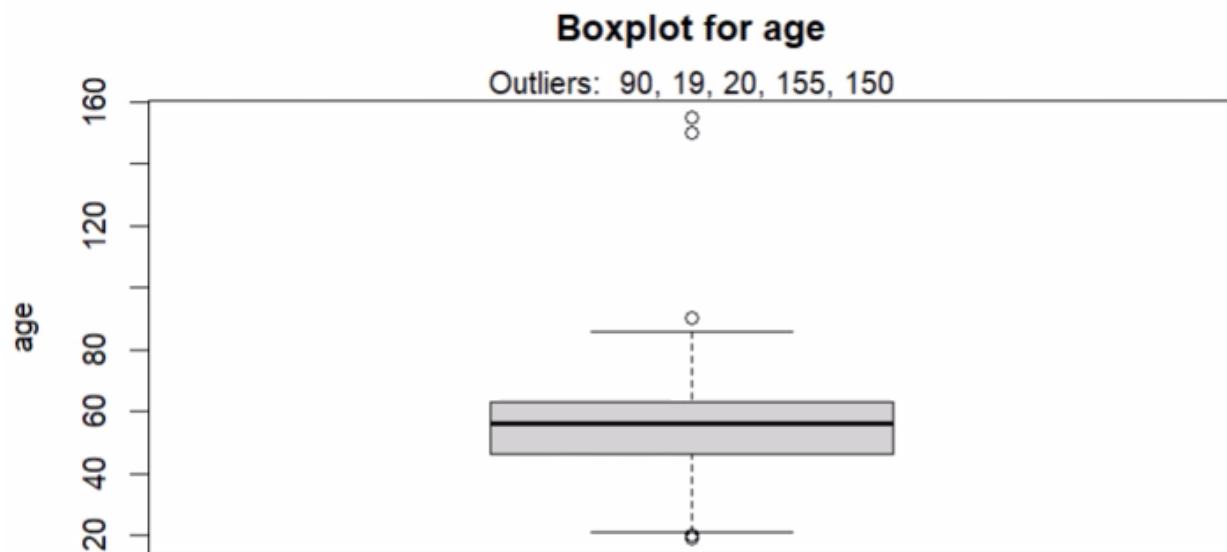
Here,We find mode value 130 and 135 of pressurehigh column by mlv () function.

Finding Outliers Values :(Boxplot)

for age column

```
a<- boxplot.stats(projectdata$age)$out
```

```
boxplot(projectdata$age,  
        ylab = "age",  
        main = "Boxplot for age"  
)  
mtext(paste("Outliers: ", paste(a, collapse = ", ")))
```



Here we perform box plot for age columns outliers. Outliers are 90,19,20,155 and 150 those are out of the range .

gender column has no outliers.

for impluse column

```
b<- boxplot.stats(projectdata$impluse)$out
```

```
b
```

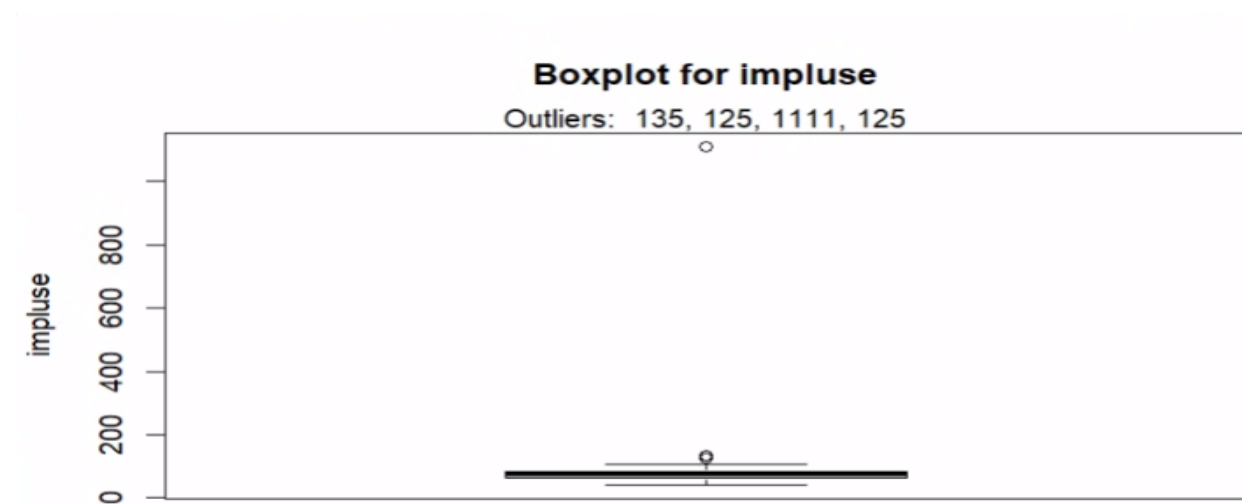
```
boxplot(projectdata$impluse,
```

```
  ylab = "impluse",
```

```
  main = "Boxplot for impluse"
```

```
)
```

```
mtext(paste("Outliers: ", paste(b, collapse = ", ")))
```



Here we perform box plot for impluse columns outliers. Outliers are 135, 125, 1111 and 125 those are out of the range .

for pressurehigh column

```
p<- boxplot.stats(projectdata$pressurehigh)$out
```

```
p
```

