American International University-Bangladesh

NAME: Sadia Islam Shafina

ID: 20-43539-1

COURSE: INTRODUCTION TO DATA SCIENCE

SECTION: C

SUBMISSION DATE: 18-07-2023

Description:

The titanic dataset is a well-known dataset in the field of data science. Mainly this dataset is a collection of titanic ship people information. It includes information such as age, gender, siblings, parents / children aboard the Titanic, Passenger fare, Port of Embarkation, Ticket class in number, categories to (man, women, children), he was alone in ship or no, survival status. The data set contain many rows and 10 column. Some data points are missing, denoted by empty cells in the data table. The dataset includes various attributes of the passengers such as their age, gender, sibsp (siblings), parch (parents / children aboard the Titanic) ,fare(Passenger fare), embarked(Port of Embarkation), class(Ticket class in number), who(categories to (man, women, children), alone(he was alone in ship or no), survival(survival) status). There are different types of attributes in our titanic dataset and they are integer, numeric, character. For this project our goal is to obtain a clean preprocessed dataset.

Project solution design:

- Import the data set(titanic)as a csv file.
- View the structure of the dataset.
- The first view row of the dataset.
- To see the column name of data set.
- Summary of dataset.
- Finding type of our column.
- Measure of center
- Measure of Spread
- Finding missing value.
- Recover gender attributes missing values with most frequent value/Mode
- Detect the outlier.
- Data cleaning (removing missing value in 2 way)
- Annotate
- Data transformation
- Visualizations.

Code and the steps of the projects:

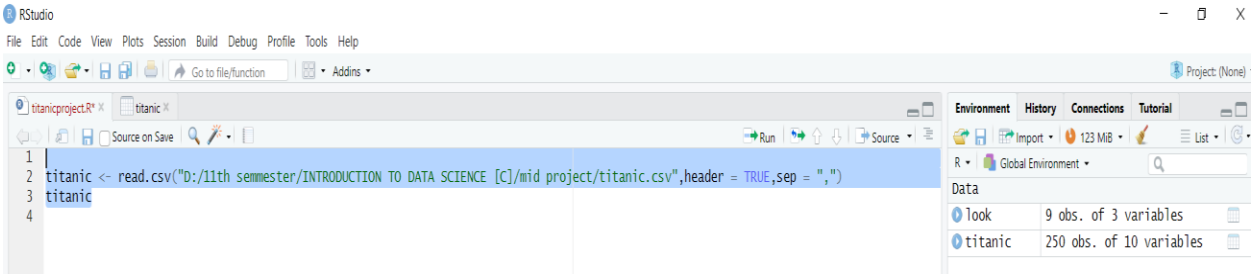
Import data:

Explanation: First of all , insert all data from the excel file and then save the file as a dataset file. Then change the format of the dataset file into CSV file. Then , I import my csv file in R Studio then I provide the following code.

Code:

> titanic <- read.csv("D:/11th semmester/INTRODUCTION TO DATA SCIENCE[C]/mid project/titanic.csv",header = TRUE,sep = ",")

> titanic



Output:

```
> titanic <- read.csv("D:/11th semmester/INTRODUCTION TO DATA SCIENCE [C]/mid project/titanic.csv")
>   View(titanic)
> titanic
   gender   age sibsp parch     fare embarked  class    who alone survived
1       0 22.00     1     0   7.2500        S  Third    man FALSE        0
2       1 38.00     1     0  71.2833        C  First woman  FALL        1
3       1 26.00     0     0   7.9250        S  Third woman  TRUE        1
4       1 35.00     1     0  53.1000        S  First woman  FALL        1
5       0 35.00     0     0   8.0500        S  Third    man  TRUE        0
6       0    NA     0     0   8.4583        Q  Third    man  TRUE        0
7       0 54.00     0     0  51.8625        S  First    man  TRUE        0
8       0  2.00     3     1  21.0750        S  Third child FALSE        0
9       1 27.00     0     2  11.1333        S  Third woman FALSE        1
10      1 14.00     1     0  30.0708        C Second child FALSE        1
11      1  4.00     1     1  16.7000        S  Third child FALSE        1
12      1 58.00     0     0  26.5500        S  First woman  TRUE        1
13     NA 20.00     0     0   8.0500        S  Third    man  TRUE        0
14      0 39.00     1     5  31.2750        S  Third    man FALSE        0
15      1 14.00     0     0   7.8542        S  Third child  TRUE        0
16      1 55.00     0     0  16.0000        S Second woman  TRUE        1
17      0  2.00     4     1  29.1250        Q  Third child FALSE        0
18      0    NA     0     0  13.0000        S Second    man  TRUE        1
19      1 31.00     1     0  18.0000        S  Third woman FALSE        0
20      1    NA     0     0   7.2250        C  Third woman  TRUE        1
21      0 35.00     0     0  26.0000        S Second    man  TRUE        0
22      0 34.00     0     0  13.0000        S Second    man  TRUE        1
23      1 15.00     0     0   8.0292        Q  Third child  TRUE        1
24      0 28.00     0     0  35.5000        S  First    man  TRUE        1
25      1  8.00     3     1  21.0750        S        child FALSE        0
26      1 38.00     1     5  31.3875        S  Third woman FALSE        1
27      0    NA     0     0   7.2250        C  Third    man  TRUE        0
28      0 19.00     3     2 263.0000        S  First    man FALSE        0
29      1    NA     0     0   7.8792        Q  Third woman  TRUE        1
30      0    NA     0     0   7.8958        S  Third    man  TRUE        0
31      0 40.00     0     0  27.7208        C  First    man  TRUE        0
```

View the structure of the dataset:

Explain: The str() function displays the structure of the dataset, including the variables, their data types, and the first few values. This will give us an overview of the dataset.

Code:

```
> str(titanic)
```

Output:

```
> str(titanic)
'data.frame':   250 obs. of  10 variables:
 $ gender  : int  0 1 1 1 0 0 0 0 1 1 ...
 $ age     : num  22 38 26 35 35 NA 54 2 27 14 ...
 $ sibsp   : int  1 1 0 1 0 0 0 3 0 1 ...
 $ parch   : int  0 0 0 0 0 0 0 1 2 0 ...
 $ fare    : num  7.25 71.28 7.92 53.1 8.05 ...
 $ embarked: chr  "S" "C" "S" "S" ...
 $ class   : chr  "Third" "First" "Third" "First" ...
 $ who     : chr  "man" "woman" "woman" "woman" ...
 $ alone   : chr  "FALSE" "FALL" "TRUE" "FALL" ...
 $ survived: int  0 1 1 1 0 0 0 0 1 1 ...
>
```

The first view row of the dataset:

Explanation: The head() function displays the first few rows of the dataset. This will help us get a sense of the data and verify that it has been imported correctly.

Code:

```
> head(titanic)
```

Output:

```
> head(titanic)
  gender age sibsp parch    fare embarked class   who alone survived
1      0  22     1     0  7.2500        S Third   man FALSE        0
2      1  38     1     0 71.2833        C First woman  FALL        1
3      1  26     0     0  7.9250        S Third woman  TRUE        1
4      1  35     1     0 53.1000        S First woman  FALL        1
5      0  35     0     0  8.0500        S Third   man  TRUE        0
6      0  NA     0     0  8.4583        Q Third   man  TRUE        0
>
```

To see the column name of data set:

Code:

```
> names(titanic)
```

output

```
> names(titanic)
 [1] "gender"   "age"      "sibsp"    "parch"    "fare"     "embarked" "class"    "who"      "alone"    "survived"
```

Summary statistics of numeric variables:

Explanation: The summary() function provides summary statistics (count, mean, median, etc.) for numeric variables in the dataset. This will give us insights into the distribution and central tendencies of the variables.

code

```
> summary(titanic)
```

Output:

```
> summary(titanic)
     gender            age            sibsp           parch            fare          embarked            class
 Min.   :0.0000   Min.   :  0.83   Min.   :0.000   Min.   :0.000   Min.   :  0.000   Length:250         Length:250
 1st Qu.:0.0000   1st Qu.: 19.00   1st Qu.:0.000   1st Qu.:0.000   1st Qu.:  8.034   Class :character   Class :character
 Median :0.0000   Median : 27.00   Median :0.000   Median :0.000   Median : 13.977   Mode  :character   Mode  :character
 Mean   :0.3629   Mean   : 33.33   Mean   :0.656   Mean   :0.392   Mean   : 26.588
 3rd Qu.:1.0000   3rd Qu.: 37.00   3rd Qu.:1.000   3rd Qu.:0.000   3rd Qu.: 29.094
 Max.   :1.0000   Max.   :455.00   Max.   :8.000   Max.   :5.000   Max.   :263.000
 NA's   :13       NA's   :48
     who                alone              survived
 Length:250         Length:250         Min.   :0.000
 Class :character   Class :character   1st Qu.:0.000
 Mode  :character   Mode  :character   Median :0.000
                                       Mean   :0.344
                                       3rd Qu.:1.000
                                       Max.   :1.000

>
```

Finding type of our column:

Explanation: using sapply, we can know which column has which type.

code

```
> sapply(titanic, class)
```

Output:

```
> sapply(titanic, class)
     gender         age       sibsp       parch        fare    embarked       class         who       alone    survived
  "integer"   "numeric"   "integer"   "integer"   "numeric" "character" "character" "character" "character"   "integer"
>
```

Measure of center (mean, median, mode)for age,sibsp,parch,fare,survived attributes.

