



## **American International University-Bangladesh (AIUB)**

**Course: INTRODUCTION TO DATA SCIENCE [D]**

**Faculty Name: TOHEDUL ISLAM**

**Student Name: Shanto Kumar Basak**

**ID: 20-42945-1**

**Submitted Date: 13th March 2023**

## 1) Print the Data Set

```
dset_1 <- read.csv("F:/Dataset_midterm.csv",header=TRUE,sep=",")  
dset_1
```

The screenshot shows the RStudio interface. The script editor contains the following code:

```
1 dset_1 <- read.csv("F:/Dataset_midterm.csv",header=TRUE,sep=",")  
2 dset_1  
3 names(dset_1)  
4 summary(dset_1)  
5 sum(is.na(dset_1))  
6  
7 dset_1$blood <- factor(dset_1$blood,  
8 levels = c("high", "normal", "low"),  
9 labels = c(1,2,3))  
10 dset_1  
11  
12 colSums(is.na(dset_1))  
13  
14 which(is.na(dset_1$age))  
15 which(is.na(dset_1$weight.kg))  
16 which(is.na(dset_1$delivery_number))  
17 which(is.na(dset_1$delivery_time))  
18 which(is.na(dset_1$blood))  
19 which(is.na(dset_1$caesarian))  
20  
21 dset_1$missing_value = is.na(dset_1$age) | is.na(dset_1$weight.kg) | is.na(dset_1$delivery_number) | is.na(dset_1$delivery_time) | is.na(dset_1$blood) | is.na(dset_1$caesarian)  
22 dset_1  
23  
24 data = filter(dset_1, dset_1$missing_value != FALSE)  
25  
26 plot(dset_1$age, dset_1$missing_value)  
27 plot(dset_1$weight.kg, dset_1$missing_value)  
28 plot(dset_1$delivery_number, dset_1$missing_value)  
29 plot(dset_1$delivery_time, dset_1$missing_value)  
30 plot(dset_1$blood, dset_1$missing_value)  
31 plot(dset_1$caesarian, dset_1$missing_value)  
32  
33
```

The console output shows the first 33 rows of the dataset:

60	60	30	67.5	2	1	high	1	NA
61	61	22	58.5	1	2	high	0	0
62	62	NA	NA	1	0	normal	0	1
63	63	32	67.0	2	0	low	0	1
64	64	32	67.0	2	0	normal	1	1
65	65	31	66.0	1	2	high	1	0
66	66	35	72.0	2	0	normal	0	1
67	67	28	62.5	3	0	normal	0	1
68	68	29	64.5	2	0	normal	1	0
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	2	low	0	1
72	72	29	65.0	1	2	low	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

## 2) Identify the Characteristics From The Data Collection.

```
names(dset_1)
```

The screenshot shows the RStudio interface. The script editor contains the same code as the previous screenshot. The console output shows the names of the variables in the dataset:

```
> names(dset_1)  
[1] "id" "Age" "weight.kg." "delivery_number" "delivery_time" "blood" "Heart" "Caesarian"
```

### 3) An overview of each attribute in the data set

summary(dset\_1)

The screenshot shows the RStudio interface. The console displays the following summary output for the dataset 'dset\_1':

```
> summary(dset_1)
      "id"      "Age"      "weight.kg."      "delivery.number"      "delivery.time"      "blood"      "Heart"      "Caesarian"
Min.   : 1.00  Min.   :18.00  Min.   : 49.00  Min.   :1.0000  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.:1.0000  1st Qu.:0.0000  Class :character  1st Qu.:0.0000  1st Qu.:0.0000
Median :40.50  Median :28.00  Median : 61.50  Median :1.5000  Median :0.0000  Mode  :character  Median :0.0000  Median :1.0000
Mean   :40.50  Mean   :29.68  Mean   : 65.13  Mean   :1.6799  Mean   :0.6234  Mean   :0.375  Mean   :0.5641
3rd Qu.:60.25  3rd Qu.:32.00  3rd Qu.: 68.00  3rd Qu.:12.0000  3rd Qu.:1.0000  3rd Qu.:1.0000  3rd Qu.:1.0000  3rd Qu.:1.0000
Max.   :80.00  Max.   :55.00  Max.   :110.00  Max.   :14.0000  Max.   :12.0000  Max.   :1.0000  Max.   :1.0000  Max.   :1.0000
NA's   : 3      NA's   :3      NA's   :12      NA's   :3
```

The Environment pane on the right shows the dataset 'dset\_1' with 80 observations and 8 variables.

