



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO®

TECNOLÓGICO NACIONAL DE MÉXICO

INSTITUTO TECNOLÓGICO DE TIJUANA

SUBDIRECCIÓN ACADÉMICA

DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN

NOMBRE DE LOS ALUMNOS:

RAYMUNDO HIRALES LAZARENO (N. CONTROL: 17212339)

GALAVIZ LONA OSCAR EDUARDO (N.CONTROL: 17212993)

Carrera: Ingeniería Informática

Semestre: 9no

MATERIA: Minería de datos

PROFESOR: JOSE CHRISTIAN ROMERO HERNANDEZ

TRABAJOS: Practica evaluatoria 3

FECHA: 7/12/21

Introduccion

La practica consiste en utilizar el modelo de naive bayes para la realizacion de esta practica, el modelo de naive bayes es proveniente del calculo probabilistico, esto quiere decir que se trata de cuanto es la probabilidad de que me toque dicho objeto con ciertas características dentro de un conjunto de objetos, esto tambien puede ser aplicado a los datos ya que cada registro de dato tiene características unicas mediante este modelo podemos determinar cuanto porcentaje de probabilidad hay de que salga dicho dato. Tambien puede haber una comparativa dentro de los propios datos de saber cual es mas probable que salga o que ocurra ya que esto aplica para todos los datos, ya que se esta implementa este modelo dentro de un repositorio. En general nos mostrara el porcentaje que tendra cada dato de salir si el analisis correspondiente es muy especifico.

Desarrollo

```
install.packages ("caret")
library(e1071)
library (caret)
install.packages ("caTools")
install.packages ("ElemStatLearn")

# Importing the dataset
data <- read.csv('Social_Network_Ads.csv')

dataset=data
dataset = data[3:5]

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

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)

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

install.packages ("e1071")

library(e1071)
classifier = naiveBayes(formula = Purchased ~ .,
                        data=training_set,
                        type='c-classification',
                        kernel = 'linear')

naiveBayes
#prediccion de resultados
y_pred=predict(classifier, newdata = test_set[-3])
```

```

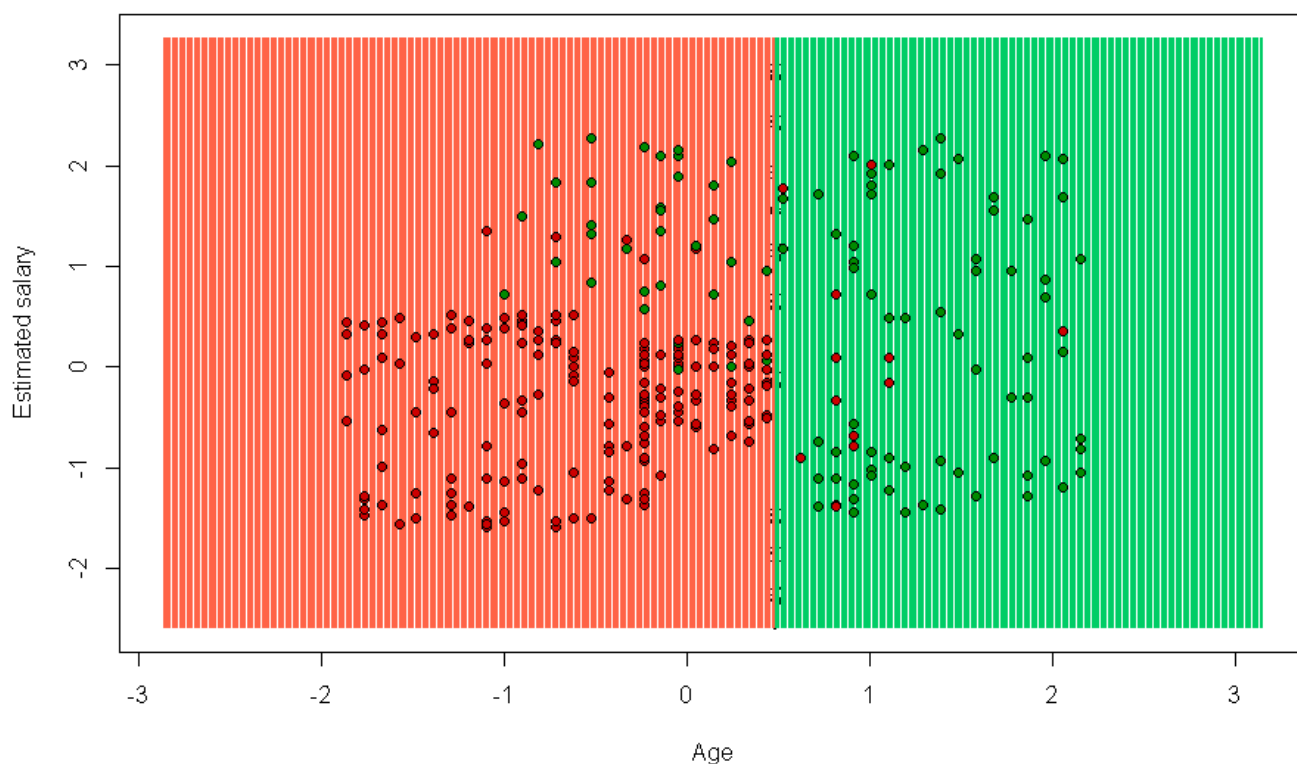
y_pred

#creando la matriz de confusion
cm = table(test_set[, 3], y_pred)
cm
```R
#visualizing the training set results
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', 'Estimated salary')
y_grid=predict(classifier,newdata=grid_set)
plot(set[, -3],
 main='Native 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'))

```

Native Bayes (Training set)



```

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'))
```

Evidence1

## Análisis de resultados

---

Podemos ver que el resultado es similar en ambos y el resultado es escalable puesto que tiene el doble de elementos, podemos decir que no varía en tamaño de elementos y se pueden usar en grandes cantidades de información, pero en pequeñas cantidades no sería tan útil este método.

## Conclusion

---

Con esta práctica nos damos cuenta de que el análisis de datos son funciones o implementan las bases de análisis y estadística ya que usan modelos probabilísticos que determinan las probabilidades de los datos, así como tener en cuenta que es importante tener en cuenta las bases de probabilidad y estadística para poder entender los datos que nos arrojan los modelos y porque de esos datos, En conclusión esta práctica en sí nos demuestra que los modelos están basados en operaciones de probabilidad y estadística, ya que estas se implementan en los datos para poder tener un resultado sobre los datos y que nos de un panorama que podamos entender y poder determinar que decisión se tiene que tomar