Explanation: The mean provides the average value, the median represents the middle value, and the mode indicates the most frequent value within each attribute. By using these measures, I gain a better understanding of the distribution and typical values of the dataset, which can be useful for making inferences and comparisons in our analysis.

code

For age:

Code:

```
find_mode <- function(x) {
  u <- unique(x)
  tab <- tabulate(match(x, u))
  u[tab == max(tab)]
}

age_mean <- mean(titanic$age, na.rm = TRUE)
age_median <- median(titanic$age, na.rm = TRUE)
age_mode <- find_mode(titanic$age)

print(age_mean)
print(age_median)
print(age_mode)
```

Output:

```
> find_mode <- function(x) {
+   u <- unique(x)
+   tab <- tabulate(match(x, u))
+   u[tab == max(tab)]
+ }
>
> age_mean <- mean(titanic$age, na.rm = TRUE)
> age_median <- median(titanic$age, na.rm = TRUE)
> age_mode <- find_mode(titanic$age)
> find_mode <- function(x) {
+   u <- unique(x)
+   tab <- tabulate(match(x, u))
+   u[tab == max(tab)]
+ }
>
> age_mean <- mean(titanic$age, na.rm = TRUE)
> age_median <- median(titanic$age, na.rm = TRUE)
> age_mode <- find_mode(titanic$age)
>
> print(age_mean)
[1] 33.32837
> print(age_median)
[1] 27
> print(age_mode)
[1] NA
> #######
```

For sibsp:

Code:

```
sibsp_mean <- mean(titanic$sibsp, na.rm = TRUE)
sibsp_median <- median(titanic$sibsp, na.rm = TRUE)
sibsp_mode <- find_mode(titanic$sibsp)
print(sibsp_mean)
print(sibsp_median)
print(sibsp_mode)
```

output:

```
>
> sibsp_mean <- mean(titanic$sibsp, na.rm = TRUE)
> sibsp_median <- median(titanic$sibsp, na.rm = TRUE)
> sibsp_mode <- find_mode(titanic$sibsp)
>
> print(sibsp_mean)
[1] 0.656
> print(sibsp_median)
[1] 0
> print(sibsp_mode)
[1] 0
>
```

For parch:

Code:

```
parch_mean <- mean(titanic$parch , na.rm = TRUE)
parch_median <- median(titanic$parch , na.rm = TRUE)
parch_mode <- find_mode(titanic$parch )

print(parch_mean)
print(parch_median)
print(parch_mode)
```

Explanation:

Output:

```
> parch_mean <- mean(titanic$parch , na.rm = TRUE)
> parch_median <- median(titanic$parch , na.rm = TRUE)
> parch_mode <- find_mode(titanic$parch )
>
> print(parch_mean)
[1] 0.392
> print(parch_median)
[1] 0
> print(parch_mode)
[1] 0
```

For Fare:

Code:

```
fare_mean <- mean(titanic$fare  , na.rm = TRUE)
fare_median <- median(titanic$fare  , na.rm = TRUE)
fare_mode <- find_mode(titanic$fare  )

print(fare_mean)
print(fare_median)
print(fare_mode)
```

output:

```
> fare_mean <- mean(titanic$fare  , na.rm = TRUE)
> fare_median <- median(titanic$fare  , na.rm = TRUE)
> fare_mode <- find_mode(titanic$fare  )
>
> print(fare_mean)
[1] 26.58762
> print(fare_median)
[1] 13.9771
> print(fare_mode)
[1] 8.05
>
```

Survived:

Code:

```
survived_mean <- mean(titanic$survived  , na.rm = TRUE)
survived_median <- median(titanic$survived  , na.rm = TRUE)
survived_mode <- find_mode(titanic$survived  )

print(survived_mean)
print(survived_median)
print(survived_mode)
```

output:

```
> survived_mean <- mean(titanic$survived  , na.rm = TRUE)
> survived_median <- median(titanic$survived  , na.rm = TRUE)
> survived_mode <- find_mode(titanic$survived  )
>
> print(survived_mean)
[1] 0.344
> print(survived_median)
[1] 0
> print(survived_mode)
[1] 0
>
```

Measure of Spread (range and standard Deviation) for age, parch, sibsp and survived attributes.

Explanation: Measures of spread, like range and standard deviation help understand data variability and dispersion within a dataset. Range provides the difference between maximum and minimum values, while standard deviation calculates the average deviation from the mean. Here I including na.rm = TRUE, any missing values in the age column will be ignored, and the range will be calculated based on the available non-missing values.

For age:

Code:

```
age_range <- range(titanic$age, na.rm = TRUE)
print(age_range)
age_sd <- sd(titanic$age, na.rm = TRUE)
print(age_sd)
```

Output:

```
> age_range <- range(titanic$age, na.rm = TRUE)
> print(age_range)
[1]   0.83 455.00
>
> age_sd <- sd(titanic$age, na.rm = TRUE)
> print(age_sd)
[1] 45.7735
> ######
```

For sibsp

Code:

```
sibsp_range <- range(titanic$sibsp, na.rm = TRUE)
print(sibsp_range)
sibsp_sd <- sd(titanic$sibsp, na.rm = TRUE)
print(sibsp_sd)
```

output:

```
> sibsp_range <- range(titanic$sibsp, na.rm = TRUE)
> print(sibsp_range)
[1] 0 8
>
> sibsp_sd <- sd(titanic$sibsp, na.rm = TRUE)
> print(sibsp_sd)
[1] 1.305558
> ######
```

For parch:

Code:

```
parch_range <- range(titanic$parch, na.rm = TRUE)
print(parch_range)

parch_sd <- sd(titanic$parch, na.rm = TRUE)
print(parch_sd)
```

output:

```
>
> parch_range <- range(titanic$parch, na.rm = TRUE)
> print(parch_range)
[1] 0 5
>
> parch_sd <- sd(titanic$parch, na.rm = TRUE)
> print(parch_sd)
[1] 0.8252637
>
```

For fare

Code:

```
fare_range <- range(titanic$fare, na.rm = TRUE)
print(fare_range)

fare_sd <- sd(titanic$fare, na.rm = TRUE)
print(fare_sd)
```

output:

```
> ##
> fare_range <- range(titanic$fare, na.rm = TRUE)
> print(fare_range)
[1]   0 263
>
> fare_sd <- sd(titanic$fare, na.rm = TRUE)
> print(fare_sd)
[1] 34.82165
```

For Survived:

Code:

```
survived_range <- range(titanic$survived, na.rm = TRUE)
print(survived_range)

survived_sd <- sd(titanic$fsurvived, na.rm = TRUE)
print(survived_sd)
```

output:

```
> survived_range <- range(titanic$survived, na.rm = TRUE)
> print(survived_range)
[1] 0 1
>
> survived_sd <- sd(titanic$fsurvived, na.rm = TRUE)
> print(survived_sd)
[1] NA
```

Find the missing value for all attributes:

Explanation: Finding and handling missing values in a dataset is important because missing data can introduce bias and affect the accuracy of our analysis and results. Using this bellow code I can know the missing value in every column.

