



DATA SCIENCE PROJECT(MIDTERM)

NAME: RABBI HASAN HIMEL

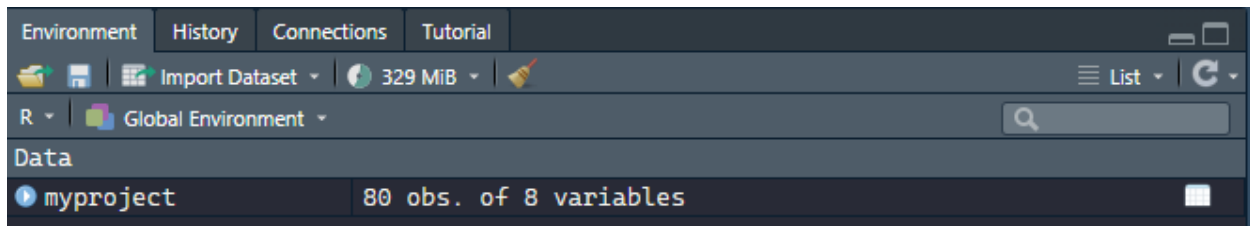
ID: 20-422404-1

SECTION: D

FACULTY: TOHEDUL ISLAM

Importing Dataset (Dataset_midterm.csv)

```
myproject<-read.csv("D:/Study/Aiub/9th Semester/Data  
Science/Project/Mid/Dataset_midterm.csv",header=TRUE, sep=",")
```



Output of the dataset:

myproject

	id	Age	weight.kg.	Delivery_number	Delivery_time	Blood	Heart	Caesarian
1	1	22	57.7	1	0	high	0	0
2	2	26	63.0	2	0	normal	0	1
3	3	26	62.0	2	1	normal	0	0
4	4	28	65.0	1	0	high	0	0
5	5	22	58.0	2	0	normal	0	1
6	6	26	63.0	1	1	low	0	0
7	7	27	64.0	2	0	normal	0	0
8	8	32	70.0	3	0	normal	0	1
9	9	28	63.5	2	0		0	0
10	10	27	64.5	1	1	normal	0	1

69	69	25	62.0	1	0	low	0	1
70	70	27	61.0	2	2	low	0	0
71	71	90	105.0	1	0	low	0	1
72	72	29	65.0	1	2		1	1
73	73	28	64.0	2	0	normal	0	0
74	74	32	69.0	3	0	normal	1	0
75	75	38	75.0	3	2	high	1	1
76	76	27	62.5	2	1	normal	0	0
77	77	33	66.0	4	0	normal	0	NA
78	78	NA	63.0	2	1	high	0	1
79	79	25	58.0	1	2	low	0	1
80	80	24	57.0	2	2	normal	0	0

Structure of the DATASET

str(myproject)

```
> View(myproject)
> str(myproject)
'data.frame': 80 obs. of 8 variables:
 $ id      : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Age     : int  22 26 26 28 22 26 27 32 28 27 ...
 $ weight.kg.: num  57.7 63 62 65 58 63 64 70 63.5 64.5 ...
 $ Delivery_number: int  1 2 2 1 2 1 2 3 2 1 ...
 $ Delivery_time : int  0 0 1 0 0 1 0 0 0 1 ...
 $ Blood      : chr  "high" "normal" "normal" "high" ...
 $ Heart      : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Caesarian   : int  0 1 0 0 1 0 0 1 0 1 ...
```

Summary of Dataset

summary(myproject)

```
> summary(myproject)
   id      Age      weight.kg.  Delivery_number Delivery_time      Blood      Heart      Caesarian
Min.   :1.00  Min.   :18.00  Min.   :49.00  Min.   :1.000  Min.   :0.0000  Length:80  Min.   :0.000  Min.   :0.0000
1st Qu.:20.75 1st Qu.:25.00 1st Qu.:61.00 1st Qu.:1.000 1st Qu.:0.0000 Class :character 1st Qu.:0.000 1st Qu.:0.0000
Median :40.50 Median :28.00 Median :63.50 Median :1.500 Median :0.0000 Mode  :character Median :0.000 Median :1.0000
Mean   :40.50 Mean   :29.68 Mean   :65.13 Mean   :1.679 Mean   :0.6234      Mean   :0.375 Mean   :0.5641
3rd Qu.:60.25 3rd Qu.:32.00 3rd Qu.:68.00 3rd Qu.:2.000 3rd Qu.:1.0000      3rd Qu.:1.000 3rd Qu.:1.0000
Max.   :80.00 Max.   :95.00 Max.   :110.00 Max.   :4.000 Max.   :2.0000      Max.   :1.000 Max.   :1.0000
      NA's   :3      NA's   :3      NA's   :2      NA's   :3      NA's   :2
```

