Question 1.

I. Read the data from “people.txt”.

```
> people<-read.csv('people.csv')
> people
  Age agegroup height  status yearsmarried
1  21    adult    6.0  single           -1
2   2    child    3.0 married            0
3  18    adult    5.7 married           20
4 221  elderly    5.0 widowed            2
5  34    child   -7.0 married            3
```

II. Create a ruleset E that contain rules to check for the following conditions:

a) The age should be in the range 0-150.

```
> E1<- editset(c("Age >=0","Age<=150"))
> E1

Edit set:
num1 : 0 <= Age
num2 : Age <= 150
```

b) The age should be greater than yearsmarried.

```
> E2<-editfile("edit.txt")
> E2

Edit set:
num1 : 0 <= Age
num2 : 0 < Height
num3 : Age <= 150
num4 : yearsmarried < Age
```

c) The status should be married or single or widowed.

```
> E3<-editfile("edit2.txt")
> E3

Data model:
dat1 : Age_Group %in% c('Adult', 'child', 'Elderly')
dat2 : Status %in% c('Married', 'single', 'Widowed')

Edit set:
cat1 : if( Age_Group == 'child' ) Status != 'Married'
```

d) If age is less than 18 the agegroup should be child, if age is between 18 and 65 the agegroup should be adult, if age is more than 65 the agegroup should be elderly.

```
> E4<-editfile("edit3.txt")
> E4

Data model:
dat6 : Age_Group %in% c('Adult', 'child', 'Elderly')
dat7 : Status %in% c('Married', 'Widowed')

Edit set:
mix1 : if ( yearsmarried + 17 <= Age ) FALSE
mix2 : if( 18 <= Age ) Age_Group != 'Adult'
mix3 : if( Age < 18 & 65 <= Age ) Age_Group != 'Adult'
mix4 : if( Age < 65 ) Age_Group != 'Adult'
```

III. Check whether ruleset E is violated by the data in the file people.txt.

a)

```
> E <- editset(c("Age >=0", " Age <=150"))
> ve1 <- violatedEdits(E,people)
> ve1
      edit
record  num1  num2
     1 FALSE FALSE
     2 FALSE FALSE
     3 FALSE FALSE
     4 FALSE  TRUE
     5 FALSE FALSE
```

b)

```
> ve4 <- violatedEdits(E,people)
> ve4
      edit
record  num1  num2  num3  num4
     1 FALSE FALSE FALSE FALSE
     2 FALSE FALSE FALSE FALSE
     3 FALSE FALSE FALSE  TRUE
     4 FALSE FALSE  TRUE FALSE
     5 FALSE  TRUE FALSE FALSE
```

c)

```
> E <- editfile("edit2.txt")
> ve2 <- violatedEdits(E,people)
> ve2
      edit
record  dat1  dat2  cat1
     1 FALSE FALSE FALSE
     2 FALSE FALSE  TRUE
     3 FALSE FALSE FALSE
     4 FALSE FALSE FALSE
     5 FALSE FALSE  TRUE
```

d)

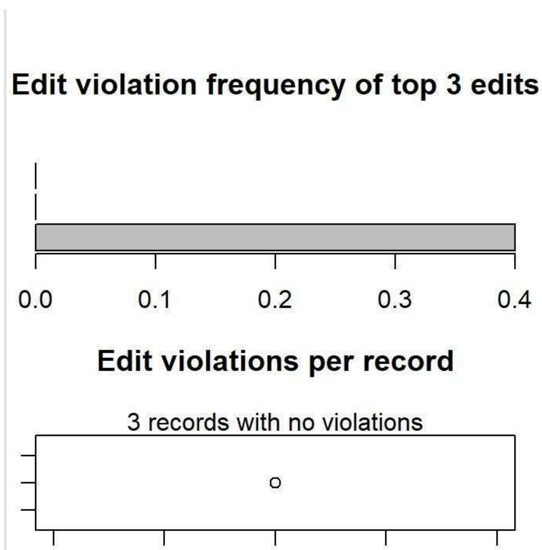
```
> ve3 <- violatedEdits(E,people)
> ve3
      edit
record  dat5  mix1  mix2  mix3
     1 FALSE  TRUE FALSE  TRUE
     2 FALSE FALSE FALSE FALSE
     3 FALSE  TRUE FALSE  TRUE
     4 FALSE FALSE FALSE FALSE
     5 FALSE FALSE FALSE FALSE
```

IV. Summarize the results obtained in part(iii)

```
> summary(ve1)
Edit violations, 5 observations, 0 completely missing (0%):

 editname freq rel
     num2    1 20%

Edit violations per record:

 errors freq rel
      0    4 80%
      1    1 20%
```

V. Visualize the results obtained in part (iii)

```
>plot(v3)
```



Question 2

Perform the following preprocessing tasks on the dirty_iris dataset

```
dirty_iris<- read.csv('iris_dirty.csv') dirty_iris
```

```
      X Sepal.Length Sepal.Width Petal.Length
1     1          5.1         3.5          1.4
2     2          4.9         3.0          1.4
3     3          4.7         3.2          1.3
4     4          4.6         3.1          1.5
5     5           NA         3.6          1.4
6     6          5.4          NA          1.7
7     7          4.6         3.4          1.4
8     8          5.0         3.4          1.5
9     9          4.4         2.9          1.4
10    10          4.9          NA          1.5
11    11          5.4         3.7          1.5
12    12          4.8         3.4          1.6
13    13          4.8         3.0          1.4
14    14          4.3         3.0          1.1
15    15          5.8         4.0          1.2
16    16          5.7         4.4          1.5
17    17          5.4         3.9          1.3
18    18          5.1         3.5          1.4
19    19          5.7         3.8          1.7
20    20          5.1         3.8          1.5
21    21           NA         3.4          1.7
22    22          5.1         3.7          1.5
23    23          4.6         3.6          1.0
24    24          5.1         3.3          1.7
25    25          4.8         3.4          1.9
26    26          5.0         3.0          1.6
27    27          5.0          NA          1.6
28    28          5.2         3.5          1.5
29    29          5.2         3.4          1.4
30    30          4.7         3.2          1.6
31    31          4.8         3.1          1.6
32    32          5.4         3.4          1.5
33    33          5.2         4.1          1.5
34    34          5.5         4.2          1.4
35    35          4.9         3.1          1.5
36   36           5.0         3.2          1.2 ......................... 12
38   38           4.9         3.6          1 ............................ 12
40   40           5.1         3.4          1 ............................ 12
42   42           4.5         2.3          1.3 43    43
4.4         3.2          1.344   44          5.0         3.5
1.................................................................... 12
45   45           5.1         3.8          1 ............................ 12

37   37           5.5          NA          1.3
```

39	39	4.4	3.0	1.3
41	41	5.0	3.5	1.3
46	46	NA	3.0	1.4
47	47	5.1	NA	1.6
48	48	4.6	3.2	1.4
49	49	5.3	3.7	1.5
50	50	5.0	3.3	1.4
51	51	7.0	3.2	4.7
52	52	6.4	3.2	4.5
53	53	6.9	NA	4.9
54	54	5.5	2.3	4.0
55	55	6.5	2.8	4.6
56	56	5.7	2.8	4.5
57	57	6.3	3.3	4.7
58	58	4.9	2.4	3.3
59	59	6.6	2.9	4.6
60	60	5.2	2.7	3.9
61	61	5.0	2.0	3.5
62	62	5.9	NA	4.2
63	63	6.0	2.2	4.0
64	64	6.1	2.9	4.7
65	65	5.6	2.9	3.6
66	66	6.7	3.1	4.4
67	67	5.6	3.0	4.5
68	68	5.8	2.7	4.1
69	69	6.2	2.2	4.5
70	70	5.6	2.5	3.9
71	71	5.9	3.2	4.8
72	72	6.1	2.8	4.0
73	73	6.3	2.5	4.9
74	74	6.1	2.8	4.7
75	75	6.4	2.9	4.3
76	76	6.6	3.0	4.4
77	77	6.8	2.8	4.8
78	78	6.7	3.0	5.0
79	79	6.0	2.9	4.5
80	80	5.7	2.6	3.5
81	81	5.5	2.4	3.8
82	82	5.5	2.4	3.7
83	83	5.8	2.7	3.9
84	84	6.0	2.7	5.1
85	85	5.4	3.0	4.5
86	86	6.0	3.4	4.5
87	87	NA	3.1	4.7
88	88	6.3	2.3	4.4
89	89	5.6	NA	4.1
90	90	5.5	2.5	4.0
91	91	5.5	2.6	4.4
92	92	6.1	3.0	4.6

```
93   93          5.8          2.6           4.0
94   94          5.0          2.3           3.3 95   95
     5.6         2.7           4.2 96   96          5.7
     3.0          4.2 97   97          5.7          2.9
     4.2 98   98          6.2          2.9          4.3
99   99          5.1          2.5           3.0
100 100          5.7          2.8           4.1
101 101          6.3          3.3           6.0
102 102          5.8          2.7           5.1
103 103          7.1          3.0           5.9
104 104          6.3          2.9           5.6
105 105          6.5          3.0           5.8
106 106          7.6          3.0           6.6
107 107          4.9          2.5           4.5
108 108          7.3          2.9           6.3
109 109          6.7          2.5           5.8
110 110          7.2          3.6           6.1
111 111          6.5          3.2           5.1
112 112          6.4          2.7           5.3
113 113          6.8          3.0           5.5
114 114          5.7          2.5           5.0
115 115          5.8           NA           5.1
116 116          6.4          3.2           5.3
117 117          6.5          3.0           5.5
118 118          7.7          3.8           6.7
119 119          7.7          2.6           6.9
120 120          6.0          2.2           5.0
121 121          6.9          3.2           5.7
122 122          5.6          2.8           4.9
123 123          7.7          2.8           6.7
124 124          6.3          2.7           4.9
125 125          6.7          3.3           5.7
126 126          7.2          3.2           6.0
127 127          6.2          2.8           4.8
128 128          6.1          3.0           4.9
129 129          6.4          2.8           5.6
130 130          7.2          3.0           5.8
131 131          7.4          2.8           6.1
132 132          7.9          3.8           6.4
133 133          6.4          2.8           5.6
134 134          6.3           NA           5.1
135 135          6.1          2.6           5.6
136 136          7.7          3.0           6.1
137 137          6.3          3.4           5.6
138 138          6.4          3.1           5.5
139 139          6.0          3.0           4.8
140 140           NA          3.1           5.4
141 141          6.7          3.1           5.6
142 142          6.9          3.1           5.1 143 143
    NA          2.7          5.1
```

```
144 144          6.8          NA          5.9
145 145           NA         3.3          5.7
146 146          6.7         3.0          5.2
147 147          6.3         2.5          5.0
148 148          6.5         3.0          5.2
149 149          6.2         3.4          5.4 150 150
    NA         3.0          5.1
    Petal.Width    Species
1           0.2     Setosa
2           0.2     setosa
3           0.2     setosa
4           0.2     setosa
5           0.2     setosa
6           0.4     setosa
7           0.3     setosa
8           0.2     SETOSA
9           0.2     setosa
10          0.1     setosa
11          0.2     setosa
12          0.2     setosa
13          0.1     setosa
14          0.1     setosa
15          0.2     setosa
16          0.4     setosa
17          0.4     setosa
18          0.3     setosa
19          0.3     setosa
20          0.3     setosa
21          0.2     setosa
22          0.4     setosa
23          0.2     setosa
24          0.5     setosa
25          0.2     setosa
26          0.2     setosa
27          0.4     setosa
28          0.2     setosa
29          0.2     setosa
30          0.2     setosa
31          0.2     setosa
32          0.4     setosa
33          0.1     setosa
34          0.2     setosa
35          0.2     setosa36          0.2     setosa
37          0.2     setosa
38          0.1     setosa
39          0.2     setosa
40          0.2     setosa
41          0.3     setosa
42          0.3     setosa
```

```
43          0.2     setosa
44          0.6     setosa
45          0.4     setosa46          0.3     setosa 47
            0.2     setosa
48          0.2     setosa
49          0.2     setosa
50          0.2     setosa
51          1.4 versicolor
52          1.5 versicolor
53          1.5 versicolor
54          1.3 versicolor
55          1.5 versicolor
56          1.3 versicolor
57          1.6 versicolor
58          1.0 versicolor
59          1.3 Versicolor
60          1.4 versicolor
61          1.0 versicolor
62          1.5 versicolor
63          1.0 versicolor
64          1.4 VERSICOLOR
65          1.3 versicolor
66          1.4 versicolor
67          1.5 versicolor
68          1.0 versicolor
69          1.5 versicolor
70          1.1 versicolor
71          1.8 versicolor
72          1.3 versicolor
73          1.5 versicolor
74          1.2 versicolor
75          1.3 versicolor
76          1.4 versicolor
77          1.4 versicolor
78          1.7 versicolor
79          1.5 versicolor
80          1.0 versicolor
81          1.1 versicolor
82          1.0 versicolor
83          1.2 versicolor
84          1.6 versicolor
85          1.5 versicolor
86          1.6 versicolor
87          1.5 versicolor
88          1.3 versicolor
89          1.3 versicolor
90          1.3 versicolor
91          1.2 versicolor92          1.4 versicolor 93
            1.2 versicolor 94          1.0 versicolor
```

```
95          1.3 versicolor
96          1.2 versicolor
97          1.3 versicolor
98          1.3 versicolor
99          1.1 versicolor
100         1.3 versicolor
101         2.5  virginica
102         1.9  virginica
103         2.1  virginica
104         1.8  virginica
105         2.2  virginica
106         2.1  virginica
107         1.7  virginica
108         1.8  VIRGINICA
109         1.8  virginica
110         2.5  virginica
111         2.0  virginica
112         1.9  virginica
113         2.1  virginica
114         2.0  virginica
115         2.4  virginica
116         2.3  virginica
117         1.8  virginica
118         2.2  virginica
119         2.3  virginica
120         1.5  virginica
121         2.3  virginica
122         2.0  virginica
123         2.0  virginica
124         1.8  virginica
125         2.1  virginica
126         1.8  virginica
127         1.8  virginica
128         1.8  virginica
129         2.1  virginica
130         1.6  virginica
131         1.9  virginica
132         2.0  virginica
133         2.2  virginica
134         1.5  virginica
135         1.4  virginica
136         2.3  virginica
137         2.4  virginica
138         1.8  virginica
139         1.8  virginica140         2.1  virginica 141
            2.4  virginica 142         2.3  virginica
143         1.9  virginica
144         2.3  virginica
145         2.5  virginica
```

```
146         2.3  virginica
147         1.9  virginica
148         2.0  virginica
149         2.3  virginica
150         1.8  virginica
```

1. Calculate the number and percentage of observations that are complete. `c<-sum(complete.cases(dirty_iris)) cat("Number of complete observations:",c,"\n") cat("Pertencage of complete observations:",c/(dim(dirty_iris)[1])*100,"\n\n")`

```
Number of complete observations: 131
Pertencage of complete observations: 87.33333
```

2. Replace all the special values in data with NA.

```
replace <- function(x){   for (i in
1:length(x)){     if(is.infinite(x[i])
| is.nan(x[i]))       x[i]=NA   }   x }
for(i in 1:length(dirty_iris))
{
dirty_iris[,i]=replace(dirty_iris[,i])
   }
dirty_iri
s
```

```
X Sepal.Length Sepal.Width Petal.Length
1     1          5.1         3.5          1.4
2     2          4.9         3.0          1.4
3     3          4.7         3.2          1.3
4     4          4.6         3.1          1.5
5     5           NA         3.6          1.4
6     6          5.4          NA          1.7
7     7          4.6         3.4          1.4
8     8          5.0         3.4          1.5
9     9          4.4         2.9          1.4
10    10          4.9          NA          1.5
11    11          5.4         3.7          1.5
12    12          4.8         3.4          1.6
13    13          4.8         3.0          1.4
14    14          4.3         3.0          1.1
15    15          5.8         4.0          1.2
16    16          5.7         4.4          1.5
17    17          5.4         3.9          1.3 18   18
      5.1         3.5          1.4
19   19          5.7         3.8          1.7
```