code

```
number_of_missing_value=colSums(is.na(titanic))
number_of_missing_value
```

Output:

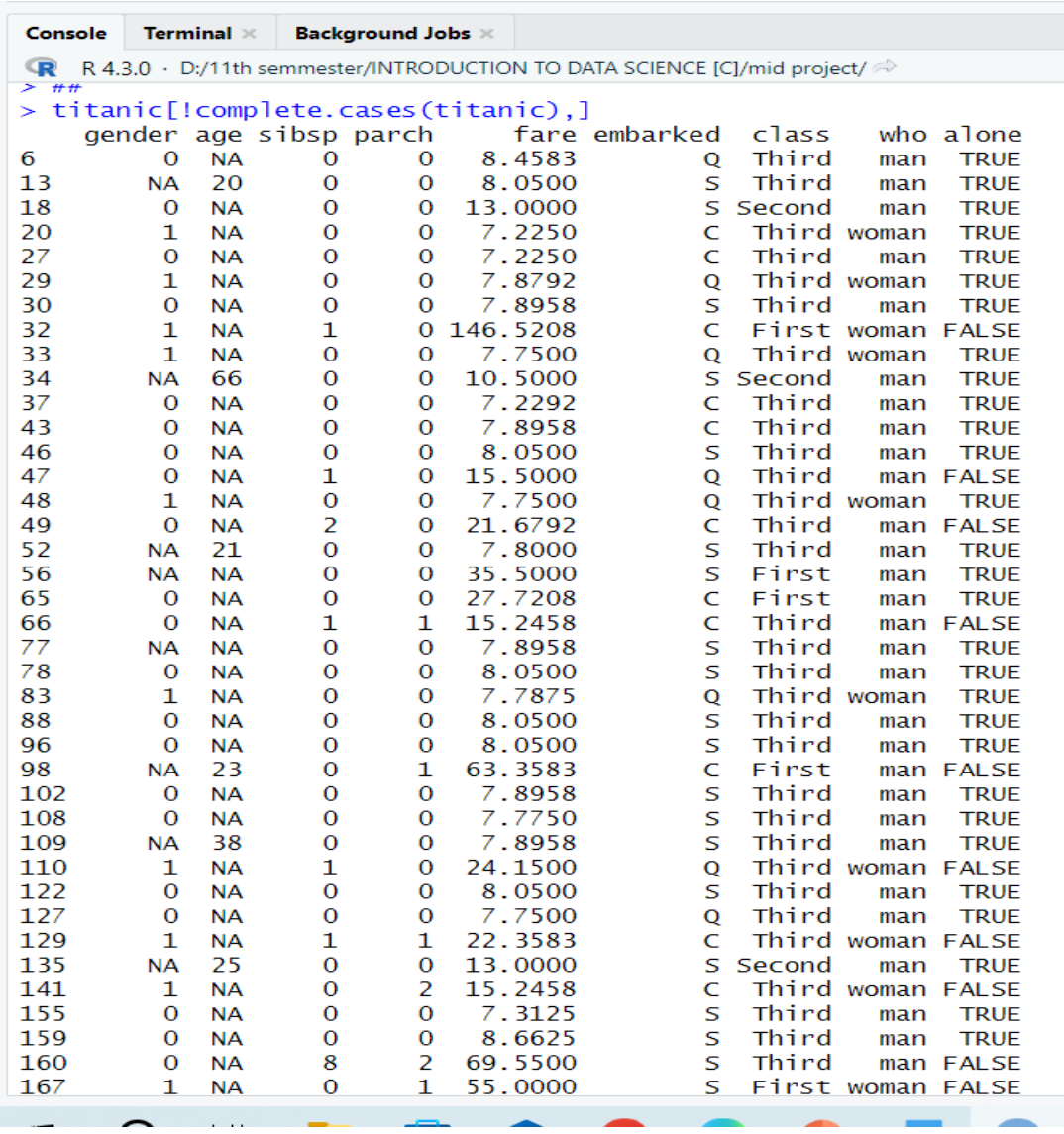
```
> number_of_missing_value=colSums(is.na(titanic))
> number_of_missing_value
  gender      age    sibsp    parch     fare embarked    class
      13       48        0        0        0        0        0
     who    alone survived
       0        0        0
>
```

Explanation: The code "titanic[!complete.cases(titanic),]" is used to subset the "titanic" dataset and extract the rows that have missing values.

Code:

```
titanic[!complete.cases(titanic),]
```

Output:



Console | Terminal | Background Jobs

R 4.3.0 · D:/11th semmester/INTRODUCTION TO DATA SCIENCE [C]/mid project/

```
> ##
> titanic[!complete.cases(titanic),]
     gender age sibsp parch     fare embarked  class    who alone
6         0  NA     0     0   8.4583        Q  Third    man  TRUE
13       NA  20     0     0   8.0500        S  Third    man  TRUE
18        0  NA     0     0  13.0000        S Second    man  TRUE
20        1  NA     0     0   7.2250        C  Third woman  TRUE
27        0  NA     0     0   7.2250        C  Third    man  TRUE
29        1  NA     0     0   7.8792        Q  Third woman  TRUE
30        0  NA     0     0   7.8958        S  Third    man  TRUE
32        1  NA     1     0 146.5208        C  First woman FALSE
33        1  NA     0     0   7.7500        Q  Third woman  TRUE
34       NA  66     0     0  10.5000        S Second    man  TRUE
37        0  NA     0     0   7.2292        C  Third    man  TRUE
43        0  NA     0     0   7.8958        C  Third    man  TRUE
46        0  NA     0     0   8.0500        S  Third    man  TRUE
47        0  NA     1     0  15.5000        Q  Third    man FALSE
48        1  NA     0     0   7.7500        Q  Third woman  TRUE
49        0  NA     2     0  21.6792        C  Third    man FALSE
52       NA  21     0     0   7.8000        S  Third    man  TRUE
56       NA  NA     0     0  35.5000        S  First    man  TRUE
65        0  NA     0     0  27.7208        C  First    man  TRUE
66        0  NA     1     1  15.2458        C  Third    man FALSE
77       NA  NA     0     0   7.8958        S  Third    man  TRUE
78        0  NA     0     0   8.0500        S  Third    man  TRUE
83        1  NA     0     0   7.7875        Q  Third woman  TRUE
88        0  NA     0     0   8.0500        S  Third    man  TRUE
96        0  NA     0     0   8.0500        S  Third    man  TRUE
98       NA  23     0     1  63.3583        C  First    man FALSE
102       0  NA     0     0   7.8958        S  Third    man  TRUE
108       0  NA     0     0   7.7750        S  Third    man  TRUE
109      NA  38     0     0   7.8958        S  Third    man  TRUE
110       1  NA     1     0  24.1500        Q  Third woman FALSE
122       0  NA     0     0   8.0500        S  Third    man  TRUE
127       0  NA     0     0   7.7500        Q  Third    man  TRUE
129       1  NA     1     1  22.3583        C  Third woman FALSE
135      NA  25     0     0  13.0000        S Second    man  TRUE
141       1  NA     0     2  15.2458        C  Third woman FALSE
155       0  NA     0     0   7.3125        S  Third    man  TRUE
159       0  NA     0     0   8.6625        S  Third    man  TRUE
160       0  NA     8     2  69.5500        S  Third    man FALSE
167       1  NA     0     1  55.0000        S  First woman FALSE
```

Explanation: Using this bellow code I can know which row have missing value.

Code:

```
missing_gender=which(is.na(titanic$gender))
missing_gender
missing_age=which(is.na(titanic$age))
missing_age
```

Output:

```
>
> missing_gender=which(is.na(titanic$gender))
> missing_gender
 [1]  13  34  52  56  77  98 109 135 177 194 210 214 246
> missing_age=which(is.na(titanic$age))
> missing_age
 [1]   6  18  20  27  29  30  32  33  37  43  46  47  48  49  56
[16]  65  66  77  78  83  88  96 102 108 110 122 127 129 141 155
[31] 159 160 167 169 177 181 182 186 187 197 199 202 215 224 230
[46] 236 241 242
>
```

Handle gender attributes invalid value

Explanation: From our data set gender column is invalid value. Using function and code we can get most frequent value from gender attribute.