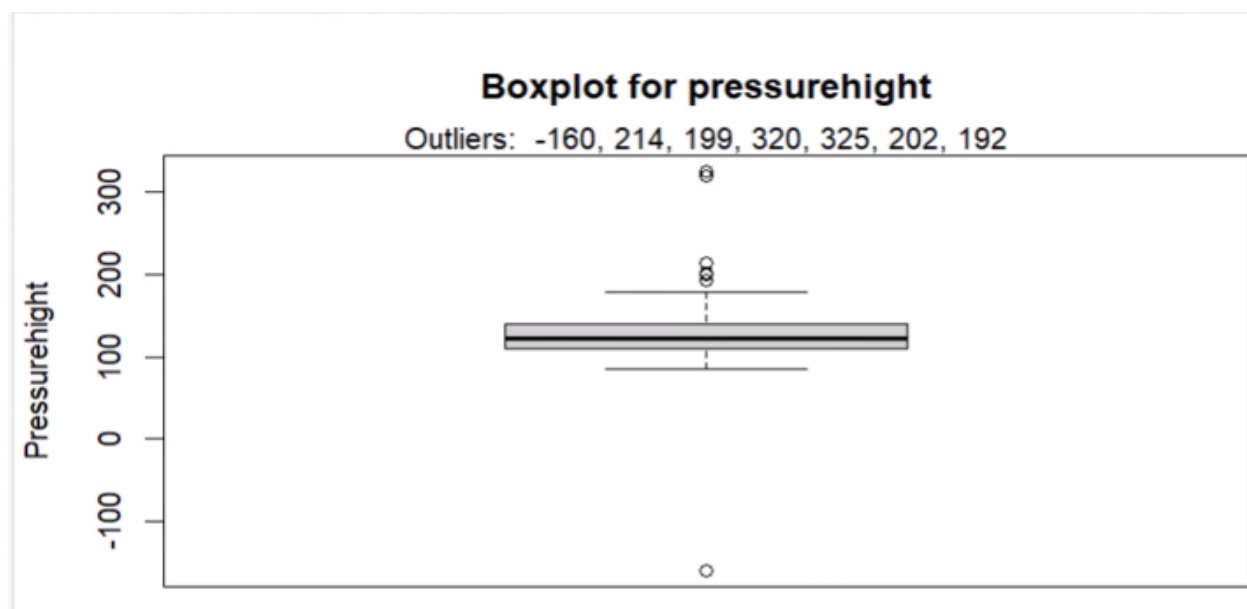
```
boxplot(projectdata$pressurehigh,
```

```
  ylab = "Pressurehigh",
```

```
  main = "Boxplot for pressurehigh"
```

```
)
```

```
mtext(paste("Outliers: ", paste(p, collapse = ", ")))
```



Box plot for pressurehigh columns outliers. Outliers are -160,214,199,320,325,202,192 those are out of the range .

for pressurelow column

```
plow<- boxplot.stats(projectdata$pressurelow)$out
```

```
plow
```

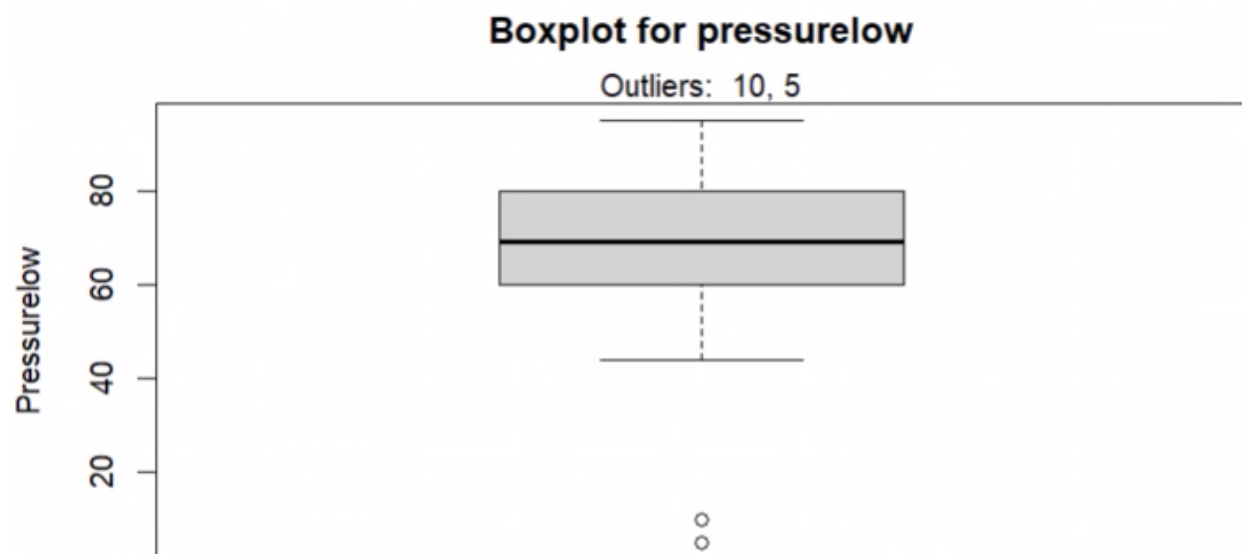
```
boxplot(projectdata$pressurelow,
```

```
  ylab = "Pressurelow",
```

```
  main = "Boxplot for pressurelow"
```

```
)
```

```
mtext(paste("Outliers: ", paste(plow, collapse = ", ")))
```



Here we perform box plot for impluse columns outliers.Outliers are 10and 5 those are out of the range.

for glucose column

```
gl<- boxplot.stats(projectdata$glucose)$out
```

```
gl
```

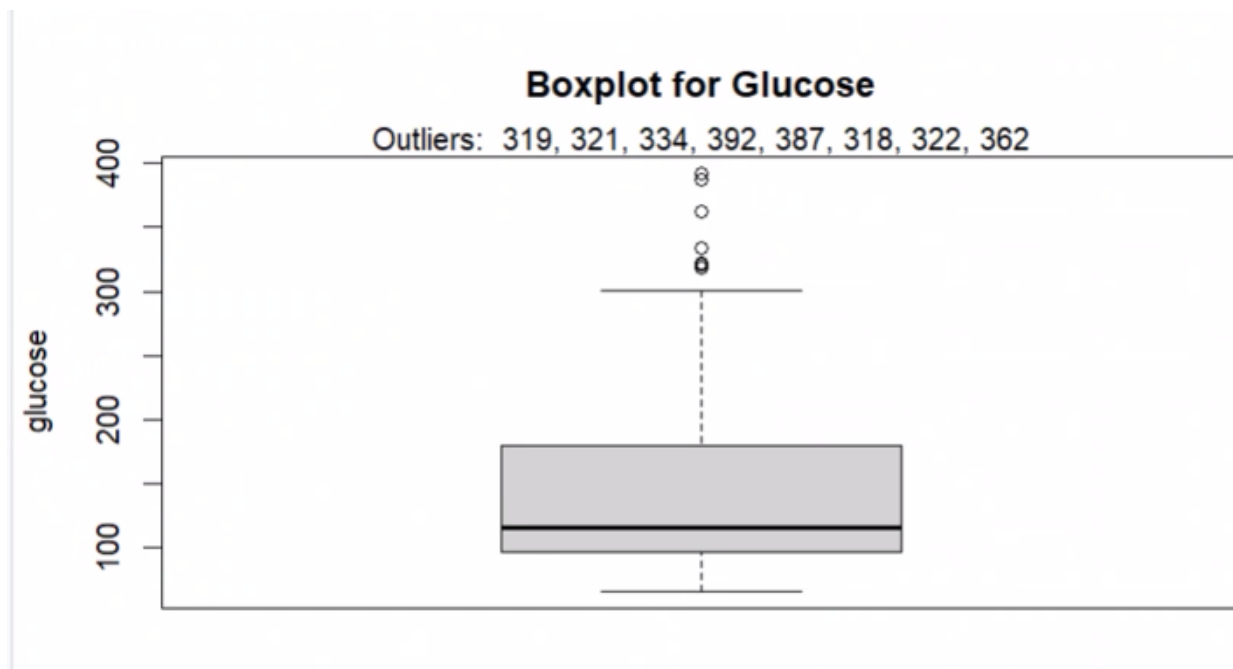
```
boxplot(projectdata$glucose,
```

```
  ylab = "glucose",
```

```
  main = "Boxplot for Glucose"
```

```
)
```

```
mtext(paste("Outliers: ", paste(gl, collapse = ", ")))
```