Attributes name of this dataset

```
> names(myproject)
[1] "id"      "Age"      "weight.kg." "Delivery_number"
[5] "Delivery_time" "Blood"    "Heart"      "Caesarian"
> |
```

Datatypes of this dataset of all attributes

```
> typeof(myproject$id)
[1] "integer"
> typeof(myproject$Age)
[1] "integer"
> typeof(myproject$weight.kg.)
[1] "double"
> typeof(myproject$Delivery_number)
[1] "integer"
> typeof(myproject$Delivery_time)
[1] "integer"
> typeof(myproject$Blood)
[1] "character"
> typeof(myproject$Heart)
[1] "integer"
> typeof(myproject$Caesarian)
[1] "integer"
```

Removing Missing values

```
remove_missing_value <- na.omit(myproject)
```

```
myproject<-remove_missing_value
```

Before removing:

	id	Age	weight.kg.	Delivery_number	Delivery_time	Blood	Heart	Caesarian
1	1	22	57.7	1	0	2	0	0
2	2	26	63.0	2	0	1	0	1
3	3	26	62.0	2	1	1	0	0
4	4	28	65.0	1	0	2	0	0
5	5	22	58.0	2	0	1	0	1
6	6	26	63.0	1	1	0	0	0
7	7	27	64.0	2	0	1	0	0
8	8	32	70.0	3	0	1	0	1
9	9	28	63.5	2	0	<NA>	0	0
10	10	27	64.5	1	1	1	0	1
11	11	36	75.0	1	0	1	0	0
12	12	33	70.0	1	1	0	0	1
13	13	23	58.0	1	1	1	0	0
14	14	20	55.0	1	0	1	1	0
15	15	29	65.0	1	NA	<NA>	1	1
16	16	25	61.5	1	2	0	0	0
17	17	25	61.5	1	0	1	0	0
18	18	20	55.5	1	2	2	0	1
19	19	37	76.0	3	0	1	1	1
20	20	24	56.6	1	2	0	1	1
21	21	26	62.0	1	1	1	0	0
22	22	33	75.0	2	0	0	1	1
23	23	25	62.0	1	1	2	0	0
24	24	27	65.0	NA	NA	0	1	1
25	25	20	55.0	1	0	2	1	1
26	26	18	49.0	NA	0	1	0	0
27	27	18	50.0	1	NA	2	1	1

After Removing:

	id	Age	weight.kg.	Delivery_number	Delivery_time	Blood	Heart	Caesarian
1	1	22	57.7	1	0	2	0	0
2	2	26	63.0	2	0	1	0	1
3	3	26	62.0	2	1	1	0	0
4	4	28	65.0	1	0	2	0	0
5	5	22	58.0	2	0	1	0	1
6	6	26	63.0	1	1	0	0	0
7	7	27	64.0	2	0	1	0	0
8	8	32	70.0	3	0	1	0	1
10	10	27	64.5	1	1	1	0	1
11	11	36	75.0	1	0	1	0	0
12	12	33	70.0	1	1	0	0	1
13	13	23	58.0	1	1	1	0	0
14	14	20	55.0	1	0	1	1	0
16	16	25	61.5	1	2	0	0	0
17	17	25	61.5	1	0	1	0	0
18	18	20	55.5	1	2	2	0	1
19	19	37	76.0	3	0	1	1	1
20	20	24	56.6	1	2	0	1	1
21	21	26	62.0	1	1	1	0	0
22	22	33	75.0	2	0	0	1	1
23	23	25	62.0	1	1	2	0	0
25	25	20	55.0	1	0	2	1	1
28	28	30	68.0	1	0	1	0	0
29	29	32	73.0	1	0	2	1	1
30	30	26	62.5	2	1	1	1	0
31	31	25	58.0	1	0	0	0	0

Measure of center (MEAN, MEDIAN and MODE) for Weight.kg.

```
mean(myproject$weight.kg.,na.rm = TRUE) #Mean
median(myproject$weight.kg.,na.rm = TRUE) #median
```

```
# For modeValue installing packages("DescTools")
library(DescTools)
modeValue <- Mode(myproject$weight.kg.)#mode
modeValue
```

```
> mean(myproject$weight.kg.,na.rm = TRUE) #Mean
[1] 65.6
> median(myproject$weight.kg.,na.rm = TRUE) #median
[1] 63
> library(DescTools)
> modeValue <- Mode(myproject$weight.kg.)#mode
> modeValue
[1] 62.5 63.0
attr(,"freq")
[1] 5
```