Code:

```
find_mode <- function(x) {
  u <- unique(x)
  tab <- tabulate(match(x, u))
  u[tab == max(tab)]
}
most_frequent_gender=find_mode(titanic$gender)
most_frequent_gender
```

output:

```
> find_mode <- function(x) {
+   u <- unique(x)
+   tab <- tabulate(match(x, u))
+   u[tab == max(tab)]
+ }
> most_frequent_gender=find_mode(titanic$gender)
> most_frequent_gender
[1] 0
>
```

Explanation: The Titanic dataset's 10th gender element is updated with the most_frequent_gender variable, replacing the existing value in the 10th row.

Code:

titanic$gender[10]<-most_frequent_gender
print(titanic)

output:

```
> titanic$gender[10]<-most_frequent_gender
> print(titanic)
   gender   age sibsp parch     fare embarked  class    who
1       0 22.00     1     0   7.2500        S  Third    man
2       1 38.00     1     0 71.2833        C  First woman
3       1 26.00     0     0   7.9250        S  Third woman
4       1 35.00     1     0 53.1000        S  First woman
5       0 35.00     0     0   8.0500        S  Third    man
6       0     NA    0     0   8.4583        Q  Third    man
7       0 54.00     0     0 51.8625        S  First    man
8       0  2.00     3     1 21.0750        S  Third child
9       1 27.00     0     2 11.1333        S  Third woman
10      0 14.00     1     0 30.0708        C Second child
11      1  4.00     1     1 16.7000        S  Third child
12      1 58.00     0     0 26.5500        S  First woman
13     NA 20.00     0     0   8.0500        S  Third    man
14      0 39.00     1     5 31.2750        S  Third    man
15      1 14.00     0     0   7.8542        S  Third child
16      1 55.00     0     0 16.0000        S Second woman
17      0  2.00     4     1 29.1250        Q  Third child
18      0     NA    0     0 13.0000        S Second    man
19      1 31.00     1     0 18.0000        S  Third woman
20      1     NA    0     0   7.2250        C  Third woman
21      0 35.00     0     0 26.0000        S Second    man
22      0 34.00     0     0 13.0000        S Second    man
23      1 15.00     0     0   8.0292        Q  Third child
24      0 28.00     0     0 35.5000        S  First    man
25      1  8.00     3     1 21.0750        S        child
26      1 38.00     1     5 31.3875        S  Third woman
27      0     NA    0     0   7.2250        C  Third    man
28      0 19.00     3     2 263.0000       S  First    man
```

Fig: Updated dataset

```
>  titanic <- read.csv("D:/11th semmester/INTRODUCTION TO DATA SCIENCE [C]/mi
> titanic
   gender   age sibsp parch     fare embarked  class    who alone survived
1       0 22.00     1     0   7.2500        S  Third    man FALSE        0
2       1 38.00     1     0 71.2833        C  First woman  FALL        1
3       1 26.00     0     0   7.9250        S  Third woman  TRUE        1
4       1 35.00     1     0 53.1000        S  First woman  FALL        1
5       0 35.00     0     0   8.0500        S  Third    man  TRUE        0
6       0     NA    0     0   8.4583        Q  Third    man  TRUE        0
7       0 54.00     0     0 51.8625        S  First    man  TRUE        0
8       0  2.00     3     1 21.0750        S  Third child FALSE        0
9       1 27.00     0     2 11.1333        S  Third woman FALSE        1
10      1 14.00     1     0 30.0708        C Second child FALSE        1
11      1  4.00     1     1 16.7000        S  Third child FALSE        1
12      1 58.00     0     0 26.5500        S  First woman  TRUE        1
13     NA 20.00     0     0   8.0500        S  Third    man  TRUE        0
14      0 39.00     1     5 31.2750        S  Third    man FALSE        0
15      1 14.00     0     0   7.8542        S  Third child  TRUE        0
16      1 55.00     0     0 16.0000        S Second woman  TRUE        1
17      0  2.00     4     1 29.1250        Q  Third child FALSE        0
18      0     NA    0     0 13.0000        S Second    man  TRUE        1
19      1 31.00     1     0 18.0000        S  Third woman FALSE        0
```

fig: previous dataset

Recover gender attributes missing values with most frequent value/Mode

Code:

titanic$gender[is.na(titanic$gender)]<-most_frequent_gender
print(titanic)

Output:

```
R 4.3.0 · D:/11th semmester/INTRODUCTION TO DATA SCIENCE [C]/mid project/
> titanic$gender[is.na(titanic$gender)]<-most_frequent_gender
> print(titanic)
   gender   age sibsp parch     fare embarked  class    who
1       0 22.00     1     0   7.2500        S  Third    man
2       1 38.00     1     0 71.2833        C  First woman
3       1 26.00     0     0   7.9250        S  Third woman
4       1 35.00     1     0 53.1000        S  First woman
5       0 35.00     0     0   8.0500        S  Third    man
6       0     NA    0     0   8.4583        Q  Third    man
7       0 54.00     0     0 51.8625        S  First    man
8       0  2.00     3     1 21.0750        S  Third child
9       1 27.00     0     2 11.1333        S  Third woman
10      0 14.00     1     0 30.0708        C Second child
11      1  4.00     1     1 16.7000        S  Third child
12      1 58.00     0     0 26.5500        S  First woman
13      0 20.00     0     0   8.0500        S  Third    man
14      0 39.00     1     5 31.2750        S  Third    man
15      1 14.00     0     0   7.8542        S  Third child
16      1 55.00     0     0 16.0000        S Second woman
17      0  2.00     4     1 29.1250        Q  Third child
18      0     NA    0     0 13.0000        S Second    man
19      1 31.00     1     0 18.0000        S  Third woman
20      1     NA    0     0   7.2250        C  Third woman
21      0 35.00     0     0 26.0000        S Second    man
22      0 34.00     0     0 13.0000        S Second    man
23      1 15.00     0     0   8.0292        Q  Third child
24      0 28.00     0     0 35.5000        S  First    man
25      1  8.00     3     1 21.0750        S        child
26      1 38.00     1     5 31.3875        S  Third woman
27      0     NA    0     0   7.2250        C  Third    man
28      0 19.00     3     2 263.0000       S  First    man
29      1     NA    0     0   7.8792        Q  Third woman
30      0     NA    0     0   7.8958        S  Third    man
31      0 40.00     0     0 27.7208        C  First    man
32      1     NA    1     0 146.5208       C  First woman
33      1     NA    0     0   7.7500        Q  Third woman
34      0 66.00     0     0  10.5000        S Second    man
```

Detect the outlier as a missing value:

Explanation: Outliers are those data points that are significantly different from the rest of the dataset. Here I take “age ” attribute as a outliner.

Code:

sort(titanic$age)

Output:

```
> sort(titanic$age)
  [1]   0.83   1.00   1.00   2.00   2.00   2.00   2.00
  [9]   3.00   3.00   4.00   4.00   4.00   4.00   5.00   5.00
 [17]   7.00   8.00   8.00   9.00   9.00   9.00  11.00  12.00
 [25]  14.00  14.00  14.00  14.50  15.00  16.00  16.00  16.00
 [33]  16.00  16.00  16.00  17.00  17.00  17.00  17.00  18.00
 [41]  18.00  18.00  18.00  18.00  18.00  19.00  19.00  19.00
 [49]  19.00  19.00  19.00  19.00  19.00  19.00  19.00  20.00
 [57]  20.00  20.00  20.00  20.50  21.00  21.00  21.00  21.00
 [65]  21.00  21.00  21.00  21.00  21.00  22.00  22.00  22.00
 [73]  22.00  22.00  22.00  22.00  22.00  22.00  23.00  23.00
 [81]  23.00  24.00  24.00  24.00  24.00  24.00  24.00  24.00
 [89]  24.00  24.00  25.00  25.00  25.00  26.00  26.00  26.00
 [97]  26.00  26.00  26.00  27.00  27.00  27.00  27.00  27.00
[105]  28.00  28.00  28.00  28.00  28.00  28.00  28.00  28.50
[113]  29.00  29.00  29.00  29.00  29.00  29.00  29.00  29.00
[121]  30.00  30.00  30.00  30.00  30.00  30.00  31.00  31.00
[129]  32.00  32.00  32.00  32.00  32.00  32.50  33.00  33.00
[137]  33.00  33.00  34.00  34.00  34.00  34.00  35.00  35.00
[145]  35.00  35.00  35.00  36.00  36.00  37.00  37.00  37.00
[153]  38.00  38.00  38.00  38.00  38.00  39.00  40.00  40.00
[161]  40.00  40.00  40.00  40.50  42.00  42.00  42.00  42.00
[169]  44.00  44.00  44.00  44.00  45.00  45.00  45.00  45.00
[177]  46.00  47.00  47.00  49.00  50.00  51.00  51.00  51.00
[185]  54.00  54.00  54.00  55.00  55.50  56.00  58.00  58.00
[193]  59.00  59.00  61.00  65.00  66.00  70.50  71.00 325.00
[201] 365.00 455.00
>
```