Here we perform box plot for impluse columns outliers.Outliers are 319,321,334,392,387,318,322,362 those are out of the range

class column has no outliers

Handeling Outliers Values :(by median)

For age column

```
projectdata$age[projectdata$age %in% boxplot(projectdata)$out] <-  
median_value
```

projectdata

135	63	1	80	140	83	116	1
136	57	1	64	117	68	94	1
137	67	1	58	119	72	109	0
138	56	2	62	109	63	362	0
139	56	1	79	85	44	97	0
140	43	2	79	89	57	98	0
141	45	2	79	87	47	82	0
142	50	1	80	98	52	110	0
143	64	1	79	99	55	105	1
144	63	2	78	116	60	180	0
145	60	1	78	96	57	116	1
146	53	2	77	105	58	92	0
147	60	2	89	95	70	93	1
148	56	2	91	100	71	116	0
149	50	1	83	95	70	94	1
150	69	1	82	86	70	87	1

Outliers replaced by median values.All the outliers index replaced by median value of age column 56.

For impluse column

```
projectdata$impluse[projectdata$impluse %in% boxplot(projectdata)$out] <-  
median_impluse
```

projectdata

Console	Terminal	Background Jobs
R 4.3.1	~/	
57	49	1
58	29	1
59	80	2
60	45	1
61	47	1
62	56	2
63	45	1
64	45	1
65	61	1
66	54	1
67	62	1
68	65	1
69	45	1
70	46	1

Outliers replaced by median values. All the outliers index replaced by median value of impulse column 74.

For pressurehigh column

```
projectdata$pressurehigh[projectdata$pressurehigh%in%
boxplot(projectdata)$out] <- ceiled_median
```

projectdata

```

> projectdata

```

	age	gender	impluse	pressurehight	pressurelow	glucose	class
1	64	1	66	160	83	160	0
2	21	1	94	98	46	296	1
3	55	1	64	122	77	270	0
4	64	1	70	120	55	270	1
5	55	1	64	112	65	300	0
6	58	2	61	112	58	87	0
7	32	2	40	179	68	102	0
8	63	1	60	122	82	87	1

Outliers replaced by median values. All the outliers index replaced by median value of pressurehigh column 122.

For pressurelow column

```
median_low <- median(projectdata$pressurelow, na.rm = TRUE)
```

```
ceiled_median_low <- ceiling(median_low)
```

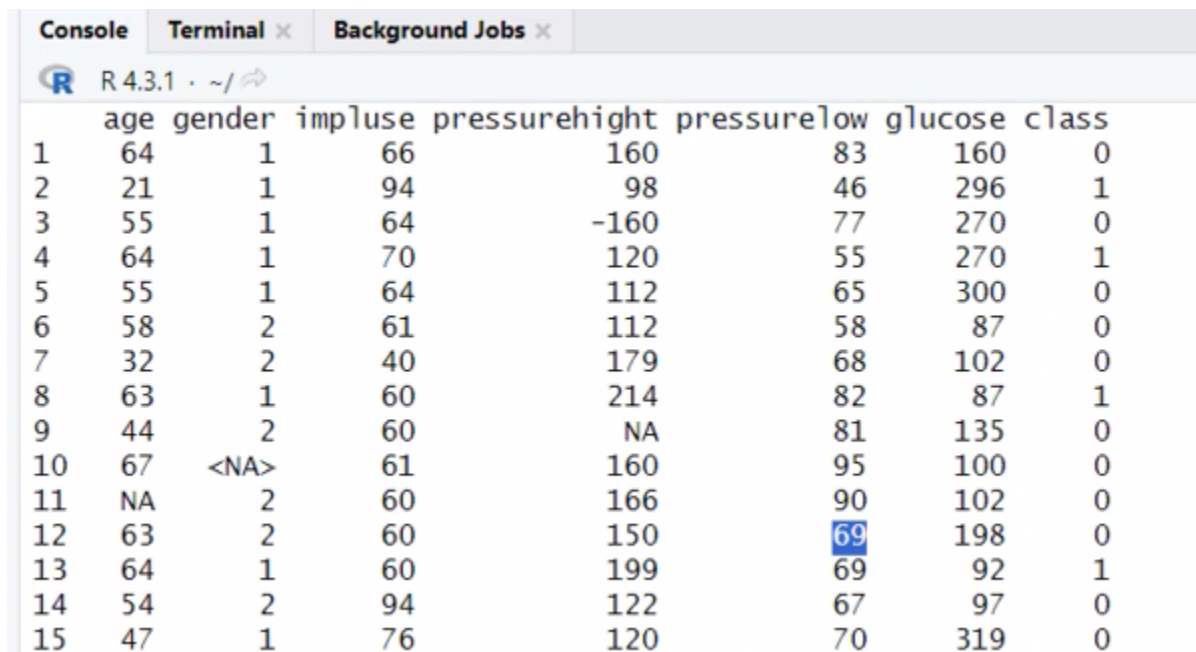
```
projectdata$pressurelow[is.na(projectdata$pressurelow)]<-  
ceiled_median_low
```

```
projectdata
```

```
projectdata$pressurelow[projectdata$pressurelow%in%
```

```
boxplot(projectdata)$out] <- ceiled_median_low
```

```
projectdata
```



	age	gender	impluse	pressurehight	pressurelow	glucose	class
1	64	1	66	160	83	160	0
2	21	1	94	98	46	296	1
3	55	1	64	-160	77	270	0
4	64	1	70	120	55	270	1
5	55	1	64	112	65	300	0
6	58	2	61	112	58	87	0
7	32	2	40	179	68	102	0
8	63	1	60	214	82	87	1
9	44	2	60	NA	81	135	0
10	67	<NA>	61	160	95	100	0
11	NA	2	60	166	90	102	0
12	63	2	60	150	69	198	0
13	64	1	60	199	69	92	1
14	54	2	94	122	67	97	0
15	47	1	76	120	70	319	0

Outliers replaced by median values. All the outliers index replaced by median value of pressurelow column 69. Index 12 and 13.