Sorting and Max value

```
[1] 58
> sort(myproject$weight.kg.)
[1] 51.0 53.0 55.0 55.0 55.5 56.6 57.0 57.0 57.7 58.0 58.0 58.0 58.0 58.5
[15] 59.0 59.0 59.0 61.0 61.5 61.5 61.5 62.0 62.0 62.0 62.0 62.5 62.5 62.5
[29] 62.5 62.5 63.0 63.0 63.0 63.0 63.0 63.5 64.0 64.0 64.5 64.5 65.0 66.0
[43] 66.0 67.0 67.0 67.5 67.5 68.0 68.0 68.0 68.5 69.0 69.0 70.0 70.0 72.0
[57] 72.5 73.0 73.0 74.0 75.0 75.0 75.0 75.0 76.0 82.0 105.0 110.0
> max(myproject$weight.kg.)
[1] 110
>
```

Measure of center (MEAN, MEDIAN and MODE) for Blood

```
library(dplyr)
myproject$weight.kg.= as.numeric(myproject$weight.kg.)
mean(myproject$weight.kg.,na.rm = TRUE) #Mean
median(myproject$weight.kg.,na.rm = TRUE) #median
# For modeValue installing packages("DescTools")
library(DescTools)
```

modeValue

```
> mean(myproject$Blood, na.rm = TRUE) #Mean
[1] 2.279412
> median(myproject$Blood, na.rm = TRUE) #median
[1] 3
> #install.packages("DescTools")
> library(DescTools)
> modeValue <- Mode(myproject$Blood)#mode
> modeValue
[1] 3
attr(,"freq")
[1] 35
```

Sort and maximum:

[illegible]

Measure of center (MEAN, MEDIAN and MODE) for AGE

```
library(dplyr)
```

#Because Blood value was characteristic, have to convert into Numeric

```
myproject$Blood= as.numeric(myproject$Age)
```

```
mean(myproject$Age,na.rm = TRUE) #Mean
```

```
median(myproject$Age,na.rm = TRUE) #median
```

```
#install.packages("DescTools")
```

```
library(DescTools)
```

```
modeValue <- Mode(myproject$Age)#mode
```

modeValue

```
> mean(myproject$Age,na.rm = TRUE) #Mean
[1] 30.10294
> median(myproject$Age,na.rm = TRUE) #median
[1] 27.5
```

Sort and maximum:

```
sort(myproject$Age)
```

```
max(myproject$Age)
```

```

> modeValue
[1] 26
attr(,"freq")
[1] 9
>
> sort(myproject$Age)
[1] 19 20 20 20 21 22 22 22 22 23 24 24 25 25 25 25 25 25 26 26 26 26 26 26 26 26
[29] 27 27 27 27 27 27 28 28 28 28 28 29 29 30 30 31 31 31 32 32 32 32 32 32 32 33 33
[57] 33 33 35 35 36 36 36 37 38 40 90 95
> max(myproject$Age)
[1] 95

```

Finding MEAN of Column (AGE & BLOOD PRESSURE)

```
install.packages("matrixStats")
```

```
library(matrixStats)
```

```
myproject$age_bloodmean = rowMeans(as.matrix(myproject[,c(2,6)]))
```

```
myproject$age_bloodmean
```

```

> myproject$age_bloodmean
[1] 12.0 14.5 14.5 15.0 12.5 13.5 15.0 17.5 15.0 19.5 17.0 13.0 11.5 13.0 14.0 11.0
[17] 20.0 12.5 14.5 17.0 13.5 11.0 16.5 17.0 14.5 13.0 21.5 17.0 15.0 14.5 15.0 18.0
[33] 17.0 17.0 13.5 14.5 11.0 19.0 12.5 19.0 15.5 17.0 14.5 18.0 11.0 16.0 18.0 16.0
[49] 14.0 16.5 48.0 14.0 12.0 16.5 17.5 16.5 19.0 15.5 16.0 13.0 14.0 45.5 15.5 17.5
[65] 20.0 15.0 13.0 13.5