Code:

summary(titanic)

output:

```
> summary(titanic)
     gender                age              sibsp
 Min.   :0.00   Min.   :  0.83   Min.   :0.000
 1st Qu.:0.00   1st Qu.: 19.00   1st Qu.:0.000
 Median :0.00   Median : 27.00   Median :0.000
 Mean   :0.34   Mean   : 33.33   Mean   :0.656
 3rd Qu.:1.00   3rd Qu.: 37.00   3rd Qu.:1.000
 Max.   :1.00   Max.   :455.00   Max.   :8.000
                NA's   :48
     parch             fare            embarked
 Min.   :0.000   Min.   :  0.000   Length:250
 1st Qu.:0.000   1st Qu.:  8.034   Class :character
 Median :0.000   Median : 13.977   Mode  :character
 Mean   :0.392   Mean   : 26.588
 3rd Qu.:0.000   3rd Qu.: 29.094
 Max.   :5.000   Max.   :263.000

    class               who                alone
Length:250         Length:250         Length:250
Class :character   Class :character   Class :character
Mode  :character   Mode  :character   Mode  :character


    survived
 Min.   :0.000
 1st Qu.:0.000
 Median :0.000
 Mean   :0.344
 3rd Qu.:1.000
 Max.   :1.000

>
```

Code:

titanic_outlier=subset(titanic, age<=19)
titanic_outlier

Output:

```
> titanic_outlier=subset(titanic, age<=19)
> titanic_outlier
    gender   age sibsp parch      fare embarked  class    who
8        0  2.00     3     1  21.0750        S  Third child
10       0 14.00     1     0  30.0708        C Second child
11       1  4.00     1     1  16.7000        S  Third child
15       1 14.00     0     0   7.8542        S  Third child
17       0  2.00     4     1  29.1250        Q  Third child
23       1 15.00     0     0   8.0292        Q  Third child
25       1  8.00     3     1  21.0750        S        child
28       0 19.00     3     2 263.0000        S  First   man
39       1 18.00     2     0  18.0000        S  Third woman
40       1 14.00     1     0  11.2417        C  Third child
44       1  3.00     1     2  41.5792        C Second child
45       1 19.00     0     0   7.8792        Q  Third woman
50       1 18.00     1     0  17.8000        S  Third woman
51       0  7.00     4     1  39.6875        S  Third child
59       1  5.00     1     2  27.7500        S Second child
60       0 11.00     5     2  46.9000        S  Third child
64       0  4.00     3     2  27.9000        S  Third child
68       0 19.00     0     0   8.1583        S  Third   man
69       1 17.00     4     2   7.9250        S  Third woman
72       1 16.00     5     2  46.9000        S  Third woman
79       0  0.83     0     2  29.0000        S Second child
85       1 17.00     0     0  10.5000        S Second woman
87       0 16.00     1     3  34.3750        S  Third   man
112      1 14.50     1     0  14.4542        C  Third child
115      1 17.00     0     0  14.4583        C  Third woman
120      1  2.00     4     2  31.2750        S  Third child
126      0 12.00     1     0  11.2417        C  Third child
137      1 19.00     0     2  26.2833        S  First woman
139      0 16.00     0     0   9.2167        S  Third   man
```

Code:

titanic_outlier_location=which(titanic$age<19)
titanic_outlier_location

output:

```
> titanic_outlier_location=which(titanic$age<19)
> titanic_outlier_location
 [1]   8  10  11  15  17  23  25  39  40  44  50  51  59  60  64
[16]  69  72  79  85  87 112 115 120 126 139 145 148 157 164 165
[31] 166 172 173 176 183 184 185 194 205 206 209 221 229 234 238
>
```

Code:

```
titanic$age[titanic_outlier_location]<-NA
print(titanic)
```

Output:

```
> titanic$age[titanic_outlier_location]<-NA
> print(titanic)
    gender  age sibsp parch     fare embarked  class   who alone survived
1        0 22.0     1     0   7.2500        S  Third   man FALSE        0
2        1 38.0     1     0  71.2833        C  First woman  FALL        1
3        1 26.0     0     0   7.9250        S  Third woman  TRUE        1
4        1 35.0     1     0  53.1000        S  First woman  FALL        1
5        0 35.0     0     0   8.0500        S  Third   man  TRUE        0
6        0   NA     0     0   8.4583        Q  Third   man  TRUE        0
7        0 54.0     0     0  51.8625        S  First   man  TRUE        0
8        0   NA     3     1  21.0750        S  Third child FALSE        0
9        1 27.0     0     2  11.1333        S  Third woman FALSE        1
10       0   NA     1     0  30.0708        C Second child FALSE        1
11       1   NA     1     1  16.7000        S  Third child FALSE        1
12       1 58.0     0     0  26.5500        S  First woman  TRUE        1
13       0 20.0     0     0   8.0500        S  Third   man  TRUE        0
14       0 39.0     1     5  31.2750        S  Third   man FALSE        0
15       1   NA     0     0   7.8542        S  Third child  TRUE        0
16       1 55.0     0     0  16.0000        S Second woman  TRUE        1
17       0   NA     4     1  29.1250        Q  Third child FALSE        0
18       0   NA     0     0  13.0000        S Second   man  TRUE        1
19       1 31.0     1     0  18.0000        S  Third woman FALSE        0
20       1   NA     0     0   7.2250        C  Third woman  TRUE        1
21       0 35.0     0     0  26.0000        S Second   man  TRUE        0
22       0 34.0     0     0  13.0000        S Second   man  TRUE        1
23       1   NA     0     0   8.0292        Q  Third child  TRUE        1
24       0 28.0     0     0  35.5000        S  First   man  TRUE        1
25       1   NA     3     1  21.0750        S        child FALSE        0
26       1 38.0     1     5  31.3875        S  Third woman FALSE        1
27       0   NA     0     0   7.2250        C  Third   man  TRUE        0
28       0 19.0     3     2 263.0000        S  First   man FALSE        0
29       1   NA     0     0   7.8792        Q  Third woman  TRUE        1
30       0   NA     0     0   7.8958        S  Third   man  TRUE        0
31       0 40.0     0     0  27.7208        C  First   man  TRUE        0
32       1   NA     1     0 146.5208        C  First woman FALSE        1
33       1   NA     0     0   7.7500        Q  Third woman  TRUE        1
```

Data cleaning

Recover missing values: In our report I use two method to remove missing value.
Explanation: Removing missing value we can get a clean dataset that's helps us to analyze our dataset most effectively.

1. Delete the rows with missing values. Using na.omit() function we can delete missing values row .Its a one kind of cleaning missing value.

