AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)

Faculty of Science and Technology (FST)

Course Title: INTRODUCTION TO DATA SCIENCE

Spring 2023-2024

Section: (A), Group: 05

Project Title: Apply data preparation steps (which can be applied) for the given data set.

Supervised By

TOHEDUL ISLAM

Faculty of Science and Technology

American International University-Bangladesh

Submitted By:

NAME	ID
Punam Das	21-44946-2
Mithi Zaman	21-44753-1

Dataset Description:

The Caesarian Section Classification Dataset is a medical dataset that includes data from 80 pregnant women, focusing on significant factors associated with delivery complications. It consists of seven attributes: 'Age', 'Gender', 'Weight', 'Delivery number', 'Delivery time', 'Blood of Pressure', and 'Heart Problem', each with specific categorical values. 'Delivery time' is categorized as Timely, Premature, and Latecomer, while 'Blood of Pressure' is classified into Low, Normal, and High states. 'Heart Problem' is divided into Apt (no problem) and Inept (problem present). The dataset also includes a class variable, 'Caesarian', indicating the outcome of whether a caesarian section was necessary (Yes) or not (No). In this dataset, 'Gender' and 'Caesarian' are categorical, while the rest are numerical. The 'Caesarian' variable is binary, reflecting the presence (Yes) or absence (No) of a caesarian section. This dataset is crucial for understanding and predicting caesarian section needs based on the given attributes.

Attributes:

Age: Represents the age of the pregnant women in the dataset, ranging from 17 to 40 years.

Gender: Specifies the sex of the individuals, categorized as Male or Female.

Weight: Indicates the weight of the individuals, which can be a critical factor in pregnancy and delivery.

Delivery Number: Denotes the count of deliveries the pregnant woman has had, ranging from 1 to 4.

Delivery Time: Categorized into three classes—0 for timely, 1 for premature, and 2 for latecomer—indicating the stage of delivery.

Blood of Pressure: Classified into three states—0 for low, 1 for normal, and 2 for high—reflecting the blood pressure condition of the individual.

Heart Problem: Divided into two categories—0 for apt (no problem) and 1 for inept (problem present)—showing the heart condition.

Caesarian (Target Variable): The outcome variable, indicating whether a caesarian section was performed (1 for Yes, 0 for No), which is the primary focus of this dataset.

PURPOSE: The Caesarian Section Classification Dataset is designed to assist medical professionals in predicting the necessity of a caesarian section during childbirth. It evaluates various factors such as age, gender, weight, delivery number, delivery time, blood pressure, and heart condition to determine the likelihood of requiring a caesarian procedure, aiming to improve the safety and outcomes for both mother and child.

Project Overview:

A critical step in data analysis is data pre-processing, which is transforming unprocessed data into a format that computers and machine learning systems can easily understand and analyze. In actuality, raw data is often jumbled with plenty of errors, require cleaning before it may be used to a particular task. Moreover, univariate analysis is required, which involves evaluating each variable in a dataset independently without taking the relationships between variables into account.

It is noticeable that the data set is not well formatted. The dataset has to be cleaned and pre-processed before using it.

Data Pre-processing:

1. Importing the Dataset:

The dataset is located in a file called mid_project 12.15.26.csv in the current working directory. To begin data pre-processing using R, the first step is to import the dataset. Once imported, the mid_project 12.15.26.csv file is transformed into an R data frame and stored in a variable named "Dataset_mid". After printing the dataset, it looks like this-

R code:

```
Dataset_mid<-read.csv("/Users/mithizaman/Documents/9th semester/INTRODUCTION TO DATA SCIENCE/Data Science/mid_project 12.15.26.csv",header = TRUE,sep = ",")
Dataset_mid
```

```
> Dataset_mid<-read.csv("/Users/mithizaman/Documents/9th semester/INTRODUCTION TO DATA SCIENCE/Data Science/mid_project 12.15.26.csv",heade
r = TRUE,sep = ",")
> Dataset_mid
   id Age  Gender weight.kg. Delivery_number Delivery_time  Blood Heart Caesarian
1   1  22  Female       57.7               1             0   high     0         0
2   2  26    Male         63               2             0 normal     0         1
3   3  26    Male         62               2             1 normal     0         0
4   4  28    Male         65               1             0   high     0         0
5   5  22  Female         58               2             0 normal     0         1
6   6  26    Male         63              NA             1    low     0         0
7   7  27  Female         64               2             0 normal     0         0
8   8  32    Male         70               3             0 normal     0         1
9   9  28  Female       63.5               2             0            0         0
10 10  NA    Male       64.5               1             1 normal     0         1
11 11  36    Male         75               1             0 normal     0         0
12 12  33                 70               1             1    low     0         1
13 13  23  female         58               1             1 normal     0         0
14 14  20    Male         55               1             0 normal     1         0
15 15  29    Male         65               1            NA            1         1
16 16  25  female       61.5               1             2    low     0         0
17 17  25    Male       61.5               1             0 normal     0         0
18 18  20    Male       55.5               1             2   high     0         1
19 19  37    Male         76               3             0 normal     1         1
20 20  24    Male       56.6               1             2    low     1         1
21 21  26    Male         62               1             1 normal     0         0
22 22  33    male        75X               2             0    low     1         1
23 23  25    Male         62               1             1   high     0         0
24 24  27    Male         65               2            NA    low     1         1
25 25  20    Male         55               1             0   high     1         1
26 26  18    Male         49               1             0 normal     0         0
27 27  18    Male         50               1            NA   high     1         1
28 28  30  Female         68               1             0 normal     0         0
29 29  32    male         73               1             0   high     1         1
30 30  26    Male       62.5              NA             1 normal     1         0
31 31  25    Male         58               1             0    low     0         0
32 32  40    Male         82               1             0 normal     1         1
```

This is the imported Dataset

Filter

	id	Age	Gender	weight.kg.	Delivery_number	Delivery_time	Blood	Heart	Caesarian
1	1	22	Female	57.7	1	0	high	0	0
2	2	26	Male	63	2	0	normal	0	1
3	3	26	Male	62	2	1	normal	0	0
4	4	28	Male	65	1	0	high	0	0
5	5	22	Female	58	2	0	normal	0	1
6	6	26	Male	63	NA	1	low	0	0
7	7	27	Female	64	2	0	normal	0	0
8	8	32	Male	70	3	0	normal	0	1
9	9	28	Female	63.5	2	0		0	0
10	10	NA	Male	64.5	1	1	normal	0	1
11	11	36	Male	75	1	0	normal	0	0
12	12	33		70	1	1	low	0	1
13	13	23	female	58	1	1	normal	0	0
14	14	20	Male	55	1	0	normal	1	0
15	15	29	Male	65	1	NA		1	1
16	16	25	female	61.5	1	2	low	0	0
17	17	25	Male	61.5	1	0	normal	0	0
18	18	20	Male	55.5	1	2	high	0	1
19	19	37	Male	76	3	0	normal	1	1
20	20	24	Male	56.6	1	2	low	1	1
21	21	26	Male	62	1	1	normal	0	0
22	22	33	male	75X	2	0	low	1	1
23	23	25	Male	62	1	1	high	0	0
24	24	27	Male	65	2	NA	low	1	1
25	25	20	Male	55	1	0	high	1	1
26	26	18	Male	49	1	0	normal	0	0
27	27	18	Male	50	1	NA	high	1	1
28	28	30	Female	68	1	0	normal	0	0
29	29	32	male	73	1	0	high	1	1

2. Dealing with Missing Values:

For checking the missing value (NA) present in column name: id[0], Age[4] Gender[0], weight.kg.[0], Delivery_number[5], Delivery_time[3], Blood[0], Heart[0], Caesarian[2]. We need to use the give code to find the missing value.

