SVM

Diego Pavez Olave

26/7/2020

```r
#SVM
# Importar el dataset
dataset = read.csv('Social_Network_Ads.csv')
dataset = dataset[, 3:5]

# Dividir los datos en conjunto de entrenamiento y conjunto de test
set.seed(123)
split = sample.split(dataset$Purchased, SplitRatio = 0.75)
training_set = subset(dataset, split == TRUE)
testing_set = subset(dataset, split == FALSE)

# Escalado de valores
training_set[,1:2] = scale(training_set[,1:2])
testing_set[,1:2] = scale(testing_set[,1:2])

# Ajustar el SVM con el conjunto de entrenamiento.
classifier = svm(formula = Purchased ~ .,
                 data = training_set,
                 type = "C-classification",
                 kernel = "linear")

# Predicción de los resultados con el conjunto de testing
y_pred = predict(classifier, newdata = testing_set[,-3])

# Crear la matriz de confusión
cm = table(testing_set[, 3], y_pred)

# Visualización del conjunto de entrenamiento
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 = 'SVM (Conjunto de Entrenamiento)',
     xlab = 'Edad', ylab = 'Sueldo Estimado',
     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'))
```
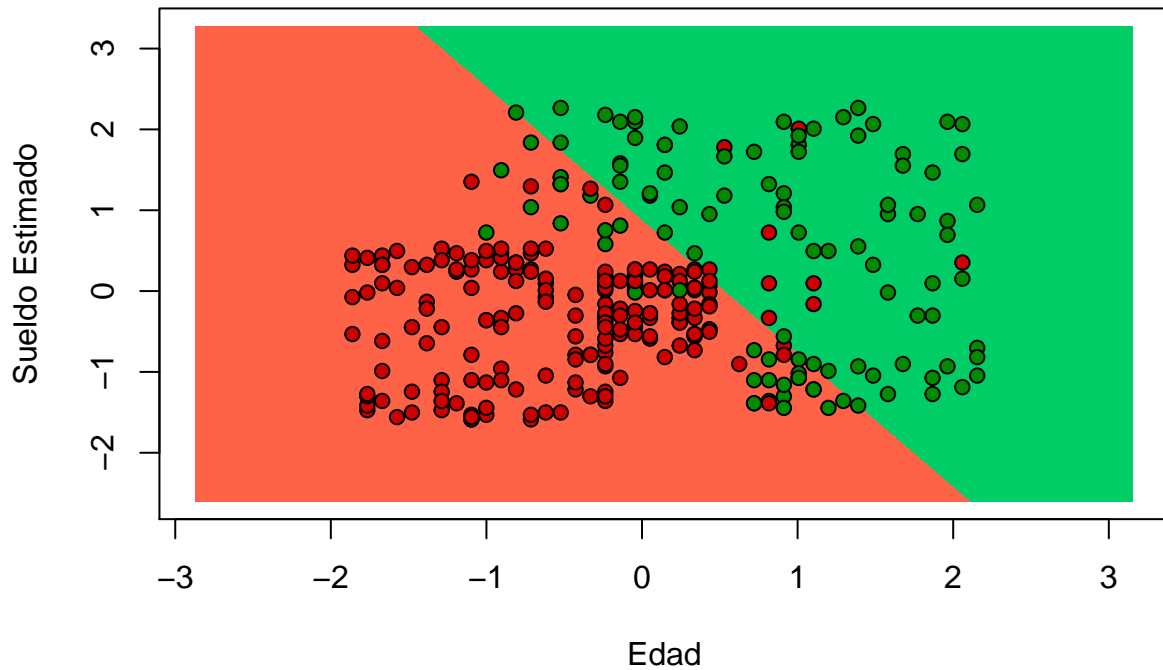
SVM (Conjunto de Entrenamiento)


```r
# Visualización del conjunto de testing
set = testing_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 = 'SVM (Conjunto de Testing)',
     xlab = 'Edad', ylab = 'Sueldo Estimado',
     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'))
```

SVM (Conjunto de Testing)