```

Row wise standard deviation (AGE & Blood-Numeric) Creating New Column(AGE BLOOD)

```
myproject$Blood <- factor(myproject$Blood, levels = c("low","high","normal"), labels = c(0,2,1))
```

```
myproject$Blood= as.numeric(myproject$Blood)
```

```
myproject$age_blood = rowSds(as.matrix(myproject[,c(2,6)]))
```

```
myproject$age_blood
```

```

> myproject$age_blood
[1] 14.14214 16.26346 16.26346 18.38478 13.43503 17.67767 16.97056 20.50610 16.97056
[10] 23.33452 22.62742 14.14214 12.02082 16.97056 15.55635 12.72792 24.04163 16.26346
[19] 16.26346 22.62742 16.26346 12.72792 19.09188 21.21320 16.26346 16.97056 26.16295
[28] 21.21320 16.97056 16.26346 18.38478 21.21320 19.79899 19.79899 17.67767 17.67767
[37] 11.31371 24.04163 13.43503 24.04163 17.67767 21.21320 16.26346 21.21320 14.14214
[46] 19.79899 24.04163 18.38478 15.55635 21.92031 66.46804 16.97056 14.14214 21.92031
[55] 20.50610 20.50610 22.62742 17.67767 18.38478 16.97056 18.38478 62.93250 17.67767
[64] 20.50610 25.45584 16.97056 16.97056 14.84924

```

standard deviation of all columns

```
myproject %>% summarise_if(is.numeric, sd)
```

```
> myproject %>% summarise_if(is.numeric, sd)
  id      Age weight.kg. Delivery_number Delivery_time      Heart Caesarian
1 23.04838 11.93282    9.651865         0.781143    0.8243443 0.4814377 0.5017529
```

MEASURE OF SPREAD (RANGE) for AGE

#(na.rm) will avoid the null values(if have)

```
range(myproject$Age, na.rm = TRUE)
```

```
> range(myproject$Age, na.rm = TRUE) #(na.rm) will avoid the null values(if have)
[1] 19 95
```

```
range_AGE <- max(myproject$Age)-min(myproject$Age)
```

```
range_AGE
```

```
> range_AGE <- max(myproject$Age)-min(myproject$Age)
> range_AGE
[1] 76
>
```

MEASURE OF SPREAD (RANGE) for Weight.KG.

range(myproject\$weight.kg., na.rm = TRUE) #(na.rm) will avoid the null values(if have)

```
range_Weight.KG. <- max(myproject$weight.kg.)-min(myproject$weight.kg.)
```

```
range_Weight.KG.
```

```
> range(myproject$weight.kg., na.rm = TRUE) #(na.rm) will avoid the null values(if have)
[1] 51 110
> range_Weight.KG. <- max(myproject$weight.kg.)-min(myproject$weight.kg.)
> range_Weight.KG.
[1] 59
```

MEASURE OF SPREAD (Standard Deviation) for AGE & Blood

```
s1 <- (myproject$weight.kg.)
```

```
sd(s1)
```

```
> sd(s1)
[1] 9.651865
```

```
sdblood <- (myproject$Blood)
```

```
sd(sdblood)
```



```
> sdblood ← (myproject$Blood)
> sd(sdblood)
[1] 0.8258074
```

##MATRIX

Select RANDOM Row from Data Set

sample_n(myproject,6)

```
> sample_n(myproject,6)
  id Age weight.kg. Delivery_number Delivery_time Blood Heart Caesarian
1 25 20      55.0             1           0      20      1         1
2 49 26      62.5             2           2      26      0         0
3 79 25      58.0             1           2      25      0         1
4 40 26      59.0             1           2      26      1         1
5 69 25      62.0             1           0      25      0         1
6 74 32      69.0             3           0      32      1         0
```

##OUTLIERS & RECOVERING THE DELETED DATA

This time replacing by NA for every missing value &

```
> myproject2←read.csv("D:/Study/Aiub/9th Semester/Data Science/Project/Mid/Dataset_midterm_Outlier.csv",header=TRUE, sep=",")
>
> myproject2
  id Age weight.kg. Delivery_number Delivery_time Blood Heart Caesarian
1  1  22      57.7             1           0    high      0         0
2  2  26      63.0             2           0 normal      0         1
3  3  26      62.0             2           1 normal      0         0
4  4  28      65.0             1           0    high      0         0
5  5  22      58.0             2           0 normal      0         1
6  6  26      63.0             1           1    low       0         0
7  7  27      64.0             2           0 normal      0         0
8  8  32      70.0             3           0 normal      0         1
9  9  28      63.5             2           0         0         0         0
10 10 27      64.5             1           1 normal      0         1
11 11 36      75.0             1           0 normal      0         0
12 12 33      70.0             1           1    low       0         1
13 13 23      58.0             1           1 normal      0         0
14 14 20      55.0             1           0 normal      1         0
15 15 29      65.0             1           NA         1         1
16 16 25      61.5             1           2    low       0         0
```