R code:

colSums(is.na(Dataset_mid)

)

```
> colSums(is.na(Dataset_mid))
          id          Age       Gender    weight.kg. Delivery_number Delivery_time        Blood        Heart    Caesarian
           0            4            0            0            5            3            0            0            2
>
```

Before, the dataset look like this –

Filter

	id	Age	Gender	weight.kg.	Delivery_number	Delivery_time	Blood	Heart	Caesarian
1	1	22	Female	57.7	1	0	high	0	0
2	2	26	Male	63	2	0	normal	0	1
3	3	26	Male	62	2	1	normal	0	0
4	4	28	Male	65	1	0	high	0	0
5	5	22	Female	58	2	0	normal	0	1
6	6	26	Male	63	NA	1	low	0	0
7	7	27	Female	64	2	0	normal	0	0
8	8	32	Male	70	3	0	normal	0	1
9	9	28	Female	63.5	2	0		0	0
10	10	NA	Male	64.5	1	1	normal	0	1
11	11	36	Male	75	1	0	normal	0	0
12	12	33		70	1	1	low	0	1
13	13	23	female	58	1	1	normal	0	0
14	14	20	Male	55	1	0	normal	1	0
15	15	29	Male	65	1	NA		1	1
16	16	25	female	61.5	1	2	low	0	0
17	17	25	Male	61.5	1	0	normal	0	0
18	18	20	Male	55.5	1	2	high	0	1
19	19	37	Male	76	3	0	normal	1	1
20	20	24	Male	56.6	1	2	low	1	1
21	21	26	Male	62	1	1	normal	0	0
22	22	33	male	75X	2	0	low	1	1
23	23	25	Male	62	1	1	high	0	0
24	24	27	Male	65	2	NA	low	1	1
25	25	20	Male	55	1	0	high	1	1
26	26	18	Male	49	1	0	normal	0	0
27	27	18	Male	50	1	NA	high	1	1
28	28	30	Female	68	1	0	normal	0	0
29	29	32	male	73	1	0	high	1	1

2.<u>Handling missing value:</u> We have two ways to handling missing values.

a.Replace by Most Frequent/Average Value.

R Code:

>Dataset_mid$Caesarian[is.na(Dataset_mid$Caesarian)]<-
mean(Dataset_mid$Caesarian, na.rm = TRUE)
> Dataset_mid

```
> Dataset_mid$Caesarian[is.na(Dataset_mid$Caesarian)] <- mean(Dataset_mid$Caesarian
> Dataset_mid
   id Age  Gender weight.kg. Delivery_number Delivery_time  Blood Heart Caesarian
1   1  22  Female       57.7               1             0   high     0 0.0000000
2   2  26    Male         63               2             0 normal     0 1.0000000
3   3  26    Male         62               2             1 normal     0 0.0000000
4   4  28    Male         65               1             0   high     0 0.0000000
5   5  22  Female         58               2             0 normal     0 1.0000000
6   6  26    Male         63              NA             1    low     0 0.0000000
7   7  27  Female         64               2             0 normal     0 0.0000000
8   8  32    Male         70               3             0 normal     0 1.0000000
9   9  28  Female       63.5               2             0            0 0.0000000
10 10  NA    Male       64.5               1             1 normal     0 1.0000000
11 11  36    Male         75               1             0 normal     0 0.0000000
12 12  33                 70               1             1    low     0 1.0000000
13 13  23  female         58               1             1 normal     0 0.0000000
14 14  20    Male         55               1             0 normal     1 0.0000000
15 15  29    Male         65               1            NA            1 1.0000000
16 16  25  female       61.5               1             2    low     0 0.0000000
17 17  25    Male       61.5               1             0 normal     0 0.0000000
18 18  20    Male       55.5               1             2   high     0 1.0000000
19 19  37    Male         76               3             0 normal     1 1.0000000
20 20  24    Male       56.6               1             2    low     1 1.0000000
21 21  26    Male         62               1             1 normal     0 0.0000000
22 22  33    male        75X               2             0    low     1 1.0000000
23 23  25    Male         62               1             1   high     0 0.0000000
24 24  27    Male         65               2            NA    low     1 1.0000000
25 25  20    Male         55               1             0   high     1 1.0000000
26 26  18    Male         49               1             0 normal     0 0.0000000
27 27  18    Male         50               1            NA   high     1 1.0000000
28 28  30  Female         68               1             0 normal     0 0.0000000
29 29  32    male         73               1             0   high     1 1.0000000
30 30  26    Male       62.5              NA             1 normal     1 0.0000000
31 31  25    Male         58               1             0    low     0 0.0000000
32 32  40    Male         82               1             0 normal     1 1.0000000
```

b.Discard Instances.

R Code:

```
Dataset_mid<- na.omit(Dataset_mid)
Dataset_mid
```

```
>  Dataset_mid<- na.omit(Dataset_mid)
> Dataset_mid
   id Age  Gender weight.kg. Delivery_number Delivery_time  Blood Heart Caesarian
1   1  22  Female       57.7               1             0   high     0 0.0000000
2   2  26    Male         63               2             0 normal     0 1.0000000
3   3  26    Male         62               2             1 normal     0 0.0000000
4   4  28    Male         65               1             0   high     0 0.0000000
5   5  22  Female         58               2             0 normal     0 1.0000000
7   7  27  Female         64               2             0 normal     0 0.0000000
8   8  32    Male         70               3             0 normal     0 1.0000000
9   9  28  Female       63.5               2             0            0 0.0000000
11 11  36    Male         75               1             0 normal     0 0.0000000
12 12  33                 70               1             1    low     0 1.0000000
13 13  23  female         58               1             1 normal     0 0.0000000
14 14  20    Male         55               1             0 normal     1 0.0000000
16 16  25  female       61.5               1             2    low     0 0.0000000
17 17  25    Male       61.5               1             0 normal     0 0.0000000
18 18  20    Male       55.5               1             2   high     0 1.0000000
19 19  37    Male         76               3             0 normal     1 1.0000000
20 20  24    Male       56.6               1             2    low     1 1.0000000
21 21  26    Male         62               1             1 normal     0 0.0000000
22 22  33    male        75X               2             0    low     1 1.0000000
23 23  25    Male         62               1             1   high     0 0.0000000
25 25  20    Male         55               1             0   high     1 1.0000000
26 26  18    Male         49               1             0 normal     0 0.0000000
28 28  30  Female         68               1             0 normal     0 0.0000000
29 29  32    male         73               1             0   high     1 1.0000000
31 31  25    Male         58               1             0    low     0 0.0000000
32 32  40    Male         82               1             0 normal     1 1.0000000
33 33  32    Male         68               2             0   high     1 1.0000000
34 34  27    Male         63               2             0 normal     1 1.0000000
35 35  26                 59               2             2 normal     0 1.0000000
37 37  33    Male         75               1             1 normal     0 0.0000000
38 38  31    Male         69               2             2 normal     0 0.0000000
39 39  31    Male         63               1             0 normal     0 0.0000000
40 40  26    Male         59               1             2    low     1 1.0000000
41 41  27    Male         63               1             0   high     1 1.0000000
43 43  36    Male         73               1             1   high     0 1.0000000
44 44  22    Male         57               1             0 normal     0 1.0000000
```

I have discarded the rows that were NA.

Here we can see that in the “Gender” column, some values are missing. We can find it out in this way-

R Code:

Dataset_mid[,3]

```
> Dataset_mid[,3]
 [1] "Female"  "Male"    "Male"    "Male"    "Female"  "Female"  "Male"    "Female"  "Male"    ""        "female"  "Male"    "female
[14] "Male"    "Male"    "Male"    "Male"    "Male"    "male"    "Male"    "Male"    "Male"    "Female"  "male"    "Male"    "Male"
[27] "Male"    "Male"    ""        "Male"    "Male"    "Male"    "Male"    "Male"    "Male"    "Male"    "Female"  "Male"    "Male"
[40] "Male"    "male"    "Male"    "Male"    "Male"    "Male"    "Male"    "Male"    "Male"    "Male"    "Male"    "Female"  "Male"
[53] "male"    "Male"    "Feemale" "Mmale"   "Female"  "Male"    "male"    "Male"    "Male"    "Male"    "Male"    "Male"    "Male"
[66] "Male"
```

As the Gender column we can overcome this problem using
The edit function()

