Instituto Tecnológico de Tijuana
Ingeniería en Informática

Subject Name:

Data Mining

Exercise:

Evaluative Practice 1 - Unit 3

Teacher:

Romero Hernandez Jose Christian

Students:

Castillo Ramirez Guadalupe 17213043

Ramos Rivera Manuel Isai 17212931

```
getwd()
setwd("F:/Data mining U1/Unidad3/PracticaEvaluatoria")
getwd()
```

Importing the dataset

```
dataset = read.csv('Social_Network_Ads.csv')
dataset = dataset[3:5]
```

Encoding the target feature as factor

```
dataset$Purchased = factor(dataset$Purchased, levels = c(0, 1))
```

Splitting the dataset into the Training set and Test set

```
install.packages('caTools')
library(caTools)
set.seed(123)
split = sample.split(dataset$Purchased, SplitRatio = 0.75)
training_set = subset(dataset, split == TRUE)
test_set = subset(dataset, split == FALSE)
```

Feature Scaling

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

Fitting Random Forest Classification to the Training set

```
install.packages('e1071')
library(e1071)
classifier = naiveBayes(x = training_set[-3],
                        y = training_set$Purchased)
```

Predicting the Test set results

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

Making the Confusion Matrix

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

```
#  We make the graph to visualize the information with the help of the ElemStatLearn library
```

Visualising the Training set results

```
install.packages('ElemStatLearn')
library(ElemStatLearn)
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', 'EstimatedSalary')
y_grid = predict(classifier, newdata = grid_set)
plot(set[, -3],
     main = 'Naive Bayes (Training set)',
     xlab = 'Age', ylab = 'Estimated Salary',
     xlim = range(X1), ylim = range(X2))
contour(X1, X2, matrix(as.numeric(y_grid), length(X1), length(X2)), add = TRUE)
points(grid_set, pch = '.', col = ifelse(y_grid == 1, 'springgreen3', 'tomato'))
points(set, pch = 21, bg = ifelse(set[, 3] == 1, 'green4', 'red3'))
```

Visualising the Test set results

```
library(ElemStatLearn)
set = test_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', 'EstimatedSalary')
y_grid = predict(classifier, newdata = grid_set)
plot(set[, -3], main = 'Naive Bayes (Test set)',
     xlab = 'Age', ylab = 'Estimated Salary',
     xlim = range(X1), ylim = range(X2))
contour(X1, X2, matrix(as.numeric(y_grid), length(X1), length(X2)), add = TRUE)
points(grid_set, pch = '.', col = ifelse(y_grid == 1, 'springgreen3', 'tomato'))
points(set, pch = 21, bg = ifelse(set[, 3] == 1, 'green4', 'red3'))
```