

Practical 2:

Aim: Classification using SVM

Requirement:

R tool

Code:

```
getwd()
```

```
read.csv()
```

```
ds=read.csv("E:/Rajdeep/bigdata pract/dataset/social.csv",TRUE,"")
```

```
ds
```

```
ds=ds[3:5]
```

```
ds
```

```
install("catools")
```

```
library(caTools)
```

```
set.seed(123)
```

```
split=sample.split(ds$Purchased, SplitRatio=0.75)
```

```
training_set=(subset(ds, split == TRUE))
```

```
test_set =(subset(ds, split == FALSE))
```

```
ds
```

```
test_set[-3]=scale(test_set[-3])
```

```
training_set[-3]=scale(training_set[-3])
```

```
test_set[-3]
```

```
training_set[-3]
```

```
install.packages('e1071')
```

```
library('e1071')
```

```
classifier=svm(formula=Purchased ~ ., data= training_set , type='C-classification',kernel='linear')
```

```
classifier
```

```
y_pred=predict(classifier, newdata=test_set[-3])
```

```
y_pred
```

```
cm=table(test_set[, 3],y_pred)
```

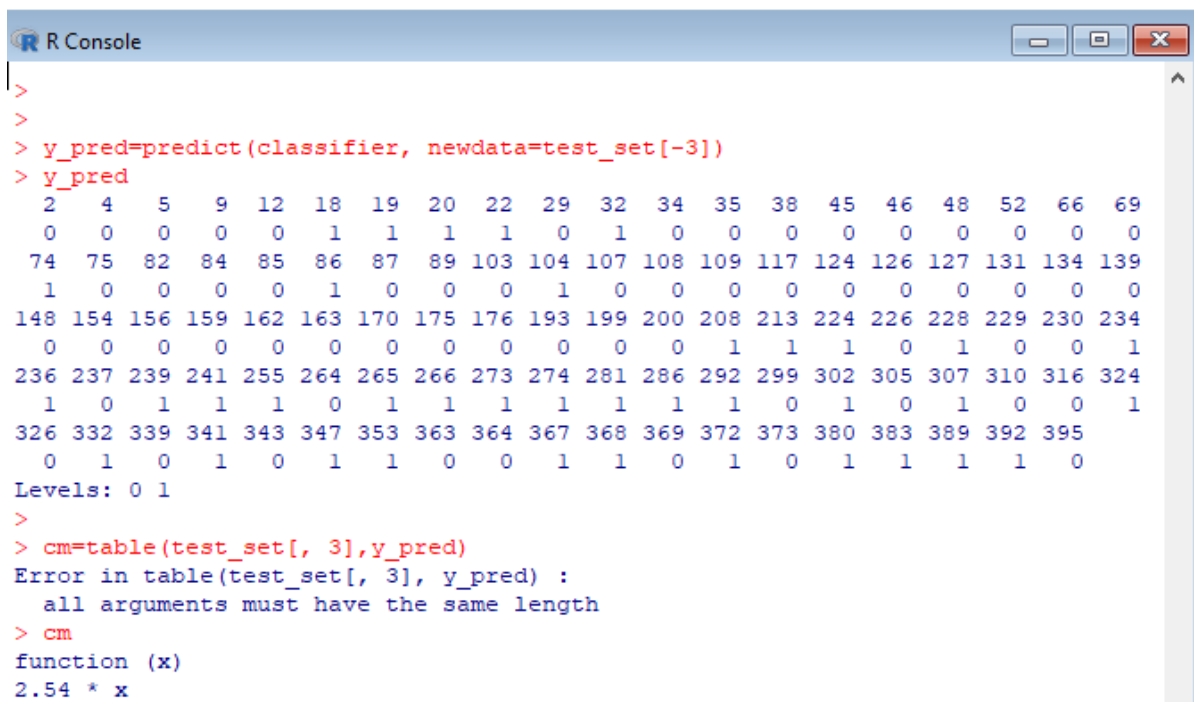
```
cm
```

```

> set = training_set
> x1 = seq(min(set[, 1]) - 1, max(set[, 1]) + 1, by = 0.01)
> x2 = seq(min(set[, 2]) - 1, max(set[, 2]) + 1, by = 0.01)

> grid_set = expand.grid(x1, x2)
> colnames(grid_set) = c('Age', 'EstimatedAge')
> y_grid = predict(classifier, newdata = grid_set)
> plot(set[, -3],
+       main = 'SVM',
+       xlab = 'Age',
+       ylab = 'EstimatedAge')

