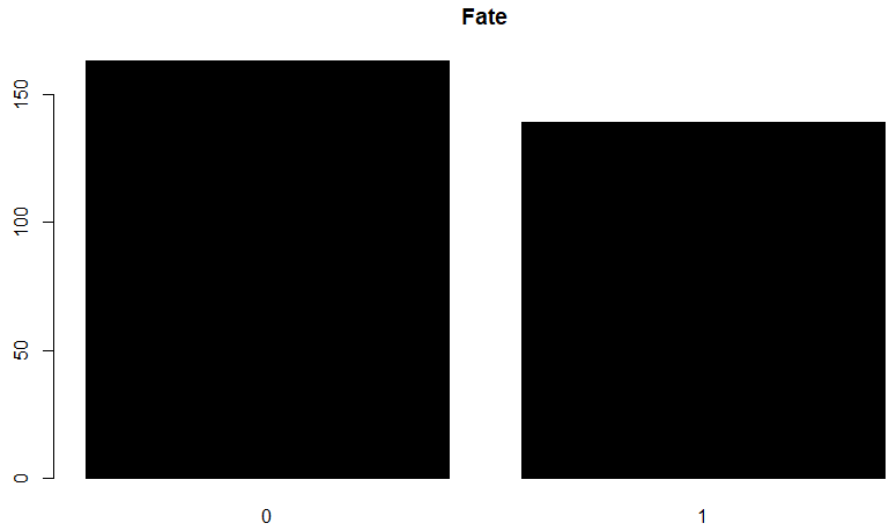
```
# Roll no. 33140
# Batch: L9
# P.S.: Application of Linear regression on Heart disease dataset to predict
        the fate (prob. of heart disease)

R version 3.6.2 (2019-12-12) -- "Dark and Stormy Night"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from G:/College/SL6/Assignment6/.RData]

> # Set working directory

> setwd("G:/College/SL6/Assignment6/")

> # Read the CSV file and analyse

> hdata <- read.csv("../../Sl-VI DataSets/HeartDisease/Cleavland.csv",header=TRUE,
                    sep=",")
> names(hdata)

 [1] "X63.0"  "X1.0"   "X1.0.1" "X145.0" "X233.0" "X1.0.2" "X2.0"   "X150.0" "X0.0"
 "X2.3"   "X3.0"   "X0.0.1" "X6.0"
[14] "X0"

> str(hdata)
'data.frame':   302 obs. of  14 variables:
 $ X63.0 : num  67 67 37 41 56 62 57 63 53 57 ...
 $ X1.0  : num  1 1 1 0 1 0 0 1 1 1 ...
 $ X1.0.1: num  4 4 3 2 2 4 4 4 4 4 ...
 $ X145.0: num  160 120 130 130 120 140 120 130 140 140 ...
 $ X233.0: num  286 229 250 204 236 268 354 254 203 192 ...
 $ X1.0.2: num  0 0 0 0 0 0 0 0 1 0 ...
 $ X2.0  : num  2 2 0 2 0 2 0 2 2 0 ...
 $ X150.0: num  108 129 187 172 178 160 163 147 155 148 ...
 $ X0.0  : num  1 1 0 0 0 0 1 0 1 0 ...
 $ X2.3  : num  1.5 2.6 3.5 1.4 0.8 3.6 0.6 1.4 3.1 0.4 ...
 $ X3.0  : num  2 2 3 1 1 3 1 2 3 2 ...
 $ X0.0.1: Factor w/ 5 levels "?","0.0","1.0",..: 5 4 2 2 2 4 2 3 2 2 ...
 $ X6.0  : Factor w/ 4 levels "?","3.0","6.0",..: 2 4 2 2 2 2 2 4 4 3 ...
 $ X0    : int  2 1 0 0 0 3 0 2 1 0 ...

> dim(hdata)
[1] 302  14

> # Change the headers
```

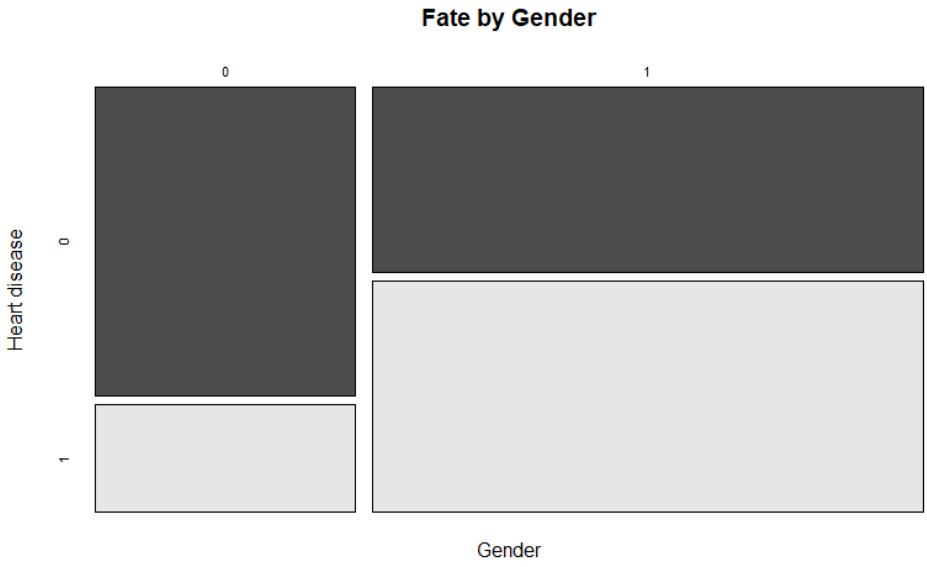
```
> names(hdata)[1] <- "age"
> names(hdata)[2] <- "sex"
> names(hdata)[3] <- "cp"
> names(hdata)[4] <- "trestbps"
> names(hdata)[5] <- "chol"
> names(hdata)[6] <- "fbs"
> names(hdata)[7] <- "restecg"
> names(hdata)[8] <- "thalach"
> names(hdata)[9] <- "exang"
> names(hdata)[10] <- "oldpeak"
> names(hdata)[11] <- "slope"
> names(hdata)[12] <- "ca"
> names(hdata)[13] <- "thal"
> names(hdata)[14] <- "num"
> hdata$ca
  [1] 3.0 2.0 0.0 0.0 0.0 2.0 0.0 1.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 2.0 2.0 0.0 0.0 0.0 0.0 0.0
 [30] 2.0 2.0 0.0 0.0 0.0 0.0 0.0 1.0 1.0 0.0 3.0 0.0 2.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0
      1.0 0.0 1.0 0.0 1.0 1.0 1.0 0.0 1.0
 [59] 1.0 0.0 0.0 3.0 0.0 1.0 2.0 0.0 0.0 0.0 0.0 0.0 2.0 2.0 2.0 1.0 0.0 1.0 1.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
 [88] 0.0 0.0 0.0 3.0 3.0 0.0 0.0 1.0 1.0 2.0 1.0 0.0 0.0 0.0 1.0 1.0 3.0 0.0 1.0 1.0
      1.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0 0.0
[117] 0.0 3.0 1.0 2.0 3.0 0.0 0.0 1.0 0.0 2.0 1.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0
      1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
[146] 3.0 0.0 0.0 1.0 0.0 0.0 0.0 1.0 1.0 3.0 0.0 2.0 2.0 1.0 0.0 3.0 0.0 0.0 2.0 0.0
      1.0 1.0 0.0 0.0 1.0 0.0 0.0 0.0 2.0
[175] 1.0 3.0 1.0 1.0 3.0 0.0 2.0 2.0 0.0 0.0 2.0 0.0 3.0 1.0 3.0 0.0 3.0 2.0 3.0 0.0
      2.0 1.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0
[204] 0.0 3.0 2.0 0.0 0.0 0.0 0.0 0.0 0.0 2.0 1.0 0.0 0.0 0.0 2.0 0.0 0.0 0.0 0.0 2.0
      2.0 0.0 0.0 1.0 1.0 1.0 0.0 0.0 3.0
[233] 1.0 1.0 2.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 2.0 0.0 0.0 1.0 1.0 2.0 0.0 0.0 1.0 1.0
      0.0 0.0 0.0 2.0 0.0 0.0 0.0 1.0 2.0
[262] 0.0 0.0 1.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0 2.0 0.0 2.0 0.0 1.0 0.0 1.0 0.0 1.0 0.0
      1.0 0.0 1.0 3.0 2.0 ?   0.0 0.0 0.0
[291] 0.0 0.0 2.0 0.0 0.0 2.0 0.0 0.0 2.0 1.0 1.0 ?

Levels: ? 0.0 1.0 2.0 3.0

> levels(hdata$ca)[levels(hdata$ca) == "?"]<-"0.0"

> hdata
   age sex cp trestbps chol fbs restecg thalach exang oldpeak slope  ca thal num
1   67   1  4      160  286   0       2     108     1     1.5     2 3.0  3.0   2
2   67   1  4      120  229   0       2     129     1     2.6     2 2.0  7.0   1
3   37   1  3      130  250   0       0     187     0     3.5     3 0.0  3.0   0
4   41   0  2      130  204   0       2     172     0     1.4     1 0.0  3.0   0
5   56   1  2      120  236   0       0     178     0     0.8     1 0.0  3.0   0
6   62   0  4      140  268   0       2     160     0     3.6     3 2.0  3.0   3
7   57   0  4      120  354   0       0     163     1     0.6     1 0.0  3.0   0
```

8	63	1	4	130	254	0	2	147	0	1.4	2	1.0	7.0	2
9	53	1	4	140	203	1	2	155	1	3.1	3	0.0	7.0	1
10	57	1	4	140	192	0	0	148	0	0.4	2	0.0	6.0	0
11	56	0	2	140	294	0	2	153	0	1.3	2	0.0	3.0	0
12	56	1	3	130	256	1	2	142	1	0.6	2	1.0	6.0	2
13	44	1	2	120	263	0	0	173	0	0.0	1	0.0	7.0	0
14	52	1	3	172	199	1	0	162	0	0.5	1	0.0	7.0	0
15	57	1	3	150	168	0	0	174	0	1.6	1	0.0	3.0	0
16	48	1	2	110	229	0	0	168	0	1.0	3	0.0	7.0	1
17	54	1	4	140	239	0	0	160	0	1.2	1	0.0	3.0	0
18	48	0	3	130	275	0	0	139	0	0.2	1	0.0	3.0	0
19	49	1	2	130	266	0	0	171	0	0.6	1	0.0	3.0	0
20	64	1	1	110	211	0	2	144	1	1.8	2	0.0	3.0	0
21	58	0	1	150	283	1	2	162	0	1.0	1	0.0	3.0	0
22	58	1	2	120	284	0	2	160	0	1.8	2	0.0	3.0	1
23	58	1	3	132	224	0	2	173	0	3.2	1	2.0	7.0	3
24	60	1	4	130	206	0	2	132	1	2.4	2	2.0	7.0	4
25	50	0	3	120	219	0	0	158	0	1.6	2	0.0	3.0	0
26	58	0	3	120	340	0	0	172	0	0.0	1	0.0	3.0	0
27	66	0	1	150	226	0	0	114	0	2.6	3	0.0	3.0	0
28	43	1	4	150	247	0	0	171	0	1.5	1	0.0	3.0	0
29	40	1	4	110	167	0	2	114	1	2.0	2	0.0	7.0	3
30	69	0	1	140	239	0	0	151	0	1.8	1	2.0	3.0	0
31	60	1	4	117	230	1	0	160	1	1.4	1	2.0	7.0	2
32	64	1	3	140	335	0	0	158	0	0.0	1	0.0	3.0	1
33	59	1	4	135	234	0	0	161	0	0.5	2	0.0	7.0	0
34	44	1	3	130	233	0	0	179	1	0.4	1	0.0	3.0	0
35	42	1	4	140	226	0	0	178	0	0.0	1	0.0	3.0	0
36	43	1	4	120	177	0	2	120	1	2.5	2	0.0	7.0	3
37	57	1	4	150	276	0	2	112	1	0.6	2	1.0	6.0	1
38	55	1	4	132	353	0	0	132	1	1.2	2	1.0	7.0	3
39	61	1	3	150	243	1	0	137	1	1.0	2	0.0	3.0	0
40	65	0	4	150	225	0	2	114	0	1.0	2	3.0	7.0	4
41	40	1	1	140	199	0	0	178	1	1.4	1	0.0	7.0	0
42	71	0	2	160	302	0	0	162	0	0.4	1	2.0	3.0	0
43	59	1	3	150	212	1	0	157	0	1.6	1	0.0	3.0	0
44	61	0	4	130	330	0	2	169	0	0.0	1	0.0	3.0	1
45	58	1	3	112	230	0	2	165	0	2.5	2	1.0	7.0	4
46	51	1	3	110	175	0	0	123	0	0.6	1	0.0	3.0	0
47	50	1	4	150	243	0	2	128	0	2.6	2	0.0	7.0	4
48	65	0	3	140	417	1	2	157	0	0.8	1	1.0	3.0	0
49	53	1	3	130	197	1	2	152	0	1.2	3	0.0	3.0	0
50	41	0	2	105	198	0	0	168	0	0.0	1	1.0	3.0	0
51	65	1	4	120	177	0	0	140	0	0.4	1	0.0	7.0	0
52	44	1	4	112	290	0	2	153	0	0.0	1	1.0	3.0	2
53	44	1	2	130	219	0	2	188	0	0.0	1	0.0	3.0	0
54	60	1	4	130	253	0	0	144	1	1.4	1	1.0	7.0	1
55	54	1	4	124	266	0	2	109	1	2.2	2	1.0	7.0	1
56	50	1	3	140	233	0	0	163	0	0.6	2	1.0	7.0	1
57	41	1	4	110	172	0	2	158	0	0.0	1	0.0	7.0	1
58	54	1	3	125	273	0	2	152	0	0.5	3	1.0	3.0	0
59	51	1	1	125	213	0	2	125	1	1.4	1	1.0	3.0	0
60	51	0	4	130	305	0	0	142	1	1.2	2	0.0	7.0	2
61	46	0	3	142	177	0	2	160	1	1.4	3	0.0	3.0	0
62	58	1	4	128	216	0	2	131	1	2.2	2	3.0	7.0	1
63	54	0	3	135	304	1	0	170	0	0.0	1	0.0	3.0	0
64	54	1	4	120	188	0	0	113	0	1.4	2	1.0	7.0	2
65	60	1	4	145	282	0	2	142	1	2.8	2	2.0	7.0	2
66	60	1	3	140	185	0	2	155	0	3.0	2	0.0	3.0	1
67	54	1	3	150	232	0	2	165	0	1.6	1	0.0	7.0	0
68	59	1	4	170	326	0	2	140	1	3.4	3	0.0	7.0	2
69	46	1	3	150	231	0	0	147	0	3.6	2	0.0	3.0	1
70	65	0	3	155	269	0	0	148	0	0.8	1	0.0	3.0	0
71	67	1	4	125	254	1	0	163	0	0.2	2	2.0	7.0	3

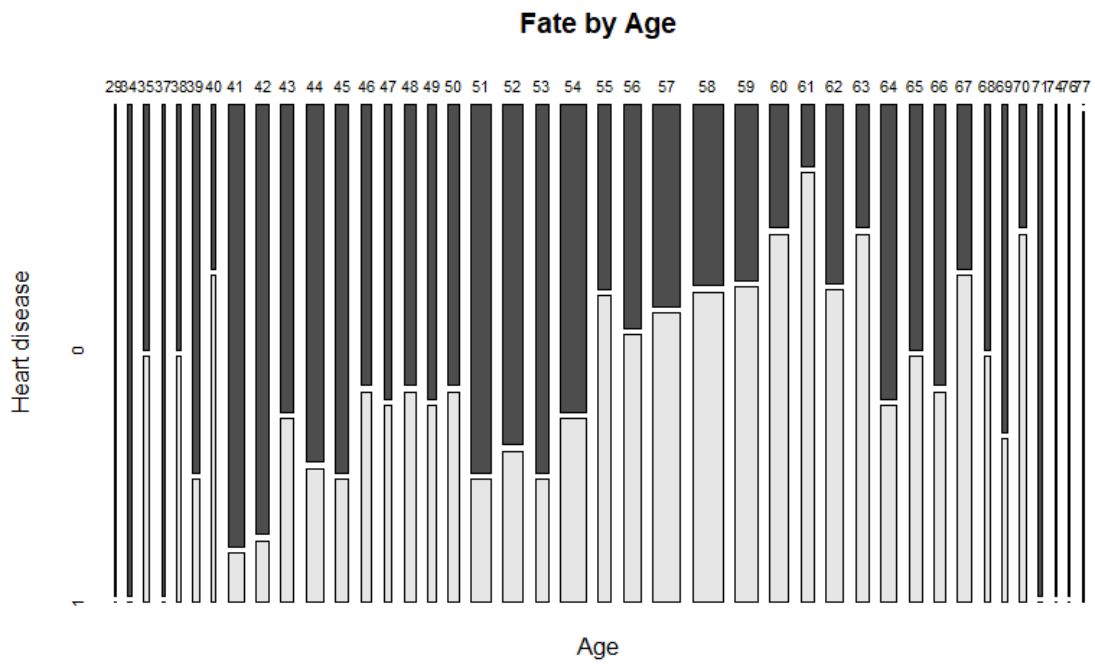
```
 [ reached 'max' / getOption("max.print") -- omitted 231 rows ]

> hdata$ca[hdata$ca == 1.0]
factor(0)
Levels: 0.0 1.0 2.0 3.0

> typeof(hdata$ca)
[1] "integer"

> nrow(hdata)
[1] 302

> # Plotting Fate vs number of records

> hdata$num[hdata$num >= 1] <- 1 # Edit the fate to 0 and 1

> barplot(table(hdata$num), main="Fate", col="black")
```




```
> # Plot Fate vs gender

> mosaicplot(hdata$sex ~ hdata$num,main="Fate by Gender",
+            shade=FALSE,color=TRUE,xlab="Gender", ylab="Heart disease")
```




```
> # Plot Fate vs Age
> mosaicplot(hdata$age ~ hdata$num,main="Fate by Age",
+            shade=FALSE,color=TRUE,xlab="Age", ylab="Heart disease")
```




```
> # Most important step, change the values of NA
> levels(hdata$thal)[levels(hdata$thal)=="?"]<-"3.0"
> # removal of additional NA
```

```
> hdata$thal

  [1] 3.0 7.0 3.0 3.0 3.0 3.0 3.0 7.0 7.0 6.0 3.0 6.0 7.0 7.0 3.0 7.0 3.0 3.0 3.0 3.0
      3.0 3.0 7.0 7.0 3.0 3.0 3.0 3.0 7.0
 [30] 3.0 7.0 3.0 7.0 3.0 3.0 7.0 6.0 7.0 3.0 7.0 7.0 3.0 3.0 3.0 7.0 3.0 7.0 3.0 3.0
      3.0 7.0 3.0 3.0 7.0 7.0 7.0 7.0 3.0
 [59] 3.0 7.0 3.0 7.0 3.0 7.0 7.0 3.0 7.0 7.0 3.0 3.0 7.0 7.0 6.0 3.0 3.0 7.0 3.0 3.0
      7.0 3.0 3.0 3.0 7.0 3.0 3.0 3.0 3.0
 [88] 3.0 3.0 3.0 7.0 7.0 3.0 3.0 7.0 7.0 7.0 3.0 3.0 3.0 3.0 3.0 3.0 7.0 7.0 7.0 7.0
      7.0 7.0 7.0 3.0 6.0 7.0 7.0 6.0 3.0
[117] 3.0 7.0 7.0 7.0 7.0 3.0 7.0 3.0 3.0 7.0 7.0 3.0 3.0 7.0 7.0 3.0 3.0 3.0 3.0 7.0
      7.0 7.0 3.0 3.0 7.0 3.0 7.0 7.0 3.0
[146] 7.0 3.0 3.0 3.0 7.0 3.0 7.0 7.0 3.0 3.0 7.0 7.0 7.0 7.0 7.0 3.0 3.0 3.0 3.0 7.0
      3.0 3.0 7.0 3.0 7.0 7.0 3.0 3.0 6.0
[175] 7.0 7.0 6.0 3.0 3.0 7.0 7.0 3.0 7.0 3.0 3.0 7.0 6.0 7.0 7.0 3.0 7.0 7.0 3.0 3.0
      3.0 3.0 3.0 3.0 3.0 3.0 3.0 7.0 7.0
[204] 7.0 7.0 7.0 7.0 3.0 3.0 3.0 7.0 3.0 7.0 3.0 7.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 7.0
      3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0
[233] 3.0 3.0 3.0 7.0 7.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 7.0 3.0 7.0 3.0 6.0 7.0 7.0
      3.0 3.0 3.0 3.0 3.0 3.0 7.0 3.0 3.0
[262] 3.0 3.0 3.0 6.0 3.0 6.0 7.0 3.0 7.0 6.0 7.0 3.0 3.0 7.0 3.0 3.0 3.0 3.0 7.0 3.0
      7.0 3.0 7.0 6.0 6.0 7.0 7.0 3.0 7.0
[291] 3.0 6.0 7.0 3.0 3.0 6.0 7.0 7.0 7.0 7.0 3.0 3.0

Levels: 3.0 6.0 7.0

> table(hdata$thal)

3.0 6.0 7.0
168  17 117

> table(hdata$ca)

0.0 1.0 2.0 3.0
179  65  38  20

> library(caTools) # import library caTools
Warning message:
package 'caTools' was built under R version 3.6.3

> n<- sapply(hdata[, c(1)], mean) # get the average values
> set.seed(123) # generate a pseudo-random number

> v3 <- hdata[c(11:14),c(2,7:9)]

> v3
   sex restecg thalach exang
11   0       2     153     0
12   1       2     142     1
13   1       0     173     0
14   1       0     162     0

> m<- sapply(v3,max)

> m
    sex restecg thalach   exang
      1       2     173       1

> set.seed(121)

> # Divide the dataset into 2/3 for training, and 1/3 for testing

> split = sample.split(hdata$num, SplitRatio = 2/3)
```

```
> train_hdata = subset(hdata, split == TRUE)

> test_hdata = subset(hdata, split == FALSE)

> # Apply linear regression for Fate vs age

> regressor=lm(formula = num~age, data=train_hdata)

> View(regressor)

> regressor

Call:
lm(formula = num ~ age, data = train_hdata)

Coefficients:
(Intercept)          age
   -0.33038      0.01453

> # Apply regression on test data
> hd_age_predict = predict(regressor, newdata=test_hdata)

> hd_age_predict
         2          4          7          8         10         18         19         20
 25         29         30         31
0.6430055 0.2652722 0.4977234 0.5848927 0.4977234 0.3669696 0.3814978 0.5994209
0.3960260 0.2507440 0.6720619 0.5413081
        33         34         42         43         45         47         52         59
61         65         68         70
0.5267799 0.3088568 0.7011183 0.5267799 0.5122517 0.3960260 0.3088568 0.4105542
0.3379132 0.5413081 0.5267799 0.6139491
        73         75         78         79         83         84         85         86
       108        109        110        111
0.6139491 0.6139491 0.3669696 0.5122517 0.6575337 0.4250824 0.3088568 0.3524414
0.5558363 0.2362158 0.5558363 0.4831952
       115        118        122        129        134        136        137        139
       141        144        147        149
0.2652722 0.5848927 0.4105542 0.5703645 0.2943286 0.6865901 0.5703645 0.4105542
0.5267799 0.5122517 0.2652722 0.5413081
       155        156        158        168        169        173        175        176
       179        184        185        186
0.6865901 0.4105542 0.5413081 0.1781030 0.3233850 0.5703645 0.4977234 0.4250824
0.4396106 0.5413081 0.5848927 0.2798004
       187        192        194        197        201        204        206        207
       214        217        219        220
0.6284773 0.2943286 0.6575337 0.3233850 0.5994209 0.2943286 0.5122517 0.3960260
0.4250824 0.3379132 0.5267799 0.2652722
       225        226        229        230        232        236        237        240
       241        247        248        252
0.1635748 0.3524414 0.6284773 0.4250824 0.3814978 0.4831952 0.3379132 0.2652722
0.2652722 0.3524414 0.4250824 0.5994209
       255        258        262        265        267        273        276        282
       284        285        288        292
0.2798004 0.6865901 0.5413081 0.2798004 0.5267799 0.7011183 0.6284773 0.4686670
0.5558363 0.5122517 0.4831952 0.3088568
       293        296        301        302
0.5848927 0.5267799 0.4977234 0.2216876
> # Round the values of fate in prediction
> round_age=hd_age_predict

> r=round(round_age)

> r
```

```
  2   4   7   8  10  18  19  20  25  29  30  31  33  34  42  43  45  47  52  59  61
65  68  70  73  75  78  79  83  84
  1   0   0   1   0   0   0   1   0   0   1   1   1   0   1   1   1   0   0   0   0
1   1   1   1   1   0   1   1   0
 85  86 108 109 110 111 115 118 122 129 134 136 137 139 141 144 147 149 155 156 158
168 169 173 175 176 179 184 185 186
  0   0   1   0   1   0   0   1   0   1   0   1   1   0   1   1   0   1   1   0   1
0   0   1   0   0   0   1   1   0
187 192 194 197 201 204 206 207 214 217 219 220 225 226 229 230 232 236 237 240 241
247 248 252 255 258 262 265 267 273
  1   0   1   0   1   0   1   0   0   0   1   0   0   0   1   0   0   0   0   0   0
0   0   1   0   1   1   0   1   1
276 282 284 285 288 292 293 296 301 302
  1   0   1   1   0   0   1   1   0   0
```

```
> table(r,test_hdata$num)

r    0  1
  0 34 20
  1 20 26
```

```
> library(e1071)
```

```
> library(caret)
```

```
> typeof(r)
[1] "double"
```

```
> levels(r)
NULL
```

```
> levels(test_hdata$num)
NULL
```

```
> str(r)
 Named num [1:100] 1 0 0 1 0 0 0 1 0 0 ...
 - attr(*, "names")= chr [1:100] "2" "4" "7" "8" ...
```

```
> r1 = as.data.frame(r)
```

```
> r1
    r
2   1
4   0
7   0
8   1
10  0
18  0
19  0
20  1
25  0
29  0
30  1
31  1
33  1
34  0
42  1
43  1
45  1
47  0
52  0
59  0
```

61 0
65 1
68 1
70 1
73 1
75 1
78 0
79 1
83 1
84 0
85 0
86 0
108 1
109 0
110 1
111 0
115 0
118 1
122 0
129 1
134 0
136 1
137 1
139 0
141 1
144 1
147 0
149 1
155 1
156 0
158 1
168 0
169 0
173 1
175 0
176 0
179 0
184 1
185 1
186 0
187 1
192 0
194 1
197 0
201 1
204 0
206 1
207 0
214 0
217 0
219 1
220 0
225 0
226 0
229 1
230 0
232 0
236 0
237 0
240 0
241 0
247 0
248 0
252 1

```
255 0
258 1
262 1
265 0
267 1
273 1
276 1
282 0
284 1
285 1
288 0
292 0
293 1
296 1
301 0
302 0
> df1=confusionMatrix(as.factor(r1$r),as.factor(test_hdata$num))




> df1
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 34 20
         1 20 26

               Accuracy : 0.6
                 95% CI : (0.4972, 0.6967)
    No Information Rate : 0.54
    P-Value [Acc > NIR] : 0.1347

                  Kappa : 0.1948

 Mcnemar's Test P-Value : 1.0000

            Sensitivity : 0.6296
            Specificity : 0.5652
         Pos Pred Value : 0.6296
         Neg Pred Value : 0.5652
             Prevalence : 0.5400
         Detection Rate : 0.3400
   Detection Prevalence : 0.5400
      Balanced Accuracy : 0.5974

       'Positive' Class : 0


>
```