R Code:

max(Dataset_mid$Gender)
Dataset_mid$Gender<-edit(Dataset_mid$Gender)
Dataset_mid

```
> max(Dataset_mid$Gender)
[1] "Mmale"
>
```

```
Dataset_mid$Gender<-edit(Dataset_mid$Gender)
Dataset_mid
   id Age  Gender weight.kg. Delivery_number Delivery_time  Blood Heart Caesarian
    1  22  Female       57.7               1             0   high     0 0.0000000
    2  26    Male         63               2             0 normal     0 1.0000000
    3  26    Male         62               2             1 normal     0 0.0000000
    4  28    Male         65               1             0   high     0 0.0000000
    5  22  Female         58               2             0 normal     0 1.0000000
    7  27  Female         64               2             0 normal     0 0.0000000
    8  32    Male         70               3             0 normal     0 1.0000000
    9  28  Female       63.5               2             0            0 0.0000000
1  11  36    Male         75               1             0 normal     0 0.0000000
2  12  33   Mmale         70               1             1    low     0 1.0000000
3  13  23  female         58               1             1 normal     0 0.0000000
4  14  20    Male         55               1             0 normal     1 0.0000000
6  16  25  female       61.5               1             2    low     0 0.0000000
7  17  25    Male       61.5               1             0 normal     0 0.0000000
8  18  20    Male       55.5               1             2   high     0 1.0000000
9  19  37    Male         76               3             0 normal     1 1.0000000
0  20  24    Male       56.6               1             2    low     1 1.0000000
1  21  26    Male         62               1             1 normal     0 0.0000000
2  22  33    male        75X               2             0    low     1 1.0000000
3  23  25    Male         62               1             1   high     0 0.0000000
5  25  20    Male         55               1             0   high     1 1.0000000
6  26  18    Male         49               1             0 normal     0 0.0000000
8  28  30  Female         68               1             0 normal     0 0.0000000
9  29  32    male         73               1             0   high     1 1.0000000
1  31  25    Male         58               1             0    low     0 0.0000000
2  32  40    Male         82               1             0 normal     1 1.0000000
3  33  32    Male         68               2             0   high     1 1.0000000
4  34  27    Male         63               2             0 normal     1 1.0000000
5  35  26   Mamle         59               2             2 normal     0 1.0000000
7  37  33    Male         75               1             1 normal     0 0.0000000
8  38  31    Male         69               2             2 normal     0 0.0000000
9  39  31    Male         63               1             0 normal     0 0.0000000
0  40  26    Male         59               1             2    low     1 1.0000000
1  41  27    Male         63               1             0   high     1 1.0000000
3  43  36    Male         73               1             1   high     0 1.0000000
```

R Code:

Dataset_mid[,4]

```
Dataset_mid[,4]
 [1] "57.7" "63"   "62"   "65"   "58"   "64"   "70"   "63.5" "75"   "70"   "58"   "55"   "61.5" "61.5" "55.5" "76"   "56.6" "62"
[19] "75X"  "62"   "55"   "49"   "68"   "73"   "58"   "82"   "68"   "63"   "59"   "75"   "69"   "63"   "59"   "63"   "73"   "57"
[37] "62.5" ""     "67.5" "62.5" ""     "68.5" "53"   "68"   "74"   "59"   "67.5" "110"  "61.5" "58.5" ""     "67"   "66"   "72"
[55] "62.5" "64.5" "62"   "61"   "65"   "64"   "69"   "75"   "62.5" "63"   "58"   "57"
data
```

As the weight.kg. column we can overcome this problem using
The edit function()

R Code:

max(Dataset_mid$weight.kg.)
Dataset_mid$weight.kg.<-edit(Dataset_mid$weight.kg.)
Dataset_mid

```
> max(Dataset_mid$weight.kg.)
[1] "82"
> Dataset_mid$weight.kg.r<-edit(Dataset_mid$weight.kg.)
data
```

31	31	25	Male	58	1	0	low	0	0.0000000
32	32	40	Male	82	1	0	normal	1	1.0000000
33	33	32	Male	68	2	0	high	1	1.0000000
34	34	27	Male	63	2	0	normal	1	1.0000000
35	35	26	Mamle	59	2	2	normal	0	1.0000000
37	37	33	Male	75	1	1	normal	0	0.0000000
38	38	31	Male	69	2	2	normal	0	0.0000000
39	39	31	Male	63	1	0	normal	0	0.0000000
40	40	26	Male	59	1	2	low	1	1.0000000
41	41	27	Male	63	1	0	high	1	1.0000000
43	43	36	Male	73	1	1	high	0	1.0000000
44	44	22	Male	57	1	0	normal	0	1.0000000
46	46	28	Female	62.5	3	0	normal	1	1.0000000
47	47	26	Male	82	1	0	normal	0	0.0000000
48	48	32	Male	67.5	2	0	high	1	1.0000000
49	49	26	Male	62.5	2	2	normal	0	0.0000000
50	50	30	male	82	2	0	low	1	1.0000000
51	51	33	Male	68.5	3	2	normal	1	0.0000000
52	52	21	Male	53	2	1	low	1	1.0000000
53	53	30	Male	68	3	2	high	0	0.0000000
54	54	35	Male	74	1	1	low	0	0.0000000
56	56	25	Male	59	2	0	normal	0	0.0000000
57	57	32	Male	67.5	3	1	low	1	1.0000000
58	58	95	Male	110	1	0	low	0	1.0000000
59	59	26	Male	61.5	1	0	high	0	1.0000000
60	60	30	Male	67.5	2	1	high	1	0.5769231
61	61	22	Male	58.5	1	2	high	0	0.0000000
62	62	160	Female	82	1	0	normal	0	1.0000000
64	64	32	Male	67	2	0	normal	1	1.0000000
65	65	31	male	66	1	2	high	1	0.0000000
66	66	35	Male	72	2	0	normal	0	1.0000000
67	67	28	Feemale	62.5	3	0	normal	0	1.0000000
68	68	29	Mmale	64.5	2	0	normal	1	0.0000000
69	69	25	Female	62	1	0	low	0	1.0000000
70	70	27	Male	61	2	2	low	0	0.0000000

R Code:

Dataset_mid[,7]

```
> Dataset_mid[,7]
 [1] "high"   "normal" "normal" "high"   "normal" "normal" "normal" ""       "normal" "low"    "normal" "normal" "low"    "normal"
[15] "high"   "normal" "low"    "normal" "low"    "high"   "high"   "normal" "normal" "high"   "low"    "normal" "high"   "normal"
[29] "normal" "normal" "normal" "normal" "low"    "high"   "high"   "normal" "normal" "normal" "high"   "normal" "low"    "normal"
[43] "low"    "high"   "low"    "normal" "low"    "low"    "high"   "high"   "normal" "normal" "high"   "normal" "normal" "normal"
[57] "low"    "low"    ""       "normal" "normal" "high"   "normal" "high"   "low"    "low"
> data
```

As the Blood column we can overcome this problem using

The edit function()

R Code:

max(Dataset_mid$Blood)

Dataset_mid$Blood<-edit(Dataset_mid$Blood)

Dataset_mid

```
> max(Dataset_mid$Blood)
[1] "normal"
> Dataset_mid$Blood<-edit(Dataset_mid$Blood)
data
```

```
Dataset_mid$Blood<-edit(Dataset_mid$Blood)
Dataset_mid
   id Age  Gender weight.kg. Delivery_number Delivery_time  Blood Heart Caesarian
    1  22  Female       57.7               1             0   high     0 0.0000000
    2  26    Male         63               2             0 normal     0 1.0000000
    3  26    Male         62               2             1 normal     0 0.0000000
    4  28    Male         65               1             0   high     0 0.0000000
    5  22  Female         58               2             0 normal     0 1.0000000
    7  27  Female         64               2             0 normal     0 0.0000000
    8  32    Male         70               3             0 normal     0 1.0000000
    9  28  Female       63.5               2             0 normal     0 0.0000000
1  11  36    Male         75               1             0 normal     0 0.0000000
2  12  33   Mmale         70               1             1    low     0 1.0000000
3  13  23  female         58               1             1 normal     0 0.0000000
4  14  20    Male         55               1             0 normal     1 0.0000000
6  16  25  female       61.5               1             2    low     0 0.0000000
7  17  25    Male       61.5               1             0 normal     0 0.0000000
8  18  20    Male       55.5               1             2   high     0 1.0000000
9  19  37    Male         76               3             0 normal     1 1.0000000
0  20  24    Male       56.6               1             2    low     1 1.0000000
1  21  26    Male         62               1             1 normal     0 0.0000000
2  22  33    male        75X               2             0    low     1 1.0000000
3  23  25    Male         62               1             1   high     0 0.0000000
5  25  20    Male         55               1             0   high     1 1.0000000
6  26  18    Male         49               1             0 normal     0 0.0000000
8  28  30  Female         68               1             0 normal     0 0.0000000
9  29  32    male         73               1             0   high     1 1.0000000
1  31  25    Male         58               1             0    low     0 0.0000000
2  32  40    Male         82               1             0 normal     1 1.0000000
3  33  32    Male         68               2             0   high     1 1.0000000
4  34  27    Male         63               2             0 normal     1 1.0000000
5  35  26   Mamle         59               2             2 normal     0 1.0000000
7  37  33    Male         75               1             1 normal     0 0.0000000
8  38  31    Male         69               2             2 normal     0 0.0000000
9  39  31    Male         63               1             0 normal     0 0.0000000
0  40  26    Male         59               1             2    low     1 1.0000000
1  41  27    Male         63               1             0   high     1 1.0000000
3  43  36    Male         73               1             1   high     0 1.0000000
4  44  22    Male         57               1             0 normal     0 1.0000000
```