For glucose column

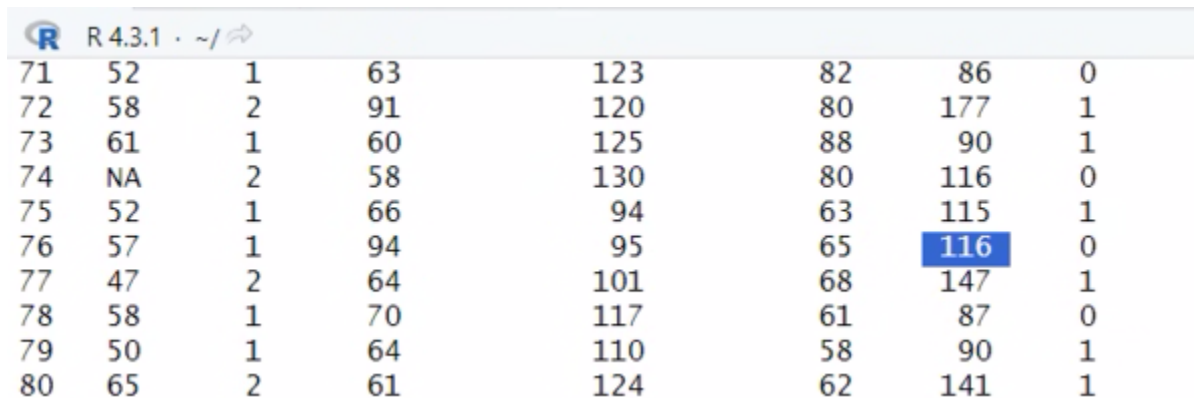
```
median_glucose <- median(projectdata$glucose, na.rm = TRUE)
```

```
ceiled_median_glucose <- ceiling(median_glucose)
```

```
projectdata$glucose[is.na(projectdata$glucose)]<- ceiled_median_glucose
```

```
projectdata$glucose[projectdata$glucose %in% boxplot(projectdata)$out]  
<- ceiled_median_glucose
```

projectdata



71	52	1	63	123	82	86	0
72	58	2	91	120	80	177	1
73	61	1	60	125	88	90	1
74	NA	2	58	130	80	116	0
75	52	1	66	94	63	115	1
76	57	1	94	95	65	116	0
77	47	2	64	101	68	147	1
78	58	1	70	117	61	87	0
79	50	1	64	110	58	90	1
80	65	2	61	124	62	141	1

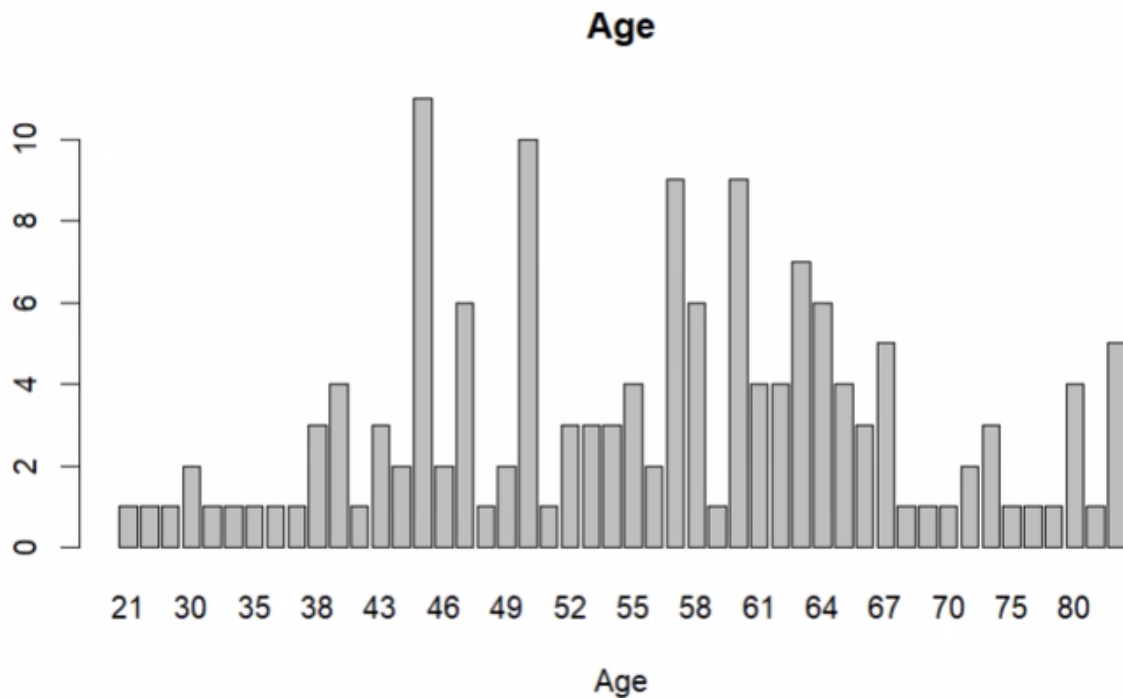
Outliers replaced by median values. All the outliers index replaced by median value of glucose column 116.

Barplot:

For age column

```
ag <- table(projectdata$age)
```

```
barplot(ag, main="Age", xlab="Age")
```