Code:

```
remove_missing <- na.omit(titanic)
print(remove_missing)
```

Output:

```
> remove_missing <- na.omit(titanic)
> print(remove_missing)
    gender   age sibsp parch     fare embarked  class    who alone survived
1        0  22.0     1     0   7.2500        S  Third    man FALSE        0
2        1  38.0     1     0  71.2833        C  First woman  FALL        1
3        1  26.0     0     0   7.9250        S  Third woman  TRUE        1
4        1  35.0     1     0  53.1000        S  First woman  FALL        1
5        0  35.0     0     0   8.0500        S  Third    man  TRUE        0
7        0  54.0     0     0  51.8625        S  First    man  TRUE        0
9        1  27.0     0     2  11.1333        S  Third woman FALSE        1
12       1  58.0     0     0  26.5500        S  First woman  TRUE        1
13       0  20.0     0     0   8.0500        S  Third    man  TRUE        0
14       0  39.0     1     5  31.2750        S  Third    man FALSE        0
16       1  55.0     0     0  16.0000        S Second woman  TRUE        1
19       1  31.0     1     0  18.0000        S  Third woman FALSE        0
21       0  35.0     0     0  26.0000        S Second    man  TRUE        0
22       0  34.0     0     0  13.0000        S Second    man  TRUE        1
24       0  28.0     0     0  35.5000        S  First    man  TRUE        1
26       1  38.0     1     5  31.3875        S  Third woman FALSE        1
28       0  19.0     3     2 263.0000        S  First    man FALSE        0
31       0  40.0     0     0  27.7208        C  First    man  TRUE        0
34       0  66.0     0     0  10.5000        S Second    man  TRUE        0
35       0  28.0     1     0  82.1708        C  First    man FALSE        0
36       0  42.0     1     0  52.0000        S  First    man FALSE        0
38       0  21.0     0     0   8.0500        S  Third    man  TRUE        0
41       1  40.0     1     0   9.4750        S  Third woman FALSE        0
42       1  27.0     1     0  21.0000        S Second woman FALSE        0
45       1  19.0     0     0   7.8792        Q  Third woman  TRUE        1
52       0  21.0     0     0   7.8000        S  Third    man  TRUE        0
53       1  49.0     1     0  76.7292        C  First woman FALSE        1
54       1  29.0     1     0  26.0000        S Second woman FALSE        1
55       0  65.0     0     1  61.9792        C  First    man FALSE        0
57       1  21.0     0     0  10.5000        S Second woman  TRUE        1
58       0  28.5     0     0   7.2292        C  Third    man  TRUE        0
```

2. Recover missing values with the mean value.

Code:

```
age_mean=mean(titanic$age,na.rm=T)
recover_missing_age_mean=titanic$age[is.na(titanic$age)]<-age_mean
recover_missing_age_mean
 print(titanic)
```

Output:

```
> age_mean=mean(titanic$age,na.rm=T)
> recover_missing_age_mean=titanic$age[is.na(titanic$age)]<-age_mean
> recover_missing_age_mean
[1] 39.94904
> print(titanic)
    gender      age sibsp parch     fare embarked  class   who alone
1        0 22.00000     1     0   7.2500        S  Third   man FALSE
2        1 38.00000     1     0  71.2833        C  First woman  FALL
3        1 26.00000     0     0   7.9250        S  Third woman  TRUE
4        1 35.00000     1     0  53.1000        S  First woman  FALL
5        0 35.00000     0     0   8.0500        S  Third   man  TRUE
6        0 39.94904     0     0   8.4583        Q  Third   man  TRUE
7        0 54.00000     0     0  51.8625        S  First   man  TRUE
8        0 39.94904     3     1  21.0750        S  Third child FALSE
9        1 27.00000     0     2  11.1333        S  Third woman FALSE
10       0 39.94904     1     0  30.0708        C Second child FALSE
11       1 39.94904     1     1  16.7000        S  Third child FALSE
12       1 58.00000     0     0  26.5500        S  First woman  TRUE
13       0 20.00000     0     0   8.0500        S  Third   man  TRUE
14       0 39.00000     1     5  31.2750        S  Third   man FALSE
15       1 39.94904     0     0   7.8542        S  Third child  TRUE
16       1 55.00000     0     0  16.0000        S Second woman  TRUE
17       0 39.94904     4     1  29.1250        Q  Third child FALSE
18       0 39.94904     0     0  13.0000        S Second   man  TRUE
19       1 31.00000     1     0  18.0000        S  Third woman FALSE
20       1 39.94904     0     0   7.2250        C  Third woman  TRUE
21       0 35.00000     0     0  26.0000        S Second   man  TRUE
22       0 34.00000     0     0  13.0000        S Second   man  TRUE
23       1 39.94904     0     0   8.0292        Q  Third child  TRUE
24       0 28.00000     0     0  35.5000        S  First   man  TRUE
25       1 39.94904     3     1  21.0750        S        child FALSE
26       1 38.00000     1     5  31.3875        S  Third woman FALSE
27       0 39.94904     0     0   7.2250        C  Third   man  TRUE
```

Data transformation:

We already known, the data transformation process includes one or more of the following steps: normalization, summarization, noise removal, smoothing, and summarizing of the data. for our data set I used normalization.

Normalization: The statistical distribution of the data is positively impacted by normalization procedures since they allow us to minimize the magnitude of the variables.in this data set I have normalized column between 3 to 5.

Code:

```
> min_max_norm <- function(x) { (x - min(x)) / (max(x) - min(x))
}
> titanic <- as.data.frame(lapply(titanic[3:5], min_max_norm))
> titanic
.
```

Output:

```
>
> min_max_norm <- function(x) { (x - min(x)) / (max(x) - min(x)) }
> titanic <- as.data.frame(lapply(titanic[3:5], min_max_norm))
> titanic
    sibsp parch       fare
1   0.125   0.0 0.02756654
2   0.125   0.0 0.27103916
3   0.000   0.0 0.03013308
4   0.125   0.0 0.20190114
5   0.000   0.0 0.03060837
6   0.000   0.0 0.03216084
7   0.000   0.0 0.19719582
8   0.375   0.2 0.08013308
9   0.000   0.4 0.04233194
10  0.125   0.0 0.11433764
11  0.125   0.2 0.06349810
12  0.000   0.0 0.10095057
13  0.000   0.0 0.03060837
14  0.125   1.0 0.11891635
15  0.000   0.0 0.02986388
16  0.000   0.0 0.06083650
17  0.500   0.2 0.11074144
18  0.000   0.0 0.04942966
19  0.125   0.0 0.06844106
20  0.000   0.0 0.02747148
21  0.000   0.0 0.09885932
```

Annotate

Explanation: Here I, use Annotate for Improve data interpretability, accuracy, and analysis for better decision-making. In our Titanic dataset have a 10 attributes . I annotate class attributes then I annotate who attribute .
Annotate First as 1, second as 2, and Third as 3 from “class” attribute and Annotate man as 1, woman as 2, and child as 3 from “who” attribute.

First Annotate:

Code:

```
titanic$class<-factor(titanic$class,levels=c("First","Second","Third"),labels=c(1,2,3))
print(titanic$class)
print(titanic)
```

Output:

```
> titanic$class<-factor(titanic$class,levels=c("First","Second","Third"),label
s=c(1,2,3))
> print(titanic$class)
  [1] 3    1    3    1    3    3    1    3    3    2    3    1    3    3
 [15] 3    2    3    2    3    3    2    2    3    1    <NA> 3    3    1
 [29] 3    3    1    1    3    2    1    1    3    3    3    3    3    2
 [43] 3    2    3    3    3    3    3    3    3    3    1    2    1    1
 [57] 2    3    2    3    3    1    1    3    1    3    2    3    3    3
 [71] 2    3    2    3    3    3    3    3    2    3    3    3    3    1
 [85] 2    3    3    3    1    3    3    3    1    3    3    3    1    1
 [99] 2    2    3    3    1    3    3    3    3    3    3    3    1    3
[113] 3    3    3    3    <NA> 2    1    3    2    3    2    2    1    3
[127] 3    3    3    3    3    3    3    2    2    2    1    1    3    1
[141] 3    3    3    3    2    2    3    3    2    2    2    1    3    3
[155] 3    1    3    3    3    3    3    2    3    3    3    3    1    3
[169] 1    3    1    3    3    3    1    3    3    1    2    3    3    2
[183] 3    2    3    1    3    1    3    3    2    2    <NA> 2    1    1
[197] 3    3    3    2    3    3    3    3    3    3    3    3    3    1
```

```
> print(titanic)
    gender   age sibsp parch     fare embarked class   who alone survived
1        0 22.00     1     0   7.2500        S     3   man FALSE        0
2        1 38.00     1     0  71.2833        C     1 woman  FALL        1
3        1 26.00     0     0   7.9250        S     3 woman  TRUE        1
4        1 35.00     1     0  53.1000        S     1 woman  FALL        1
5        0 35.00     0     0   8.0500        S     3   man  TRUE        0
6        0    NA     0     0   8.4583        Q     3   man  TRUE        0
7        0 54.00     0     0  51.8625        S     1   man  TRUE        0
8        0  2.00     3     1  21.0750        S     3 child FALSE        0
9        1 27.00     0     2  11.1333        S     3 woman FALSE        1
10       1 14.00     1     0  30.0708        C     2 child FALSE        1
11       1  4.00     1     1  16.7000        S     3 child FALSE        1
12       1 58.00     0     0  26.5500        S     1 woman  TRUE        1
13      NA 20.00     0     0   8.0500        S     3   man  TRUE        0
14       0 39.00     1     5  31.2750        S     3   man FALSE        0
15       1 14.00     0     0   7.8542        S     3 child  TRUE        0
16       1 55.00     0     0  16.0000        S     2 woman  TRUE        1
17       0  2.00     4     1  29.1250        Q     3 child FALSE        0
18       0    NA     0     0  13.0000        S     2   man  TRUE        1
19       1 31.00     1     0  18.0000        S     3 woman FALSE        0
20       1    NA     0     0   7.2250        C     3 woman  TRUE        1
21       0 35.00     0     0  26.0000        S     2   man  TRUE        0
22       0 34.00     0     0  13.0000        S     2   man  TRUE        1
23       1 15.00     0     0   8.0292        Q     3 child  TRUE        1
24       0 28.00     0     0  35.5000        S     1   man  TRUE        1
```