20	20	5.1	3.8	1.5
21	21	NA	3.4	1.7
22	22	5.1	3.7	1.5
23	23	4.6	3.6	1.0
24	24	5.1	3.3	1.7
25	25	4.8	3.4	1.9
26	26	5.0	3.0	1.6
27	27	5.0	NA	1.6
28	28	5.2	3.5	1.5
29	29	5.2	3.4	1.4
30	30	4.7	3.2	1.6
31	31	4.8	3.1	1.6
32	32	5.4	3.4	1.5
33	33	5.2	4.1	1.5
34	34	5.5	4.2	1.4
35	35	4.9	3.1	1.5
36	36	5.0	3.2	1.2
37	37	5.5	NA	1.3
38	38	4.9	3.6	1.4
39	39	4.4	3.0	1.3
40	40	5.1	3.4	1.5
41	41	5.0	3.5	1.3
42	42	4.5	2.3	1.3
43	43	4.4	3.2	1.3
44	44	5.0	3.5	1.6
45	45	5.1	3.8	1.9
46	46	NA	3.0	1.4
47	47	5.1	NA	1.6
48	48	4.6	3.2	1.4
49	49	5.3	3.7	1.5
50	50	5.0	3.3	1.4
51	51	7.0	3.2	4.7
52	52	6.4	3.2	4.5
53	53	6.9	NA	4.9
54	54	5.5	2.3	4.0
55	55	6.5	2.8	4.6
56	56	5.7	2.8	4.5
57	57	6.3	3.3	4.7
58	58	4.9	2.4	3.3
59	59	6.6	2.9	4.6
60	60	5.2	2.7	3.9
61	61	5.0	2.0	3.5
62	62	5.9	NA	4.2 63 63
	6.0	2.2	4.0	
64	64	6.1	2.9	4.7
65	65	5.6	2.9	3.6
66	66	6.7	3.1	4.4
67	67	5.6	3.0	4.5
68	68	5.8	2.7	4.1

69	69	6.2	2.2	4.5
70	70	5.6	2.5	3.9
71	71	5.9	3.2	4.8
72	72	6.1	2.8	4.0
73	73	6.3	2.5	4.9
74	74	6.1	2.8	4.7
75	75	6.4	2.9	4.3
76	76	6.6	3.0	4.4
77	77	6.8	2.8	4.8
78	78	6.7	3.0	5.0
79	79	6.0	2.9	4.5
80	80	5.7	2.6	3.5
81	81	5.5	2.4	3.8
82	82	5.5	2.4	3.7
83	83	5.8	2.7	3.9
84	84	6.0	2.7	5.1
85	85	5.4	3.0	4.5
86	86	6.0	3.4	4.5
87	87	NA	3.1	4.7
88	88	6.3	2.3	4.4
89	89	5.6	NA	4.1
90	90	5.5	2.5	4.0
91	91	5.5	2.6	4.4
92	92	6.1	3.0	4.6
93	93	5.8	2.6	4.0
94	94	5.0	2.3	3.3
95	95	5.6	2.7	4.2
96	96	5.7	3.0	4.2
97	97	5.7	2.9	4.2
98	98	6.2	2.9	4.3
99	99	5.1	2.5	3.0
100	100	5.7	2.8	4.1
101	101	6.3	3.3	6.0
102	102	5.8	2.7	5.1
103	103	7.1	3.0	5.9
104	104	6.3	2.9	5.6
105	105	6.5	3.0	5.8
106	106	7.6	3.0	6.6
107	107	4.9	2.5	4.5
108	108	7.3	2.9	6.3
109	109	6.7	2.5	5.8
110	110	7.2	3.6	6.1
111	111	6.5	3.2	5.1
112	112	6.4	2.7	5.3
113	113	6.8	3.0	5.5
114	114	5.7	2.5	5.0
115	115	5.8	NA	5.1
116	116	6.4	3.2	5.3
117	117	6.5	3.0	5.5

```
118  118          7.7         3.8          6.7
119  119          7.7         2.6          6.9
120  120          6.0         2.2          5.0
121  121          6.9         3.2          5.7
122  122          5.6         2.8          4.9
123  123          7.7         2.8          6.7
124  124          6.3         2.7          4.9
125  125          6.7         3.3          5.7
126  126          7.2         3.2          6.0
127  127          6.2         2.8          4.8
128  128          6.1         3.0          4.9
129  129          6.4         2.8          5.6
130  130          7.2         3.0          5.8
131  131          7.4         2.8          6.1
132  132          7.9         3.8          6.4
133  133          6.4         2.8          5.6
134  134          6.3          NA          5.1
135  135          6.1         2.6          5.6
136  136          7.7         3.0          6.1
137  137          6.3         3.4          5.6
138  138          6.4         3.1          5.5
139  139          6.0         3.0          4.8
140  140           NA         3.1          5.4
141  141          6.7         3.1          5.6
142  142          6.9         3.1          5.1
143  143           NA         2.7          5.1
144  144          6.8          NA          5.9
145  145           NA         3.3          5.7
146  146          6.7         3.0          5.2
147  147          6.3         2.5          5.0
148  148          6.5         3.0          5.2
149  149          6.2         3.4          5.4
150  150           NA         3.0          5.1
    Petal.Width    Species
1           0.2     Setosa
2           0.2     setosa
3           0.2     setosa
4           0.2     setosa
5           0.2     setosa
6           0.4     setosa
7           0.3     setosa
8           0.2     SETOSA
9           0.2     setosa
10          0.1     setosa
11          0.2     setosa12          0.2     setosa 13
            0.1     setosa
14          0.1     setosa
15          0.2     setosa
16          0.4     setosa
```

```
17          0.4     setosa
18          0.3     setosa
19          0.3     setosa
20          0.3     setosa
21          0.2     setosa
22          0.4     setosa
23          0.2     setosa
24          0.5     setosa
25          0.2     setosa
26          0.2     setosa
27          0.4     setosa
28          0.2     setosa
29          0.2     setosa
30          0.2     setosa
31          0.2     setosa
32          0.4     setosa
33          0.1     setosa
34          0.2     setosa
35          0.2     setosa
36          0.2     setosa
37          0.2     setosa
38          0.1     setosa
39          0.2     setosa
40          0.2     setosa
41          0.3     setosa
42          0.3     setosa
43          0.2     setosa
44          0.6     setosa
45          0.4     setosa
46          0.3     setosa
47          0.2     setosa
48          0.2     setosa
49          0.2     setosa
50          0.2     setosa
51          1.4 versicolor
52          1.5 versicolor
53          1.5 versicolor
54          1.3 versicolor
55          1.5 versicolor
56          1.3 versicolor
57          1.6 versicolor
58          1.0 versicolor59          1.3 Versicolor 60
            1.4 versicolor
61          1.0 versicolor
62          1.5 versicolor
63          1.0 versicolor
64          1.4 VERSICOLOR
65          1.3 versicolor
66          1.4 versicolor
```

```
67          1.5 versicolor
68          1.0 versicolor
69          1.5 versicolor
70          1.1 versicolor
71          1.8 versicolor
72          1.3 versicolor
73          1.5 versicolor
74          1.2 versicolor
75          1.3 versicolor
76          1.4 versicolor
77          1.4 versicolor
78          1.7 versicolor
79          1.5 versicolor
80          1.0 versicolor
81          1.1 versicolor
82          1.0 versicolor
83          1.2 versicolor
84          1.6 versicolor
85          1.5 versicolor
86          1.6 versicolor
87          1.5 versicolor
88          1.3 versicolor
89          1.3 versicolor
90          1.3 versicolor
91          1.2 versicolor
92          1.4 versicolor
93          1.2 versicolor
94          1.0 versicolor
95          1.3 versicolor
96          1.2 versicolor
97          1.3 versicolor
98          1.3 versicolor
99          1.1 versicolor
100         1.3 versicolor
101         2.5  virginica
102         1.9  virginica
103         2.1  virginica
104         1.8  virginica
105         2.2  virginica106         2.1  virginica 107
            1.7  virginica
108         1.8  VIRGINICA
109         1.8  virginica110         2.5  virginica
111         2.0  virginica
112         1.9  virginica
113         2.1  virginica
114         2.0  virginica
115         2.4  virginica116         2.3  virginica 117
            1.8  virginica
118         2.2  virginica
119         2.3  virginica
```

```
120         1.5  virginica
121         2.3  virginica
122         2.0  virginica
123         2.0  virginica
124         1.8  virginica
125         2.1  virginica
126         1.8  virginica
127         1.8  virginica
128         1.8  virginica
129         2.1  virginica
130         1.6  virginica
131         1.9  virginica
132         2.0  virginica
133         2.2  virginica
134         1.5  virginica
135         1.4  virginica
136         2.3  virginica
137         2.4  virginica
138         1.8  virginica
139         1.8  virginica
140         2.1  virginica
141         2.4  virginica
142         2.3  virginica
143         1.9  virginica
144         2.3  virginica
145         2.5  virginica
146         2.3  virginica
147         1.9  virginica
148         2.0  virginica
149         2.3  virginica
150         1.8  virginica
```

3. Define these rules in a separate text file and read them.

(Use editfile function in R (package editrules). Use similar function in Python). Print the resulting constraint object.

–Species should be one of the following values: setosa, versicolor or virginica.
–All measured numerical properties of an iris should be positive.
–The petal length of an iris is atleast 2 times its petal width.
–The sepal length of an iris cannot exceed 30cm.
–The sepals of an iris are longer than its petals.

```
> E<-editfile("editQ2.txt")
> E

Data model:
dat1 : Species %in% c('setosa', 'versicolor', 'virgin
ica')

Edit set:
num1 : 0 <= Sepal.Length
num2 : 0 <= Sepal.Width
num3 : 0 <= Petal.Length
num4 : 0 <= Petal.Width
num5 : 2*Petal.Width <= Petal.Length
num6 : Sepal.Length <= 30
num7 : Petal.Length < Sepal.Length
num8 : Petal.width < Sepal.Width
```

4. Determine how often each rule is broken (violatedEdits). Also summarize and plot the result `ve <- violatedEdits(E,dirty_iris) ve`

```
 edit
record  num1  num2  num3  num4  num5  num6
1   FALSE FALSE FALSE FALSE FALSE FALSE
      2      FALSE FALSE FALSE FALSE FALSE FALSE
      3      FALSE FALSE FALSE FALSE FALSE FALSE
      4      FALSE FALSE FALSE FALSE FALSE FALSE
      5      NA FALSE FALSE FALSE FALSE    NA
      6      FALSE    NA FALSE FALSE FALSE FALSE
      7      FALSE FALSE FALSE FALSE FALSE FALSE
      8      FALSE FALSE FALSE FALSE FALSE FALSE
      9      FALSE FALSE FALSE FALSE FALSE FALSE
      10     FALSE    NA FALSE FALSE FALSE FALSE
      11     FALSE FALSE FALSE FALSE FALSE FALSE
      12     FALSE FALSE FALSE FALSE FALSE FALSE
      13     FALSE FALSE FALSE FALSE FALSE FALSE 14  FALSE
             FALSE FALSE FALSE FALSE FALSE
15     FALSE FALSE FALSE FALSE FALSE FALSE
16     FALSE FALSE FALSE FALSE FALSE FALSE
17     FALSE FALSE FALSE FALSE FALSE FALSE
18     FALSE FALSE FALSE FALSE FALSE FALSE
19     FALSE FALSE FALSE FALSE FALSE FALSE
20     FALSE FALSE FALSE FALSE FALSE FALSE
21     NA FALSE FALSE FALSE FALSE    NA
22     FALSE FALSE FALSE FALSE FALSE FALSE
23     FALSE FALSE FALSE FALSE FALSE FALSE
24     FALSE FALSE FALSE FALSE FALSE FALSE
25     FALSE FALSE FALSE FALSE FALSE FALSE
26     FALSE FALSE FALSE FALSE FALSE FALSE
27     FALSE    NA FALSE FALSE FALSE FALSE
```

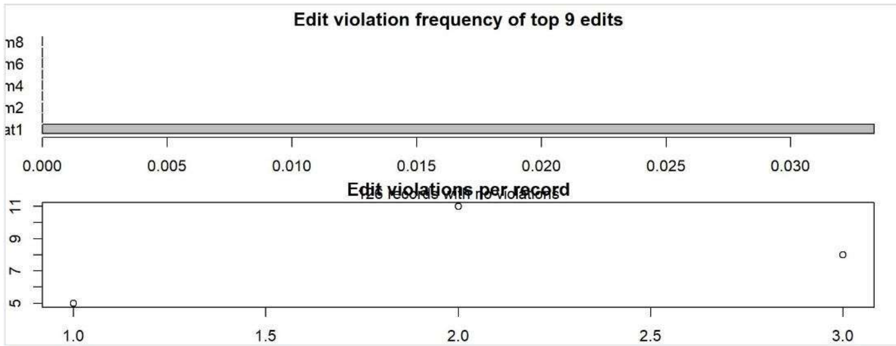
```
28     FALSE FALSE FALSE FALSE FALSE FALSE
29     FALSE FALSE FALSE FALSE FALSE FALSE
30     FALSE FALSE FALSE FALSE FALSE FALSE
31     FALSE FALSE FALSE FALSE FALSE FALSE
32     FALSE FALSE FALSE FALSE FALSE FALSE
33     FALSE FALSE FALSE FALSE FALSE FALSE
34     FALSE FALSE FALSE FALSE FALSE FALSE
35     FALSE FALSE FALSE FALSE FALSE FALSE
36     FALSE FALSE FALSE FALSE FALSE FALSE
37     FALSE    NA FALSE FALSE FALSE FALSE
38     FALSE FALSE FALSE FALSE FALSE FALSE
39     FALSE FALSE FALSE FALSE FALSE FALSE
40     FALSE FALSE FALSE FALSE FALSE FALSE
41     FALSE FALSE FALSE FALSE FALSE FALSE
42     FALSE FALSE FALSE FALSE FALSE FALSE
43     FALSE FALSE FALSE FALSE FALSE FALSE
44     FALSE FALSE FALSE FALSE FALSE FALSE
45     FALSE FALSE FALSE FALSE FALSE FALSE
46     NA FALSE FALSE FALSE FALSE    NA
47     FALSE    NA FALSE FALSE FALSE FALSE
48     FALSE FALSE FALSE FALSE FALSE FALSE
49     FALSE FALSE FALSE FALSE FALSE FALSE
50     FALSE FALSE FALSE FALSE FALSE FALSE
51     FALSE FALSE FALSE FALSE FALSE FALSE
52     FALSE FALSE FALSE FALSE FALSE FALSE
53     FALSE    NA FALSE FALSE FALSE FALSE
54     FALSE FALSE FALSE FALSE FALSE FALSE
55     FALSE FALSE FALSE FALSE FALSE FALSE
56     FALSE FALSE FALSE FALSE FALSE FALSE
57     FALSE FALSE FALSE FALSE FALSE FALSE
58     FALSE FALSE FALSE FALSE FALSE FALSE
59     FALSE FALSE FALSE FALSE FALSE FALSE
60     FALSE FALSE FALSE FALSE FALSE FALSE
61     FALSE FALSE FALSE FALSE FALSE FALSE
62     FALSE    NA FALSE FALSE FALSE FALSE
63     FALSE FALSE FALSE FALSE FALSE FALSE
64     FALSE FALSE FALSE FALSE FALSE FALSE
65     FALSE FALSE FALSE FALSE FALSE FALSE
66     FALSE FALSE FALSE FALSE FALSE FALSE
67     FALSE FALSE FALSE FALSE FALSE FALSE
68     FALSE FALSE FALSE FALSE FALSE FALSE
69     FALSE FALSE FALSE FALSE FALSE FALSE
70     FALSE FALSE FALSE FALSE FALSE FALSE
71     FALSE FALSE FALSE FALSE FALSE FALSE
72     FALSE FALSE FALSE FALSE FALSE FALSE
73     FALSE FALSE FALSE FALSE FALSE FALSE
74     FALSE FALSE FALSE FALSE FALSE FALSE
75     FALSE FALSE FALSE FALSE FALSE FALSE
76     FALSE FALSE FALSE FALSE FALSE FALSE
```

```
77     FALSE FALSE FALSE FALSE FALSE FALSE
78     FALSE FALSE FALSE FALSE FALSE FALSE
79     FALSE FALSE FALSE FALSE FALSE FALSE
80     FALSE FALSE FALSE FALSE FALSE FALSE
81     FALSE FALSE FALSE FALSE FALSE FALSE
82     FALSE FALSE FALSE FALSE FALSE FALSE
83     FALSE FALSE FALSE FALSE FALSE FALSE
84     FALSE FALSE FALSE FALSE FALSE FALSE
85     FALSE FALSE FALSE FALSE FALSE FALSE
86     FALSE FALSE FALSE FALSE FALSE FALSE
87     NA FALSE FALSE FALSE FALSE    NA
88     FALSE FALSE FALSE FALSE FALSE FALSE
89     FALSE    NA FALSE FALSE FALSE FALSE
90     FALSE FALSE FALSE FALSE FALSE FALSE
91     FALSE FALSE FALSE FALSE FALSE FALSE
92     FALSE FALSE FALSE FALSE FALSE FALSE
93     FALSE FALSE FALSE FALSE FALSE FALSE
94     FALSE FALSE FALSE FALSE FALSE FALSE
95     FALSE FALSE FALSE FALSE FALSE FALSE
96     FALSE FALSE FALSE FALSE FALSE FALSE
97     FALSE FALSE FALSE FALSE FALSE FALSE
98     FALSE FALSE FALSE FALSE FALSE FALSE
99     FALSE FALSE FALSE FALSE FALSE FALSE
100    FALSE FALSE FALSE FALSE FALSE FALSE
101    FALSE FALSE FALSE FALSE FALSE FALSE
102    FALSE FALSE FALSE FALSE FALSE FALSE
103    FALSE FALSE FALSE FALSE FALSE FALSE
104    FALSE FALSE FALSE FALSE FALSE FALSE
105    FALSE FALSE FALSE FALSE FALSE FALSE
106    FALSE FALSE FALSE FALSE FALSE FALSE
107    FALSE FALSE FALSE FALSE FALSE FALSE
108    FALSE FALSE FALSE FALSE FALSE FALSE
109    FALSE FALSE FALSE FALSE FALSE FALSE
110    FALSE FALSE FALSE FALSE FALSE FALSE    111 FALSE
       FALSE FALSE FALSE FALSE FALSE       edit record  num7
       num8  dat1 1   FALSE FALSE  TRUE
   2   FALSE FALSE FALSE
   3   FALSE FALSE FALSE4   FALSE FALSE
       FALSE 5      NA FALSE FALSE
6      FALSE    NA FALSE
7      FALSE FALSE FALSE
8      FALSE FALSE  TRUE
9      FALSE FALSE FALSE
10     FALSE    NA FALSE
11     FALSE FALSE FALSE
12     FALSE FALSE FALSE
13     FALSE FALSE FALSE
14     FALSE FALSE FALSE
15     FALSE FALSE FALSE
16     FALSE FALSE FALSE
```