Count of NA values of each columns

colSums(is.na(myproject2))

```
> colSums(is.na(myproject2))
  id      Age      weight.kg. Delivery_number
0      3          3          2
Delivery_time      Blood      Heart      Caesarian
3          0          0          2
```

Finding which are the NA values of each attributes

```

which(is.na(myproject2$Age))
which(is.na(myproject2$weight.kg.))
which(is.na(myproject2$Delivery_number))
which(is.na(myproject2$Delivery_time))
which(is.na(myproject2$Blood))
which(is.na(myproject2$Caesarian))
which(is.na(myproject2$Heart))

```

```

> which(is.na(myproject2$Age))
[1] 50 62 78
> which(is.na(myproject2$Age))
[1] 50 62 78
> which(is.na(myproject2$weight.kg.))
[1] 47 50 62
> which(is.na(myproject2$Delivery_number))
[1] 24 26
> which(is.na(myproject2$Delivery_time))
[1] 15 24 27
> which(is.na(myproject2$Blood))
integer(0)
> which(is.na(myproject2$Caesarian))
[1] 60 77
> which(is.na(myproject2$Heart))
integer(0)

```

TOTAL NUMBER OF NA value Individual Column or Attributes

```
sum(is.na(myproject2$weight.kg.))
```

```

3      0
> sum(is.na(myproject2$weight.kg.))
[1] 3
> View(myproject2)

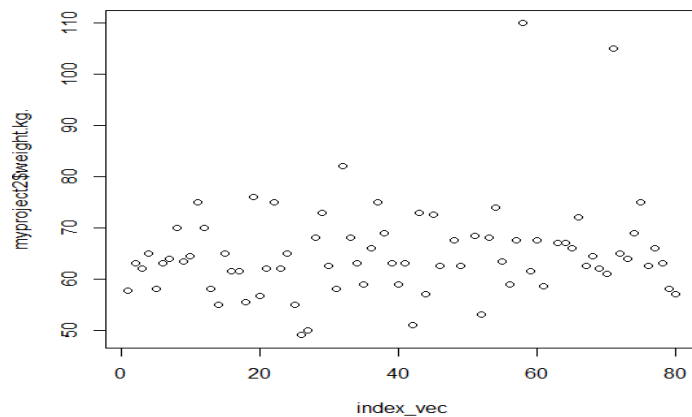
```

FINDING THE OUTLIER OF WEIGHT.KG.

```
index_vec = c(1:80)
```

```
plot(index_vec,myproject2$weight.kg.)
```

```
myproject2$weight.kg.[myproject2$weight.kg.<=999] <- NA
```

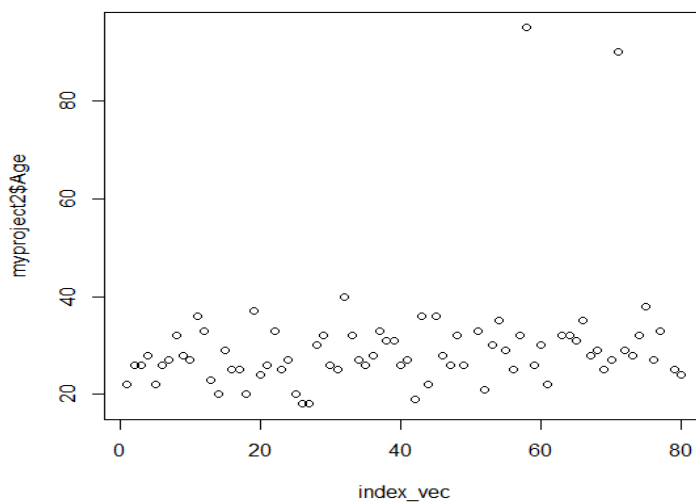


FINDING THE OUTLIER OF AGE

```
index_vec = c(1:80)
```

```
plot(index_vec,myproject2$Age)
```

```
myproject2$Age[myproject2$Age<=999] <- NA
```