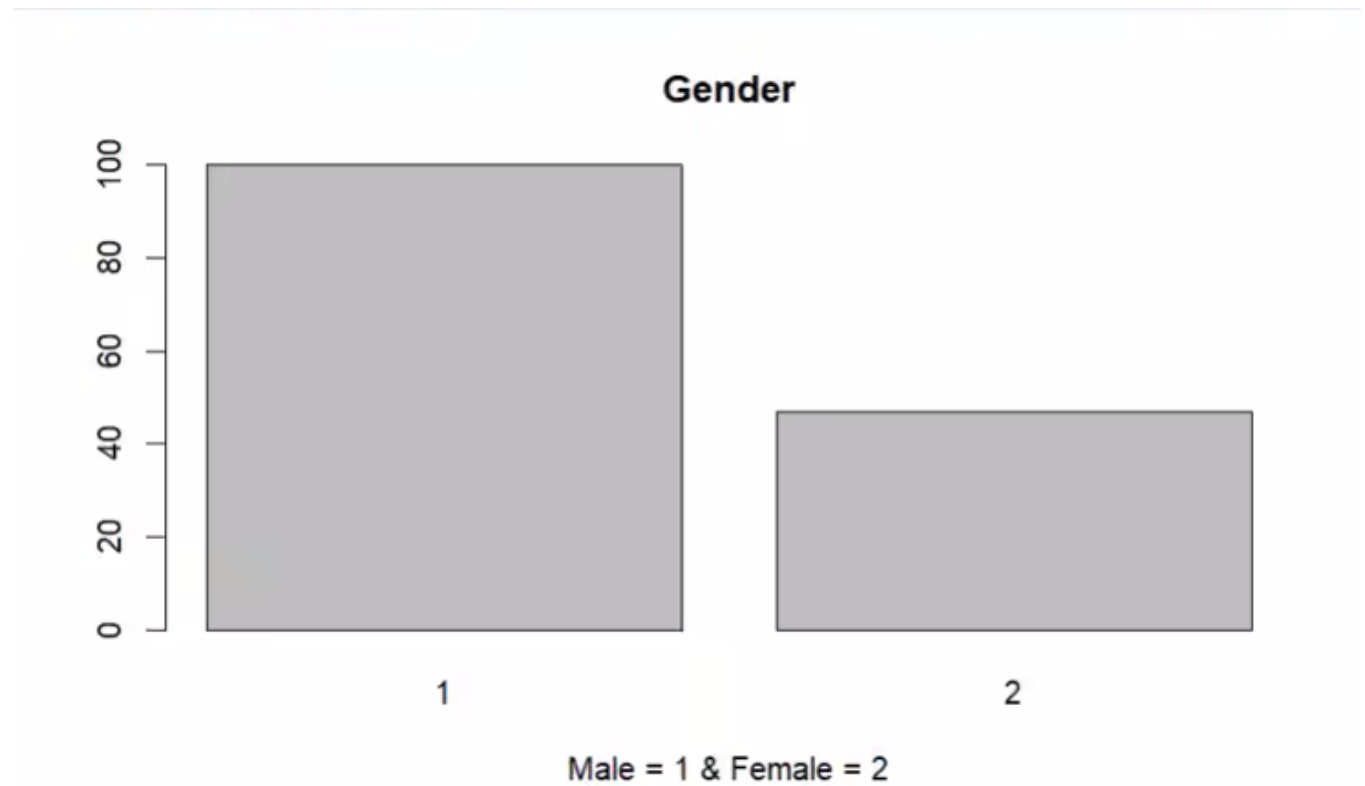


Barplot for age column.

For gender column

```
bar <- table(projectdata$gender)
```

```
barplot(bar, main="Gender", xlab="Male = 1 & Female = 2")
```

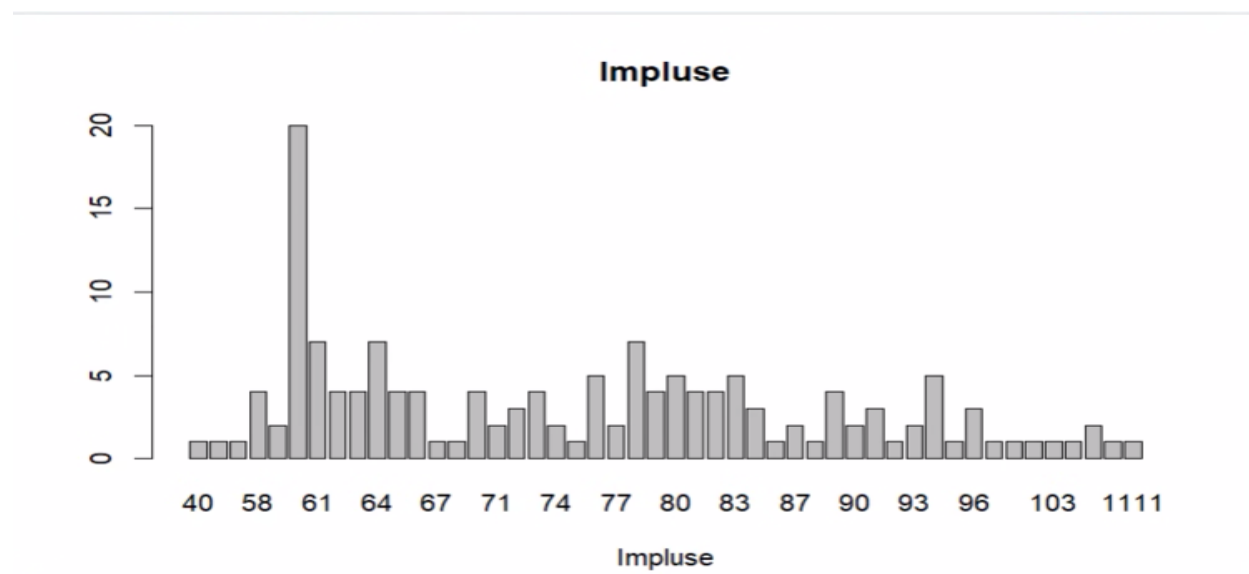


Barplot for gender column. Male represent 1 and Female represent 2.

For impulse column

```
imp<- table(projectdata$impluse)
```

```
barplot(imp, main="Impluse", xlab="Impluse")
```

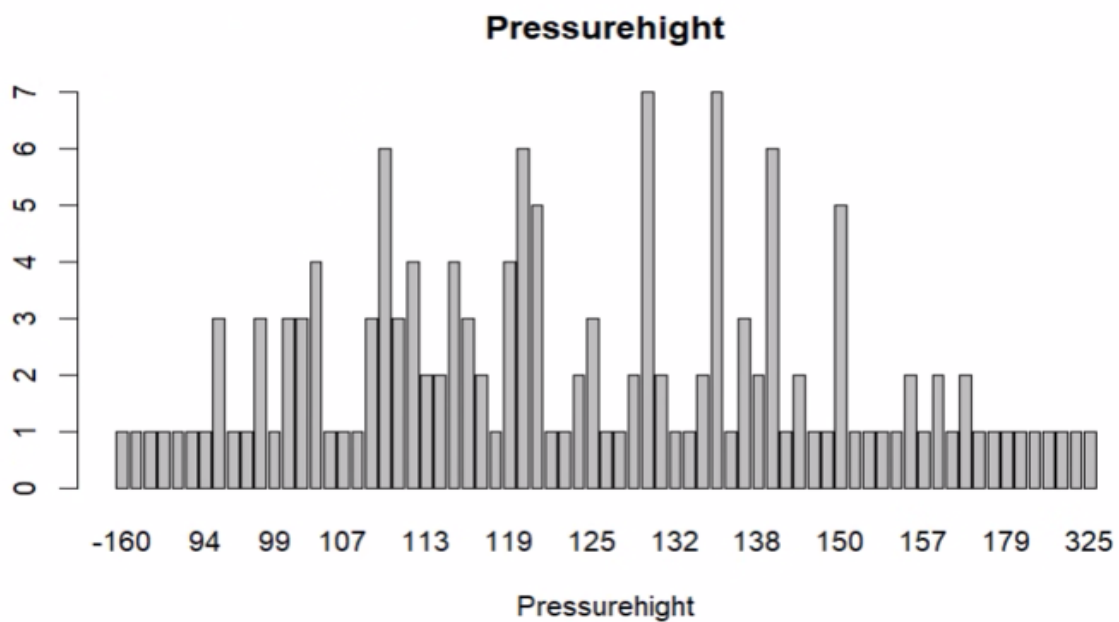


Barplot for impulse column.

For pressurehigh column

```
high <- table(projectdata$pressurehigh)
```

```
barplot(high, main="Pressurehigh", xlab="Pressurehigh")
```

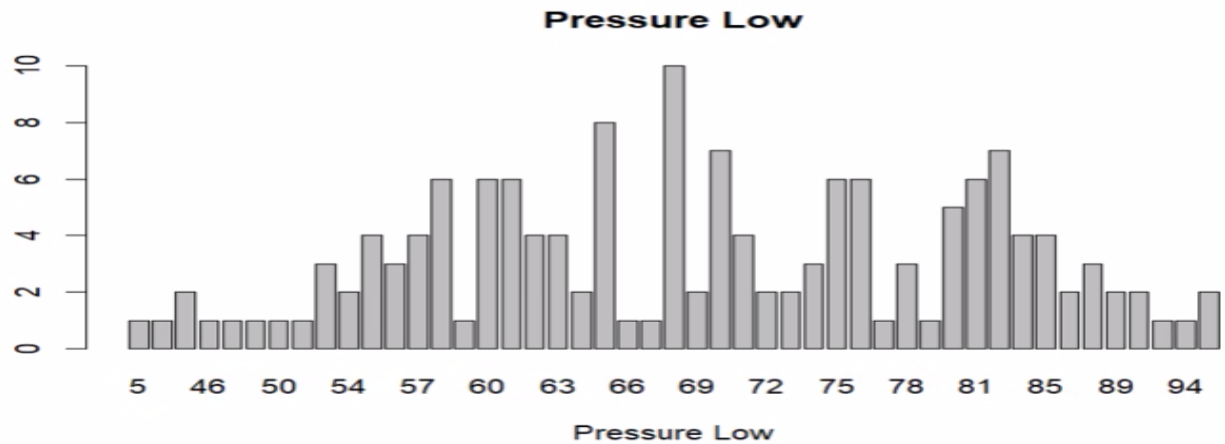


Barplot for pressurehigh column.

For pressurelow column

```
lo <- table(projectdata$pressurelow)
```

```
barplot(lo, main="Pressure Low", xlab="Pressure Low")
```

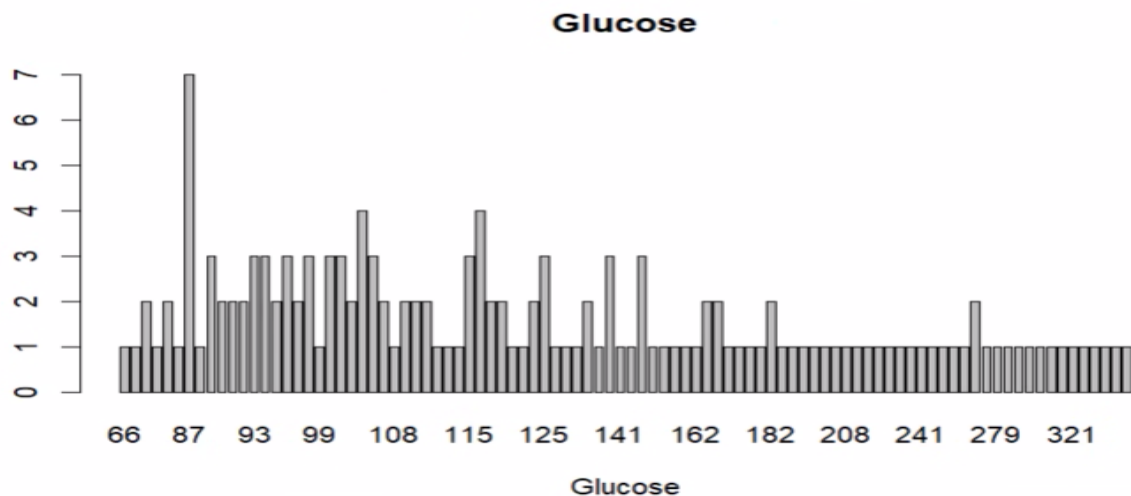


Barplot for pressurelow column.

For glucose column

```
glu <- table(projectdata$glucose)
```

```
barplot(glu, main="Glucose", xlab="Glucose")
```

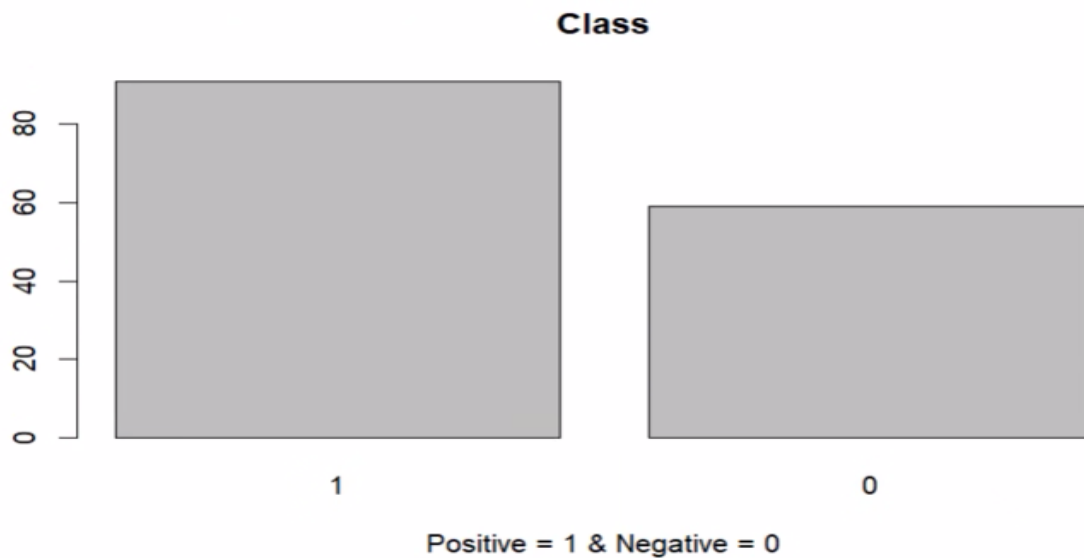


Barplot for glucose column.

For class column

```
cls <- table(projectdata$class)
```

```
barplot(cls, main="Class", xlab="Positive = 1 & Negative = 0")
```

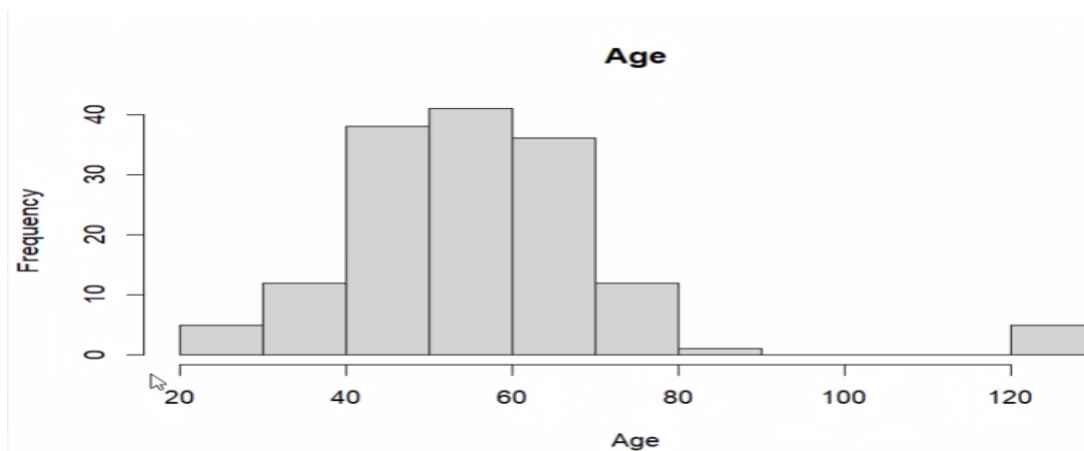


Barplot for class column. 1 represent positive and 0 represent negative.

Histogram:

For age column

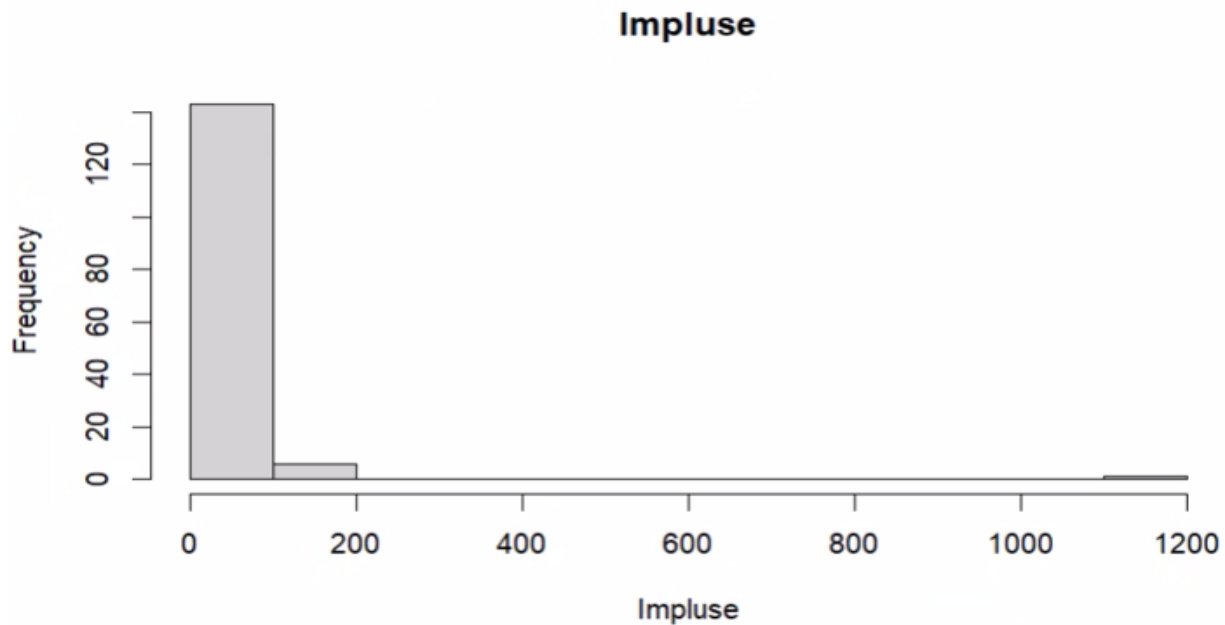
```
hist(projectdata$age, main="Age", xlab="Age")
```



Histogram for age column.

For impluse column

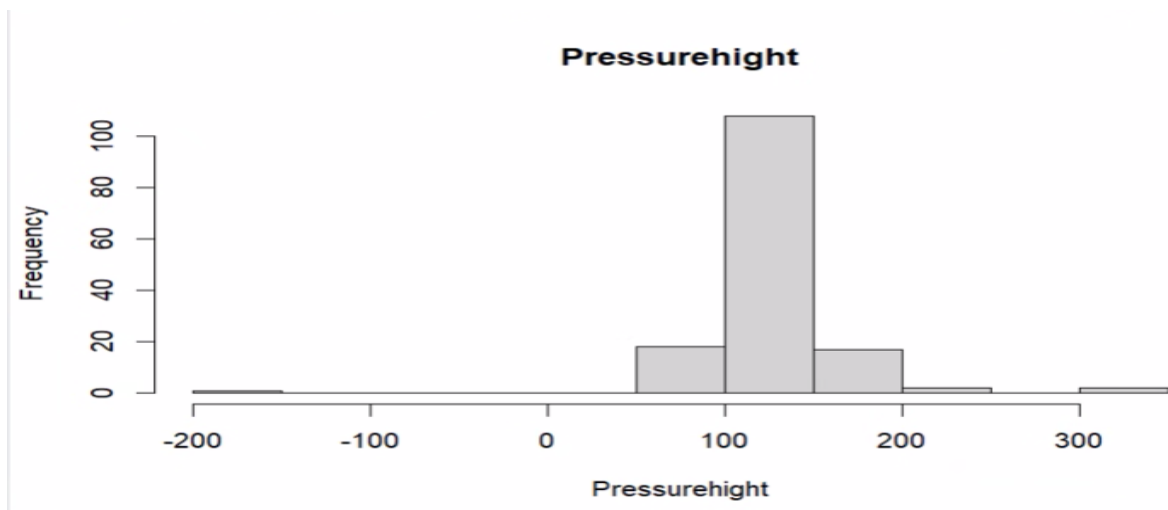
```
hist(projectdata$impluse, main="Impluse", xlab="Impluse")
```



Histogram for impluse column.

For pressurehigh column

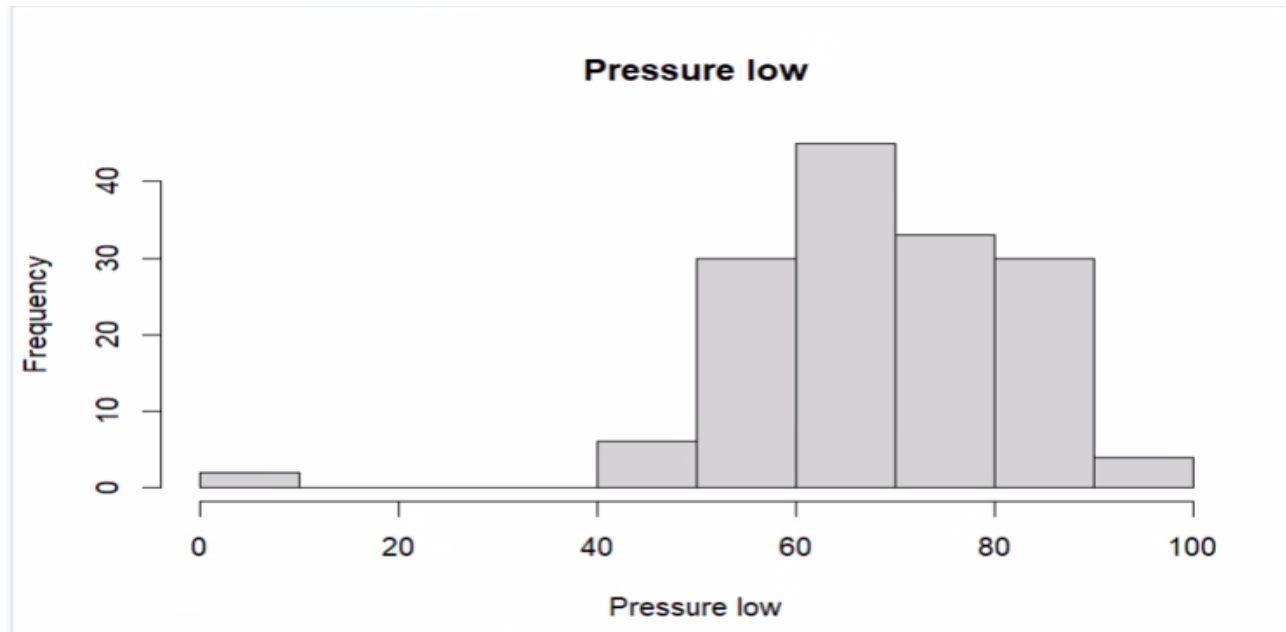
```
hist(projectdata$pressurehigh,main="Pressurehigh",xlab="Pressurehigh")
```



Histogram for pressurehigh column.

For pressurelow column

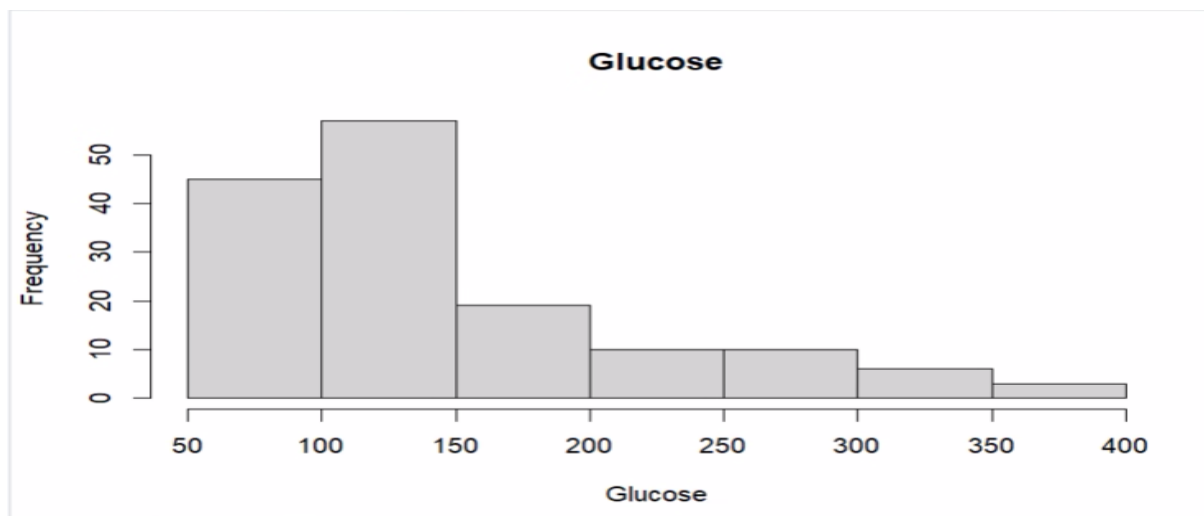
```
hist(projectdata$pressurelow, main="Pressure low", xlab="Pressure low")
```



Histogram for pressurelow column.

For glucose column

```
hist(projectdata$glucose, main="Glucose", xlab="Glucose")
```



Histogram for glucose column.

Standard Deviation:

```
s<- projectdata$age
sd(s)
im<- projectdata$impluse
sd(im)
hi<- projectdata$pressurehight
sd(hi)
lo<- projectdata$pressurelow
sd(lo)
glu<- projectdata$glucose
sd(glu)
```

```
> s<- projectdata$age
> sd(s)
[1] 16.9973
> |
```

```
> im<- projectdata$impluse
> sd(im)
[1] 85.87672
> |
```

```
> hi<- projectdata$pressurehight
> sd(hi)
[1] 39.9979
> |
```

```
> lo<- projectdata$pressurelow
> sd(lo)
[1] 13.65114
> |
```

```
> glu<- projectdata$glucose
> sd(glu)
[1] 73.10635
```

sd() Function return the standard deviation of a column. We have apply standard deviation for each column.