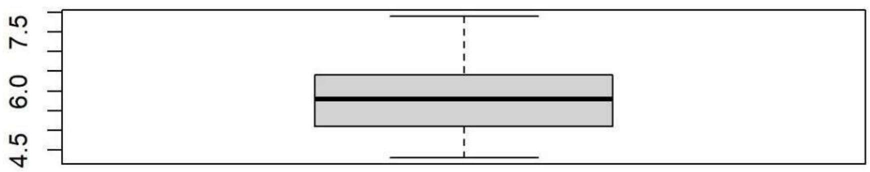
```
17     FALSE FALSE FALSE
18     FALSE FALSE FALSE
19     FALSE FALSE FALSE
20     FALSE FALSE FALSE
21     NA FALSE FALSE
22     FALSE FALSE FALSE
23     FALSE FALSE FALSE
24     FALSE FALSE FALSE
25     FALSE FALSE FALSE
26     FALSE FALSE FALSE
27     FALSE    NA FALSE
28     FALSE FALSE FALSE
29     FALSE FALSE FALSE
30     FALSE FALSE FALSE
31     FALSE FALSE FALSE
32     FALSE FALSE FALSE
33     FALSE FALSE FALSE
34     FALSE FALSE FALSE
35     FALSE FALSE FALSE
36     FALSE FALSE FALSE
37     FALSE    NA FALSE
38     FALSE FALSE FALSE
39     FALSE FALSE FALSE
40     FALSE FALSE FALSE
41     FALSE FALSE FALSE
42     FALSE FALSE FALSE
43     FALSE FALSE FALSE
44     FALSE FALSE FALSE
45     FALSE FALSE FALSE
46     NA FALSE FALSE
47     FALSE    NA FALSE
48     FALSE FALSE FALSE
49     FALSE FALSE FALSE
50     FALSE FALSE FALSE
51     FALSE FALSE FALSE
52     FALSE FALSE FALSE
53     FALSE    NA FALSE54  FALSE FALSE FALSE 55  FALSE
       FALSE FALSE
56     FALSE FALSE FALSE
57     FALSE FALSE FALSE
58     FALSE FALSE FALSE
59     FALSE FALSE  TRUE
60     FALSE FALSE FALSE
61     FALSE FALSE FALSE
62     FALSE    NA FALSE
63     FALSE FALSE FALSE
64     FALSE FALSE  TRUE
65     FALSE FALSE FALSE
66     FALSE FALSE FALSE
```

```
67     FALSE FALSE FALSE
68     FALSE FALSE FALSE
69     FALSE FALSE FALSE
70     FALSE FALSE FALSE
71     FALSE FALSE FALSE
72     FALSE FALSE FALSE
73     FALSE FALSE FALSE
74     FALSE FALSE FALSE
75     FALSE FALSE FALSE
76     FALSE FALSE FALSE
77     FALSE FALSE FALSE
78     FALSE FALSE FALSE
79     FALSE FALSE FALSE
80     FALSE FALSE FALSE
81     FALSE FALSE FALSE
82     FALSE FALSE FALSE
83     FALSE FALSE FALSE
84     FALSE FALSE FALSE
85     FALSE FALSE FALSE
86     FALSE FALSE FALSE
87     NA FALSE FALSE
88     FALSE FALSE FALSE
89     FALSE    NA FALSE
90     FALSE FALSE FALSE
91     FALSE FALSE FALSE
92     FALSE FALSE FALSE
93     FALSE FALSE FALSE
94     FALSE FALSE FALSE
95     FALSE FALSE FALSE
96     FALSE FALSE FALSE
97     FALSE FALSE FALSE
98     FALSE FALSE FALSE
99     FALSE FALSE FALSE
100    FALSE FALSE FALSE
101    FALSE FALSE FALSE
102    FALSE FALSE FALSE
103    FALSE FALSE FALSE104 FALSE FALSE FALSE 105 FALSE
       FALSE FALSE
   106 FALSE FALSE FALSE
   107 FALSE FALSE FALSE
   108 FALSE FALSE  TRUE
   109 FALSE FALSE FALSE
   110 FALSE FALSE FALSE
   111 FALSE FALSE FALSE
```

```
Plot(ve)
```



5. Find outliers in sepal length using boxplot and boxplot.stats
```
boxplot(iris$Sepal.Length)
```



```
> boxplot.stats((iris$Sepal.Length))
$stats
[1] 4.3 5.1 5.8 6.4 7.9

$n
[1] 150

$conf
[1] 5.632292 5.967708

$out
numeric(0)
```

Question 3

Load the data from wine dataset. Check whether all attributes are standardized or not (mean is 0 and standard deviation is 1). If not, standardize the attributes.

```
wine <- as.data.frame(read.csv("wine.csv"))
```

```
fixed_acidity volatile_acidity citric_acid
1             7.4             0.700        0.00
2             7.8             0.880        0.00
```

3	7.8	0.760	0.04
4	11.2	0.280	0.56
5	7.4	0.700	0.00
6	7.4	0.660	0.00
7	7.9	0.600	0.06
8	7.3	0.650	0.00
9	7.8	0.580	0.02
10	7.5	0.500	0.36
11	6.7	0.580	0.08
12	7.5	0.500	0.36
13	5.6	0.615	0.00
14	7.8	0.610	0.29
15	8.9	0.620	0.18
16	8.9	0.620	0.19
17	8.5	0.280	0.56
18	8.1	0.560	0.28
19	7.4	0.590	0.08
20	7.9	0.320	0.51
21	8.9	0.220	0.48
22	7.6	0.390	0.31
23	7.9	0.430	0.21
24	8.5	0.490	0.11
25	6.9	0.400	0.14
26	6.3	0.390	0.16
27	7.6	0.410	0.24
28	7.9	0.430	0.21
29	7.1	0.710	0.00
30	7.8	0.645	0.00
31	6.7	0.675	0.07
32	6.9	0.685	0.00
33	8.3	0.655	0.12
34	6.9	0.605	0.12

35	5.2	0.320	0.25
36	7.8	0.645	0.00
37	7.8	0.600	0.14
38	8.1	0.380	0.28
39	5.7	1.130	0.09
40	7.3	0.450	0.36
41	7.3	0.450	0.36
42	8.8	0.610	0.30
43	7.5	0.490	0.20
44	8.1	0.660	0.22
45	6.8	0.670	0.02
46	4.6	0.520	0.15
47	7.7	0.935	0.43
48	8.7	0.290	0.52
49	6.4	0.400	0.23
50	5.6	0.310	0.37
51	8.8	0.660	0.26
52	6.6	0.520	0.04
53	6.6	0.500	0.04
54	8.6	0.380	0.36
55	7.6	0.510	0.15
56	7.7	0.620	0.04
57	10.2	0.420	0.57
58	7.5	0.630	0.12
59	7.8	0.590	0.18
60	7.3	0.390	0.31
61	8.8	0.400	0.40
62	7.7	0.690	0.49
63	7.5	0.520	0.16
64	7.0	0.735	0.05
65	7.2	0.725	0.05
66	7.2	0.725	0.05

```
67          7.5           0.520        0.11
68          6.6           0.705        0.07
69          9.3           0.320        0.57
70          8.0           0.705        0.05
71          7.7           0.630        0.08
72          7.7           0.670        0.23
73          7.7           0.690        0.22
74          8.3           0.675        0.26
75          9.7           0.320        0.54
76          8.8           0.410        0.64    residual_sugar
            chlorides free_sulfur_dioxide 1             1.90
            0.076                  11
2           2.60     0.098                  25
3           2.30     0.092                  15
4           1.90     0.075                  17
5           1.90     0.076                  11
6           1.80     0.075                  13
7           1.60     0.069                  15
8           1.20     0.065                  15
9           2.00     0.073                   9
10          6.10     0.071                  17
11          1.80     0.097                  15
12          6.10     0.071                  17
13          1.60     0.089                  16
14          1.60     0.114                   9 15            3.80
            0.176                  52
16          3.90     0.170                  51
17          1.80     0.092                  35
18          1.70     0.368                  16
19          4.40     0.086                   6
20          1.80     0.341                  17
21          1.80     0.077                  29
```

22	2.30	0.082	23
23	1.60	0.106	10
24	2.30	0.084	9
25	2.40	0.085	21
26	1.40	0.080	11
27	1.80	0.080	4
28	1.60	0.106	10
29	1.90	0.080	14
30	2.00	0.082	8
31	2.40	0.089	17
32	2.50	0.105	22
33	2.30	0.083	15
34	10.70	0.073	40
35	1.80	0.103	13
36	5.50	0.086	5
37	2.40	0.086	3
38	2.10	0.066	13
39	1.50	0.172	7
40	5.90	0.074	12
41	5.90	0.074	12
42	2.80	0.088	17
43	2.60	0.332	8
44	2.20	0.069	9
45	1.80	0.050	5
46	2.10	0.054	8
47	2.20	0.114	22
48	1.60	0.113	12
49	1.60	0.066	5
50	1.40	0.074	12
51	1.70	0.074	4
52	2.20	0.069	8
53	2.10	0.068	6

```
54           3.00    0.081                 30
55           2.80    0.110                 33
56           3.80    0.084                 25
57           3.40    0.070                  4
58           5.10    0.111                 50
59           2.30    0.076                 17
60           2.40    0.074                  9
61           2.20    0.079                 19
62           1.80    0.115                 20
63           1.90    0.085                 12
64           2.00    0.081                 13
65           4.65    0.086                  4
66           4.65    0.086                  4
67           1.50    0.079                 11
68           1.60    0.076                  6
69           2.00    0.074                 27
70           1.90    0.074                  8
71           1.90    0.076                 15
72           2.10    0.088                 17
73           1.90    0.084                 18
74           2.10    0.084                 11
75           2.50    0.094                 28
76           2.20    0.093                  9
   total_sulfur_dioxide density   pH sulphates
1                    34  0.9978 3.51      0.56
2                    67  0.9968 3.20      0.68
                     3                    54
                     0.9970 3.26      0.65
4                    60  0.9980 3.16      0.58
5                    34  0.9978 3.51      0.56
6                    40  0.9978 3.51      0.56
7                    59  0.9964 3.30      0.46
```

8	21	0.9946	3.39	0.47
9	18	0.9968	3.36	0.57
10	102	0.9978	3.35	0.80
11	65	0.9959	3.28	0.54
12	102	0.9978	3.35	0.80
13	59	0.9943	3.58	0.52
14	29	0.9974	3.26	1.56
15	145	0.9986	3.16	0.88
16	148	0.9986	3.17	0.93
17	103	0.9969	3.30	0.75
18	56	0.9968	3.11	1.28
19	29	0.9974	3.38	0.50
20	56	0.9969	3.04	1.08
21	60	0.9968	3.39	0.53
22	71	0.9982	3.52	0.65
23	37	0.9966	3.17	0.91
24	67	0.9968	3.17	0.53
25	40	0.9968	3.43	0.63
26	23	0.9955	3.34	0.56
27	11	0.9962	3.28	0.59
28	37	0.9966	3.17	0.91
29	35	0.9972	3.47	0.55
30	16	0.9964	3.38	0.59
31	82	0.9958	3.35	0.54
32	37	0.9966	3.46	0.57
33	113	0.9966	3.17	0.66
34	83	0.9993	3.45	0.52
35	50	0.9957	3.38	0.55
36	18	0.9986	3.40	0.55
37	15	0.9975	3.42	0.60
38	30	0.9968	3.23	0.73
39	19	0.9940	3.50	0.48

40	87	0.9978	3.33	0.83	
41	87	0.9978	3.33	0.83	
42	46	0.9976	3.26	0.51	
43	14	0.9968	3.21	0.90	
44	23	0.9968	3.30	1.20	
45	11	0.9962	3.48	0.52	
46	65	0.9934	3.90	0.56	
47	114	0.9970	3.25	0.73	
48	37	0.9969	3.25	0.58	
49	12	0.9958	3.34	0.56	
50	96	0.9954	3.32	0.58	
51	23	0.9971	3.15	0.74	
52	15	0.9956	3.40	0.63	
53	14	0.9955	3.39	0.64	
54	119	0.9970	3.20	0.56	
55	73	0.9955	3.17	0.63	
56	45	0.9978	3.34	0.53	
57	10	0.9971	3.04	0.63	
58	110	0.9983	3.26	0.77	
59	54	0.9975	3.43	0.59	
60	46	0.9962	3.41	0.54	
61	52	0.9980	3.44	0.64	
62	112	0.9968	3.21	0.71	
63	35	0.9968	3.38	0.62	
64	54	0.9966	3.39	0.57	
65	11	0.9962	3.41	0.39	66
	11	0.9962	3.41	0.39	
67	39	0.9968	3.42	0.58	
68	15	0.9962	3.44	0.58	
69	65	0.9969	3.28	0.79	
70	19	0.9962	3.34	0.95	
71	27	0.9967	3.32	0.54	

72 96 0.9962 3.32 0.48

73 94 0.9961 3.31 0.48

74 43 0.9976 3.31 0.53

75 83 0.9984 3.28 0.82 76

42 0.9986 3.54 0.66 alcohol quality

style 1 9.4 5 red

2	9.8	5	red
3	9.8	5	red
4	9.8	6	red
5	9.4	5	red
6	9.4	5	red
7	9.4	5	red
8	10.0	7	red
9	9.5	7	red
10	10.5	5	red
11	9.2	5	red
12	10.5	5	red
13	9.9	5	red
14	9.1	5	red
15	9.2	5	red
16	9.2	5	red
17	10.5	7	red
18	9.3	5	red
19	9.0	4	red
20	9.2	6	red
21	9.4	6	red
22	9.7	5	red
23	9.5	5	red
24	9.4	5	red
25	9.7	6	red
26	9.3	5	red
27	9.5	5	red