Detecting the outlier as a missing value:

```
myproject2$Age[myproject2$Age==""] <- NA
```

```
myproject2$weight.kg.[myproject2$weight.kg==""] <- NA
```

```
myproject2$Delivery_number[myproject2$Delivery_number==""] <- NA
```

```
myproject2$Delivery_time[myproject2$Delivery_time==""] <- NA
```

```
myproject2$Blood[myproject2$Blood==""] <- NA
```

```
myproject2$Heart[myproject2$Heart==""] <- NA
```

```
myproject2$Caesarian[myproject2$Caesarian==""] <- NA
```

	id	Age	weight.kg.	Delivery_number	Delivery_time	Blood	Heart	Caesarian
1	1	22	57.7	1	0	high	0	0
2	2	26	63.0	2	0	normal	0	1
3	3	26	62.0	2	1	normal	0	0
4	4	28	65.0	1	0	high	0	0
5	5	22	58.0	2	0	normal	0	1
6	6	26	63.0	1	1	low	0	0
7	7	27	64.0	2	0	normal	0	0
8	8	32	70.0	3	0	normal	0	1
9	9	28	63.5	2	0	NA	0	0
10	10	27	64.5	1	1	normal	0	1
11	11	36	75.0	1	0	normal	0	0
12	12	33	70.0	1	1	low	0	1
13	13	23	58.0	1	1	normal	0	0
14	14	20	55.0	1	0	normal	1	0
15	15	29	65.0	1	NA	NA	1	1
16	16	25	61.5	1	2	low	0	0
17	17	25	61.5	1	0	normal	0	0
18	18	20	55.5	1	2	high	0	1
19	19	37	76.0	3	0	normal	1	1

Annotating Low as 1, Normal as 2 , and High as 3 from “Type” attribute:

```
myproject2$BP_RATE = factor(myproject2$Blood,  
                             levels = c("low","normal","high"),  
                             labels = c(1,2,3))
```

	id	Age	weight.kg.	Delivery_number	Delivery_time	Blood	Heart	Caesarian	BP_RATE
1	1	22	57.7	1	0	high	0	0	3
2	2	26	63.0	2	0	normal	0	1	2
3	3	26	62.0	2	1	normal	0	0	2
4	4	28	65.0	1	0	high	0	0	3
5	5	22	58.0	2	0	normal	0	1	2
6	6	26	63.0	1	1	low	0	0	1
7	7	27	64.0	2	0	normal	0	0	2
8	8	32	70.0	3	0	normal	0	1	2
9	9	28	63.5	2	0	NA	0	0	NA
10	10	27	64.5	1	1	normal	0	1	2
11	11	36	75.0	1	0	normal	0	0	2
12	12	33	70.0	1	1	low	0	1	1
13	13	23	58.0	1	1	normal	0	0	2
14	14	20	55.0	1	0	normal	1	0	2
15	15	29	65.0	1	NA	NA	1	1	NA
16	16	25	61.5	1	2	low	0	0	1
17	17	25	61.5	1	0	normal	0	0	2
18	18	20	55.5	1	2	high	0	1	3
19	19	37	76.0	3	0	normal	1	1	2

RECOVERING MISSING THE DATA BY FOLLOWING STRATEGIES

1. Removing the rows which has the values

```
myproject2.1<- na.omit(myproject2.1)
```

```
myproject2.1
```

	id	Age	weight.kg.	Delivery_number	Delivery_time	Blood	Heart	Caesarian	BP_RATE
1	1	22.00000	57.70000	1	0	high	0	0	3
2	2	26.00000	63.00000	2	0	normal	0	1	2
3	3	26.00000	62.00000	2	1	normal	0	0	2
4	4	28.00000	65.00000	1	0	high	0	0	3
5	5	22.00000	58.00000	2	0	normal	0	1	2
6	6	26.00000	63.00000	1	1	low	0	0	1
7	7	27.00000	64.00000	2	0	normal	0	0	2
8	8	32.00000	70.00000	3	0	normal	0	1	2
10	10	27.00000	64.50000	1	1	normal	0	1	2
11	11	36.00000	75.00000	1	0	normal	0	0	2
12	12	33.00000	70.00000	1	1	low	0	1	1
13	13	23.00000	58.00000	1	1	normal	0	0	2
14	14	20.00000	55.00000	1	0	normal	1	0	2
16	16	25.00000	61.50000	1	2	low	0	0	1

2. Recovering the MISSING VALUEs with the help of mean values

```
myproject2.1 <- myproject2
```

```
mean5 <- mean(myproject2.1$Age, na.rm = TRUE)
```

```
mean5
```

```
mean6 <- mean(myproject2.1$weight.kg., na.rm = TRUE)
```

```
mean6
```

```
mean7 <- mean(myproject2.1$Blood, na.rm = TRUE)
```

```
mean7
```

```
mean8 <- mean(myproject2.1$Caesarian, na.rm = TRUE)
```

```
mean8
```

```
mean9 <- mean(myproject2.1$Delivery_number, na.rm = TRUE)
```

```
mean9
```

```
mean10 <- mean(myproject2.1$Delivery_time, na.rm = TRUE)
```

```
mean10
```

```
myproject2.1$Age[is.na(myproject2.1$Age)] <- mean5
```

```
myproject2.1$weight.kg.[is.na(myproject2.1$weight.kg.)] <- mean6
```

```
myproject2.1$Blood[is.na(myproject2.1$Blood)] <- mean7
```

```
myproject2.1$Caesarian[is.na(myproject2.1$Caesarian)] <- mean8
```

```
myproject2.1$Delivery_number[is.na(myproject2.1$Delivery_number)] <- mean9
```

```
myproject2.1$Delivery_time[is.na(myproject2.1$Delivery_time)] <- mean10
```

```
myproject2.1
```

```
> myproject2.1
```