3.Invalid Value: We can see in the dataset there many invalid values in attribute Gender, Age and Weight .So we need to correct or reject the invalid values. Before, the dataset look like this those are the invalid values that we can see

R Code:

Dataset_mid$Gender

```
› Dataset_mid$Gender
 [1] "Female"  "Male"    "Male"    "Male"    "Female"  "Female"  "Male"    "Female"  "Male"    "Mmale"   "female"  "Male"
[13] "female"  "Male"    "Male"    "Male"    "Male"    "Male"    "male"    "Male"    "Male"    "Male"    "Female"  "male"
[25] "Male"    "Male"    "Male"    "Male"    "Mmale"   "Male"    "Male"    "Male"    "Male"    "Male"    "Male"    "Male"
[37] "Female"  "Male"    "Male"    "Male"    "male"    "Male"    "Male"    "Male"    "Male"    "Male"    "Male"    "Male"
[49] "Male"    "Male"    "Female"  "Male"    "male"    "Male"    "Feemale" "Mmale"   "Female"  "Male"    "male"    "Male"
[61] "Male"    "Male"    "Male"    "Male"    "Male"    "Male"
```

R Code:

invalid_indices<-grep("Mmale",Dataset_mid$Gender)
Dataset_mid$Gender[invalid_indices]<-"Male"
Dataset_mid

38	38	31	Male	69	2	2	normal	0	0.0000000
39	39	31	Male	63	1	0	normal	0	0.0000000
40	40	26	Male	59	1	2	low	1	1.0000000
41	41	27	Male	63	1	0	high	1	1.0000000
43	43	36	Male	73	1	1	high	0	1.0000000
44	44	22	Male	57	1	0	normal	0	1.0000000
46	46	28	Female	62.5	3	0	normal	1	1.0000000
47	47	26	Male	82	1	0	normal	0	0.0000000
48	48	32	Male	67.5	2	0	high	1	1.0000000
49	49	26	Male	62.5	2	2	normal	0	0.0000000
50	50	30	male	82	2	0	low	1	1.0000000
51	51	33	Male	68.5	3	2	normal	1	0.0000000
52	52	21	Male	53	2	1	low	1	1.0000000
53	53	30	Male	68	3	2	high	0	0.0000000
54	54	35	Male	74	1	1	low	0	0.0000000
56	56	25	Male	59	2	0	normal	0	0.0000000
57	57	32	Male	67.5	3	1	low	1	1.0000000
58	58	95	Male	110	1	0	low	0	1.0000000
59	59	26	Male	61.5	1	0	high	0	1.0000000
60	60	30	Male	67.5	2	1	high	1	0.5769231
61	61	22	Male	58.5	1	2	high	0	0.0000000
62	62	160	Female	82	1	0	normal	0	1.0000000
64	64	32	Male	67	2	0	normal	1	1.0000000
65	65	31	male	66	1	2	high	1	0.0000000
66	66	35	Male	72	2	0	normal	0	1.0000000
67	67	28	Feemale	62.5	3	0	normal	0	1.0000000
68	68	29	Male	64.5	2	0	normal	1	0.0000000
69	69	25	Female	62	1	0	low	0	1.0000000
70	70	27	Male	61	2	2	low	0	0.0000000

R Code:

```
invalid_indices<-grep("Feemale",Dataset_mid$Gender)
Dataset_mid$Gender[invalid_indices]<-"Female"
Dataset_mid
```

38	38	31	Male	69	2	2	normal	0	0.0000000		
39	39	31	Male	63	1	0	normal	0	0.0000000		
40	40	26	Male	59	1	2	low	1	1.0000000		
41	41	27	Male	63	1	0	high	1	1.0000000		
43	43	36	Male	73	1	1	high	0	1.0000000		
44	44	22	Male	57	1	0	normal	0	1.0000000		
46	46	28	Female	62.5	3	0	normal	1	1.0000000		
47	47	26	Male	82	1	0	normal	0	0.0000000		
48	48	32	Male	67.5	2	0	high	1	1.0000000		
49	49	26	Male	62.5	2	2	normal	0	0.0000000		
50	50	30	male	82	2	0	low	1	1.0000000		
51	51	33	Male	68.5	3	2	normal	1	0.0000000		
52	52	21	Male	53	2	1	low	1	1.0000000		
53	53	30	Male	68	3	2	high	0	0.0000000		
54	54	35	Male	74	1	1	low	0	0.0000000		
56	56	25	Male	59	2	0	normal	0	0.0000000		
57	57	32	Male	67.5	3	1	low	1	1.0000000		
58	58	95	Male	110	1	0	low	0	1.0000000		
59	59	26	Male	61.5	1	0	high	0	1.0000000		
60	60	30	Male	67.5	2	1	high	1	0.5769231		
61	61	22	Male	58.5	1	2	high	0	0.0000000		
62	62	160	Female	82	1	0	normal	0	1.0000000		
64	64	32	Male	67	2	0	normal	1	1.0000000		
65	65	31	male	66	1	2	high	1	0.0000000		
66	66	35	Male	72	2	0	normal	0	1.0000000		
67	67	28	Female	62.5	3	0	normal	0	1.0000000		
68	68	29	Male	64.5	2	0	normal	1	0.0000000		
69	69	25	Female	62	1	0	low	0	1.0000000		
70	70	27	Male	61	2	2	low	0	0.0000000		
72	72	29	male	65	1	2	normal	1	1.0000000		
73	73	165	Male	64	2	0	normal	0	0.0000000		

R Code:

Dataset_mid$Gender

```
> Dataset_mid$Gender
 [1] "Female" "Male"   "Male"   "Male"   "Female" "Female" "Male"   "Female" "Male"   "Male"   "female" "Male"   "female"
[14] "Male"   "Male"   "Male"   "Male"   "Male"   "male"   "Male"   "Male"   "Male"   "Female" "male"   "Male"   "Male"
[27] "Male"   "Male"   "Male"   "Male"   "Male"   "Male"   "Male"   "Male"   "Male"   "Male"   "Female" "Male"   "Male"
[40] "Male"   "male"   "Male"   "Male"   "Male"   "Male"   "Male"   "Male"   "Male"   "Male"   "Male"   "Male"   "Female"
[53] "Male"   "male"   "Male"   "Female" "Male"   "Female" "Male"   "male"   "Male"   "Male"   "Male"   "Male"   "Male"
[66] "Male"   "Male"   "Male"
```

R Code:

```
Dataset_mid$Gender<-edit(Dataset_mid$Gender)
Dataset_mid
```

```
Dataset_mid$Gender<-edit(Dataset_mid$Gender)
Dataset_mid
   id Age Gender weight.kg. Delivery_number Delivery_time  Blood Heart Caesarian
1   1  22 Female       57.7               1             0   high     0 0.0000000
2   2  26   Male         63               2             0 normal     0 1.0000000
3   3  26   Male         62               2             1 normal     0 0.0000000
4   4  28   Male         65               1             0   high     0 0.0000000
5   5  22 Female         58               2             0 normal     0 1.0000000
7   7  27 Female         64               2             0 normal     0 0.0000000
8   8  32   Male         70               3             0 normal     0 1.0000000
9   9  28 Female       63.5               2             0 normal     0 0.0000000
1  11  36   Male         75               1             0 normal     0 0.0000000
2  12  33   Male         70               1             1    low     0 1.0000000
3  13  23 Female         58               1             1 normal     0 0.0000000
4  14  20   Male         55               1             0 normal     1 0.0000000
6  16  25 Female       61.5               1             2    low     0 0.0000000
7  17  25   Male       61.5               1             0 normal     0 0.0000000
8  18  20   Male       55.5               1             2   high     0 1.0000000
9  19  37   Male         76               3             0 normal     1 1.0000000
0  20  24   Male       56.6               1             2    low     1 1.0000000
```

R Code:

Dataset_mid$weight.kg.

```
> Dataset_mid$weight.kg.
 [1] "57.7" "63"   "62"   "65"   "58"   "64"   "70"   "63.5" "75"   "70"   "58"   "55"   "61.5" "61.5" "55.5" "76"   "56.6" "62"
[19] "75X"  "62"   "55"   "49"   "68"   "73"   "58"   "82"   "68"   "63"   "59"   "75"   "69"   "63"   "59"   "63"   "73"   "57"
[37] "62.5" "82"   "67.5" "62.5" "82"   "68.5" "53"   "68"   "74"   "59"   "67.5" "110"  "61.5" "58.5" "82"   "67"   "66"   "72"
[55] "62.5" "64.5" "62"   "61"   "65"   "64"   "69"   "75"   "62.5" "63"   "58"   "57"
```

R Code:

```
invalid_indices<-grep("75X",Dataset_mid$weight.kg.)
Dataset_mid$weight.kg.[invalid_indices]<-75
Dataset_mid
```

```
Dataset_mid$weight.kg.[invalid_indices] <- 75
Dataset_mid
 id Age  Gender weight.kg. Delivery_number Delivery_time  Blood Heart Caesarian
  1  22  Female       57.7               1             0   high     0 0.0000000
  2  26    Male         63               2             0 normal     0 1.0000000
  3  26    Male         62               2             1 normal     0 0.0000000
  4  28    Male         65               1             0   high     0 0.0000000
  5  22  Female         58               2             0 normal     0 1.0000000
  7  27  Female         64               2             0 normal     0 0.0000000
  8  32    Male         70               3             0 normal     0 1.0000000
  9  28  Female       63.5               2             0 normal     0 0.0000000
 11  36    Male         75               1             0 normal     0 0.0000000
 12  33   Mmale         70               1             1    low     0 1.0000000
 13  23  female         58               1             1 normal     0 0.0000000
 14  20    Male         55               1             0 normal     1 0.0000000
 16  25  female       61.5               1             2    low     0 0.0000000
 17  25    Male       61.5               1             0 normal     0 0.0000000
 18  20    Male       55.5               1             2   high     0 1.0000000
 19  37    Male         76               3             0 normal     1 1.0000000
 20  24    Male       56.6               1             2    low     1 1.0000000
 21  26    Male         62               1             1 normal     0 0.0000000
 22  33    male         75               2             0    low     1 1.0000000
 23  25    Male         62               1             1   high     0 0.0000000
 25  20    Male         55               1             0   high     1 1.0000000
```

4-Dealing with Data types and Conversion:

As we can see that in Casarian columns contain decimal place data. So, to overcome it. We will use the below code to round it up.

```
64 64  32   Male         67               2             0 normal     1 1.0000000
65 65  31   Male         66               1             2   high     1 0.0000000
66 66  35   Male         72               2             0 normal     0 1.0000000
67 67  28 Female       62.5               3             0 normal     0 1.0000000
68 68  29   Male       64.5               2             0 normal     1 0.0000000
69 69  25 Female         62               1             0    low     0 1.0000000
70 70  27   Male         61               2             2    low     0 0.0000000
72 72  29   Male         65               1             2 normal     1 1.0000000
73 73 165   Male         64               2             0 normal     0 0.0000000
74 74  32   Male         69               3             0 normal     1 0.0000000
75 75  38   Male         75               3             2   high     1 1.0000000
76 76  27   Male       62.5               2             1 normal     0 0.0000000
77 77  33   Male         66               4             0 normal     0 0.5769231
78 78 150   Male         63               2             1   high     0 1.0000000
79 79  25   Male         58               1             2    low     0 1.0000000
80 80  30   Male         57               2             1    low     1 1.0000000
```

R Code:

```
Dataset_mid$Caesarian<- as.numeric(format(round(Dataset_mid$Caesarian    ,0)))
print(Dataset_mid)
```

```
Dataset_mid$Caesarian<- as.numeric(format(round(Dataset_mid$Caesarian,0)))
print(Dataset_mid)
   id Age Gender weight.kg. Delivery_number Delivery_time  Blood Heart Caesarian
1   1  22 Female       57.7               1             0   high     0         0
2   2  26   Male       63.0               2             0 normal     0         1
3   3  26   Male       62.0               2             1 normal     0         0
4   4  28   Male       65.0               1             0   high     0         0
5   5  22 Female       58.0               2             0 normal     0         1
7   7  27 Female       64.0               2             0 normal     0         0
8   8  32   Male       70.0               3             0 normal     0         1
9   9  28 Female       63.5               2             0 normal     0         0
11 11  36   Male       75.0               1             0 normal     0         0
12 12  33   Male       70.0               1             1    low     0         1
13 13  23 female       58.0               1             1 normal     0         0
14 14  20   Male       55.0               1             0 normal     1         0
16 16  25 female       61.5               1             2    low     0         0
17 17  25   Male       61.5               1             0 normal     0         0
18 18  20   Male       55.5               1             2   high     0         1
19 19  37   Male       76.0               3             0 normal     1         1
20 20  24   Male       56.6               1             2    low     1         1
```

Then we need to check all the Data type of all the attributes using str() function

```
str(Dataset_mid)
data.frame':   68 obs. of  9 variables:
$ id             : int  1 2 3 4 5 7 8 9 11 12 ...
$ Age            : int  22 26 26 28 22 27 32 28 36 33 ...
$ Gender         : chr  "Female" "Male" "Male" "Male" ...
$ weight.kg.     : chr  "57.7" "63" "62" "65" ...
$ Delivery_number: int  1 2 2 1 2 2 3 2 1 1 ...
$ Delivery_time  : int  0 0 1 0 0 0 0 0 0 1 ...
$ Blood          : chr  "high" "normal" "normal" "high" ...
$ Heart          : int  0 0 0 0 0 0 0 0 0 0 ...
$ Caesarian      : num  0 1 0 0 1 0 1 0 0 1 ...
- attr(*, "na.action")= 'omit' Named int [1:12] 6 10 15 24 27 30 36 42 45 55 ..
 ..- attr(*, "names")= chr [1:12] "6" "10" "15" "24" ...
```

So we can see Weight.kg. Attribute the data type should be numeric lets convert it

R Code

Dataset_mid

Dataset_mid$weight.kg. <- as.numeric(format(Dataset_mid$weight.kg.,0))

```
Dataset_mid$weight.kg. <- as.numeric(format(Dataset_mid$weight.kg.,0))
Dataset_mid
   id Age Gender weight.kg. Delivery_number Delivery_time  Blood Heart Caesarian
1   1  22 Female       57.7               1             0   high     0         0
2   2  26   Male       63.0               2             0 normal     0         1
3   3  26   Male       62.0               2             1 normal     0         0
4   4  28   Male       65.0               1             0   high     0         0
5   5  22 Female       58.0               2             0 normal     0         1
7   7  27 Female       64.0               2             0 normal     0         0
8   8  32   Male       70.0               3             0 normal     0         1
9   9  28 Female       63.5               2             0 normal     0         0
11 11  36   Male       75.0               1             0 normal     0         0
12 12  33   Male       70.0               1             1    low     0         1
13 13  23 Female       58.0               1             1 normal     0         0
14 14  20   Male       55.0               1             0 normal     1         0
16 16  25 Female       61.5               1             2    low     0         0
17 17  25   Male       61.5               1             0 normal     0         0
18 18  20   Male       55.5               1             2   high     0         1
19 19  37   Male       76.0               3             0 normal     1         1
20 20  24   Male       56.6               1             2    low     1         1
21 21  26   Male       62.0               1             1 normal     0         0
22 22  33   Male       75.0               2             0    low     1         1
```

Then we can convert Attribute Heart data type int to chr and also change there labels (0,1)to(Positive,Negative)

R CODE:

Dataset_mid$Heart <- factor(Dataset_mid$Heart,levels = c(1,0),labels = c("Positive", "Negative"))

```
Dataset_mid$Heart <- factor(Dataset_mid$Heart,levels = c(1,0),labels = c("Positive"
Dataset_mid
  id Age Gender weight.kg. Delivery_number Delivery_time  Blood    Heart Caesarian
   1  22 Female       57.7               1             0   high Negative         0
   2  26   Male       63.0               2             0 normal Negative         1
   3  26   Male       62.0               2             1 normal Negative         0
   4  28   Male       65.0               1             0   high Negative         0
   5  22 Female       58.0               2             0 normal Negative         1
   7  27 Female       64.0               2             0 normal Negative         0
   8  32   Male       70.0               3             0 normal Negative         1
   9  28 Female       63.5               2             0 normal Negative         0
  11  36   Male       75.0               1             0 normal Negative         0
  12  33   Male       70.0               1             1    low Negative         1
  13  23 Female       58.0               1             1 normal Negative         0
  14  20   Male       55.0               1             0 normal Positive         0
  16  25 Female       61.5               1             2    low Negative         0
  17  25   Male       61.5               1             0 normal Negative         0
  18  20   Male       55.5               1             2   high Negative         1
  19  37   Male       76.0               3             0 normal Positive         1
  20  24   Male       56.6               1             2    low Positive         1
  21  26   Male       62.0               1             1 normal Negative         0
  22  33   Male       75.0               2             0    low Positive         1
  23  25   Male       62.0               1             1   high Negative         0
```

5.Finding Mean, Median, Variance and Standard Deviation.

To find out the exploration of the Age attribute, we have to use the below code written in R.

R Code:

mean(Dataset_mid$Age)

median(Dataset_mid$Age)

sd (Dataset_mid$Age)

```
> mean(Dataset_mid$Age)
[1] 35.01471
> median(Dataset_mid$Age)
[1] 28
> sd (Dataset_mid$Age)
[1] 28.30616
```

R Code:

mean(Dataset_mid$weight.kg.)

median(Dataset_mid$weight.kg.)

sd (Dataset_mid$weight.kg.)

```
> mean(Dataset_mid$weight.kg.)
[1] 65.67353
> median(Dataset_mid$weight.kg.)
[1] 63.25
> sd (Dataset_mid$weight.kg.)
[1] 9.061772
```

R Code:

mean(Dataset_mid$Delivery_number)

median(Dataset_mid$Delivery_number)

sd(Dataset_mid$Delivery_number)

```
> mean(Dataset_mid$Delivery_number)
[1] 1.661765
> median(Dataset_mid$Delivery_number)
[1] 1.5
> sd (Dataset_mid$Delivery_number)
[1] 0.7651026
```

R Code:

mean(Dataset_mid$Delivery_time)

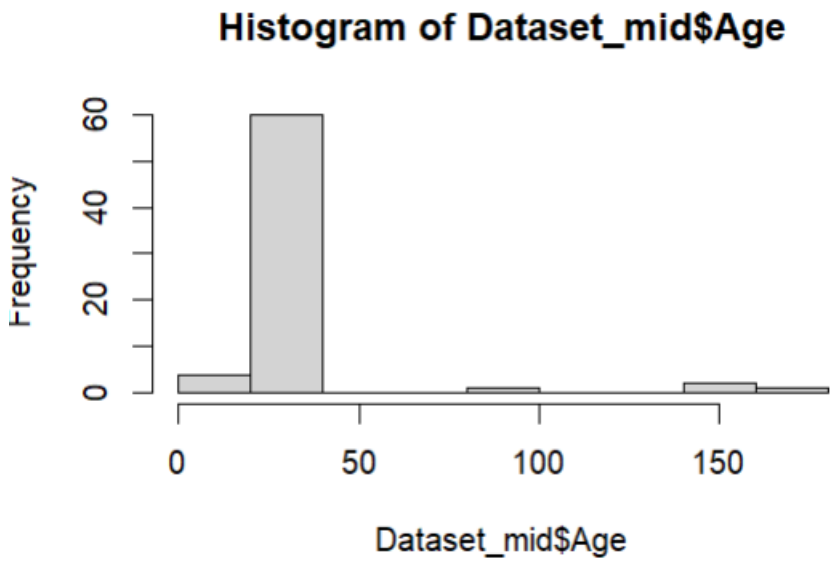
median(Dataset_mid$Delivery_time)

sd(Dataset_mid$Delivery_time)

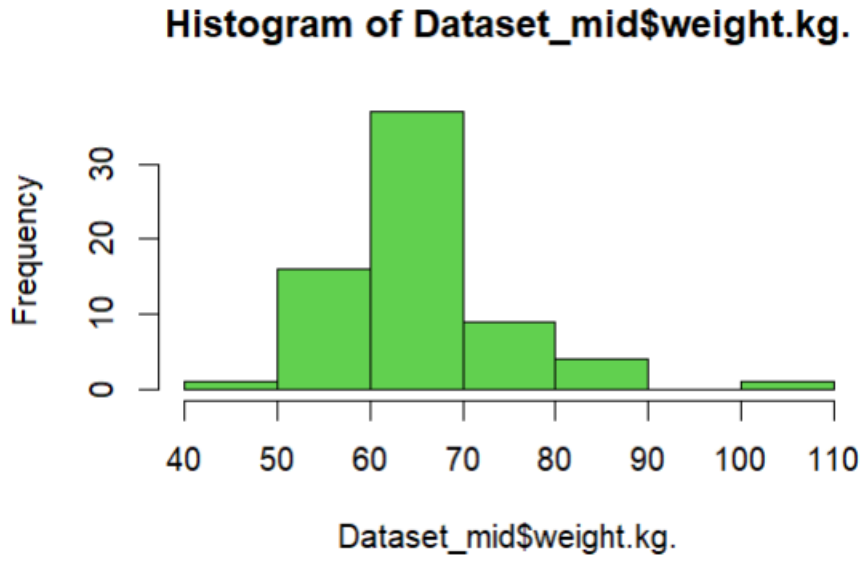
```
> mean(Dataset_mid$Delivery_time)
[1] 0.6470588
> median(Dataset_mid$Delivery_time)
[1] 0
> sd(Dataset_mid$Delivery_time)
[1] 0.8243443
```

6.Now, we draw a histogram for Age, Weight.kg.,Delivery_number ,Delivery_time and Caearian attributes for analysis.

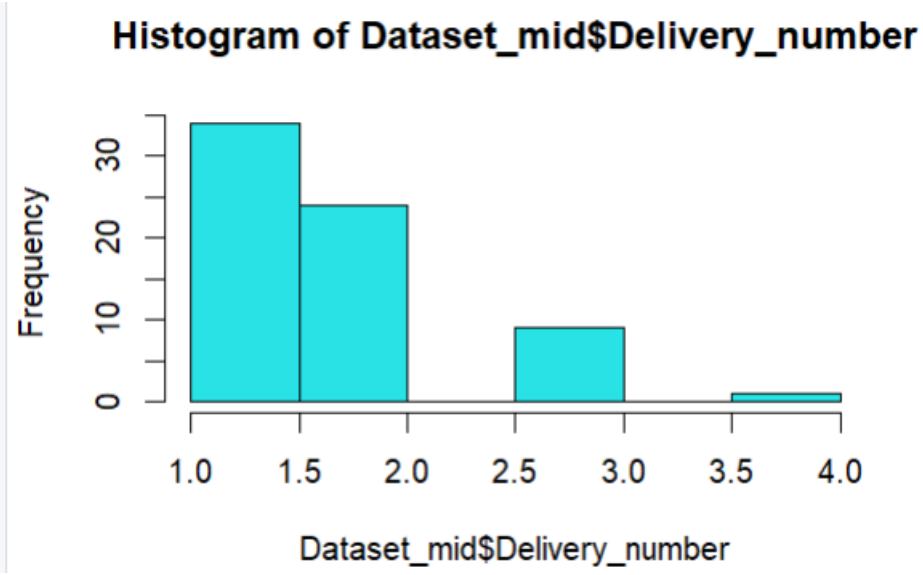
hist(Dataset_mid$Age)
hist(Dataset_mid$weight.kg.,col=3)
hist(Dataset_mid$Delivery_number ,col=5)
hist(Dataset_mid$Delivery_time ,col=7)
hist(Dataset_mid$Caesarian,col=4)



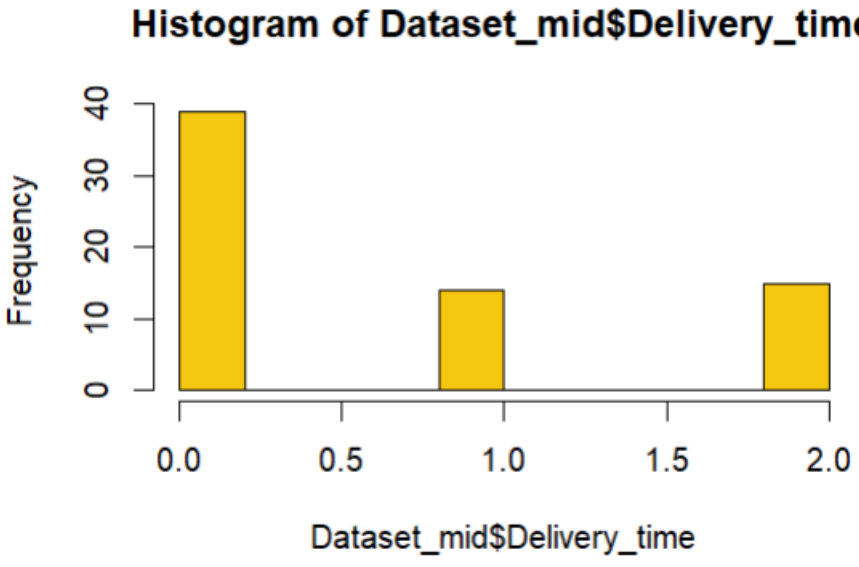
From the above histogram, we can see that 60% People Age is between 20 to 40. Other people Age is likes up to 18 people are adult and the rest are the left-over people. From the rest there are also outliers.



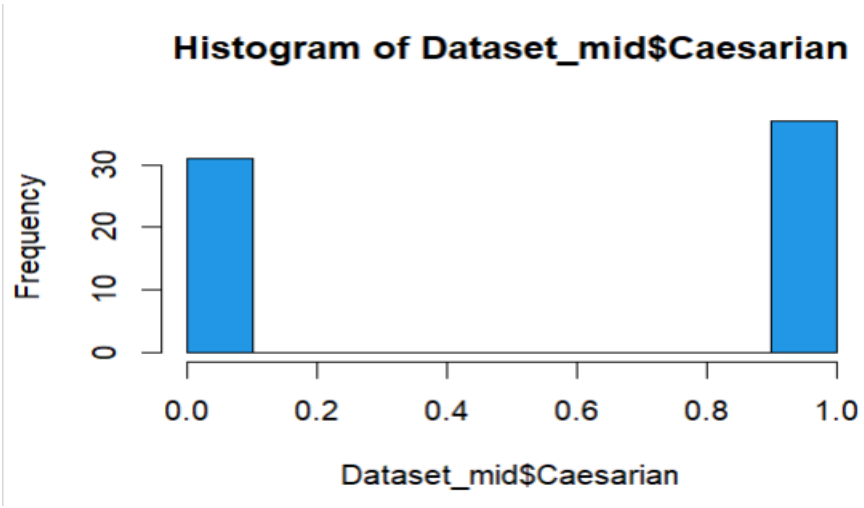
The following histogram shows the weight obtained by the people. The maximum number of people is between 60 and 70. Then nearly 15% people weigh from 50 to 60. There is a presence of outliers in the histogram.



From the above histogram, we can see that the highest number between 1.0 and 1.5, there are nearly 25% of people who deliver number from 1.5 to 2.0. And the lowest number is between 2.5 and 3.0. Finally, there is a presence of outliers in the histogram.



In the following histogram, we can see 40% of people delivering time between 0.0 and 0.2. Then a few have delivery times within 1.0. Moreover, there is a presence of outliers too.



In the above histogram, we can see that there are 30% of people having their caesarian is within nearly 0.1 and other people's caesarian is between 0.9 to 1.0.

6.Standard deviation of each attribute:

Here, we also downloaded “dplyr” and “matrixStats” package. To find out the standard deviation of each attribute.

R Code:

install.packages("dplyr")

install.packages("matrixStats")

library(matrixStats)

library(dplyr)

Dataset_mid %>% summarise_if(is.numeric, sd)

```
> Dataset_mid %>% summarise_if(is.numeric, sd)
        id      Age weight.kg. Delivery_number Delivery_time
1 23.68275 28.30616   9.061772       0.7651026     0.8243443
      Heart Caesarian
1 0.4857495 0.5017529
>
```

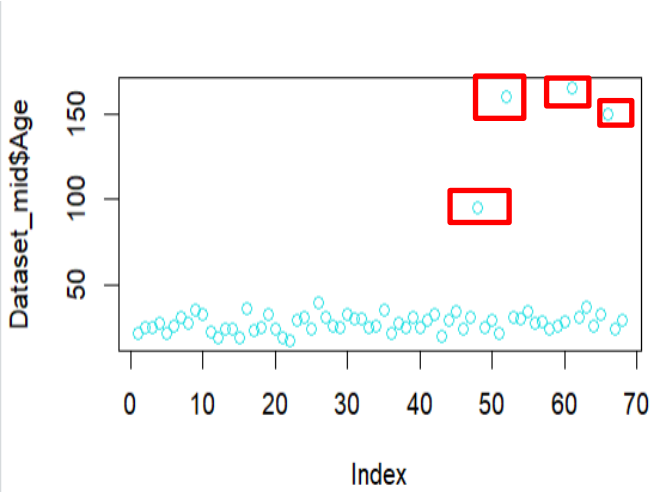
7.Dealing with Outliers:

Data, which are different from the rest of the dataset, known as OUTLIERS. To check the outliers, we have applied the below code:

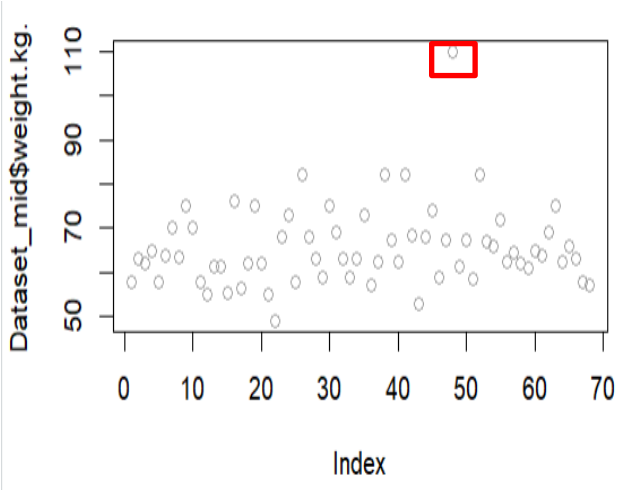
R Code:

```
plot(Dataset_mid$Age,col=5)
plot(Dataset_mid$weight.kg.,col=8)
plot(Dataset_mid$Delivery_number,
col=7)
```

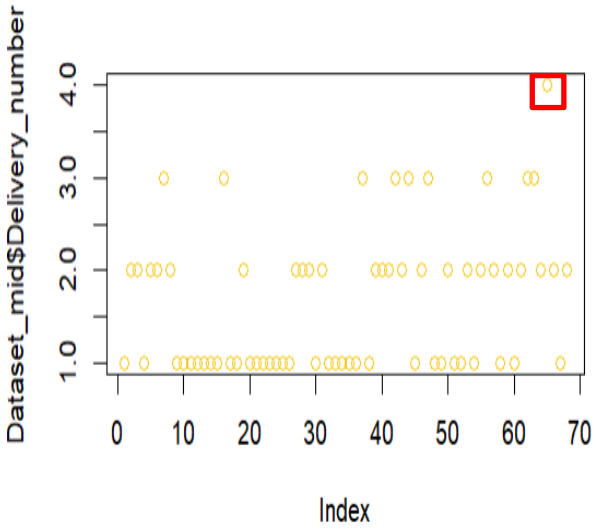
For Age



For Weight



For Delivery_number



8.Removing Outliers:

we find that the presence of outliers the value of mean mode, variance and standard deviation is bigger. So that we must remove these outliers.

Remove outliers from age attribute:

R Code:

```
Q1 <- quantile(Dataset_mid$Age, 0.25)
Q3 <- quantile(Dataset_mid$Age, 0.75)
IQR <- Q3 - Q1

lower_bound <- Q1 - 1.5 * IQR
upper_bound <- Q3 + 1.5 * IQR

Dataset_mid <- Dataset_mid[Dataset_mid$Age >= lower_bound &
Dataset_mid$Age <= upper_bound,]
plot(Dataset_mid$Age,col=5)
```

plot(Dataset$age,col=4)

Remove outliers from weight attribute:

R Code:

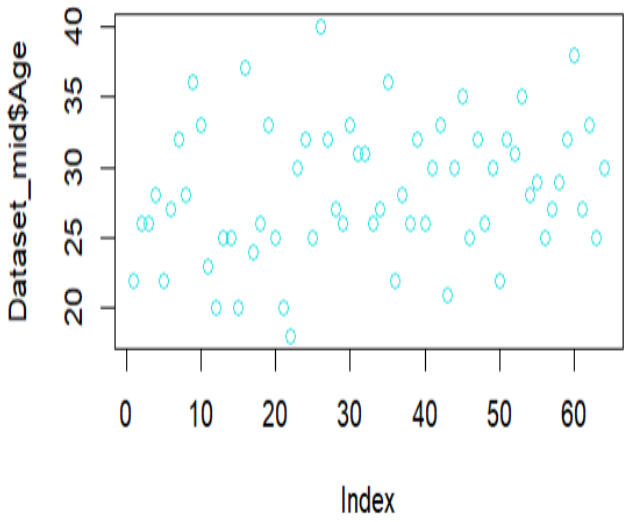
```
Q1 <- quantile(Dataset_mid$weight.kg., 0.25)
Q3 <- quantile(Dataset_mid$weight.kg., 0.75)
IQR <- Q3 - Q1

lower_bound <- Q1 - 1.5 * IQR
upper_bound <- Q3 + 1.5 * IQR

Dataset_mid <- Dataset_mid[Dataset_mid$weight.kg.>= lower_bound &
Dataset_mid$weight.kg. <= upper_bound,]
plot(Dataset_mid$weight.kg.,col=8)
```

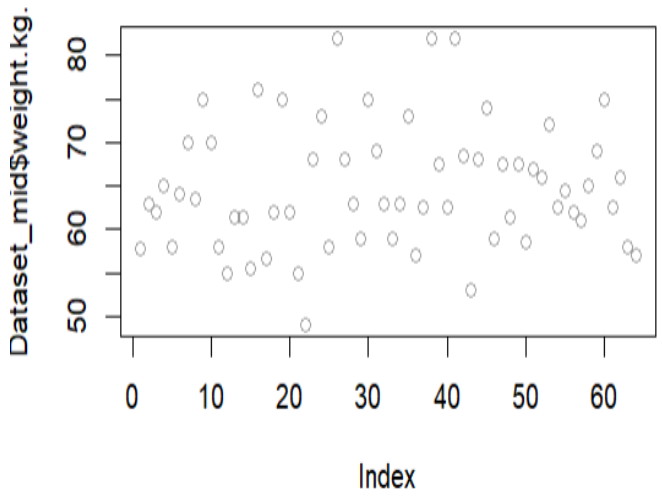
Remove outliers from Delivery_number attribute:

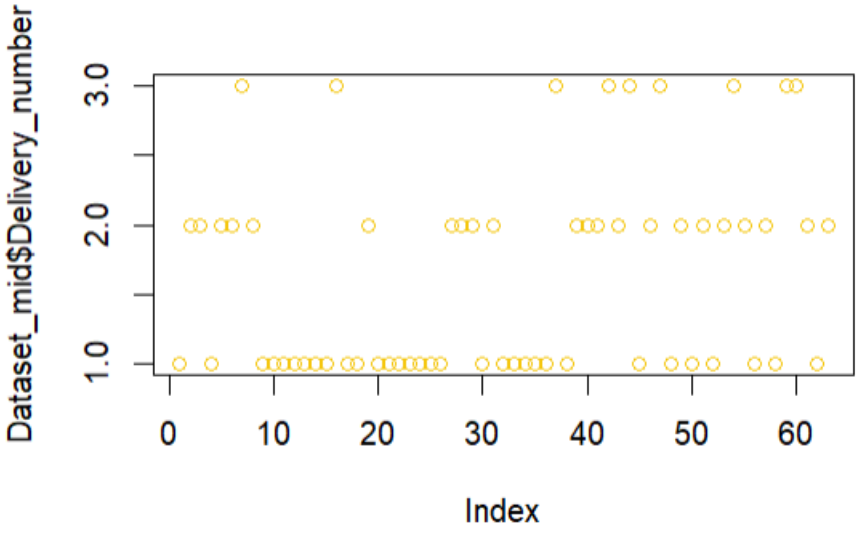
R Code:

```
Q1 <- quantile(Dataset_mid$Delivery_number, 0.25)

Q3 <- quantile(Dataset_mid$Delivery_number, 0.75)

IQR <- Q3 - Q1


lower_bound <- Q1 - 1.5 * IQR

upper_bound <- Q3 + 1.5 * IQR


Dataset_mid <- Dataset_mid[Dataset_mid$Delivery_number>= lower_bound
& Dataset_mid$Delivery_number <= upper_bound,]

plot(Dataset_mid$Delivery_number ,col=7)
```

Outputs:


Dataset_mid$Age
20
25
30
35
40
0
10
20
30
40
50
60
Index


Dataset_mid$weight.kg.
50
60
70
80
0
10
20
30
40
50
60
Index


Dataset_mid$Delivery_number
1.0
2.0
3.0
0
10
20
30
40
50
60
Index

Discussion & Conclusion:

The given dataset was very messy and there was many missing values and invalid values and also have outliers in many attributes. Moreover, there was a combination of categorical and numerical value. The dataset was like this-

Filter

	id	Age	Gender	weight.kg.	Delivery_number	Delivery_time	Blood	Heart	Caesarian
1	1	22	Female	57.7	1	0	high	0	0
2	2	26	Male	63	2	0	normal	0	1
3	3	26	Male	62	2	1	normal	0	0
4	4	28	Male	65	1	0	high	0	0
5	5	22	Female	58	2	0	normal	0	1
6	6	26	Male	63	NA	1	low	0	0
7	7	27	Female	64	2	0	normal	0	0
8	8	32	Male	70	3	0	normal	0	1
9	9	28	Female	63.5	2	0		0	0
10	10	NA	Male	64.5	1	1	normal	0	1
11	11	36	Male	75	1	0	normal	0	0
12	12	33		70	1	1	low	0	1
13	13	23	female	58	1	1	normal	0	0
14	14	20	Male	55	1	0	normal	1	0
15	15	29	Male	65	1	NA		1	1
16	16	25	female	61.5	1	2	low	0	0
17	17	25	Male	61.5	1	0	normal	0	0
18	18	20	Male	55.5	1	2	high	0	1
19	19	37	Male	76	3	0	normal	1	1
20	20	24	Male	56.6	1	2	low	1	1
21	21	26	Male	62	1	1	normal	0	0
22	22	33	male	75X	2	0	low	1	1
23	23	25	Male	62	1	1	high	0	0
24	24	27	Male	65	2	NA	low	1	1
25	25	20	Male	55	1	0	high	1	1
26	26	18	Male	49	1	0	normal	0	0
27	27	18	Male	50	1	NA	high	1	1
28	28	30	Female	68	1	0	normal	0	0
29	29	32	male	73	1	0	high	1	1

After Applying data preparation steps for the given data set., we got the dataset looks like this-

Dataset_mid

id	Age	Gender	weight.kg.	Delivery_number	Delivery_time	Blood	Heart	Caesarian
1	22	Female	57.7	1	0	high	Negative	0
2	26	Male	63.0	2	0	normal	Negative	1
3	26	Male	62.0	2	1	normal	Negative	0
4	28	Male	65.0	1	0	high	Negative	0
5	22	Female	58.0	2	0	normal	Negative	1
7	27	Female	64.0	2	0	normal	Negative	0
8	32	Male	70.0	3	0	normal	Negative	1
9	28	Female	63.5	2	0	normal	Negative	0
11	36	Male	75.0	1	0	normal	Negative	0
12	33	Male	70.0	1	1	low	Negative	1
13	23	Female	58.0	1	1	normal	Negative	0
14	20	Male	55.0	1	0	normal	Positive	0
16	25	Female	61.5	1	2	low	Negative	0
17	25	Male	61.5	1	0	normal	Negative	0
18	20	Male	55.5	1	2	high	Negative	1
19	37	Male	76.0	3	0	normal	Positive	1
20	24	Male	56.6	1	2	low	Positive	1
21	26	Male	62.0	1	1	normal	Negative	0
22	33	Male	75.0	2	0	low	Positive	1
23	25	Male	62.0	1	1	high	Negative	0
25	20	Male	55.0	1	0	high	Positive	1
26	18	Male	49.0	1	0	normal	Negative	0
28	30	Female	68.0	1	0	normal	Negative	0
29	32	Male	73.0	1	0	high	Positive	1
31	25	Male	58.0	1	0	low	Negative	0
32	40	Male	82.0	1	0	normal	Positive	1
33	32	Male	68.0	2	0	high	Positive	1
34	27	Male	63.0	2	0	normal	Positive	1
35	26	Male	59.0	2	2	normal	Negative	1
37	33	Male	75.0	1	1	normal	Negative	0
38	31	Male	69.0	2	2	normal	Negative	0
39	31	Male	63.0	1	0	normal	Negative	0
40	26	Male	59.0	1	2	low	Positive	1
41	27	Male	63.0	1	0	high	Positive	1
43	36	Male	73.0	1	1	high	Negative	1
44	22	Male	57.0	1	0	normal	Negative	1