### 4) From the Blood attribute, annotate high as 1, normal as 2, and low as 3.

dset\_1\$Blood <- factor(dset\_1\$Blood,

levels = c("high", "normal", "low"),

labels = c(1,2,3))

dset\_1

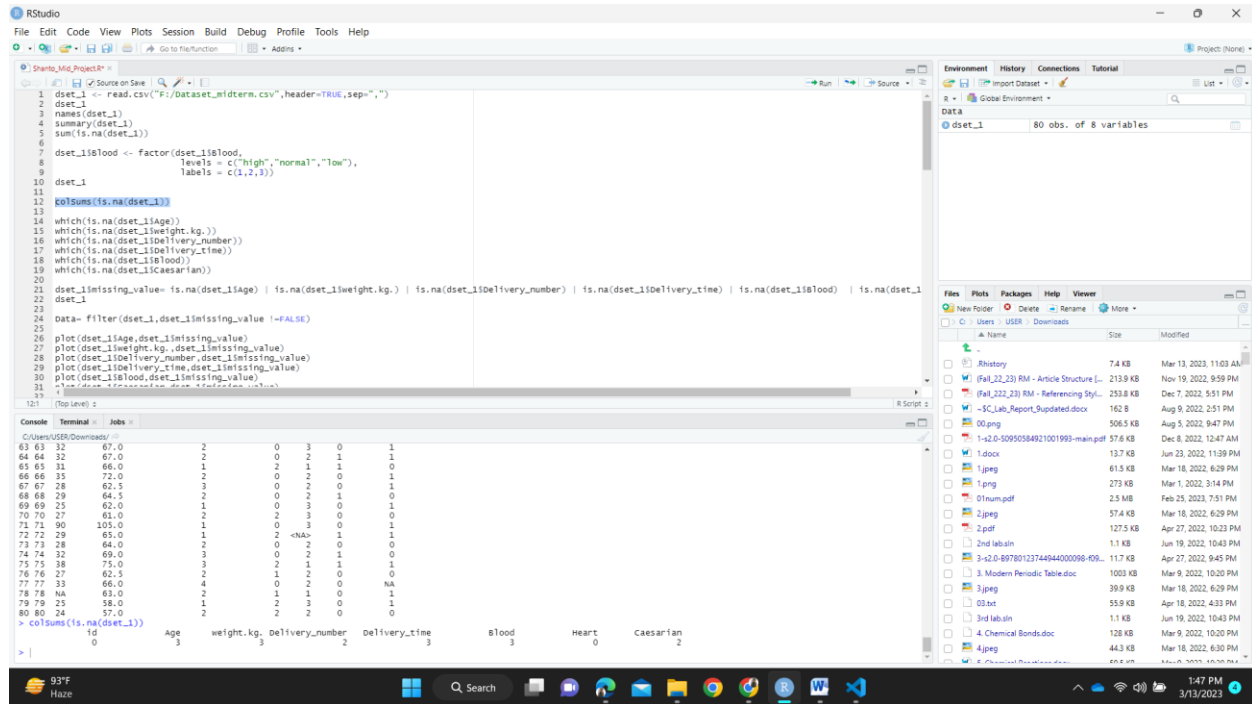
The screenshot shows the RStudio interface after the 'Blood' attribute has been annotated. The console displays the following summary output for the dataset 'dset\_1':

```
> summary(dset_1)
      "id"      "Age"      "weight.kg."      "delivery.number"      "delivery.time"      "blood"      "Heart"      "Caesarian"
Min.   : 1.00  Min.   :18.00  Min.   : 49.00  Min.   :1.0000  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.:1.0000  1st Qu.:0.0000  Class :character  1st Qu.:0.0000  1st Qu.:0.0000
Median :40.50  Median :28.00  Median : 61.50  Median :1.5000  Median :0.0000  Mode  :character  Median :0.0000  Median :1.0000
Mean   :40.50  Mean   :29.68  Mean   : 65.13  Mean   :1.6799  Mean   :0.6234  Mean   :0.375  Mean   :0.5641
3rd Qu.:60.25  3rd Qu.:32.00  3rd Qu.: 68.00  3rd Qu.:12.0000  3rd Qu.:1.0000  3rd Qu.:1.0000  3rd Qu.:1.0000  3rd Qu.:1.0000
Max.   :80.00  Max.   :55.00  Max.   :110.00  Max.   :14.0000  Max.   :12.0000  Max.   :1.0000  Max.   :1.0000  Max.   :1.0000
NA's   : 3      NA's   :3      NA's   :12      NA's   :3
```

The Environment pane on the right shows the dataset 'dset\_1' with 80 observations and 8 variables.

