Lab 5: Data Preprocessing in R

Use following data for this exercise:

titanic_df<-read.csv("D:/Titanic.csv")

marks <- c(22,NA,45,30,NA,50,20)

rr1. Naming and renaming variables, adding a new variable.

1. Load titanic data in R environment and 1) Display first 5 rows 2) Display last 5 rows

```
> titanic_df<-read.csv("D:/Titanic.csv")
>
> titanic_df
   PassengerId Survived Pclass
1            1        0      3
2            2        1      1       Cumings, Mrs. Johı
3            3        1      3
4            4        1      1              Futrelle, ı
5            5        0      3
6            6        0      3
7            7        0      1
8            8        0      3
9            9        1      3         Johnson, Mrs. Oı
10          10        1      2                       Nı
> head(titanic_df,5)
  PassengerId Survived Pclass                                                Name    Sex Age SibSp
1           1        0      3                             Braund, Mr. Owen Harris   male  22     1
2           2        1      1 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1
3           3        1      3                              Heikkinen, Miss. Laina female  26     0
4           4        1      1        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1
5           5        0      3                            Allen, Mr. William Henry   male  35     0
  Parch           Ticket    Fare Cabin Embarked
1     0        A/5 21171  7.2500              S
2     0         PC 17599 71.2833   C85        C
3     0 STON/O2. 3101282  7.9250              S
4     0           113803 53.1000  C123        S
5     0           373450  8.0500              S
> tail(titanic_df,5)
    PassengerId Survived Pclass                                     Name    Sex Age SibSp Parch
887         887        0      2                   Montvila, Rev. Juozas   male  27     0     0
888         888        1      1            Graham, Miss. Margaret Edith female  19     0     0
889         889        0      3 Johnston, Miss. Catherine Helen "Carrie" female  NA     1     2
890         890        1      1                   Behr, Mr. Karl Howell   male  26     0     0
891         891        0      3                     Dooley, Mr. Patrick   male  32     0     0
        Ticket  Fare Cabin Embarked
887     211536 13.00              S
888     112053 30.00   B42        S
889 W./C. 6607 23.45              S
890     111369 30.00  C148        C
891     370376  7.75              Q
>
```

2. Display first 5 columns of titanic dataset.

```
> df<-titanic_df
> df[,1:5]
   PassengerId Survived Pclass                                                Name    Sex
1            1        0      3                             Braund, Mr. Owen Harris   male
2            2        1      1 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female
3            3        1      3                              Heikkinen, Miss. Laina female
4            4        1      1        Futrelle, Mrs. Jacques Heath (Lily May Peel) female
5            5        0      3                            Allen, Mr. William Henry   male
6            6        0      3                                    Moran, Mr. James   male
7            7        0      1                             McCarthy, Mr. Timothy J   male
8            8        0      3                      Palsson, Master. Gosta Leonard   male
9            9        1      3   Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg) female
10          10        1      2                 Nasser, Mrs. Nicholas (Adele Achem) female
11          11        1      3                     Sandstrom, Miss. Marguerite Rut female
12          12        1      1                            Bonnell, Miss. Elizabeth female
13          13        0      3                      Saundercock, Mr. William Henry   male
```

3. Rename the column Embarked with name Location of titanic dataframe.

```
> New_dataframe<-rename(titanic_df, "location"="Embarked")
> New_dataframe
   PassengerId Survived Pclass                                                Name    Sex
1            1        0      3                                             Braun
2            2        1      1                 Cumings, Mrs. John Bradley (Flore
3            3        1      3                                              Heik
4            4        1      1                        Futrelle, Mrs. Jacques Hea
5            5        0      3                                            Allen,
83          83        1      3                                          McDermott, Miss.
     Age SibSp Parch           Ticket     Fare       Cabin location
1  22.00     1     0        A/5 21171   7.2500                   S
2  38.00     1     0         PC 17599  71.2833         C85       C
3  26.00     0     0 STON/O2. 3101282   7.9250                   S
4  35.00     1     0           113803  53.1000        C123       S
5  35.00     0     0           373450   8.0500                   S
6     NA     0     0           330877   8.4583                   Q
```

4. Load titanic data with user defined column name.

```
> titanic_df = read.csv(file="d:/titanic.csv", col.names=c("passengers", "survied","Pclass","Name"))
> titanic_df
   passengers survied Pclass                                                Name
1           1       0      3                             Braund, Mr. Owen Harris
2           2       1      1 Cumings, Mrs. John Bradley (Florence Briggs Thayer)
3           3       1      3                              Heikkinen, Miss. Laina
4           4       1      1        Futrelle, Mrs. Jacques Heath (Lily May Peel)
5           5       0      3                            Allen, Mr. William Henry
6           6       0      3                                    Moran, Mr. James
```

5. Load first 5 column data in dataframe titanic1 and rest of the columns in titanic2 and merge this two dataframe in titanic3.

```
> titanicdf3<-merge(titanic_df1,titanic_df2)
> titanicdf3
   PassengerId Parch           Ticket     Fare       Cabin Embarked Survived Pclass
1            1     0        A/5 21171   7.2500                   S        0      3
2            2     0         PC 17599  71.2833         C85       C        0      3
3            3     0 STON/O2. 3101282   7.9250                   S        0      3
4            4     0           113803  53.1000        C123       S        0      3
5            5     0           373450   8.0500                   S        0      3
6            6     0           330877   8.4583                   Q        0      3
7            7     0            17463  51.8625         E46       S        0      3
8            8     1           349909  21.0750                   S        0      3
9            9     2           347742  11.1333                   S        0      3
10          10     0           237736  30.0708                   C        0      3
11          11     1          PP 9549  16.7000          G6       S        0      3
12          12     0           113783  26.5500        C103       S        0      3
13          13     0        A/5. 2151   8.0500                   S        0      3
```

2. Dealing with Missing Data

1. Missing data are represented by NA values in R, and so we wish to check how many NA elements there are in the marks vector. Also calculate how many non NA elements are there in the vector.

```
> marks
[1] 22 NA 45 30 NA 50 20
> sum(is.na(marks))
[1] 2
> sum(!is.na(marks))
[1] 5
>
```

2. Display vector marks with values that are not NA.

```
> marks[! d]
[1] 22 45 30 50 20
>
```

3. Calculate mean and median of given marks vector.

```
> mean(marks,na.rm=T)
[1] 33.4
> median(marks,na.rm=T)
[1] 30
>
```

4. Check the complete case of titanic dataframe – (Where no NA in column values)

```
> titanic_df[complete.cases(titanic_df),]
    passengers survied Pclass
1            1       0      3
2            2       1      1       Cumings, Mrs. Jc
3            3       1      3

250        250       0      2                              Carter, Rev. Ernest Cou
 [ reached 'max' / getOption("max.print") -- omitted 641 rows ]
>
```

5. Check the total missing values of cabin column of titanic dataframe without using function complete.cases function.

```
> is.na(titanic_df$cabin)
  [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
 [31] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
 [61] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
 [91] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[121] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[151] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[181] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[211] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[241] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[271] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[301] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[331] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[361] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[391] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[421] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[451] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[481] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[511] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[541] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[571] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[601] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[631] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[661] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
[691] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FAL
```

6. Replace missing value of age column with

   1) mean

```
> df1<-titanic_df$Age
> df1
  [1] 22.00 38.00 26.00 35.00 35.00    NA 54.00  2.00 27.00 14.00  4.00 58.00 20.00 39.00 14.00 55.00  2
 [31] 40.00    NA    NA 66.00 28.00 42.00    NA 21.00 18.00 14.00 40.00 27.00    NA  3.00 19.00    NA
 [61] 22.00 38.00 45.00  4.00    NA    NA 29.00 19.00 17.00 26.00 32.00 16.00 21.00 26.00 32.00 25.00
 [91] 29.00 20.00 46.00 26.00 59.00    NA 71.00 23.00 34.00 34.00 28.00    NA 21.00 33.00 37.00 28.00 21
[121] 21.00    NA 32.50 32.50 54.00 12.00    NA 24.00    NA 45.00 33.00 20.00 47.00 29.00 25.00 23.00 19
[151] 51.00 22.00 55.50 40.50    NA 51.00 16.00 30.00    NA    NA 44.00 40.00 26.00 17.00  1.00  9.00
[181]    NA    NA  9.00  1.00  4.00    NA    NA 45.00 40.00 36.00 32.00 19.00 19.00  3.00 44.00 58.00
[211] 24.00 35.00 22.00 30.00    NA 31.00 27.00 42.00 32.00 30.00 16.00 27.00 51.00    NA 38.00 22.00 19
[241]    NA    NA 29.00 22.00 30.00 44.00 25.00 24.00 37.00 54.00    NA 29.00 62.00 30.00 41.00 29.00
[271]    NA 25.00 41.00 37.00    NA 63.00 45.00    NA  7.00 35.00 65.00 28.00 16.00 19.00    NA 33.00 30
[301]    NA    NA 19.00    NA    NA  0.92    NA 17.00 30.00 30.00 24.00 18.00 26.00 28.00 43.00 26.00 24
[331]    NA 45.50 38.00 16.00    NA    NA 29.00 41.00 45.00 45.00  2.00 24.00 28.00 25.00 36.00 24.00 40
[361] 40.00 29.00 45.00 35.00    NA 30.00 60.00    NA    NA 24.00 25.00 18.00 19.00 22.00  3.00    NA 22
[391] 36.00 21.00 28.00 23.00 24.00 22.00 31.00 46.00 23.00 28.00 39.00 26.00 21.00 28.00 20.00 34.00 51
[421]    NA 21.00 29.00 28.00 18.00    NA 28.00 19.00    NA 32.00 28.00    NA 42.00 17.00 50.00 14.00 21
[451] 36.00    NA 30.00 49.00    NA 29.00 65.00    NA 50.00    NA 48.00 34.00 47.00 48.00    NA 38.00
[481]  9.00    NA 50.00 63.00 25.00    NA 35.00 58.00 30.00  9.00    NA 21.00 55.00 71.00 21.00    NA 54
[511] 29.00    NA 36.00 54.00 24.00 47.00 34.00    NA 36.00 32.00 30.00 22.00    NA 44.00    NA 40.50 50
[541] 36.00  9.00 11.00 32.00 50.00 64.00 19.00    NA 33.00  8.00 17.00 27.00    NA 22.00 22.00 62.00 48
[571] 62.00 53.00 36.00    NA 16.00 19.00 34.00 39.00    NA 32.00 25.00 39.00 54.00 36.00    NA 18.00 47
[601] 24.00    NA    NA 44.00 35.00 36.00 30.00 27.00 22.00 40.00 39.00    NA    NA    NA 35.00 24.00 34
```

```
> impute(df1,fun=mean)
       1        2        3        4        5        6        7
22.00000 38.00000 26.00000 35.00000 35.00000 29.69912* 54.00000
      19       20       21       22       23       24       25
31.00000 29.69912* 35.00000 34.00000 15.00000 28.00000  8.00000
      37       38       39       40       41       42       43
29.69912* 21.00000 18.00000 14.00000 40.00000 27.00000 29.69912*
      55       56       57       58       59       60       61
65.00000 29.69912* 21.00000 28.50000  5.00000 11.00000 22.00000
      73       74       75       76       77       78       79
21.00000 26.00000 32.00000 25.00000 29.69912* 29.69912*  0.83000
      91       92       93       94       95       96       97
```

   ii) median

```
> impute(df1,fun=median)
     1      2      3      4      5      6      7      8      9     10
 22.00  38.00  26.00  35.00  35.00 28.00*  54.00   2.00  27.00  14.00
    27     28     29     30     31     32     33     34     35     36
28.00*  19.00 28.00* 28.00*  40.00 28.00* 28.00*  66.00  28.00  42.00 2
    53     54     55     56     57     58     59     60     61     62
 49.00  29.00  65.00 28.00*  21.00  28.50   5.00  11.00  22.00  38.00
    79     80     81     82     83     84     85     86     87     88
  0.83  30.00  22.00  29.00 28.00*  28.00  17.00  33.00  16.00 28.00*
   105    106    107    108    109    110    111    112    113    114
 37.00  28.00  21.00 28.00*  38.00 28.00*  47.00  14.50  22.00  20.00
   131    132    133    134    135    136    137    138    139    140
 33.00  20.00  47.00  29.00  25.00  23.00  19.00  37.00  16.00  24.00 2
   157    158    159    160    161    162    163    164    165    166
 16.00  30.00 28.00* 28.00*  44.00  40.00  26.00  17.00   1.00   9.00 2
   183    184    185    186    187    188    189    190    191    192
  9.00   1.00   4.00 28.00* 28.00*  45.00  40.00  36.00  32.00  19.00
   209    210    211    212    213    214    215    216    217    218
 16.00  40.00  24.00  35.00  22.00  30.00 28.00*  31.00  27.00  42.00
```

iii)mode

```
> fac<-factor(titanic_df$age)
> fac<-factor(titanic_df$Age)
> impute(fac,fun=mode)
   1    2    3    4    5    6    7    8    9   10   11   12   13
  22   38   26   35   35  24*   54    2   27   14    4   58   20
  38   39   40   41   42   43   44   45   46   47   48   49   50
  21   18   14   40   27  24*    3   19  24*  24*  24*  24*   18
  75   76   77   78   79   80   81   82   83   84   85   86   87
  32   25  24*  24* 0.83   30   22   29  24*   28   17   33   16  2
 112  113  114  115  116  117  118  119  120  121  122  123  124  1
14.5   22   20   17   21 70.5   29   24    2   21  24* 32.5 32.5
 149  150  151  152  153  154  155  156  157  158  159  160  161  1
36.5   42   51   22 55.5 40.5  24*   51   16   30  24*  24*   44
```

3. Dealing with categorical data.
   1. Create category Nationality vector ("Indian", "Chinese", "Indian", "Chinese", "Indian", "Indian") and Mark vector (50, 44, 51, 32, 40, 41)

```
> Nationality<-c("Indian", "Chinese", "Indian", "Chinese", "Indian",  "Indian")
> Nationality
[1] "Indian"  "Chinese" "Indian"  "Chinese" "Indian"  "Indian"
> Mark<-c(50, 44, 51, 32, 40, 41)
>
> Mark
[1] 50 44 51 32 40 41
>
```

   2. Check the class of nationality vector and convert it into factor

```
> fac<-factor(Nationality)
> fac
[1] Indian  Chinese Indian  Chinese Indian  Indian
Levels: Chinese Indian
>
```

3. Display Category wise average Mark using above vector data Nationality and Mark (Hint: tapply function).

```
> tapply(Mark,fac,mean)
Chinese  Indian
   38.0    45.5
>
```