28	9.5	5	red				
29	9.4	5	red				
30	9.8	6	red				
31	10.1	5	red				
32	10.6	6	red				
33	9.8	5	red				
34	9.4	6	red				
35	9.2	5	red				
36	9.6	6	red				
37	10.8	6	red				
38	9.7	7	red				
39	9.8	4	red				
40	10.5	5	red				
41	10.5	5	red				
42	9.3	4	red				
43	10.5	6	red				
44	10.3	5	red				
45	9.5	5	red				
46	13.1	4	red				
47	9.2	5	red				
48	9.5	5	red				
49	9.2	5	red				
50	9.2	5	red				
51	9.2	5	red				
52	9.4	6	red	53	9.4	6	red
54	9.4	5	red				
55	10.2	6	red				
56	9.5	5	red				
57	9.6	5	red				
58	9.4	5	red				
59	10.0	5	red				
60	9.4	6	red				

```
61     9.2       5   red
62     9.3       5   red
63     9.5       7   red
64     9.8       5   red
65     10.9       5   red
66     10.9       5   red
67     9.6       5   red
68     10.7       5   red
69     10.7       5   red
70     10.5       6   red
71     9.5       6   red
72     9.5       5   red
73     9.5       5   red
74     9.2       4   red
75     9.6       5   red
76     10.5       5   red
 [ reached 'max' / getOption("max.print") -- omitted 6421 rows ]
```

```
wine[1:(length(wine)-1)]
```

```
flag<-1 j<-1 for(j in
1:length(wine))
{
  if(mean(wine[,j])!=0 || sd(wine[,j])!=1)
{
    flag<-0
}   j<-j+1
}
if(flag==1)
{
  cat("Dataset is normalized")
} if(flag==0){
  cat("Dataset is not normalized","\n")
cat("Normalizing the dataset:","\n")   data_std<-
function(X){
    (x-mean(x))/sd(X)
  }
}
```

```
Dataset is not normalized

Normalizing the dataset:


wine1<-data.frame(sapply(wine,data_std))
i<-1
for(i in 1:length(wine)-1){
  cat("The mean of ",names(wine[i]),"is ",as.integer(mean(wine1[,i]))," and 
standard
                                deviation is ",as.integer(sd(wine1[,i])))
cat("\n")   i=i+1
}
```

```
The  mean  of   fixed_acidity  is   0   and  standard
deviation is  0 The mean of  volatile_acidity is  0
and standard                                deviation is
1  The  mean  of   citric_acid  is   0   and  standard
deviation is  0 The mean of  residual_sugar is  0
and standard                                deviation is
1  The  mean  of   chlorides  is   0   and  standard
deviation is  1 The mean of  free_sulfur_dioxide is
0  and standard                                deviation
is  0
The mean of  total_sulfur_dioxide is  0  and standard
                                deviation is  1
The mean of  density is  0  and standard
deviation is  1
The mean of  pH is  0  and standard
                                deviation is  1
The mean of  sulphates is  0  and standard
deviation is  0 The mean of  alcohol is  0
and standard                      deviation
is   0 The mean of  quality is  0  and
standard
                                deviation is  0
```

Question 4

Run Apriori algorithm to find frequent item sets and association rules

4.1 Use minimum support as 50% and minimum confidence as 75%

4.2 Use minimum support as 60% and minimum confidence as 60%

```
library(arules)
```

```
 library(datasets)
data("Groceries")
inspect(Groceries[1:10])
```

```
     items                  [1]
{citrus                  fruit,
semi-finished            bread,
margarine,
ready soups}                [2]
{tropical                fruit,
yogurt,
coffee}                     [3]
{whole milk}                [4]
{pip                     fruit,
yogurt,                   cream
cheese ,                   meat
spreads}            [5]  {other
vegetables,               whole
milk,                 condensed
milk,                 long life
bakery  product}  [6]    {whole
milk,                   butter,
yogurt,                   rice,
abrasive cleaner}           [7]
{rolls/buns}
[8]  {other vegetables,
      UHT-milk,
rolls/buns,
bottled             beer,
liquor       (appetizer)}
[9]       {pot    plants}
[10]     {whole     milk,
cereals}
```

```
r<-apriori(Groceries,parameter=list(support=0.05,conf=0.1,minlen=2))
```

```
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport
0.1    0.1    1 none FALSE            TRUE
maxtime support minlen maxlen target  ext
5    0.05      2     10  rules TRUE

Algorithmic control:
 filter tree heap memopt load sort verbose
0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 491
```

```
 set item appearances ...[0 item(s)] done [0.00s]. set transactions
...[169 item(s), 9835 transaction(s)] done [0.01s]. sorting and
recoding items ... [28 item(s)] done [0.00s]. creating transaction
tree ... done [0.01s]. checking subsets of size 1 2 done [0.00s].
writing ... [6 rule(s)] done [0.00s]. creating S4 object  ... done
[0.00s].
 inspect(r)
lhs                  rhs                 [1]
{yogurt}           => {whole milk}
[2] {whole milk}       => {yogurt}
[3] {rolls/buns}       => {whole milk}
[4] {whole milk}       => {rolls/buns}
[5] {other vegetables} => {whole milk}       [6] {whole milk}
    => {other vegetables}     support    confidence coverage
    lift     count [1] 0.05602440 0.4016035  0.1395018 1.571735
    551
[2] 0.05602440 0.2192598  0.2555160 1.571735 551
[3] 0.05663447 0.3079049  0.1839349 1.205032 557
[4] 0.05663447 0.2216474  0.2555160 1.205032 557
[5] 0.07483477 0.3867578  0.1934926 1.513634 736
[6] 0.07483477 0.2928770  0.2555160 1.513634 736
```

```
data("Adult") inspect(Adult[1:10])

rul<-apriori(Adult,parameter=list(support=0.6,conf=0.6,minlen=2))
inspect(rul)
```

```
Parameter specification:
 confidence minval smax arem  aval originalSupport
0.6    0.1    1 none FALSE            TRUE
maxtime support minlen maxlen target  ext
5     0.6      2     10  rules TRUE

Algorithmic control:
 filter tree heap memopt load sort verbose
0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 29305
 set item appearances ...[0 item(s)] done [0.00s]. set transactions
...[115 item(s), 48842 transaction(s)] done [0.09s]. sorting and
recoding items ... [6 item(s)] done [0.01s]. creating transaction
tree ... done [0.03s]. checking subsets of size 1 2 3 4 done
[0.00s]. writing ... [41 rule(s)] done [0.00s]. creating S4 object
... done [0.00s].
> inspect(rul)
     lhs                                  rhs
support confidence  coverage      lift count
```

```
[1]  {sex=Male}                      => {capital-gain=None}
     0.6050735  0.9051455 0.6684820 0.9866565 29553
[2]  {capital-gain=None}             => {sex=Male}
     0.6050735  0.6595621 0.9173867 0.9866565 29553
[3]  {sex=Male}                      => {capital-loss=None}
     0.6331027  0.9470750 0.6684820 0.9934931 30922
[4]  {capital-loss=None}             => {sex=Male}
0.6331027  0.6641323 0.9532779 0.9934931 30922
[5]  {workclass=Private}             => {native-country=United-
     States} 0.6171942  0.8890757 0.6941976 0.9906971 30145
[6]  {native-country=United-States} => {workclass=Private}
     0.6171942  0.6877396 0.8974243 0.9906971 30145
[7]  {workclass=Private}             => {capital-gain=None}
     0.6413742  0.9239073 0.6941976 1.0071078 31326
[8]  {capital-gain=None}             => {workclass=Private}
     0.6413742  0.6991318 0.9173867 1.0071078 31326
[9]  {workclass=Private}             => {capital-loss=None}
     0.6639982  0.9564974 0.6941976 1.0033773 32431
[10] {capital-loss=None}             => {workclass=Private}
0.6639982  0.6965421 0.9532779 1.0033773 32431
[11] {race=White}                    => {native-country=United-
     States}
0.7881127  0.9217231 0.8550428 1.0270761 38493
[12] {native-country=United-States} => {race=White}
     0.7881127  0.8781940 0.8974243 1.0270761 38493
[13] {race=White}                    => {capital-gain=None}
     0.7817862  0.9143240 0.8550428 0.9966616 38184
[14] {capital-gain=None}             => {race=White}
     0.7817862  0.8521883 0.9173867 0.9966616 38184
[15] {race=White}                    => {capital-loss=None}
     0.8136849  0.9516307 0.8550428 0.9982720 39742
[16] {capital-loss=None}             => {race=White}
     0.8136849  0.8535653 0.9532779 0.9982720 39742
[17] {native-country=United-States} => {capital-gain=None}
     0.8219565  0.9159062 0.8974243 0.9983862 40146
[18] {capital-gain=None}             => {native-country=United-
     States} 0.8219565  0.8959761 0.9173867 0.9983862 40146
[19] {native-country=United-States} => {capital-loss=None}
     0.8548380  0.9525461 0.8974243 0.9992323 41752
[20] {capital-loss=None}             => {native-country=United-
     States} 0.8548380  0.8967354 0.9532779 0.9992323 41752
[21] {capital-gain=None}             => {capital-loss=None}
     0.8706646  0.9490705 0.9173867 0.9955863 42525
[22] {capital-loss=None}             => {capital-gain=None}
0.8706646  0.9133376 0.9532779 0.9955863 42525
[23] {workclass=Private,
     capital-gain=None}            => {capital-loss=None}
0.6111748  0.9529145 0.6413742 0.9996188 29851
[24] {workclass=Private,
     capital-loss=None}            => {capital-gain=None}
```

```
0.6111748  0.9204465 0.6639982 1.0033354 29851
[25] {capital-gain=None,
      capital-loss=None}             => {workclass=Private}
0.6111748  0.7019636 0.8706646 1.0111869 29851
[26] {race=White,
      native-country=United-States} => {capital-gain=None}
0.7194628  0.9128933 0.7881127 0.9951019 35140
[27] {race=White,
      capital-gain=None}             => {native-country=United-
      States}
0.7194628  0.9202807 0.7817862 1.0254689 35140
[28] {capital-gain=None,
      native-country=United-States} => {race=White}
0.7194628  0.8753051 0.8219565 1.0236975 35140
[29] {race=White,
      native-country=United-States} => {capital-loss=None}
0.7490480  0.9504325 0.7881127 0.9970152 36585
[30] {race=White,
      capital-loss=None}             => {native-country=United-
      States}
0.7490480  0.9205626 0.8136849 1.0257830 36585
[31] {capital-loss=None,
      native-country=United-States} => {race=White}
0.7490480  0.8762454 0.8548380 1.0247972 36585
[32] {race=White,
      capital-gain=None}             => {capital-loss=None}
0.7404283  0.9470983 0.7817862 0.9935175 36164
[33] {race=White,
      capital-loss=None}             => {capital-gain=None}
0.7404283  0.9099693 0.8136849 0.9919147 36164
[34] {capital-gain=None,
      capital-loss=None}             => {race=White}
0.7404283  0.8504174 0.8706646 0.9945905 36164
[35] {capital-gain=None,
      native-country=United-States} => {capital-loss=None}
0.7793702  0.9481891 0.8219565 0.9946618 38066
[36] {capital-loss=None,
      native-country=United-States} => {capital-gain=None}
0.7793702  0.9117168 0.8548380 0.9938195 38066
[37] {capital-gain=None,
      capital-loss=None}             => {native-country=United-
      States}
0.7793702  0.8951440 0.8706646 0.9974590 38066
[38] {race=White,
      capital-gain=None,
      native-country=United-States} => {capital-loss=None}
0.6803980  0.9457029 0.7194628 0.9920537 33232
[39] {race=White,
      capital-loss=None,
      native-country=United-States} => {capital-gain=None}
```

```
0.6803980  0.9083504 0.7490480 0.9901500 33232
[40] {race=White,
     capital-gain=None,
     capital-loss=None}            => {native-country=United-
     States}
0.6803980  0.9189249 0.7404283 1.0239581 33232
[41] {capital-gain=None,
     capital-loss=None,
     native-country=United-States} => {race=White}
     0.6803980  0.8730100 0.7793702 1.0210133 33232
```

Question 5

Use Naive bayes, K-nearest, and Decision tree classification algorithms and build classifiers.

Divide the dataset into training and test set. Compare the accuracy of the different classifiers under the following situations:

5.1

a) Training set=75% Test set= 25%
b) Training set = 66.6% (2/3rd of total), Test set = 33.3%

5.2 Training set is chosen by
i) holdout method
ii) Random subsampling
iii) Cross-Validation.
Compare the accuracy of the classifiers obtained. 5.3 Data is scaled to standard format.

NAÏVE BAYES

```
library(caret) library(klaR)
library(rpart)
library(rpart.plot)
library(class) library(plyr)
```

a) TRAINING SET 75%

```
       s<-createDataPartition(iris$Species,p =
0.75,list=F) iris_train<- iris[s,] iris_train
```

OUTPUT:

```
Sepal.Length Sepal.Width Petal.Length
1            5.1         3.5          1.4
2            4.9         3.0          1.4
3            4.7         3.2          1.3
6            5.4         3.9          1.7
7            4.6         3.4          1.4
```

8	5.0	3.4	1.5
9	4.4	2.9	1.4
10	4.9	3.1	1.5
12	4.8	3.4	1.6
14	4.3	3.0	1.1
15	5.8	4.0	1.2
16	5.7	4.4	1.5
17	5.4	3.9	1.3
18	5.1	3.5	1.4
19	5.7	3.8	1.7
21	5.4	3.4	1.7
22	5.1	3.7	1.5
23	4.6	3.6	1.0
24	5.1	3.3	1.7
25	4.8	3.4	1.9
27	5.0	3.4	1.6
28	5.2	3.5	1.5
29	5.2	3.4	1.4
30	4.7	3.2	1.6
32	5.4	3.4	1.5
34	5.5	4.2	1.4
35	4.9	3.1	1.5
36	5.0	3.2	1.2
39	4.4	3.0	1.3
40	5.1	3.4	1.5
42	4.5	2.3	1.3
43	4.4	3.2	1.3
44	5.0	3.5	1.6
45	5.1	3.8	1.9
46	4.8	3.0	1.4
48	4.6	3.2	1.4
49	5.3	3.7	1.5
50	5.0	3.3	1.4
51	7.0	3.2	4.7
52	6.4	3.2	4.5
53	6.9	3.1	4.9
54	5.5	2.3	4.0
56	5.7	2.8	4.5
61	5.0	2.0	3.5
62	5.9	3.0	4.2
63	6.0	2.2	4.0
64	6.1	2.9	4.7
65	5.6	2.9	3.6
66	6.7	3.1	4.4
67	5.6	3.0	4.5
68	5.8	2.7	4.1
69	6.2	2.2	4.5
70	5.6	2.5	3.9
71	5.9	3.2	4.8

72	6.1	2.8	4.0
73	6.3	2.5	4.9
74	6.1	2.8	4.7
75	6.4	2.9	4.3
76	6.6	3.0	4.4
77	6.8	2.8	4.8
79	6.0	2.9	4.5
80	5.7	2.6	3.5
81	5.5	2.4	3.8
82	5.5	2.4	3.7
83	5.8	2.7	3.9
84	6.0	2.7	5.1
87	6.7	3.1	4.7
89	5.6	3.0	4.1
90	5.5	2.5	4.0
93	5.8	2.6	4.0
94	5.0	2.3	3.3
95	5.6	2.7	4.2
97	5.7	2.9	4.2
98	6.2	2.9	4.3
99	5.1	2.5	3.0
100	5.7	2.8	4.1
101	6.3	3.3	6.0
102	5.8	2.7	5.1
103	7.1	3.0	5.9
104	6.3	2.9	5.6
105	6.5	3.0	5.8
106	7.6	3.0	6.6
107	4.9	2.5	4.5
108	7.3	2.9	6.3
109	6.7	2.5	5.8
110	7.2	3.6	6.1
111	6.5	3.2	5.1
112	6.4	2.7	5.3
115	5.8	2.8	5.1
116	6.4	3.2	5.3
119	7.7	2.6	6.9
120	6.0	2.2	5.0
121	6.9	3.2	5.7
122	5.6	2.8	4.9
123	7.7	2.8	6.7
124	6.3	2.7	4.9
125	6.7	3.3	5.7
127	6.2	2.8	4.8
128	6.1	3.0	4.9
130	7.2	3.0	5.8
131	7.4	2.8	6.1
132	7.9	3.8	6.4
133	6.4	2.8	5.6

```
136          7.7         3.0          6.1
138          6.4         3.1          5.5
140          6.9         3.1          5.4
142          6.9         3.1          5.1
143          5.8         2.7          5.1
144          6.8         3.2          5.9
146          6.7         3.0          5.2
147          6.3         2.5          5.0
148          6.5         3.0          5.2
149          6.2         3.4          5.4
150          5.9         3.0          5.1
    Petal.Width    Species
1           0.2     setosa
2           0.2     setosa
3           0.2     setosa
6           0.4     setosa
7           0.3     setosa
8           0.2     setosa
9           0.2     setosa
10          0.1     setosa
12          0.2     setosa
14          0.1     setosa
15          0.2     setosa
16          0.4     setosa
17          0.4     setosa
18          0.3     setosa
19          0.3     setosa
21          0.2     setosa
22          0.4     setosa
23          0.2     setosa
24          0.5     setosa
25          0.2     setosa
27          0.4     setosa
28          0.2     setosa
29          0.2     setosa
30          0.2     setosa
32          0.4     setosa
34          0.2     setosa
35          0.2     setosa
36          0.2     setosa
39          0.2     setosa
40          0.2     setosa
42          0.3     setosa
43          0.2     setosa
44          0.6     setosa
45          0.4     setosa
46          0.3     setosa
48          0.2     setosa
49          0.2     setosa
50          0.2     setosa
```

```
51          1.4 versicolor
52          1.5 versicolor
53          1.5 versicolor
54          1.3 versicolor
56          1.3 versicolor
61          1.0 versicolor
62          1.5 versicolor
63          1.0 versicolor
64          1.4 versicolor
65          1.3 versicolor
66          1.4 versicolor
67          1.5 versicolor
68          1.0 versicolor
69          1.5 versicolor
70          1.1 versicolor
71          1.8 versicolor
72          1.3 versicolor
73          1.5 versicolor74          1.2 versicolor 75
            1.3 versicolor
76          1.4 versicolor
77          1.4 versicolor
79          1.5 versicolor 80          1.0 versicolor
81          1.1 versicolor
82          1.0 versicolor
83          1.2 versicolor
84          1.6 versicolor
87          1.5 versicolor
89          1.3 versicolor
90          1.3 versicolor
93          1.2 versicolor
94          1.0 versicolor
95          1.3 versicolor
97          1.3 versicolor
98          1.3 versicolor
99          1.1 versicolor
100         1.3 versicolor
101         2.5  virginica
102         1.9  virginica
103         2.1  virginica
104         1.8  virginica
105         2.2  virginica
106         2.1  virginica
107         1.7  virginica
108         1.8  virginica
109         1.8  virginica
110         2.5  virginica
111         2.0  virginica
112         1.9  virginica
115         2.4  virginica
```

```
116          2.3  virginica
119          2.3  virginica
120          1.5  virginica
121          2.3  virginica
122          2.0  virginica
123          2.0  virginica
124          1.8  virginica
125          2.1  virginica
127          1.8  virginica
128          1.8  virginica
130          1.6  virginica
131          1.9  virginica
132          2.0  virginica
133          2.2  virginica
136          2.3  virginica
138          1.8  virginica
140          2.1  virginica
142          2.3  virginica
143          1.9  virginica
144          2.3  virginica
146          2.3  virginica
147          1.9  virginica148         2.0  virginica
149          2.3  virginica
150          1.8  virginica
```

```
iris_test<- iris[-s,] iris_test
```

OUTPUT:

```
    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
4            4.6         3.1          1.5         0.2     setosa
5            5.0         3.6          1.4         0.2     setosa
11           5.4         3.7          1.5         0.2     setosa
13           4.8         3.0          1.4         0.1     setosa
20           5.1         3.8          1.5         0.3     setosa
26           5.0         3.0          1.6         0.2     setosa
31           4.8         3.1          1.6         0.2     setosa
33           5.2         4.1          1.5         0.1     setosa
37           5.5         3.5          1.3         0.2     setosa
38           4.9         3.6          1.4         0.1     setosa
41           5.0         3.5          1.3         0.3     setosa
47           5.1         3.8          1.6         0.2     setosa
55           6.5         2.8          4.6         1.5 versicolor
57           6.3         3.3          4.7         1.6 versicolor
58           4.9         2.4          3.3         1.0 versicolor
59           6.6         2.9          4.6         1.3 versicolor
60           5.2         2.7          3.9         1.4 versicolor
78           6.7         3.0          5.0         1.7 versicolor
85           5.4         3.0          4.5         1.5 versicolor
86           6.0         3.4          4.5         1.6 versicolor
```

```
88           6.3         2.3          4.4         1.3 versicolor
91           5.5         2.6          4.4         1.2 versicolor
92           6.1         3.0          4.6         1.4 versicolor
96           5.7         3.0          4.2         1.2 versicolor
113          6.8         3.0          5.5         2.1  virginica
114          5.7         2.5          5.0         2.0  virginica
117          6.5         3.0          5.5         1.8  virginica
118          7.7         3.8          6.7         2.2  virginica
126          7.2         3.2          6.0         1.8  virginica
129          6.4         2.8          5.6         2.1  virginica
134          6.3         2.8          5.1         1.5  virginica
135          6.1         2.6          5.6         1.4  virginica
137          6.3         3.4          5.6         2.4  virginica
139          6.0         3.0          4.8         1.8  virginica
141          6.7         3.1          5.6         2.4  virginica
145          6.7         3.3          5.7         2.5  virginica
```

```
model<- NaiveBayes(Species~., data=iris_train)
model
apriori
grouping
    setosa versicolor  virginica
0.3333333  0.3333333  0.3333333

$tables
$tables$Sepal.Length
[,1]      [,2] setosa
4.997368 0.3809567 versicolor
5.936842 0.5004976 virginica
6.605263 0.6685795

$tables$Sepal.Width
[,1]      [,2] setosa
3.410526 0.3881971 versicolor
2.739474 0.3071574 virginica
2.952632 0.3125629

$tables$Petal.Length
[,1]      [,2] setosa
1.460526 0.1910702 versicolor
4.218421 0.4780904 virginica
5.550000 0.5764523

$tables$Petal.Width
[,1]      [,2] setosa
0.2631579 0.1100885 versicolor
1.3052632 0.1944464 virginica
2.0342105 0.2507031
```

```
$levels
[1] "setosa"     "versicolor" "virginica"

$call
NaiveBayes.default(x = X, grouping = Y)

$x
    Sepal.Length Sepal.Width Petal.Length Petal.Width
1            5.1         3.5          1.4         0.2
2            4.9         3.0          1.4         0.2
3            4.7         3.2          1.3         0.2 6
             5.4         3.9          1.7         0.4
7            4.6         3.4          1.4         0.3
8            5.0         3.4          1.5         0.2
9            4.4         2.9          1.4         0.2
10           4.9         3.1          1.5         0.1
12           4.8         3.4          1.6         0.2 14
4.3         3.0          1.1         0.1
15           5.8         4.0          1.2         0.2
16           5.7         4.4          1.5         0.4
17           5.4         3.9          1.3         0.4 18
             5.1         3.5          1.4         0.3
19           5.7         3.8          1.7         0.3
21           5.4         3.4          1.7         0.2
22           5.1         3.7          1.5         0.4
23           4.6         3.6          1.0         0.2
24           5.1         3.3          1.7         0.5
25           4.8         3.4          1.9         0.2
27           5.0         3.4          1.6         0.4
28           5.2         3.5          1.5         0.2
29           5.2         3.4          1.4         0.2
30           4.7         3.2          1.6         0.2
32           5.4         3.4          1.5         0.4
34           5.5         4.2          1.4         0.2
35           4.9         3.1          1.5         0.2
36           5.0         3.2          1.2         0.2
39           4.4         3.0          1.3         0.2
40           5.1         3.4          1.5         0.2
42           4.5         2.3          1.3         0.3
43           4.4         3.2          1.3         0.2
44           5.0         3.5          1.6         0.6
45           5.1         3.8          1.9         0.4
46           4.8         3.0          1.4         0.3
48           4.6         3.2          1.4         0.2
49           5.3         3.7          1.5         0.2
50           5.0         3.3          1.4         0.2
51           7.0         3.2          4.7         1.4
52           6.4         3.2          4.5         1.5
53           6.9         3.1          4.9         1.5
54           5.5         2.3          4.0         1.3
```

56	5.7	2.8	4.5	1.3	
61	5.0	2.0	3.5	1.0	
62	5.9	3.0	4.2	1.5	
63	6.0	2.2	4.0	1.0	
64	6.1	2.9	4.7	1.4	
65	5.6	2.9	3.6	1.3	
66	6.7	3.1	4.4	1.4	
67	5.6	3.0	4.5	1.5	
68	5.8	2.7	4.1	1.0	
69	6.2	2.2	4.5	1.5	
70	5.6	2.5	3.9	1.1	
71	5.9	3.2	4.8	1.8	
72	6.1	2.8	4.0	1.3	
73	6.3	2.5	4.9	1.5	74
	6.1	2.8	4.7	1.2	75
	6.4	2.9	4.3	1.3	
76	6.6	3.0	4.4	1.4	77
6.8	2.8	4.8	1.4 79		6.0
2.9	4.5	1.5			
80	5.7	2.6	3.5	1.0	
81	5.5	2.4	3.8	1.1	
82	5.5	2.4	3.7	1.0	
83	5.8	2.7	3.9	1.2	
84	6.0	2.7	5.1	1.6	
87	6.7	3.1	4.7	1.5	
89	5.6	3.0	4.1	1.3	
90	5.5	2.5	4.0	1.3	
93	5.8	2.6	4.0	1.2	
94	5.0	2.3	3.3	1.0	
95	5.6	2.7	4.2	1.3	
97	5.7	2.9	4.2	1.3	
98	6.2	2.9	4.3	1.3	
99	5.1	2.5	3.0	1.1	
100	5.7	2.8	4.1	1.3	
101	6.3	3.3	6.0	2.5	
102	5.8	2.7	5.1	1.9	
103	7.1	3.0	5.9	2.1	
104	6.3	2.9	5.6	1.8	
105	6.5	3.0	5.8	2.2	
106	7.6	3.0	6.6	2.1	
107	4.9	2.5	4.5	1.7	
108	7.3	2.9	6.3	1.8	
109	6.7	2.5	5.8	1.8	
110	7.2	3.6	6.1	2.5	
111	6.5	3.2	5.1	2.0	
112	6.4	2.7	5.3	1.9	
115	5.8	2.8	5.1	2.4	
116	6.4	3.2	5.3	2.3	
119	7.7	2.6	6.9	2.3	
120	6.0	2.2	5.0	1.5	

```
121          6.9         3.2          5.7         2.3
122          5.6         2.8          4.9         2.0
123          7.7         2.8          6.7         2.0
124          6.3         2.7          4.9         1.8
125          6.7         3.3          5.7         2.1
127          6.2         2.8          4.8         1.8
128          6.1         3.0          4.9         1.8
130          7.2         3.0          5.8         1.6
131          7.4         2.8          6.1         1.9
132          7.9         3.8          6.4         2.0
133          6.4         2.8          5.6         2.2 136
             7.7         3.0          6.1         2.3
138          6.4         3.1          5.5         1.8
140          6.9         3.1          5.4         2.1
142          6.9         3.1          5.1         2.3
143          5.8         2.7          5.1         1.9 144
             6.8         3.2          5.9         2.3 146
             6.7         3.0          5.2         2.3
147          6.3         2.5          5.0         1.9
148          6.5         3.0          5.2         2.0
149          6.2         3.4          5.4         2.3
150          5.9         3.0          5.1         1.8

$usekernel
[1] FALSE

$varnames
[1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"
 attr(,"class")
[1] "NaiveBayes"
```

```
pred<- predict(model, iris_test) pred$class
```

```
4          5          11          13          20          26          31
33         37
    setosa     setosa     setosa     setosa     setosa     setosa
setosa     setosa     setosa
        38         41         47         55         57         58
59         60         78
    setosa     setosa     setosa versicolor versicolor versicolor
versicolor versicolor  virginica
        85         86         88         91         92         96
113        114        117
versicolor versicolor versicolor versicolor versicolor versicolor
virginica  virginica  virginica
       118        126        129        134        135        137
139        141        145
 virginica  virginica  virginica versicolor versicolor  virginica
virginica  virginica  virginica
Levels: setosa versicolor virginica
```

```
conf<- confusionMatrix(pred$class ,iris_test$Species)$table
conf
```

```
Reference
Prediction   setosa versicolor virginica
setosa         12          0         0
versicolor      0         11         2
virginica       0          1        10
```

```
conf1<-as.matrix(conf)
d<-diag(conf1) s<-sum(d)
s1<-sum(conf1)
accuracy1<-(s/s1)*100
accuracy1
91.66667
```

b) TRAINING SET 66.6%(2/3)

```
s<-createDataPartition(iris$Species,p = 0.66,list=F)
iris_train<- iris[s,] iris_train
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
11	5.4	3.7	1.5	0.2	setosa
12	4.8	3.4	1.6	0.2	setosa
13	4.8	3.0	1.4	0.1	setosa
14	4.3	3.0	1.1	0.1	setosa
15	5.8	4.0	1.2	0.2	setosa
16	5.7	4.4	1.5	0.4	setosa
17	5.4	3.9	1.3	0.4	setosa
19	5.7	3.8	1.7	0.3	setosa
20	5.1	3.8	1.5	0.3	setosa
21	5.4	3.4	1.7	0.2	setosa
24	5.1	3.3	1.7	0.5	setosa
26	5.0	3.0	1.6	0.2	setosa
28	5.2	3.5	1.5	0.2	setosa
29	5.2	3.4	1.4	0.2	setosa
30	4.7	3.2	1.6	0.2	setosa
31	4.8	3.1	1.6	0.2	setosa
32	5.4	3.4	1.5	0.4	setosa
33	5.2	4.1	1.5	0.1	setosa
35	4.9	3.1	1.5	0.2	setosa
37	5.5	3.5	1.3	0.2	setosa

```
39          4.4        3.0         1.3        0.2     setosa
40          5.1        3.4         1.5        0.2     setosa
41          5.0        3.5         1.3        0.3     setosa
42          4.5        2.3         1.3        0.3
            setosa45         5.1        3.8         1.9
            0.4     setosa
46          4.8        3.0         1.4        0.3     setosa
48          4.6        3.2         1.4        0.2     setosa
51          7.0        3.2         4.7        1.4 versicolor
52          6.4        3.2         4.5        1.5 versicolor
53          6.9        3.1         4.9        1.5 versicolor
54          5.5        2.3         4.0        1.3
            versicolor55         6.5        2.8         4.6
            1.5 versicolor 57         6.3        3.3
            4.7        1.6 versicolor 58        4.9
            2.4         3.3        1.0 versicolor
61          5.0        2.0         3.5        1.0 versicolor
64          6.1        2.9         4.7        1.4 versicolor
65          5.6        2.9         3.6        1.3 versicolor
66          6.7        3.1         4.4        1.4 versicolor
67          5.6        3.0         4.5        1.5 versicolor
68          5.8        2.7         4.1        1.0 versicolor
69          6.2        2.2         4.5        1.5 versicolor
70          5.6        2.5         3.9        1.1 versicolor
73          6.3        2.5         4.9        1.5 versicolor
74          6.1        2.8         4.7        1.2 versicolor
75          6.4        2.9         4.3        1.3 versicolor
77          6.8        2.8         4.8        1.4 versicolor
78          6.7        3.0         5.0        1.7 versicolor
79          6.0        2.9         4.5        1.5 versicolor
81          5.5        2.4         3.8        1.1 versicolor
82          5.5        2.4         3.7        1.0 versicolor
86          6.0        3.4         4.5        1.6 versicolor
87          6.7        3.1         4.7        1.5 versicolor
89          5.6        3.0         4.1        1.3 versicolor
90          5.5        2.5         4.0        1.3 versicolor
92          6.1        3.0         4.6        1.4 versicolor
93          5.8        2.6         4.0        1.2 versicolor
94          5.0        2.3         3.3        1.0 versicolor
95          5.6        2.7         4.2        1.3 versicolor
98          6.2        2.9         4.3        1.3 versicolor
100         5.7        2.8         4.1        1.3 versicolor
102         5.8        2.7         5.1        1.9  virginica
103         7.1        3.0         5.9        2.1  virginica
104         6.3        2.9         5.6        1.8  virginica
106         7.6        3.0         6.6        2.1  virginica
109         6.7        2.5         5.8        1.8  virginica
110         7.2        3.6         6.1        2.5  virginica
111         6.5        3.2         5.1        2.0  virginica
112         6.4        2.7         5.3        1.9  virginica
```

```
113           6.8         3.0          5.5         2.1  virginica
114           5.7         2.5          5.0         2.0  virginica
115           5.8         2.8          5.1         2.4  virginica
116           6.4         3.2          5.3         2.3  virginica
117           6.5         3.0          5.5         1.8  virginica
118           7.7         3.8          6.7         2.2  virginica
119           7.7         2.6          6.9         2.3
              virginica120          6.0         2.2          5.0
              1.5  virginica
121           6.9         3.2          5.7         2.3  virginica
122           5.6         2.8          4.9         2.0  virginica
125           6.7         3.3          5.7         2.1  virginica
129           6.4         2.8          5.6         2.1  virginica
130           7.2         3.0          5.8         1.6  virginica
131           7.4         2.8          6.1         1.9  virginica
132           7.9         3.8          6.4         2.0  virginica
133           6.4         2.8          5.6         2.2  virginica
134           6.3         2.8          5.1         1.5  virginica
137           6.3         3.4          5.6         2.4  virginica
138           6.4         3.1          5.5         1.8
              virginica139          6.0         3.0          4.8
              1.8  virginica 144          6.8         3.2
              5.9         2.3  virginica
146           6.7         3.0          5.2         2.3  virginica
148           6.5         3.0          5.2         2.0  virginica
149           6.2         3.4          5.4         2.3  virginica
150           5.9         3.0          5.1         1.8  virginica
```

```
iris_test<- iris[-s,] iris_test
```

```
    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
1             5.1         3.5          1.4         0.2     setosa
8             5.0         3.4          1.5         0.2     setosa
9             4.4         2.9          1.4         0.2     setosa
10            4.9         3.1          1.5         0.1     setosa
18            5.1         3.5          1.4         0.3     setosa
22            5.1         3.7          1.5         0.4     setosa
23            4.6         3.6          1.0         0.2     setosa
25            4.8         3.4          1.9         0.2     setosa
27            5.0         3.4          1.6         0.4     setosa
34            5.5         4.2          1.4         0.2     setosa
36            5.0         3.2          1.2         0.2     setosa
38            4.9         3.6          1.4         0.1     setosa
43            4.4         3.2          1.3         0.2     setosa
44            5.0         3.5          1.6         0.6     setosa
47            5.1         3.8          1.6         0.2     setosa
49            5.3         3.7          1.5         0.2     setosa
50            5.0         3.3          1.4         0.2     setosa
56            5.7         2.8          4.5         1.3 versicolor
59            6.6         2.9          4.6         1.3 versicolor
```

```
60          5.2         2.7         3.9         1.4 versicolor
62          5.9         3.0         4.2         1.5 versicolor
63          6.0         2.2         4.0         1.0 versicolor
71          5.9         3.2         4.8         1.8 versicolor
72          6.1         2.8         4.0         1.3 versicolor
76          6.6         3.0         4.4         1.4 versicolor
80          5.7         2.6         3.5         1.0 versicolor
83          5.8         2.7         3.9         1.2 versicolor
84          6.0         2.7         5.1         1.6
            versicolor85          5.4         3.0         4.5
            1.5 versicolor
88          6.3         2.3         4.4         1.3 versicolor
91          5.5         2.6         4.4         1.2 versicolor
96          5.7         3.0         4.2         1.2 versicolor
97          5.7         2.9         4.2         1.3 versicolor
99          5.1         2.5         3.0         1.1 versicolor
101         6.3         3.3         6.0         2.5  virginica
105         6.5         3.0         5.8         2.2  virginica
107         4.9         2.5         4.5         1.7  virginica
108         7.3         2.9         6.3         1.8  virginica
123         7.7         2.8         6.7         2.0  virginica
124         6.3         2.7         4.9         1.8
            virginica126         7.2         3.2         6.0
            1.8  virginica
127         6.2         2.8         4.8         1.8  virginica
128         6.1         3.0         4.9         1.8  virginica
135         6.1         2.6         5.6         1.4  virginica
136         7.7         3.0         6.1         2.3  virginica
140         6.9         3.1         5.4         2.1  virginica
141         6.7         3.1         5.6         2.4  virginica
142         6.9         3.1         5.1         2.3  virginica
143         5.8         2.7         5.1         1.9  virginica
145         6.7         3.3         5.7         2.5  virginica
147         6.3         2.5         5.0         1.9  virginica

model<- NaiveBayes(Species~., data=iris_train)
model  $apriori grouping
    setosa versicolor  virginica
0.3333333  0.3333333  0.3333333

$tables
$tables$Sepal.Length
[,1]      [,2] setosa
5.033333 0.3845994 versicolor
5.987879 0.5577620 virginica
6.600000 0.6093029

$tables$Sepal.Width
[,1]      [,2] setosa
3.406061 0.4167879 versicolor
```

```
2.775758 0.3400646 virginica
3.003030 0.3513233

$tables$Petal.Length
[,1]      [,2] setosa
1.469697 0.1667424 versicolor
4.284848 0.4657797 virginica
5.578788 0.5295975

$tables$Petal.Width
                [,1]       [,2] setosa
0.2484848 0.09721501 versicolor
1.3303030 0.19603069 virginica
2.0333333 0.25819889


$levels
[1] "setosa"     "versicolor" "virginica"

$call
NaiveBayes.default(x = X, grouping = Y)

$x
    Sepal.Length Sepal.Width Petal.Length Petal.Width
2            4.9         3.0          1.4         0.2
3            4.7         3.2          1.3         0.2
4            4.6         3.1          1.5         0.2
5            5.0         3.6          1.4         0.2
6            5.4         3.9          1.7         0.4
7            4.6         3.4          1.4         0.3
11           5.4         3.7          1.5         0.2
12           4.8         3.4          1.6         0.2
13           4.8         3.0          1.4         0.1
14           4.3         3.0          1.1         0.1
15           5.8         4.0          1.2         0.2
16           5.7         4.4          1.5         0.4
17           5.4         3.9          1.3         0.4
19           5.7         3.8          1.7         0.3
20           5.1         3.8          1.5         0.3
21           5.4         3.4          1.7         0.2
24           5.1         3.3          1.7         0.5
26           5.0         3.0          1.6         0.2
28           5.2         3.5          1.5         0.2
29           5.2         3.4          1.4         0.2
30           4.7         3.2          1.6         0.2
31           4.8         3.1          1.6         0.2
32           5.4         3.4          1.5         0.4
33           5.2         4.1          1.5         0.1
35           4.9         3.1          1.5         0.2
37           5.5         3.5          1.3         0.2
```

39	4.4	3.0	1.3	0.2
40	5.1	3.4	1.5	0.2
41	5.0	3.5	1.3	0.3
42	4.5	2.3	1.3	0.3
45	5.1	3.8	1.9	0.4
46	4.8	3.0	1.4	0.3
48	4.6	3.2	1.4	0.2
51	7.0	3.2	4.7	1.4
52	6.4	3.2	4.5	1.5
53	6.9	3.1	4.9	1.5
54	5.5	2.3	4.0	1.3
55	6.5	2.8	4.6	1.5
57	6.3	3.3	4.7	1.6
58	4.9	2.4	3.3	1.0
61	5.0	2.0	3.5	1.0
64	6.1	2.9	4.7	1.4
65	5.6	2.9	3.6	1.3
66	6.7	3.1	4.4	1.4
67	5.6	3.0	4.5	1.5
68	5.8	2.7	4.1	1.0
69	6.2	2.2	4.5	1.5
70	5.6	2.5	3.9	1.1
73	6.3	2.5	4.9	1.5
74	6.1	2.8	4.7	1.2
75	6.4	2.9	4.3	1.3
77	6.8	2.8	4.8	1.4
78	6.7	3.0	5.0	1.7
79	6.0	2.9	4.5	1.5
81	5.5	2.4	3.8	1.1
82	5.5	2.4	3.7	1.0
86	6.0	3.4	4.5	1.6
87	6.7	3.1	4.7	1.5
89	5.6	3.0	4.1	1.3
90	5.5	2.5	4.0	1.3
92	6.1	3.0	4.6	1.4
93	5.8	2.6	4.0	1.2
94	5.0	2.3	3.3	1.0
95	5.6	2.7	4.2	1.3
98	6.2	2.9	4.3	1.3
100	5.7	2.8	4.1	1.3
102	5.8	2.7	5.1	1.9
103	7.1	3.0	5.9	2.1
104	6.3	2.9	5.6	1.8
106	7.6	3.0	6.6	2.1
109	6.7	2.5	5.8	1.8
110	7.2	3.6	6.1	2.5
111	6.5	3.2	5.1	2.0
112	6.4	2.7	5.3	1.9
113	6.8	3.0	5.5	2.1

```
114          5.7         2.5         5.0         2.0
115          5.8         2.8         5.1         2.4
116          6.4         3.2         5.3         2.3
117          6.5         3.0         5.5         1.8
118          7.7         3.8         6.7         2.2
119          7.7         2.6         6.9         2.3
120          6.0         2.2         5.0         1.5
121          6.9         3.2         5.7         2.3
122          5.6         2.8         4.9         2.0
125          6.7         3.3         5.7         2.1
129          6.4         2.8         5.6         2.1
130          7.2         3.0         5.8         1.6
131          7.4         2.8         6.1         1.9
132          7.9         3.8         6.4         2.0 133
             6.4         2.8         5.6         2.2
134          6.3         2.8         5.1         1.5
137          6.3         3.4         5.6         2.4
138          6.4         3.1         5.5         1.8
139          6.0         3.0         4.8         1.8
144          6.8         3.2         5.9         2.3
146          6.7         3.0         5.2         2.3
148          6.5         3.0         5.2         2.0
149          6.2         3.4         5.4         2.3
150          5.9         3.0         5.1         1.8

$usekernel
[1] FALSE

$varnames
[1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"
 attr(,"class")
[1] "NaiveBayes"

pred<- predict(model, iris_test) pred$class
        1          8          9         10         18         22
23         25         27
    setosa     setosa     setosa     setosa     setosa     setosa
setosa     setosa     setosa
        34         36         38         43         44         47
49         50         56
    setosa     setosa     setosa     setosa     setosa     setosa
setosa     setosa versicolor
        59         60         62         63         71         72
76         80         83
versicolor versicolor versicolor versicolor  virginica versicolor
versicolor versicolor versicolor
        84         85         88         91         96         97
99        101        105
versicolor versicolor versicolor versicolor versicolor versicolor
versicolor  virginica  virginica
```

```
       107        108        123        124        126        127
128        135        136
versicolor  virginica  virginica  virginica  virginica  virginica
virginica versicolor  virginica
       140        141        142        143        145        147
virginica  virginica  virginica  virginica  virginica  virginica
Levels: setosa versicolor virginica
```

```
conf<- confusionMatrix(pred$class ,iris_test$Species)$table
conf Reference
Prediction   setosa versicolor virginica
setosa           17          0         0
versicolor        0         16         2
virginica         0          1        15
```

```
  conf1<-
as.matrix(conf) d<-
diag(conf1) s<-sum(d)
s1<-sum(conf1)
accuracy2<-(s/s1)*100
accuracy2
94.11765
```

```
if(accuracy1>accuracy2)
{
  cat("Training set of 75% records is better")
}
if(accuracy2>accuracy1)
{
  cat("Training set of 66.6% records is better")
}

Training set of 66.6% records is better
```

HOLD OUT METHOD

```
 s<-sample(150,75)
iris_train<-iris[s,]
iris_test<-iris[-s,]
model<-NaiveBayes(Species~.,data=iris_train) pred<-
predict(model,iris_test)
conf<-confusionMatrix(pred$class ,iris_test$Species)$table
conf1<-as.matrix(conf)
acchld<-((sum(diag(conf1)))/sum(conf1))*100 cat("Accuracy
of holdout method is ",acchld)

Accuracy of holdout method is  94.66667
```

RANDOM SUBSAMPLING

```
 i<-75 j<-1
acc<-c() for(i
in 75:100)
{   s<-sample(150,i)
iris_train<-iris[s,]
iris_test<-iris[-s,]
  model<-NaiveBayes(Species~.,data=iris_train)   pred<-
predict(model,iris_test)
  conf<-confusionMatrix(pred$class,iris_test$Species)$table
conf1<-as.matrix(conf)
  acc[j]<-c((sum(diag(conf1))/sum(conf1))*100,acc)
j=j+1 } accrs<-mean(acc)
cat("Accuracy of random subsampling is ",accrs)

Accuracy of random subsampling is  95.5321
```

CROSS VALIDATION METHOD

```
 x=iris[,-5]
y=iris$Species
model = train(x,y,'nb',trControl=trainControl(method='cv',number=10))
cons<-table(predict(model$finalModel,x)$class,y) cons1<-as.matrix(cons)
accv<-(sum(diag(cons1))/sum(cons1))*100 cat("Accuracy
of cross validation is ",accv)

Accuracy of cross validation is  96
```

COMPARISON

```
greatest<-max(acchld,accrs,accv) if(greatest==acchld)
{
  cat("Holdout method does best classification")
}
if(greatest==accrs)
{
  cat("Random subsampling method does best classification")
}
if(greatest==accv)
{
  cat("Cross validation method does best classification")
}
Cross validation method does best classification
```

DATA SCALING

```
data_std <- function(x)
{
  (x-mean(x))/sd(x)
}
sapply(iris[,-5],data_std)
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
[1,]	-0.89767388	1.01560199	-1.33575163	-1.3110521482
[2,]	-1.13920048	-0.13153881	-1.33575163	-1.3110521482
[3,]	-1.38072709	0.32731751	-1.39239929	-1.3110521482
[4,]	-1.50149039	0.09788935	-1.27910398	-1.3110521482
[5,]	-1.01843718	1.24503015	-1.33575163	-1.3110521482
[6,]	-0.53538397	1.93331463	-1.16580868	-1.0486667950
[7,]	-1.50149039	0.78617383	-1.33575163	-1.1798594716
[8,]	-1.01843718	0.78617383	-1.27910398	-1.3110521482
[9,]	-1.74301699	-0.36096697	-1.33575163	-1.3110521482
[10,]	-1.13920048	0.09788935	-1.27910398	-1.4422448248
[11,]	-0.53538397	1.47445831	-1.27910398	-1.3110521482
[12,]	-1.25996379	0.78617383	-1.22245633	-1.3110521482
[13,]	-1.25996379	-0.13153881	-1.33575163	-1.4422448248
[14,]	-1.86378030	-0.13153881	-1.50569459	-1.4422448248
[15,]	-0.05233076	2.16274279	-1.44904694	-1.3110521482
[16,]	-0.17309407	3.08045544	-1.27910398	-1.0486667950
[17,]	-0.53538397	1.93331463	-1.39239929	-1.0486667950
[18,]	-0.89767388	1.01560199	-1.33575163	-1.1798594716
[19,]	-0.17309407	1.70388647	-1.16580868	-1.1798594716
[20,]	-0.89767388	1.70388647	-1.27910398	-1.1798594716
[21,]	-0.53538397	0.78617383	-1.16580868	-1.3110521482
[22,]	-0.89767388	1.47445831	-1.27910398	-1.0486667950
[23,]	-1.50149039	1.24503015	-1.56234224	-1.3110521482
[24,]	-0.89767388	0.55674567	-1.16580868	-0.9174741184
[25,]	-1.25996379	0.78617383	-1.05251337	-1.3110521482
[26,]	-1.01843718	-0.13153881	-1.22245633	-1.3110521482
[27,]	-1.01843718	0.78617383	-1.22245633	-1.0486667950
[28,]	-0.77691058	1.01560199	-1.27910398	-1.3110521482
[29,]	-0.77691058	0.78617383	-1.33575163	-1.3110521482
[30,]	-1.38072709	0.32731751	-1.22245633	-1.3110521482
[31,]	-1.25996379	0.09788935	-1.22245633	-1.3110521482
[32,]	-0.53538397	0.78617383	-1.27910398	-1.0486667950
[33,]	-0.77691058	2.39217095	-1.27910398	-1.4422448248
[34,]	-0.41462067	2.62159911	-1.33575163	-1.3110521482
[35,]	-1.13920048	0.09788935	-1.27910398	-1.3110521482
[36,]	-1.01843718	0.32731751	-1.44904694	-1.3110521482
[37,]	-0.41462067	1.01560199	-1.39239929	-1.3110521482
[38,]	-1.13920048	1.24503015	-1.33575163	-1.4422448248
[39,]	-1.74301699	-0.13153881	-1.39239929	-1.3110521482
[40,]	-0.89767388	0.78617383	-1.27910398	-1.3110521482
[41,]	-1.01843718	1.01560199	-1.39239929	-1.1798594716
[42,]	-1.62225369	-1.73753594	-1.39239929	-1.1798594716
[43,]	-1.74301699	0.32731751	-1.39239929	-1.3110521482
[44,]	-1.01843718	1.01560199	-1.22245633	-0.7862814418
[45,]	-0.89767388	1.70388647	-1.05251337	-1.0486667950
[46,]	-1.25996379	-0.13153881	-1.33575163	-1.1798594716
[47,]	-0.89767388	1.70388647	-1.22245633	-1.3110521482

```
 [48,]  -1.50149039  0.32731751  -1.33575163 -1.3110521482  [49,]
-0.65614727  1.47445831  -1.27910398 -1.3110521482
 [50,]  -1.01843718  0.55674567  -1.33575163 -1.3110521482
[51,]    1.39682886  0.32731751   0.53362088  0.2632599711
[52,]    0.67224905  0.32731751   0.42032558  0.3944526477
[53,]   1.27606556  0.09788935   0.64691619  0.3944526477
 [54,]  -0.41462067 -1.73753594   0.13708732  0.1320672944
 [55,]   0.79301235 -0.59039513   0.47697323  0.3944526477
[56,]  -0.17309407 -0.59039513   0.42032558  0.1320672944
 [57,]   0.55148575  0.55674567   0.53362088  0.5256453243
 [58,]  -1.13920048 -1.50810778  -0.25944625 -0.2615107354
 [59,]   0.91377565 -0.36096697   0.47697323  0.1320672944
 [60,]  -0.77691058 -0.81982329   0.08043967  0.2632599711
 [61,]  -1.01843718 -2.42582042  -0.14615094 -0.2615107354
 [62,]   0.06843254 -0.13153881   0.25038262  0.3944526477
 [63,]   0.18919584 -1.96696410   0.13708732 -0.2615107354
 [64,]   0.30995914 -0.36096697   0.53362088  0.2632599711
 [65,]  -0.29385737 -0.36096697  -0.08950329  0.1320672944
 [66,]   1.03453895  0.09788935   0.36367793  0.2632599711
 [67,]  -0.29385737 -0.13153881   0.42032558  0.3944526477
 [68,]  -0.05233076 -0.81982329   0.19373497 -0.2615107354
 [69,]   0.43072244 -1.96696410   0.42032558  0.3944526477
 [70,]  -0.29385737 -1.27867961   0.08043967 -0.1303180588
 [71,]   0.06843254  0.32731751   0.59026853  0.7880306775
 [72,]   0.30995914 -0.59039513   0.13708732  0.1320672944
 [73,]   0.55148575 -1.27867961   0.64691619  0.3944526477
 [74,]   0.30995914 -0.59039513   0.53362088  0.0008746178
 [75,]   0.67224905 -0.36096697   0.30703027  0.1320672944
 [76,]   0.91377565 -0.13153881   0.36367793  0.2632599711
 [77,]   1.15530226 -0.59039513   0.59026853  0.2632599711
 [78,]   1.03453895 -0.13153881   0.70356384  0.6568380009
 [79,]   0.18919584 -0.36096697   0.42032558  0.3944526477
 [80,]  -0.17309407 -1.04925145  -0.14615094 -0.2615107354
 [81,]  -0.41462067 -1.50810778   0.02379201 -0.1303180588
 [82,]  -0.41462067 -1.50810778  -0.03285564 -0.2615107354
 [83,]  -0.05233076 -0.81982329   0.08043967  0.0008746178
 [84,]   0.18919584 -0.81982329   0.76021149  0.5256453243
 [85,]  -0.53538397 -0.13153881   0.42032558  0.3944526477
 [86,]   0.18919584  0.78617383   0.42032558  0.5256453243
 [87,]   1.03453895  0.09788935   0.53362088  0.3944526477
 [88,]   0.55148575 -1.73753594   0.36367793  0.1320672944
 [89,]  -0.29385737 -0.13153881   0.19373497  0.1320672944
 [90,]  -0.41462067 -1.27867961   0.13708732  0.1320672944
 [91,]  -0.41462067 -1.04925145   0.36367793  0.0008746178
 [92,]   0.30995914 -0.13153881   0.47697323  0.2632599711
 [93,]  -0.05233076 -1.04925145   0.13708732  0.0008746178
[94,]  -1.01843718 -1.73753594  -0.25944625 -0.2615107354
 [95,]  -0.29385737 -0.81982329   0.25038262  0.1320672944
 [96,]  -0.17309407 -0.13153881   0.25038262  0.0008746178
 [97,]  -0.17309407 -0.36096697   0.25038262  0.1320672944
```

```
 [98,]   0.43072244 -0.36096697   0.30703027  0.1320672944  [99,]
-0.89767388 -1.27867961  -0.42938920 -0.1303180588 [100,]  -
0.17309407 -0.59039513   0.19373497  0.1320672944 [101,]
0.55148575  0.55674567   1.27004036  1.7063794137
[102,]  -0.05233076 -0.81982329   0.76021149  0.9192233541
[103,]   1.51759216 -0.13153881   1.21339271  1.1816087073
[104,]   0.55148575 -0.36096697   1.04344975  0.7880306775
[105,]   0.79301235 -0.13153881   1.15674505  1.3128013839
[106,]   2.12140867 -0.13153881   1.60992627  1.1816087073
[107,]  -1.13920048 -1.27867961   0.42032558  0.6568380009
[108,]   1.75911877 -0.36096697   1.43998331  0.7880306775
[109,]   1.03453895 -1.27867961   1.15674505  0.7880306775
[110,]   1.63835547  1.24503015   1.32668801  1.7063794137
[111,]   0.79301235  0.32731751   0.76021149  1.0504160307
[112,]   0.67224905 -0.81982329   0.87350679  0.9192233541
[113,]   1.15530226 -0.13153881   0.98680210  1.1816087073
[114,]  -0.17309407 -1.27867961   0.70356384  1.0504160307
[115,]  -0.05233076 -0.59039513   0.76021149  1.5751867371
[116,]   0.67224905  0.32731751   0.87350679  1.4439940605
[117,]   0.79301235 -0.13153881   0.98680210  0.7880306775
[118,]   2.24217198  1.70388647   1.66657392  1.3128013839
[119,]   2.24217198 -1.04925145   1.77986923  1.4439940605
[120,]   0.18919584 -1.96696410   0.70356384  0.3944526477
[121,]   1.27606556  0.32731751   1.10009740  1.4439940605
[122,]  -0.29385737 -0.59039513   0.64691619  1.0504160307
[123,]   2.24217198 -0.59039513   1.66657392  1.0504160307
[124,]   0.55148575 -0.81982329   0.64691619  0.7880306775
[125,]   1.03453895  0.55674567   1.10009740  1.1816087073
[126,]   1.63835547  0.32731751   1.27004036  0.7880306775
[127,]   0.43072244 -0.59039513   0.59026853  0.7880306775
[128,]   0.30995914 -0.13153881   0.64691619  0.7880306775
[129,]   0.67224905 -0.59039513   1.04344975  1.1816087073
[130,]   1.63835547 -0.13153881   1.15674505  0.5256453243
[131,]   1.87988207 -0.59039513   1.32668801  0.9192233541
[132,]   2.48369858  1.70388647   1.49663097  1.0504160307
[133,]   0.67224905 -0.59039513   1.04344975  1.3128013839
[134,]   0.55148575 -0.59039513   0.76021149  0.3944526477
[135,]   0.30995914 -1.04925145   1.04344975  0.2632599711
[136,]   2.24217198 -0.13153881   1.32668801  1.4439940605
[137,]   0.55148575  0.78617383   1.04344975  1.5751867371
[138,]   0.67224905  0.09788935   0.98680210  0.7880306775
[139,]   0.18919584 -0.13153881   0.59026853  0.7880306775
[140,]   1.27606556  0.09788935   0.93015445  1.1816087073
[141,]   1.03453895  0.09788935   1.04344975  1.5751867371
[142,]   1.27606556  0.09788935   0.76021149  1.4439940605
[143,]  -0.05233076 -0.81982329   0.76021149  0.9192233541
[144,]   1.15530226  0.32731751   1.21339271  1.4439940605
[145,]   1.03453895  0.55674567   1.10009740  1.7063794137
[146,]   1.03453895 -0.13153881   0.81685914  1.4439940605
[147,]   0.55148575 -1.27867961   0.70356384  0.9192233541
```

```
[148,]   0.79301235 -0.13153881   0.81685914  1.0504160307 [149,]
0.43072244  0.78617383   0.93015445  1.4439940605 [150,]
0.06843254 -0.13153881   0.76021149  0.7880306775
 s<-sample(150,90)
iris_train<-iris[s,]
iris_test<-iris[-s,]
model<-NaiveBayes(Species~.,data=iris_train) pred<-
predict(model,iris_test)
conf<-confusionMatrix(pred$class ,iris_test$Species)$table
conf1<-as.matrix(conf)
accsd<-((sum(diag(conf1)))/sum(conf1))*100
cat("Accuracy of standardized dataset is",accsd,"%")

Accuracy of standardized dataset is 98.33333 %
```

K-NEAREST NEIGHBOUR

a) **TRAINING SET IS 75%**

```
 iris_train<-iris_norm[1:113,]
iris_test<-iris_norm[114:150,]
iris_pred<-knn(iris_train,iris_test,iris[1:113,5],k=13)
con<-table(iris_pred,iris[114:150,5]) accuracy<-
((sum(diag(con)))/sum(con))*100
cat("Accuracy with 75% training set is ",accuracy,"%")

Accuracy with 75% training set is  51.35135 %
```

b) **TRAINING SET IS 66.6%**

```
iris_train<-iris_norm[1:100,] iris_test<-
iris_norm[101:150,]
iris_pred<-knn(iris_train,iris_test,iris[1:100,5],k=13)
con<-table(iris_pred,iris[101:150,5]) accu66<-
((sum(diag(con)))/sum(con))*100
cat("Accuracy with 66.6% training set is ",accu66,"%")

Accuracy with 66.6% training set is  0 %
```

HOLD OUT METHOD

```
iris_train<-iris_norm[1:75,] iris_test<-iris_norm[76:150,]
iris_pred<-knn(iris_train,iris_test,iris[1:75,5],k=13)
con<-table(iris_pred,iris[76:150,5]) acchld<-
```

```
((sum(diag(con)))/sum(con))*100 cat("Accuracy with hold
out is ",acchld,"%")
```

```
Accuracy with hold out is  33.33333 %
```

RANDOM SUBSAMPLING

a) TRAINING SET IS 80%

```
 iris_train<-iris_norm[1:80,]
iris_test<-iris_norm[81:150,]
iris_pred<-knn(iris_train,iris_test,iris[1:80,5],k=13)
con<-table(iris_pred,iris[81:150,5]) accu1<-
((sum(diag(con)))/sum(con))*100
```

b) TRAINING SET IS 90%

```
 iris_train<-iris_norm[1:90,]
iris_test<-iris_norm[91:150,]
iris_pred<-knn(iris_train,iris_test,iris[1:90,5],k=13)
con<-table(iris_pred,iris[91:150,5]) accu2<-
((sum(diag(con)))/sum(con))*100 accrs<-max(accu1,accu2)
cat("Accuracy with random subsampling is",accrs)
```

```
Accuracy with random subsampling is 28.57143
```

CROSS VALIDATION

```
 x=iris[,-5]
y=as.factor(iris$Species)
res<-knn.cv(x,y,1:length(y))
con<-table(res,y)
accucs<-((sum(diag(con)))/sum(con))*100 cat("Accuracy
with cross validation is",accucs)
```

```
Accuracy with cross validation is 96
```

COMPARISON

```
greatest<-max(acchld,accrs,accucs) if(greatest==acchld)
{
  cat("Holdout method does best classification")
} if(greatest==accrs)
{
  cat("Random subsampling method does best classification")
}
if(greatest==accucs)
{
  cat("cross validation method does best classification")
}
cross validation method does best classification
```

DATA SCALING

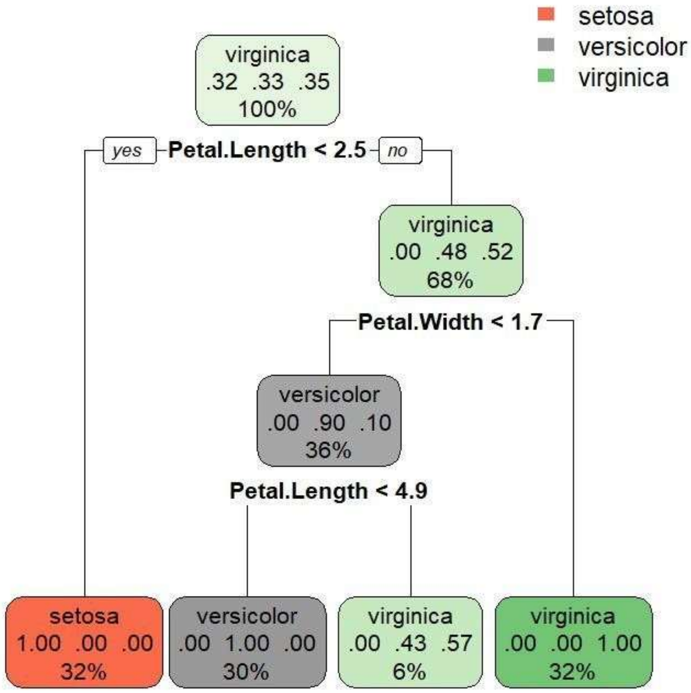
```
data_std <- function(x)
{
  (x-mean(x))/sd(x)
}
sapply(iris[,-5],data_std)
 iris_train<-iris_norm[1:100,]
iris_test<-iris_norm[101:150,]
iris_pred<-knn(iris_train,iris_test,iris[1:100,5],k=13)
con<-table(iris_pred,iris[101:150,5]) accusd<-
((sum(diag(con)))/sum(con))*100 cat("Accuracy of sd
is",accusd)
```

```
Accuracy of sd is 0
```

DECISION TREE

a) **TRAINING SET IS 75%**

```
s<-sample(150,113) iris_train<-iris[s,]
iris_test<-iris[-s,]
dtm<-rpart(Species~.,iris_train,method="class")
rpart.plot(dtm)
```




```
  p<-predict(dtm,iris_test,type="class")
cn<-confusionMatrix(iris_test[,5],p)$table
print(cn)
```

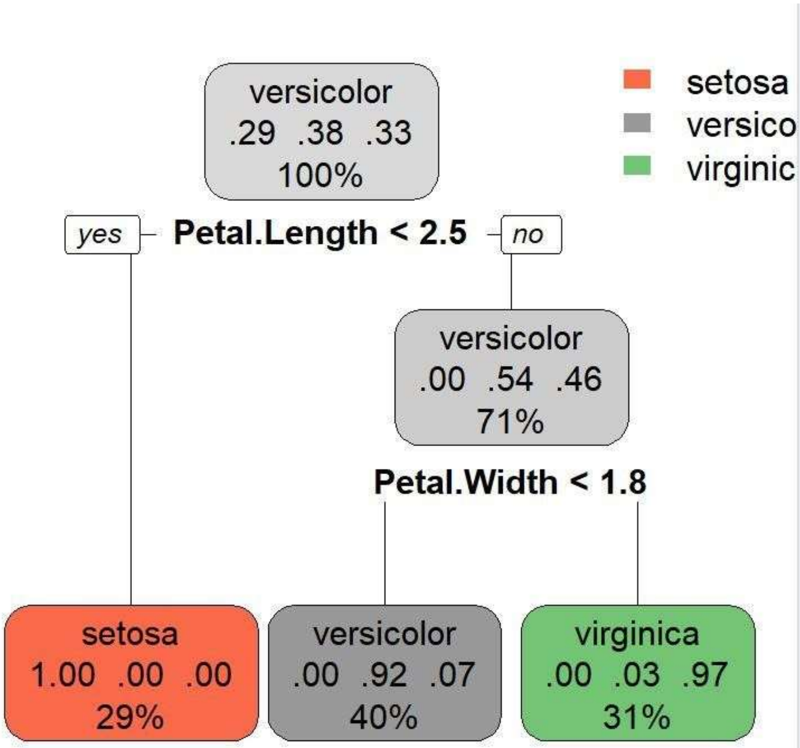
```
Reference
Prediction   setosa versicolor virginica
setosa         14          0         0
versicolor      0         11         2
virginica       0          0        10
```

```
  cn1<-as.matrix(cn) accu<-
(sum(diag(cn1))/sum(cn1))*100
cat("The accuracy with 75% training data is ",accu,"%")
```

```
The accuracy with 75% training data is  94.59459 %
```

b) <u>TRAINING SET IS 66.6%</u>

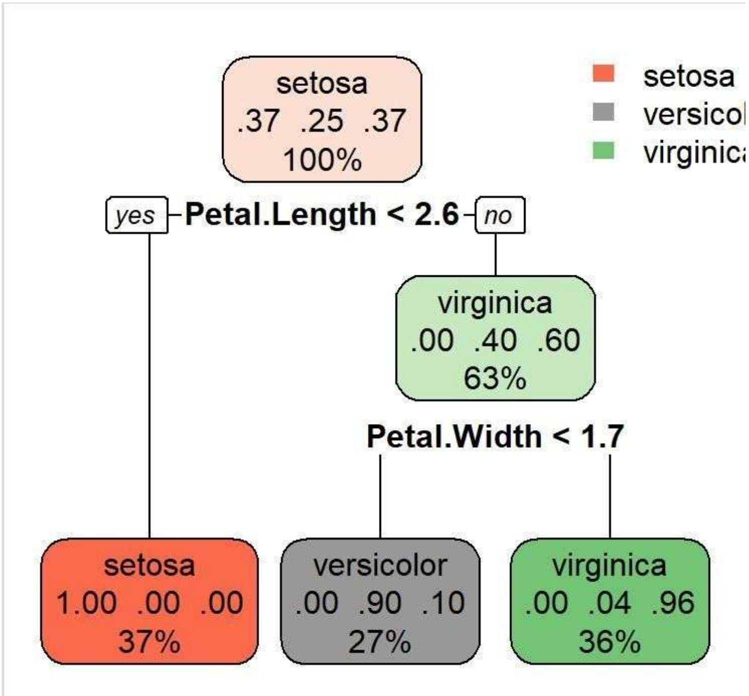
```
s<-sample(150,100) iris_train<-iris[s,]
iris_test<-iris[-s,]
dtm<-rpart(Species~.,iris_train,method="class")
rpart.plot(dtm)
```




```
p<-predict(dtm,iris_test,type="class")
cn<-confusionMatrix(iris_test[,5],p)$table
print(cn)

      Reference
Prediction   setosa versicolor virginica
setosa          21          0         0
versicolor       0         12         0
virginica        0          2        15

cn1<-as.matrix(cn)
accu6<-(sum(diag(cn1))/sum(cn1))*100
cat("The accuracy with 66.6% training data is ",accu6,"%") The
    accuracy with 66.6% training data is  96 %
```

HOLD-OUT METHOD

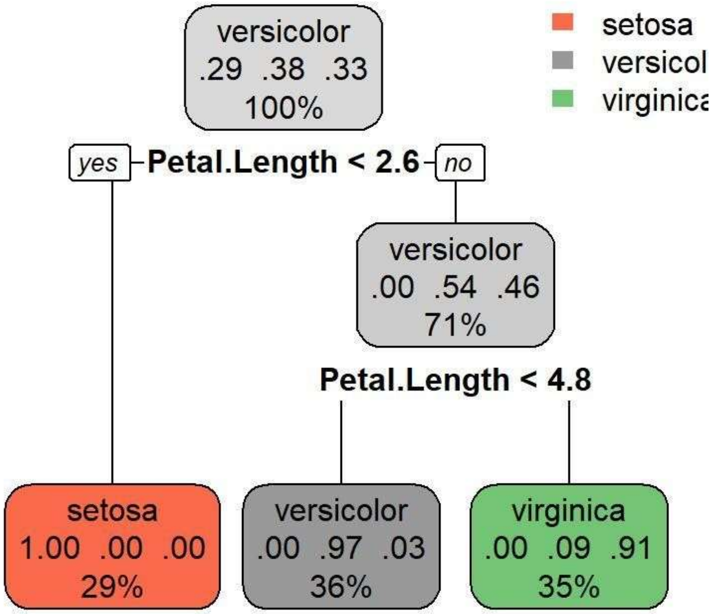
```
s<-sample(150,75) iris_train<-iris[s,]
iris_test<-iris[-s,]
dtm<-rpart(Species~.,iris_train,method="class")
rpart.plot(dtm)
```




```
 p<-predict(dtm,iris_test,type="class")
cn<-confusionMatrix(iris_test[,5],p)$table
cn1<-as.matrix(cn)
acchld<-(sum(diag(cn1))/sum(cn1))*100
cat("The accuracy with hold out method is ",acchld,"%")

The accuracy with hold out method is  96 %
```

RANDOM SUBSAMPLING

```
 i<-75 j<-1
acc<-c() for(i
in 75:100)
{   s<-sample(150,i)
iris_train<-iris[s,]
iris_test<-iris[-s,]
  dtm<-rpart(Species~.,iris_train,method="class")
rpart.plot(dtm)
  p<-predict(dtm,iris_test,type="class")   cn<-
confusionMatrix(iris_test[,5],p)$table   cn1<-as.matrix(cn)
  acc[j]<-c((sum(diag(cn1))/sum(cn1))*100,acc)
j=j+1 }
acrs<-mean(acc)
cat("The accuracy with random subsampling method is ",acrs,"%")
```




```
The accuracy with random subsampling method is  94.11932 %
```

CROSS VALIDATION

```
set.seed(123)
form <- "Species ~ Sepal.Length + Sepal.Width + Petal.Length + Petal.Width"
folds <- split(iris, cut(sample(1:nrow(iris)),10))
errs <- rep(NA, length(folds)) i<-1 for (i in
1:length(folds))
{   test <- ldply(folds[i], data.frame)   train <-
ldply(folds[-i], data.frame)   tmp.model <-
rpart(form , train, method = "class")
  tmp.predict <- predict(tmp.model, newdata = test, type = "class")
conf.mat <- table(test$Species, tmp.predict)   errs[i] <-1-
sum(diag(conf.mat))/sum(conf.mat)
}
accv<-(1-mean(errs))*100
cat("The average accutracy using k-fold cross validation is",accv,"%")
```

```
The average accutracy using k-fold cross validation is 94 %
```

COMPARISON

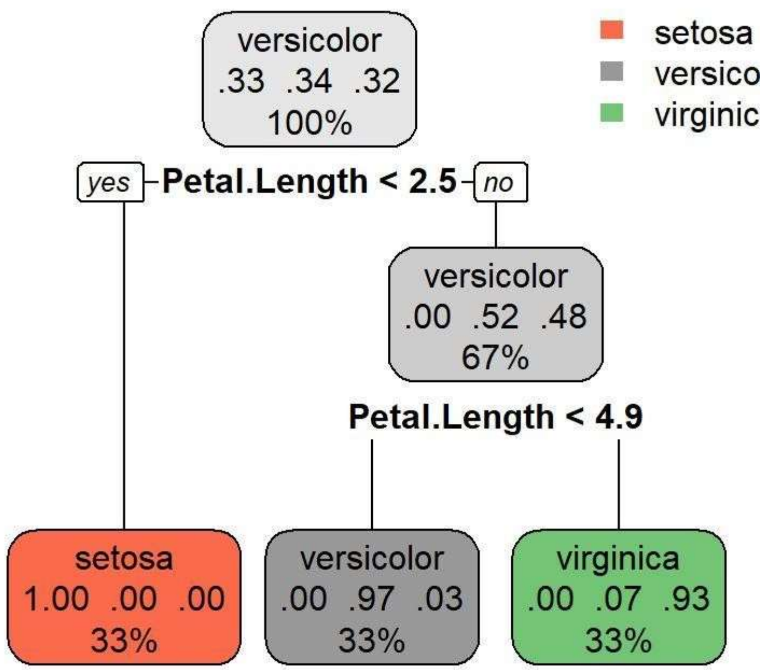
```
greatest<-max(acchld,acrs,accv) if(greatest==acchld)
{
  cat("Holdout method does best classification")
}
if(greatest==acrs)
```

```
{
  cat("Random subsampling method does best classification")
}
if(greatest==accv)
{
  cat("Cross validation method does best classification")
}
```

```
Holdout method does best classification
```

DATA SCALING

```
data_std <- function(x)
{
  (x-mean(x))/sd(x)
}
sapply(iris[,-5],data_std)
  s<-sample(150,90)
iris_train<-iris[s,]
iris_test<-iris[-s,]
dtm<-rpart(Species~.,iris_train,method="class")
rpart.plot(dtm)
```




```
  p<-predict(dtm,iris_test,type="class")
cn<-confusionMatrix(iris_test[,5],p)$table
cn1<- as.matrix(cn)
accsd<-(sum(diag(cn1))/sum(cn1))*100
cat("The accuracy in standardised iris data is ",accsd,"%")
```

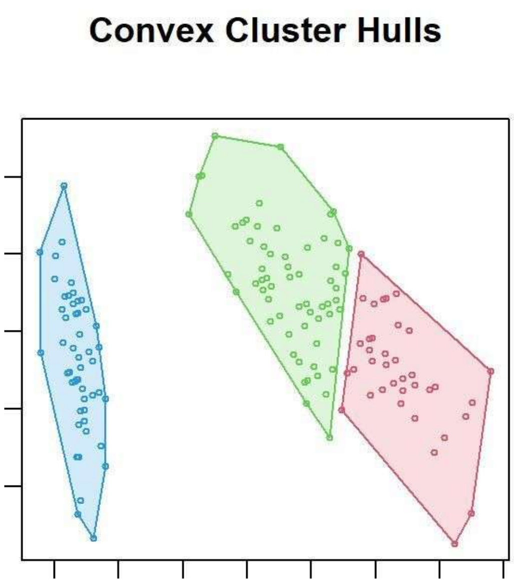
```
The accuracy in standardised iris data is  95 %
```

Question 6

Use Simple Kmeans, DBScan, Hierachical clustering algorithms for clustering. Compare the performance of clusters by changing the parameters involved in the algorithm

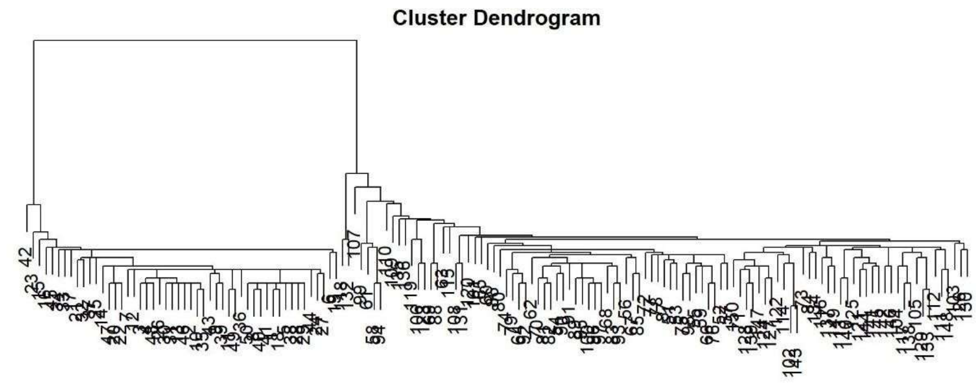
K-means Clustering

```
library(cluster) library(dbscan)
 ir <- iris k<-
kmeans(ir[1:4],3)
clusplot(ir[1:4], k$cluster, labels=2, lines=T)
hullplot(ir[1:4],k$cluster)
```

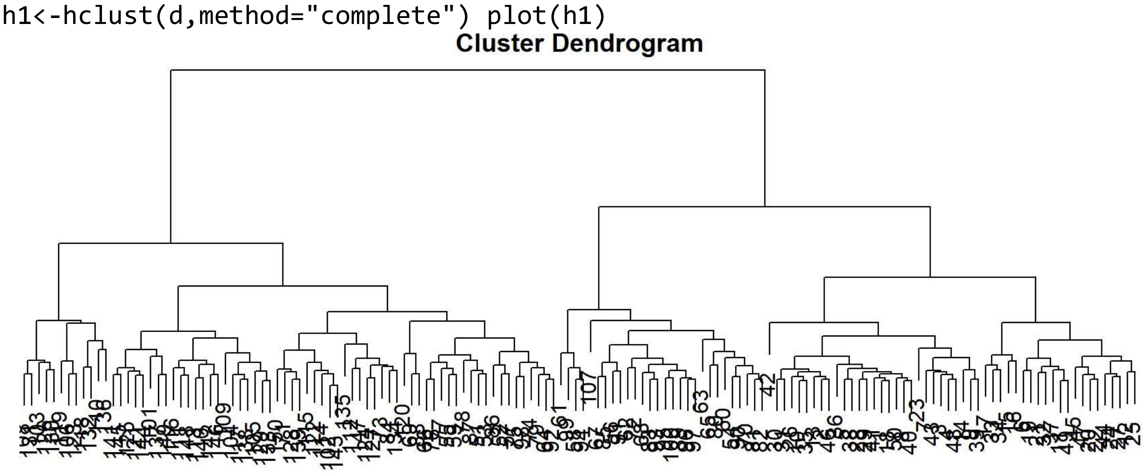


Hierachical Clustering

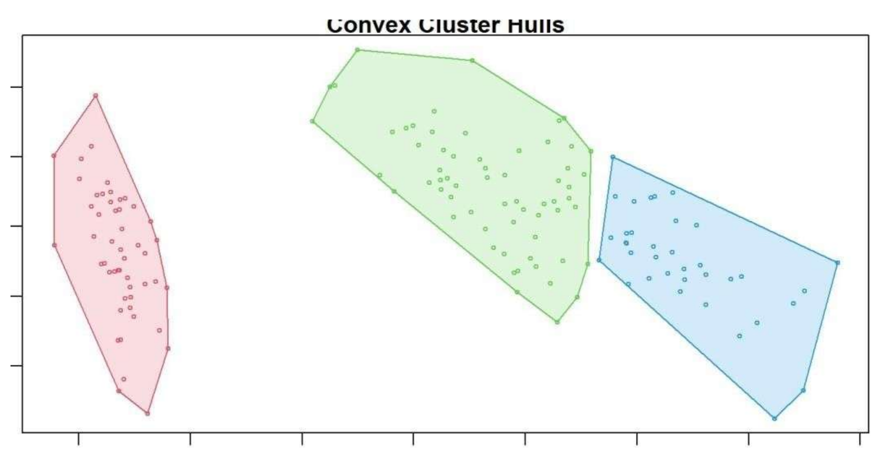
```
  d<- dist(ir[1:4],method =
"euclidean") h <- hclust(d,method =
"single") plot(h)
```




```
h1<-hclust(d,method="complete") plot(h1)
```

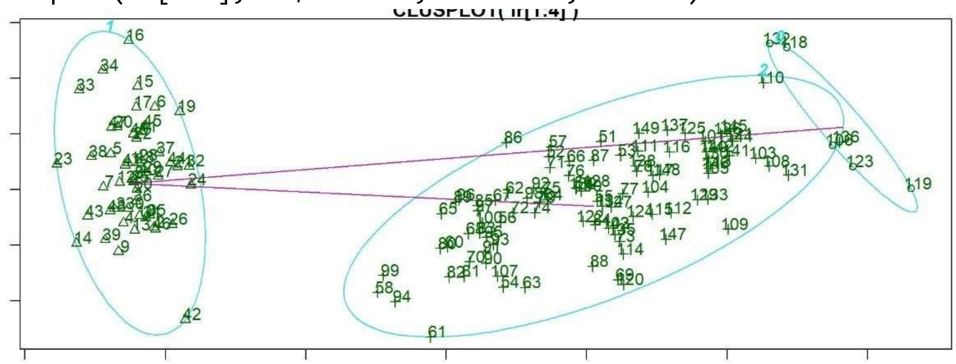



```
h2 <- hclust(d,method="average")
h2=cutree(h2,k=3) hullplot(ir[1:4],h2)
```



DB scan

```
d <- dist(ir[1:4],method ="euclidean") dbc<-
dbscan(ir[1:4],eps=0.8,minPts=10) dbc<-
dbscan(d,eps=0.8,minPts=10)
clusplot(ir[1:4],dbc$cluster, labels=2,lines=T)
```



```
hullplot(ir[1:4],dbc$cluster)
```