5) For each attribute, locate the missing value.

```
colSums(is.na(dset_1))
```



6) Determine the exact row number where an attribute has no values.

```
which(is.na(dset_1$Age))
```

```
which(is.na(dset_1$weight.kg))
```

```
which(is.na(dset_1$Delivery_num
```

```
ber))
```

```
which(is.na(dset_1$Delivery_time
```

```
)) which(is.na(dset_1$Blood))
```

```
which(is.na(dset_1$Caesarian))
```

The RStudio interface shows the following code in the script editor:

```

1 dset_1 <- read.csv("F:/Dataset_midterm.csv", header=TRUE, sep=",")
2 dset_1
3 names(dset_1)
4 summary(dset_1)
5 sum(is.na(dset_1))
6
7 dset_1$blood <- factor(dset_1$blood,
8                       levels = c("high", "normal", "low"),
9                       labels = c(1,2,3))
10 dset_1
11
12 colSums(is.na(dset_1))
13
14 which(is.na(dset_1$age))
15 which(is.na(dset_1$weight.kg))
16 which(is.na(dset_1$delivery_number))
17 which(is.na(dset_1$delivery_time))
18 which(is.na(dset_1$blood))
19 which(is.na(dset_1$caesarian))
20
21 dset_1$missing_value = is.na(dset_1$age) | is.na(dset_1$weight.kg) | is.na(dset_1$delivery_number) | is.na(dset_1$delivery_time) | is.na(dset_1$blood) | is.na(dset_1$caesarian)
22 dset_1
23
24 data = filter(dset_1, dset_1$missing_value != FALSE)
25
26 plot(dset_1$age, dset_1$missing_value)
27 plot(dset_1$weight.kg, dset_1$missing_value)
28 plot(dset_1$delivery_number, dset_1$missing_value)
29 plot(dset_1$delivery_time, dset_1$missing_value)
30 plot(dset_1$blood, dset_1$missing_value)
31 # You should also check for outliers
32
33 #>
34
35 #>

```

The console output shows the summary of the dataset:

```

75 75 38 75.0 3 2 1 1 1
76 76 27 62.5 2 1 2 0 0
77 77 33 66.0 4 0 2 0 NA
78 78 NA 63.0 2 1 1 0 1
79 79 25 58.0 1 2 3 0 1
80 80 24 57.0 2 2 2 0 0
> colSums(is.na(dset_1))
      id      Age      weight.kg      delivery_number      delivery_time      blood      Heart      Caesarian
      0         3         3         2         3         3         0         2

```

The Environment pane shows the dataset 'dset\_1' with 80 observations and 8 variables.

## 7) Find the outlier by looking for a missing value.

dset\_1\$missing\_value= is.na(dset\_1\$Age) | is.na(dset\_1\$weight.kg.) |  
is.na(dset\_1\$Delivery\_number) | is.na(dset\_1\$Delivery\_time) | is.na(dset\_1\$Blood) |  
is.na(dset\_1\$Caesarian)

dset\_1

The RStudio interface shows the following code in the script editor:

```

1 dset_1 <- read.csv("F:/Dataset_midterm.csv", header=TRUE, sep=",")
2 dset_1
3 names(dset_1)
4 summary(dset_1)
5 sum(is.na(dset_1))
6
7 dset_1$blood <- factor(dset_1$blood,
8                       levels = c("high", "normal", "low"),
9                       labels = c(1,2,3))
10 dset_1
11
12 colSums(is.na(dset_1))
13
14 which(is.na(dset_1$age))
15 which(is.na(dset_1$weight.kg))
16 which(is.na(dset_1$delivery_number))
17 which(is.na(dset_1$delivery_time))
18 which(is.na(dset_1$blood))
19 which(is.na(dset_1$caesarian))
20
21 dset_1$missing_value = is.na(dset_1$age) | is.na(dset_1$weight.kg.) | is.na(dset_1$Delivery_number) | is.na(dset_1$Delivery_time) | is.na(dset_1$Blood) | is.na(dset_1$Caesarian)
22 dset_1
23
24 data = filter(dset_1, dset_1$missing_value != FALSE)
25
26 install.packages("matrixStats")
27 library(matrixStats)
28 library("dplyr")
29 library("dplyr")
30
31 #>
32
33 #>

```

The console output shows the summary of the dataset after filtering:

```

62 62 NA NA 1 0 2 0 1 TRUE
63 63 32 67.0 2 0 2 0 1 FALSE
64 64 32 67.0 2 0 2 1 1 FALSE
65 65 31 66.0 1 2 1 1 0 FALSE
66 66 35 72.0 2 0 2 0 1 FALSE
67 67 28 62.5 2 0 2 0 1 FALSE
68 68 29 64.5 2 0 2 1 0 FALSE
69 69 25 62.0 1 0 3 0 1 FALSE
70 70 27 63.0 2 2 3 0 0 FALSE
71 71 90 105.0 1 0 3 0 1 FALSE
72 72 29 63.0 2 0 <NA> 1 TRUE
73 73 28 64.0 2 0 2 0 0 FALSE
74 74 32 69.0 3 0 2 1 0 FALSE
75 75 38 75.0 3 2 1 1 0 FALSE
76 76 27 62.5 2 1 2 0 0 FALSE
77 77 33 66.0 4 0 2 0 NA TRUE
78 78 NA 63.0 2 1 1 0 1 TRUE
79 79 25 58.0 1 2 3 0 1 FALSE
80 80 24 57.0 2 2 2 0 0 FALSE
> data = filter(dset_1, dset_1$missing_value != FALSE)
>

```

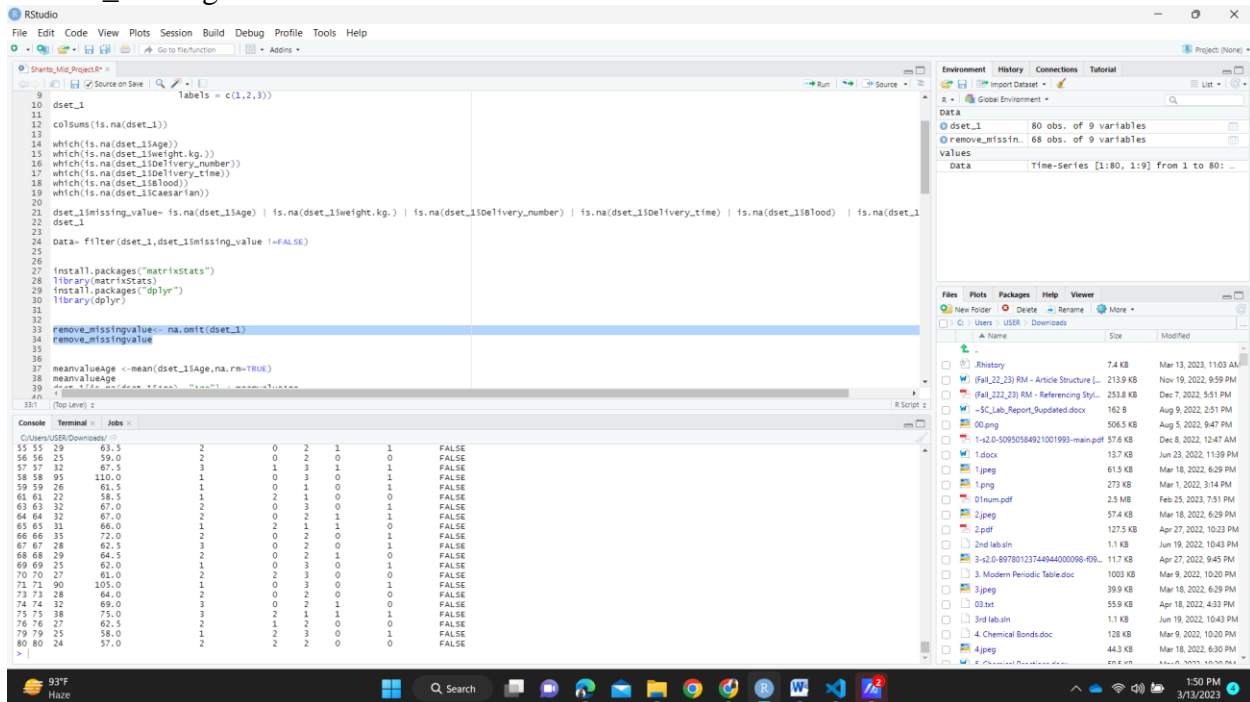
The Environment pane shows the dataset 'dset\_1' with 80 observations and 8 variables.

## 8) Use the following methods to recover missing values.

### I. Delete the rows with empty values from the data collection.

```
remove_missingvalue<- na.omit(dset_1)
```

```
remove_missingvalue
```



### II. Use the mean value to fill in any gaps in data.

```
meanvalueAge <-mean(dset_1$Age,na.rm=TRUE)
```

```
meanvalueAge
```

```
dset_1[is.na(dset_1$Age), "Age"] <-meanvalueAge
```

```
dset_1
```

```
meanvalueWeight <-mean(dset_1$weight.kg.,na.rm=TRUE)
```

```
meanvalueWeight
```

```
dset_1[is.na(dset_1$weight.kg.), "weight.kg."] <-meanvalueWeight
```

```
dset_1
```

```
meanvalueDelivery_number <-mean(dset_1$Delivery_number,na.rm=TRUE)
```

```
meanvalueDelivery_number
```

```
dset_1[is.na(dset_1$Delivery_number), "Delivery_number"] <-meanvalueDelivery_number
```

```
dset_1
```

```
meanvalueDelivery_time <-mean(dset_1$Delivery_time,na.rm=TRUE)
```

```
meanvalueDelivery_time
```

```
dset_1[is.na(dset_1$Delivery_time), "Delivery_time"] <-meanvalueDelivery_time
```

```
dset_1
```

```
meanvalueCaesarian <-mean(dset_1$Caesarian,na.rm=TRUE)
```

```
meanvalueCaesarian
```

```
dset_1[is.na(dset_1$Caesarian), "Caesarian"] <-meanvalueCaesarian
```

```
dset_1
```

The screenshot displays the RStudio interface. The script editor on the left contains R code for cleaning the 'dset\_1' dataset. The console on the bottom left shows the output of the code, which is a table with 9 columns and 80 rows of data. The right-hand side of the window shows the Environment pane with variables 'dset\_1' and 'remove\_missin', and the Files pane showing a list of files in the user's Downloads folder.

```
33 remove_missingvalue <- na.omit(dset_1)
34 remove_missingvalue
35
36
37 meanvalueAge <-mean(dset_1$Age,na.rm=TRUE)
38 meanvalueAge
39 dset_1[is.na(dset_1$Age), "Age"] <-meanvalueAge
40 dset_1
41
42 meanvalueWeight <-mean(dset_1$Weight.kg,na.rm=TRUE)
43 meanvalueWeight
44 dset_1[is.na(dset_1$Weight.kg), "Weight.kg"] <-meanvalueWeight
45 dset_1
46
47 meanvalueDelivery_number <-mean(dset_1$Delivery_number,na.rm=TRUE)
48 meanvalueDelivery_number
49 dset_1[is.na(dset_1$Delivery_number), "Delivery_number"] <-meanvalueDelivery_number
50 dset_1
51
52 meanvalueDelivery_time <-mean(dset_1$Delivery_time,na.rm=TRUE)
53 meanvalueDelivery_time
54 dset_1[is.na(dset_1$Delivery_time), "Delivery_time"] <-meanvalueDelivery_time
55 dset_1
56
57 meanvalueCaesarian <-mean(dset_1$Caesarian,na.rm=TRUE)
58 meanvalueCaesarian
59 dset_1[is.na(dset_1$Caesarian), "Caesarian"] <-meanvalueCaesarian
60 dset_1
61
62
63
64
371 [Top line] >
```

60	60	30.00000	67.50000	2.000000	1.0000000	1	1	0.5641026
61	61	22.00000	58.50000	1.000000	2.0000000	1	0	0.0000000
62	62	29.67532	65.12727	1.000000	0.0000000	2	0	1.0000000
63	63	32.00000	67.00000	2.000000	0.0000000	3	0	1.0000000
64	64	32.00000	67.00000	2.000000	0.0000000	2	1	1.0000000
65	65	31.00000	66.00000	1.000000	2.0000000	1	1	0.0000000
66	66	35.00000	72.00000	2.000000	0.0000000	2	0	1.0000000
67	67	28.00000	62.50000	3.000000	0.0000000	2	0	1.0000000
68	68	29.00000	64.50000	2.000000	0.0000000	2	1	0.0000000
69	69	25.00000	62.00000	1.000000	0.0000000	3	0	1.0000000
70	70	27.00000	63.00000	2.000000	2.0000000	3	0	0.0000000
71	71	90.00000	105.00000	1.000000	0.0000000	3	0	1.0000000
72	72	29.00000	65.00000	2.000000	2.0000000	2	1	1.0000000
73	73	28.00000	64.00000	2.000000	0.0000000	2	0	0.0000000
74	74	32.00000	69.00000	3.000000	0.0000000	2	1	0.0000000
75	75	38.00000	75.00000	3.000000	2.0000000	1	1	1.0000000
76	76	27.00000	62.50000	2.000000	1.0000000	2	0	0.0000000
77	77	33.00000	66.00000	4.000000	0.0000000	2	0	0.5641026
78	78	29.67532	63.00000	2.000000	1.0000000	1	0	1.0000000
79	79	25.00000	58.00000	1.000000	2.0000000	3	0	1.0000000
80	80	24.00000	57.00000	2.000000	2.0000000	2	0	0.0000000

### III. Use the mode value to recover missing values.

```
modeAge=names(sort(table(dset_1$Age)))[1]
```

```
modeAge
```

```
dset_1[is.na(dset_1$Age), "Age"] <-modeAge
```

```
dset_1
```

```
modeweight=names(sort(table(dset_1$weight.kg.)))[1]
```

```
modeweight
```

```
dset_1[is.na(dset_1$weight.kg.), "weight.kg."] <-modeweight
```

```
dset_1
```

```
modeDelivery_number=names(sort(table(dset_1$Delivery_number)))[1]
```

```
modeDelivery_number
```

```
dset_1[is.na(dset_1$Delivery_number), "Delivery_number"] <-modeDelivery_number
```

```
dset_1
```

```
modeDelivery_time=names(sort(table(dset_1$Delivery_time)))[1]
```

```
modeDelivery_time
```

```
dset_1[is.na(dset_1$Delivery_time), "Delivery_time"] <-modeDelivery_time
```

```
dset_1
```

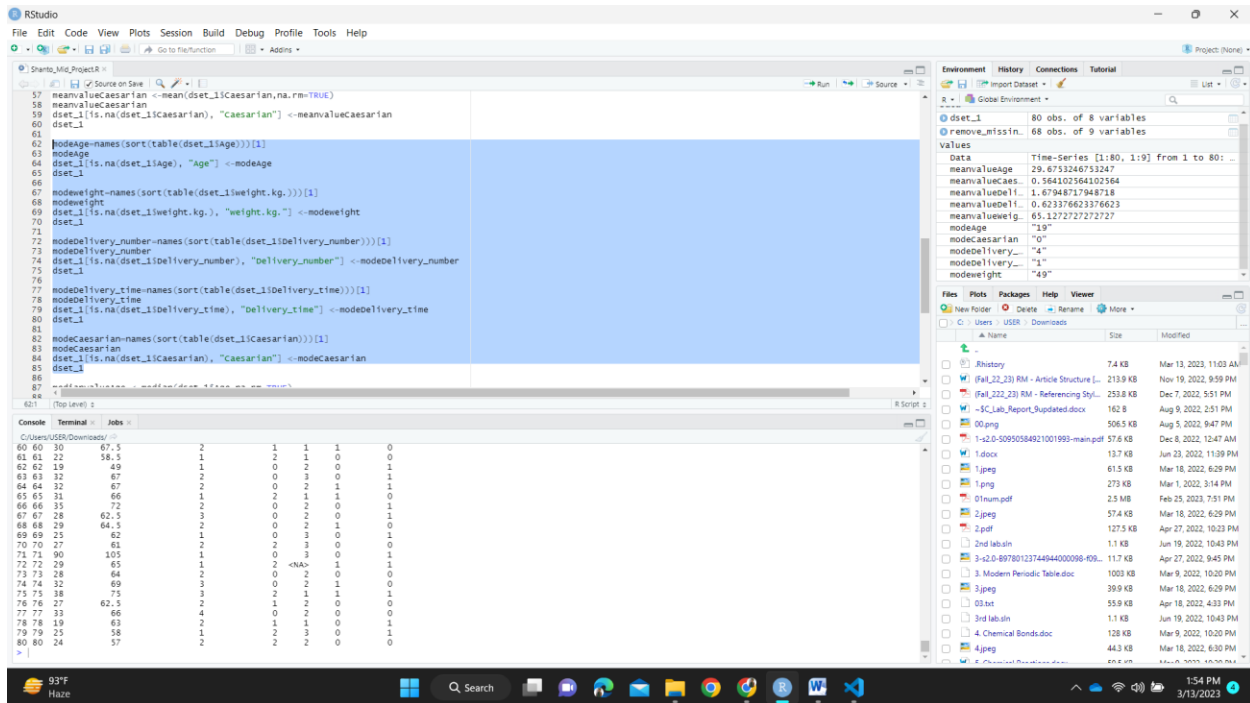
```
modeCaesarian=names(sort(table(dset_1$Caesarian)))[1]
```

```
modeCaesarian
```

```
dset_1[is.na(dset_1$Caesarian), "Caesarian"] <-modeCaesarian
```

```
dset_1
```





#### IV. Use the median value to fill in any missing values.

```
medianvalueAge <- median(dset_1$Age, na.rm=TRUE)
```

```
medianvalueAge
```

```
dset_1[is.na(dset_1$Age), "Age"] <- medianvalueAge
```

```
dset_1
```

```
medianvalueweight <- median(dset_1$weight.kg., na.rm=TRUE)
```

```
medianvalueweight
```

```
dset_1[is.na(dset_1$weight.kg.), "weight.kg."] <- medianvalueweight
```

```
dset_1
```

```
medianvalueDelivery_number <- median(dset_1$Delivery_number, na.rm=TRUE)
```

```
medianvalueDelivery_number
```

```
dset_1[is.na(dset_1$Delivery_number), "Delivery_number"] <- medianvalueDelivery_number
```

```
dset_1
```

```
medianvalueDelivery_time <- median(dset_1$Delivery_time, na.rm=TRUE)
```

```
medianvalueDelivery_time
```

```
dset_1[is.na(dset_1$Delivery_time), "Delivery_time"] <- medianvalueDelivery_time
```

```
dset_1
```

```
medianvalueCaesarian <- median(dset_1$Caesarian, na.rm=TRUE)
```

```
medianvalueCaesarian
```

```
dset_1[is.na(dset_1$Caesarian), "Caesarian"] <- medianvalueCaesarian
```

```
dset_1
```

The screenshot shows the RStudio interface with a script editor on the left containing R code for data cleaning, a console window at the bottom showing the output of the code, and a file explorer on the right.

**R Code in Script Editor:**

```
83 nodeCaesarian
84 dset_1[is.na(dset_1$Caesarian), "Caesarian"] <- medianvalueCaesarian
85 dset_1
86
87 medianvalueAge <- median(dset_1$Age, na.rm=TRUE)
88 medianvalueAge
89 dset_1[is.na(dset_1$Age), "Age"] <- medianvalueAge
90 dset_1
91
92 medianvalueWeight <- median(dset_1$Weight.kg, na.rm=TRUE)
93 medianvalueWeight
94 dset_1[is.na(dset_1$Weight.kg), "Weight.kg"] <- medianvalueWeight
95 dset_1
96
97 medianvalueDelivery_number <- median(dset_1$Delivery_number, na.rm=TRUE)
98 medianvalueDelivery_number
99 dset_1[is.na(dset_1$Delivery_number), "Delivery_number"] <- medianvalueDelivery_number
100 dset_1
101
102 medianvalueDelivery_time <- median(dset_1$Delivery_time, na.rm=TRUE)
103 medianvalueDelivery_time
104 dset_1[is.na(dset_1$Delivery_time), "Delivery_time"] <- medianvalueDelivery_time
105 dset_1
106
107 medianvalueCaesarian <- median(dset_1$Caesarian, na.rm=TRUE)
108 medianvalueCaesarian
109 dset_1[is.na(dset_1$Caesarian), "Caesarian"] <- medianvalueCaesarian
110 dset_1
111
112
```

**Console Output:**

```
60 60 30 67.5 2 1 1 1 0
61 61 22 58.5 1 2 1 0 0
62 62 19 49 1 0 2 0 1
63 63 32 67 2 0 3 0 1
64 64 32 67 2 0 2 1 1
65 65 31 66 1 2 1 1 0
66 66 35 72 2 0 2 0 1
67 67 28 62.5 3 0 2 0 1
68 68 29 64.5 2 0 2 1 0
69 69 25 62 1 0 3 0 1
70 70 27 61 2 2 3 0 1
71 71 30 105 1 0 3 0 1
72 72 29 65 1 2 <NA> 1 1
73 73 28 64 2 0 2 0 0
74 74 32 69 3 0 2 1 0
75 75 38 75 3 2 1 1 1
76 76 27 62.5 4 0 2 0 0
77 77 33 66 2 2 3 0 1
78 78 19 63 2 1 1 0 1
79 79 25 58 1 2 3 0 1
80 80 24 57 2 2 2 0 0
```

**Environment Window:**

Variable	Value
meanvalueAge	29.6753246753247
meanvalueCaes.	0.5641021564102156
meanvalueDel1	1.67948717948718
meanvalueDelg	0.623376623376623
meanvalueDelw	65.1272727272727
medianvalueAge	NA_real_
medianvalueCa.	NA_real_
medianvalueDel	NA_real_
medianvalueDe.	NA_real_
medianvalueDee	NA_real_
modeAge	"19"
modeCaesarian	"0"
modeDelivery_n	"4"
modeDelivery_t	"1"
modeWeight	"49"

**Files Window:**

Name	Size	Modified
1-hist	7.4 KB	Mar 13, 2023, 11:03 AM
1-hist_21_23	215.9 KB	Nov 19, 2022, 9:59 PM
1-hist_21_23	255.8 KB	Dec 7, 2022, 5:51 PM
1-hist_Report_Updated.docx	162.8 KB	Aug 9, 2022, 2:51 PM
1-hist	506.5 KB	Aug 5, 2022, 9:47 AM
1-hist_21_23	57.6 KB	Dec 8, 2022, 12:47 AM
1-hist	13.7 KB	Jun 23, 2022, 11:39 PM
1-hist	61.5 KB	Mar 18, 2022, 6:29 PM
1-hist	279 KB	Mar 1, 2022, 3:14 PM
1-hist	2.5 MB	Feb 25, 2023, 7:51 PM
1-hist	57.4 KB	Mar 18, 2022, 6:29 PM
1-hist	127.5 KB	Apr 27, 2022, 10:23 PM
1-hist	1.1 KB	Jun 19, 2022, 10:43 PM
1-hist	11.7 KB	Apr 27, 2022, 9:45 PM
1-hist	1003 KB	Mar 9, 2022, 10:20 PM
1-hist	39.9 KB	Mar 18, 2022, 6:29 PM
1-hist	55.9 KB	Apr 18, 2022, 4:33 PM
1-hist	1.1 KB	Jun 19, 2022, 10:43 PM
1-hist	128 KB	Mar 9, 2022, 10:20 PM
1-hist	44.3 KB	Mar 18, 2022, 6:30 PM
1-hist	60.8 KB	Mar 18, 2022, 6:30 PM

## 9) Range of all attribute

```
idRange=max(dset_1$id, na.rm=TRUE)-min(dset_1$id, na.rm=TRUE)
```

```
print(idRange)
```

```
AgeRange=max(dset_1$Age, na.rm=TRUE)-min(dset_1$Age, na.rm=TRUE)
```

```
print(AgeRange)
```

```
weightRange=max(dset_1$weight.kg., na.rm=TRUE)-min(dset_1$weight.kg., na.rm=TRUE)
```

```
print(weightRange)
```

```
Delivery_numberRange=max(dset_1$Delivery_number, na.rm=TRUE)-min(dset_1$Delivery_number,  
na.rm=TRUE)
```

```
print(Delivery_numberRange)
```

```
Delivery_timeRange=max(dset_1$Delivery_time, na.rm=TRUE)-min(dset_1$Delivery_time,  
na.rm=TRUE)
```

```
print(Delivery_timeRange)
```

```
CaesarianRange=max(dset_1$Caesarian, na.rm=TRUE)-min(dset_1$Caesarian, na.rm=TRUE)
```

```
print(CaesarianRange)
```

```
HeartRange=max(dset_1$Heart, na.rm=TRUE)-min(dset_1$Heart, na.rm=TRUE)
```

```
print(HeartRange)
```

The screenshot displays the RStudio environment with a script editor, console, and environment pane. The script editor contains R code for calculating the range of various attributes in a dataset named 'dset\_1'. The code uses the `max()` and `min()` functions with `na.rm=TRUE` to handle missing values. The attributes being calculated are `id`, `Age`, `weight.kg.`, `Delivery_number`, `Delivery_time`, `Caesarian`, and `Heart`. The results are printed to the console.

```
# RStudio Interface
```

**Script Editor:**

```
102 medianvalueDelivery_time <- median(dset_1$Delivery_time, na.rm=TRUE)
103 medianvalueDelivery_time
104 dset_1[is.na(dset_1$Delivery_time), "Delivery_time"] <- medianvalueDelivery_time
105 dset_1
106
107 medianvalueCaesarian <- median(dset_1$Caesarian, na.rm=TRUE)
108 medianvalueCaesarian
109 dset_1[is.na(dset_1$Caesarian), "Caesarian"] <- medianvalueCaesarian
110 dset_1
111
112 idRange=max(dset_1$id, na.rm=TRUE)-min(dset_1$id, na.rm=TRUE)
113 print(idRange)
114 AgeRange=max(dset_1$Age, na.rm=TRUE)-min(dset_1$Age, na.rm=TRUE)
115 print(AgeRange)
116 weightRange=max(dset_1$weight.kg., na.rm=TRUE)-min(dset_1$weight.kg., na.rm=TRUE)
117 print(weightRange)
118 Delivery_numberRange=max(dset_1$Delivery_number, na.rm=TRUE)-min(dset_1$Delivery_number, na.rm=TRUE)
119 print(Delivery_numberRange)
120 Delivery_timeRange=max(dset_1$Delivery_time, na.rm=TRUE)-min(dset_1$Delivery_time, na.rm=TRUE)
121 print(Delivery_timeRange)
122 CaesarianRange=max(dset_1$Caesarian, na.rm=TRUE)-min(dset_1$Caesarian, na.rm=TRUE)
123 print(CaesarianRange)
124 HeartRange=max(dset_1$Heart, na.rm=TRUE)-min(dset_1$Heart, na.rm=TRUE)
125 print(HeartRange)
126
127
```

**Console Output:**

```
> idRange=max(dset_1$id, na.rm=TRUE)-min(dset_1$id, na.rm=TRUE)
[1] 79
> AgeRange=max(dset_1$Age, na.rm=TRUE)-min(dset_1$Age, na.rm=TRUE)
[1] 77
> weightRange=max(dset_1$weight.kg., na.rm=TRUE)-min(dset_1$weight.kg., na.rm=TRUE)
[1] 61
> Delivery_numberRange=max(dset_1$Delivery_number, na.rm=TRUE)-min(dset_1$Delivery_number, na.rm=TRUE)
[1] 3
> Delivery_timeRange=max(dset_1$Delivery_time, na.rm=TRUE)-min(dset_1$Delivery_time, na.rm=TRUE)
[1] 2
> CaesarianRange=max(dset_1$Caesarian, na.rm=TRUE)-min(dset_1$Caesarian, na.rm=TRUE)
[1] 1
> HeartRange=max(dset_1$Heart, na.rm=TRUE)-min(dset_1$Heart, na.rm=TRUE)
[1] 1
```

**Environment Pane:**

Variable	Value
meanvalueCaes.	0.564102564102564
meanvalueDel1.	1.67948717948718
meanvalueDel1.	0.623746623746623
meanvalueweig.	65.1272727272727
medianvalueAge	NA_real_
medianvalueCa.	NA_real_
medianvalueDe.	NA_real_
medianvalueDe.	NA_real_
modeAge	"19"
modeCaesarian	"0"
modeDelivery...	"4"
modeDelivery...	"1"
modeweight	"49"
weightrange	61