Second Annotate:

Code:

```
titanic$who<-factor(titanic$who,levels=c("man","woman","child"),labels=c(1,2,3))
print(titanic$who)
print(titanic)
```

Output :

```
> titanic$who<-factor(titanic$who,levels=c("man","woman","child"),labels=c(1,
2,3))
> print(titanic$who)
  [1] 1 2 2 2 1 1 1 3 2 3 3 2 1 1 3 2 3 1 2 2 1 1 3 1 3 2 1 1 2 1 1 2 2 1 1
 [36] 1 1 1 2 3 2 2 1 3 2 1 1 2 1 2 3 1 2 2 1 1 2 1 3 3 1 2 1 3 1 1 2 1 2 1
 [71] 1 2 1 1 1 1 1 1 3 2 1 1 2 1 2 2 1 1 2 1 1 1 1 1 1 1 2 1 2 1 1 1 1
[106] 1 2 1 1 2 1 3 1 2 2 1 1 1 1 3 1 1 1 2 1 3 1 1 2 1 1 1 2 2 1 1 2 1 1 1
[141] 2 2 2 1 1 1 1 3 1 1 1 2 1 1 1 1 2 1 1 1 1 2 1 1 3 3 2 2 1 1 1 3 3 1 1
[176] 1 1 2 1 1 2 1 3 3 3 1 2 1 1 1 2 1 2 3 2 2 1 1 2 2 1 1 1 1 1 3 1 1 2 1
[211] 1 2 1 1 1 2 2 1 2 1 1 1 1 1 1 1 1 1 1 2 2 1 1 3 1 2 1 3 1 1 2 2 1 1 1
[246] 1 2 2 1 1
Levels: 1 2 3
> print(titanic)
    gender   age sibsp parch     fare embarked class who alone survived
1        0 22.00     1     0   7.2500        S     3   1 FALSE        0
2        1 38.00     1     0  71.2833        C     1   2  FALL        1
3        1 26.00     0     0   7.9250        S     3   2  TRUE        1
4        1 35.00     1     0  53.1000        S     1   2  FALL        1
5        0 35.00     0     0   8.0500        S     3   1  TRUE        0
6        0    NA     0     0   8.4583        Q     3   1  TRUE        0
7        0 54.00     0     0  51.8625        S     1   1  TRUE        0
8        0  2.00     3     1  21.0750        S     3   3 FALSE        0
9        1 27.00     0     2  11.1333        S     3   2 FALSE        1
10       1 14.00     1     0  30.0708        C     2   3 FALSE        1
11       1  4.00     1     1  16.7000        S     3   3 FALSE        1
12       1 58.00     0     0  26.5500        S     1   2  TRUE        1
13      NA 20.00     0     0   8.0500        S     3   1  TRUE        0
14       0 39.00     1     5  31.2750        S     3   1 FALSE        0
15       1 14.00     0     0   7.8542        S     3   3  TRUE        0
16       1 55.00     0     0  16.0000        S     2   2  TRUE        1
17       0  2.00     4     1  29.1250        Q     3   3 FALSE        0
18       0    NA     0     0  13.0000        S     2   1  TRUE        1
19       1 31.00     1     0  18.0000        S     3   2 FALSE        0
20       1    NA     0     0   7.2250        C     3   2  TRUE        1
21       0 35.00     0     0  26.0000        S     2   1  TRUE        0
22       0 34.00     0     0  13.0000        S     2   1  TRUE        1
23       1 15.00     0     0   8.0292        Q     3   3  TRUE        1
24       0 28.00     0     0  35.5000        S     1   1  TRUE        1
25       1  8.00     3     1  21.0750        S  <NA>   3 FALSE        0
```

Visualizations:

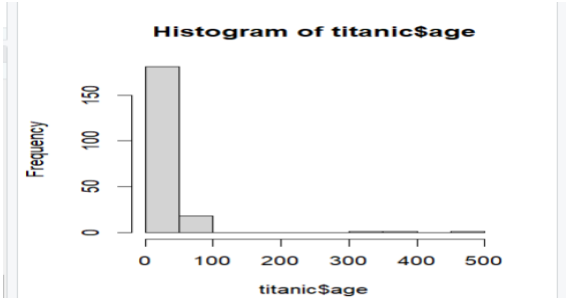
According for our Titanic data set here I create Histograms, bar plots, and scatter plots are commonly used visualizations in data analysis to gain insights and understand patterns in the dataset. Here I also create histogram for

Histogram for continuous variables (age):

code

```
> hist(titanic$age)
```

output

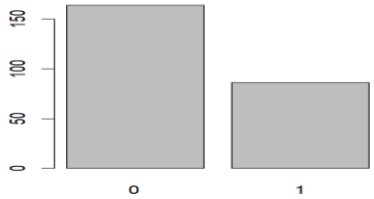


Bar plot for categorical variables (survived):

code

```
> barplot(table(titanic$survived))
```

Output:

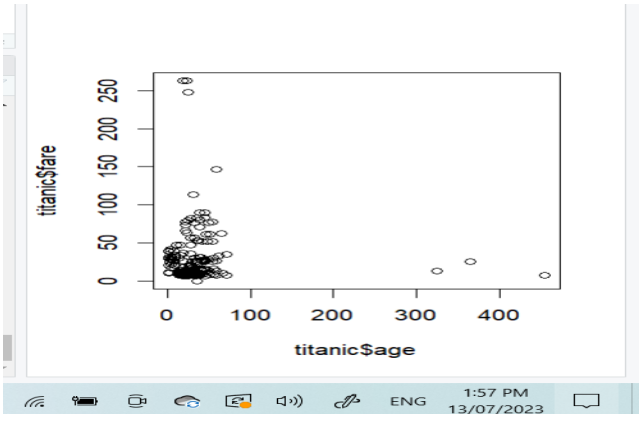


scatter plot to explore the relationship between age and fare:

code

```
> plot(titanic$age, titanic$fare)
```

Output



Visualization:

Standard deviation measures the difference in a data set from the mean, with high deviation indicating wide data points and low deviation indicating closer points.