```

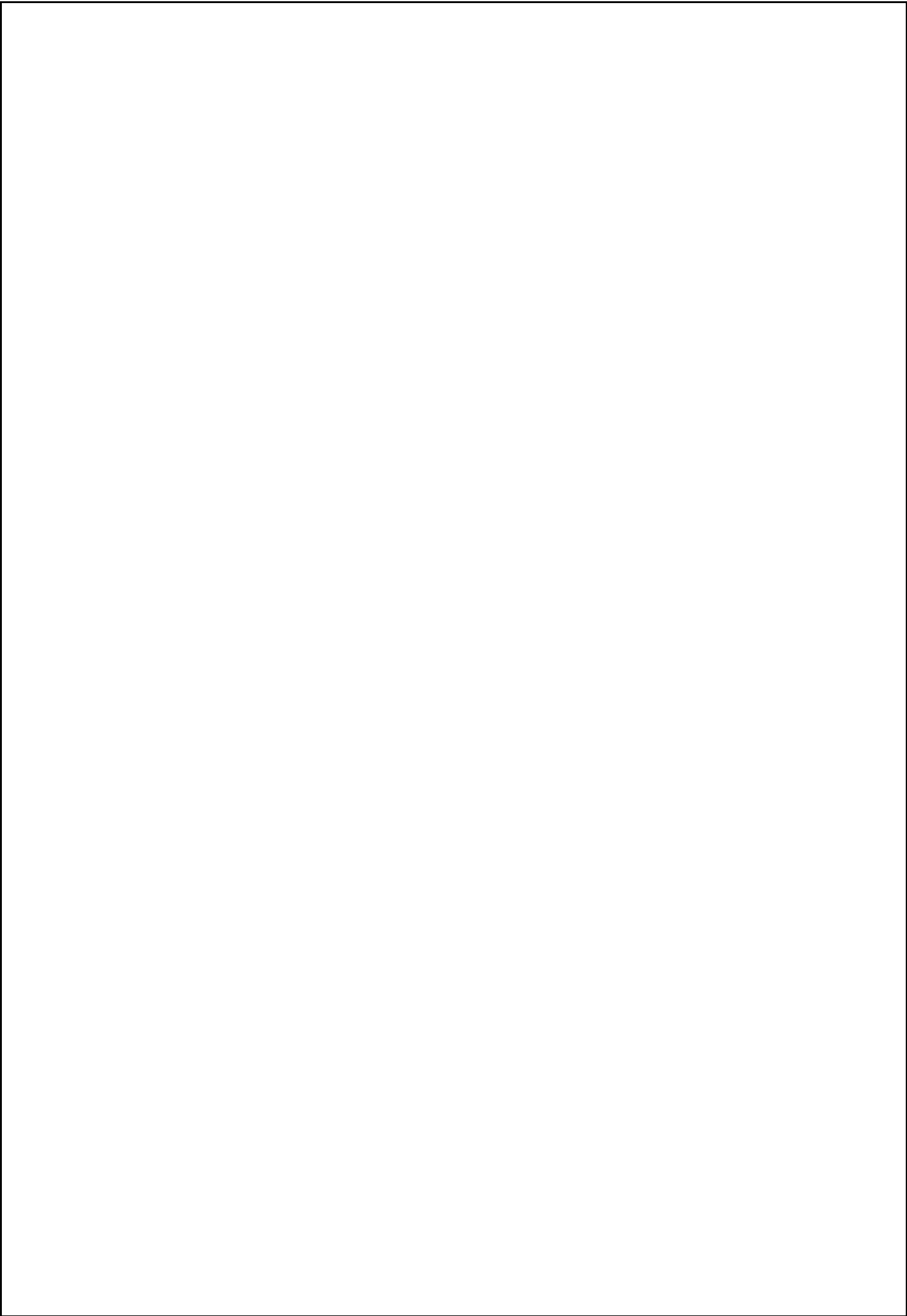


```

>
>
> y_pred=predict(classifier, newdata=test_set[-3])
> y_pred
 2   4   5   9  12  18  19  20  22  29  32  34  35  38  45  46  48  52  66  69
0   0   0   0   0   1   1   1   1   0   1   0   0   0   0   0   0   0   0
74  75  82  84  85  86  87  89 103 104 107 108 109 117 124 126 127 131 134 139
1   0   0   0   0   1   0   0   0   1   0   0   0   0   0   0   0   0   0
148 154 156 159 162 163 170 175 176 193 199 200 208 213 224 226 228 229 230 234
0   0   0   0   0   0   0   0   0   0   0   0   1   1   1   0   1   0   1
236 237 239 241 255 264 265 266 273 274 281 286 292 299 302 305 307 310 316 324
1   0   1   1   1   0   1   1   1   1   1   1   1   0   1   0   1   0   0   1
326 332 339 341 343 347 353 363 364 367 368 369 372 373 380 383 389 392 395
0   1   0   1   0   1   1   0   0   1   1   0   1   0   1   1   1   1   0
Levels: 0 1
>
> cm=table(test_set[, 3],y_pred)
Error in table(test_set[, 3], y_pred) :
  all arguments must have the same length
> cm
function (x)
2.54 * x

```

Output:



Practical 8:

Apriori algorithm

Aim: Perform Apriori algorithm using Groceries dataset from the R arules package.