	id	Age	weight.kg.	Delivery_number	Delivery_time	Blood	Heart	Caesarian
1	1	22.00000	57.70000	1.000000	0.0000000	3.000000	0	0.0000000
2	2	26.00000	63.00000	2.000000	0.0000000	2.000000	0	1.0000000
3	3	26.00000	62.00000	2.000000	1.0000000	2.000000	0	0.0000000
4	4	28.00000	65.00000	1.000000	0.0000000	3.000000	0	0.0000000
5	5	22.00000	58.00000	2.000000	0.0000000	2.000000	0	1.0000000
6	6	26.00000	63.00000	1.000000	1.0000000	1.000000	0	0.0000000
7	7	27.00000	64.00000	2.000000	0.0000000	2.000000	0	0.0000000
8	8	32.00000	70.00000	3.000000	0.0000000	2.000000	0	1.0000000
9	9	28.00000	63.50000	2.000000	0.0000000	2.025974	0	0.0000000
10	10	27.00000	64.50000	1.000000	1.0000000	2.000000	0	1.0000000
11	11	36.00000	75.00000	1.000000	0.0000000	2.000000	0	0.0000000
12	12	33.00000	70.00000	1.000000	1.0000000	1.000000	0	1.0000000
13	13	23.00000	58.00000	1.000000	1.0000000	2.000000	0	0.0000000
14	14	20.00000	55.00000	1.000000	0.0000000	2.000000	1	0.0000000
15	15	29.00000	65.00000	1.000000	0.6233766	2.025974	1	1.0000000
16	16	25.00000	61.50000	1.000000	2.0000000	1.000000	0	0.0000000
17	17	25.00000	61.50000	1.000000	0.0000000	2.000000	0	0.0000000
18	18	20.00000	55.50000	1.000000	2.0000000	3.000000	0	1.0000000
19	19	37.00000	76.00000	3.000000	0.0000000	2.000000	1	1.0000000
20	20	24.00000	56.60000	1.000000	2.0000000	1.000000	1	1.0000000
21	21	26.00000	62.00000	1.000000	1.0000000	2.000000	0	0.0000000
22	22	33.00000	75.00000	2.000000	0.0000000	1.000000	1	1.0000000
23	23	25.00000	62.00000	1.000000	1.0000000	3.000000	0	0.0000000
24	24	27.00000	65.00000	1.679487	0.6233766	1.000000	1	1.0000000
25	25	20.00000	55.00000	1.000000	0.0000000	3.000000	1	1.0000000

3. Recovering the MISSING VALUEs with the help of mode values

```
myproject2.1 <- myproject2
```

```
mode5 <- Mode(myproject2.1$Age, na.rm = TRUE)
```

```
mode5
```

```
mode6 <- Mode(myproject2.1$weight.kg., na.rm = TRUE)
```

```
mode6
```

```
mode7 <- Mode(myproject2.1$Blood, na.rm = TRUE)
```

```
mode7
```

```
mode8 <- Mode(myproject2.1$Caesarian, na.rm = TRUE)
```

```
mode8
```

```
mode9 <- Mode(myproject2.1$Delivery_number, na.rm = TRUE)
```

```
mode9
```

```
mode10 <- Mode(myproject2.1$Delivery_time, na.rm = TRUE)
```

```
mode10
```

```

myproject2.1$Age[is.na(myproject2.1$Age)] <- mode5
myproject2.1$weight.kg.[is.na(myproject2.1$weight.kg.)] <- mode6
myproject2.1$Blood[is.na(myproject2.1$Blood)] <- mode7
myproject2.1$Caesarian[is.na(myproject2.1$Caesarian)] <- mode8
myproject2.1$Delivery_number[is.na(myproject2.1$Delivery_number)] <- mode9
myproject2.1$Delivery_time[is.na(myproject2.1$Delivery_time)] <- mode10
myproject2.1

```

```

> myproject2.1
  id Age weight.kg. Delivery_number Delivery_time Blood Heart Caesarian
1   1  22    57.7           1           0      3      0         0
2   2  26    63.0           2           0      2      0         1
3   3  26    62.0           2           1      2      0         0
4   4  28    65.0           1           0      3      0         0
5   5  22    58.0           2           0      2      0         1
6   6  26    63.0           1           1      1      0         0
7   7  27    64.0           2           0      2      0         0
8   8  32    70.0           3           0      2      0         1
9   9  28    63.5           2           0      2      0         0
10  10 27    64.5           1           1      2      0         1
11  11 36    75.0           1           0      2      0         0
12  12 33    70.0           1           1      1      0         1
13  13 23    58.0           1           1      2      0         0
14  14 20    55.0           1           0      2      1         0
15  15 29    65.0           1           0      2      1         1
16  16 25    61.5           1           2      1      0         0
17  17 25    61.5           1           0      2      0         0
18  18 20    55.5           1           2      3      0         1
19  19 37    76.0           3           0      2      1         1
20  20 24    56.6           1           2      1      1         1
21  21 26    62.0           1           1      2      0         0
22  22 33    75.0           2           0      1      1         1
23  23 25    62.0           1           1      3      0         0
24  24 27    65.0           1           0      1      1         1
25  25 20    55.0           1           0      3      1         1
26  26 18    49.0           1           0      2      0         0

```

4. Recovering the MISSING VALUES with the help of median values

```

myproject2.1 <- myproject2
Median5 <- Median(myproject2.1$Age, na.rm = TRUE)
Median5
Median6 <- Median(myproject2.1$weight.kg., na.rm = TRUE)
Median6
Median7 <- Median(myproject2.1$Blood, na.rm = TRUE)

```


Median7

```
Median8 <- Median(myproject2.1$Caesarian, na.rm = TRUE)
```

Median8

```
Median9 <- Median(myproject2.1$Delivery_number, na.rm = TRUE)
```

Median9

```
Median10 <- Median(myproject2.1$Delivery_time, na.rm = TRUE)
```

Median10

```
myproject2.1$Age[is.na(myproject2.1$Age)] <- Median5
```

```
myproject2.1$weight.kg.[is.na(myproject2.1$weight.kg.)] <- Median6
```

```
myproject2.1$Blood[is.na(myproject2.1$Blood)] <- Median7
```

```
myproject2.1$Caesarian[is.na(myproject2.1$Caesarian)] <- Median8
```

```
myproject2.1$Delivery_number[is.na(myproject2.1$Delivery_number)] <- Median9
```

```
myproject2.1$Delivery_time[is.na(myproject2.1$Delivery_time)] <- Median10
```

myproject2.1

```
> myproject2.1
```

	id	Age	weight.kg.	Delivery_number	Delivery_time	Blood	Heart	Caesarian
1	1	22	57.7	1	0	3	0	0
2	2	26	63.0	2	0	2	0	1
3	3	26	62.0	2	1	2	0	0
4	4	28	65.0	1	0	3	0	0
5	5	22	58.0	2	0	2	0	1
6	6	26	63.0	1	1	1	0	0
7	7	27	64.0	2	0	2	0	0
8	8	32	70.0	3	0	2	0	1
9	9	28	63.5	2	0	2	0	0
10	10	27	64.5	1	1	2	0	1
11	11	36	75.0	1	0	2	0	0
12	12	33	70.0	1	1	1	0	1
13	13	23	58.0	1	1	2	0	0
14	14	20	55.0	1	0	2	1	0
15	15	29	65.0	1	0	2	1	1
16	16	25	61.5	1	2	1	0	0
17	17	25	61.5	1	0	2	0	0
18	18	20	55.5	1	2	3	0	1
19	19	37	76.0	3	0	2	1	1
20	20	24	56.6	1	2	1	1	1
21	21	26	62.0	1	1	2	0	0
22	22	33	75.0	2	0	1	1	1
23	23	25	62.0	1	1	3	0	0
24	24	27	65.0	1	0	1	1	1
25	25	20	55.0	1	0	3	1	1
26	26	18	49.0	1	0	2	0	0
27	27	18	50.0	1	0	3	1	1
28	28	30	68.0	1	0	2	0	0
29	29	32	73.0	1	0	3	1	1
30	30	26	62.5	2	1	2	1	0
31	31	25	58.0	1	0	1	0	0
32	32	40	82.0	1	0	2	1	1
33	33	32	68.0	2	0	3	1	1

THE END