Here I create a histigram for age ,

For age

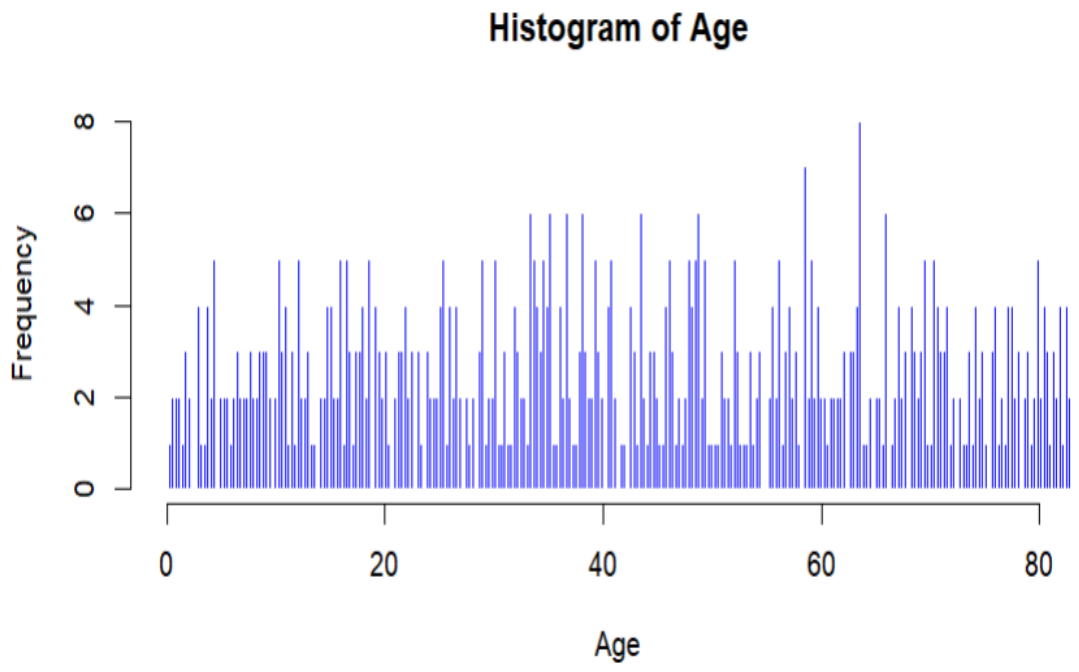
Code:

```
mean_val <- 33.32837
sd_val <- 45.7735
age_range <- c(0, 83, 455)


age_data <- runif(1000, min = age_range[1], max = age_range[2])


hist(age_data, breaks = age_range[3],
    main = "Histogram of Age",
    xlab = "Age", ylab = "Frequency",
    col = "blue", border = "white")
```

Output:



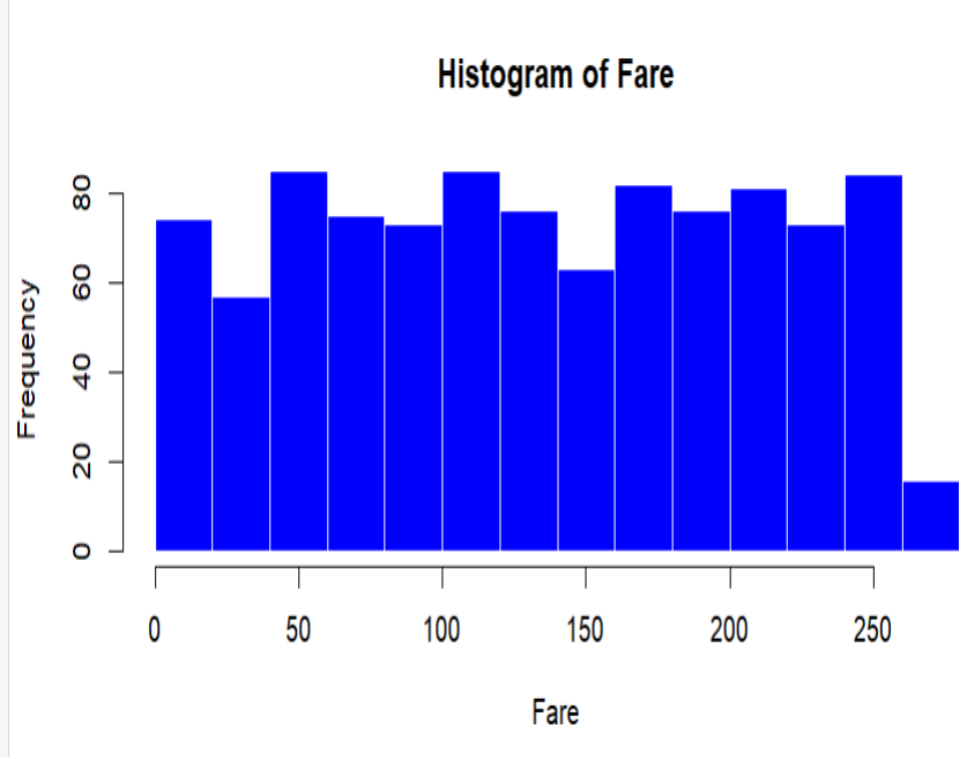
fare:
Code:

```
mean_val <- 26.58762
sd_val <- 34.82165
fare_range <- c(0, 263)

fare_data <- runif(1000, min = fare_range[1], max = fare_range[2])

hist(fare_data,
    main = "Histogram of Fare",
    xlab = "Fare", ylab = "Frequency",
    col = "blue", border = "white")
```

output:



For parch:

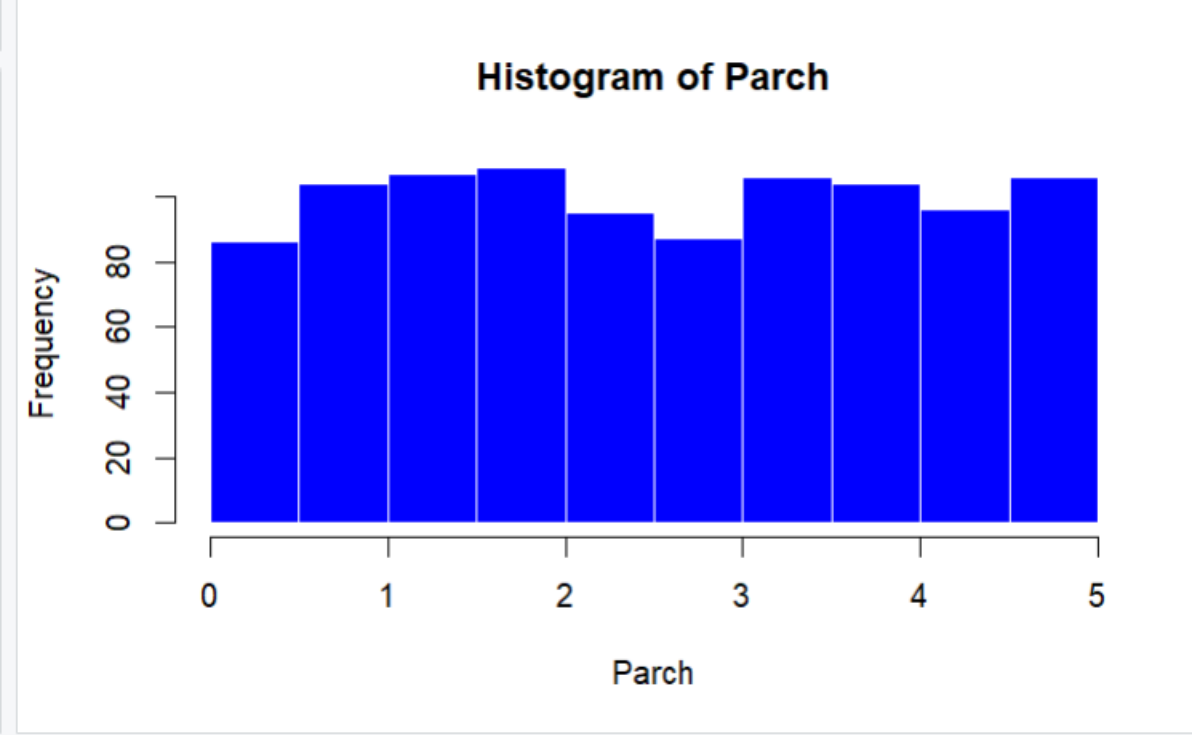
Code:

```
mean_val <- 0.392
sd_val <- 0.8252637
parch_range <- c(0, 5)

parch_data <- runif(1000, min = parch_range[1], max = parch_range[2])

hist(parch_data,
    main = "Histogram of Parch",
    xlab = "Parch", ylab = "Frequency",
    col = "blue", border = "white")
```

output:



For suevived:

Code:

```
mean_val <- 0.344
survived_range <- c(0, 1)

survived_data <- sample(survived_range, 1000, replace = TRUE, prob = c(1 - mean_val, mean_val))

hist(survived_data, breaks = c(survived_range, survived_range[2] + 1),
    main = "Histogram of Survived",
    xlab = "Survived", ylab = "Frequency",
    col = "blue", border = "white")
```

Output: