Nombre: Idalia Alejandra Delgado Moreno
Grupo: 31

Introducción

¿Qué es una regresión lógistica

Los arboles de decisión son representaciones gráficas de posibles soluciones a una decisión basadas en ciertas condiciones, es un algoritmo de aprendizaje supervisado y es de los más usados en machine learning. Sus decisiones son del tipo IF THIS, THEN THAT. Su estructura principal son los nodos y tiene uno principal llamado raíz (root) y luego se descomponen en más ramas planteando una condición de puede ser cierta o falsa. Cada nodo se bifurca en dos y así hasta llegar a las hojas, que son los nodos finales y que equivalen a la solución.
Ejemplo:

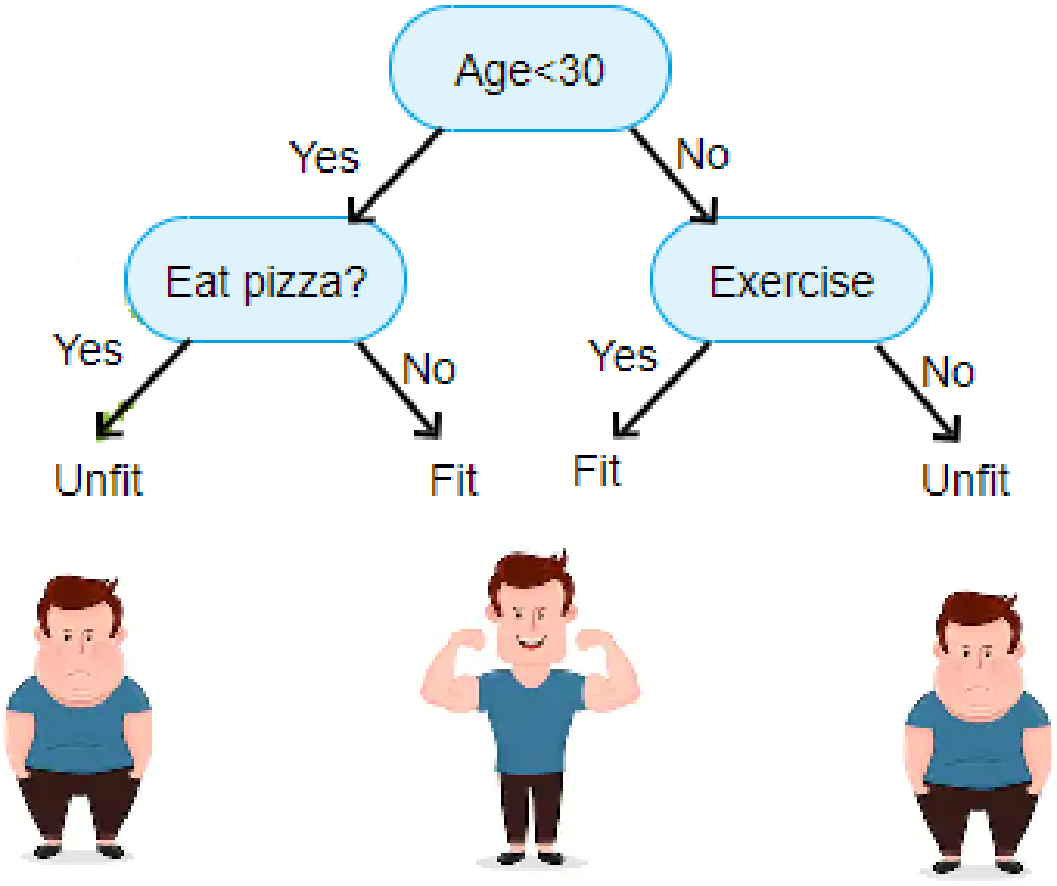

Figure 1: Árbol de Decisión generado.

Metodolog'ia

Para realizar esta actividad, hay que seguir las instrucciones de PDF que nos solicitó el profesor de la página 57 a la 79.

***Importante** Hay que tener Graphviz instalado y crear una variable de entorno. Además de cualquier otra librería con la que no se cuente.

1. Importamos las librerías necesarias para el análisis:

```
#Imports necesarios

import numpy as np
import pandas as pd
from sklearn.tree import DecisionTreeClassifier, export_graphviz
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
plt.rcParams['figure.figsize'] = (16, 9)
plt.style.use('ggplot')

from sklearn import tree
from sklearn.metrics import accuracy_score
from sklearn.model_selection import KFold, cross_val_score
from IPython.display import Image as PImage
from subprocess import check_call
from PIL import Image, ImageDraw, ImageFont
from IPython.display import display
import pydot
from sklearn.tree import export_graphviz
```

2. Cargamos el dataset y visualizamos los primeros registros:

```
artists_billboard = pd.read_csv("artists_billboard_fix3.csv")
print(artists_billboard.head())
```

3. Generamos estadísticas básicas del dataset:

```
print(artists_billboard.shape)
print(artists_billboard.groupby('top').size())
```

4. Visualizamos la distribución de datos con gráficos de conteo:

```
sns.catplot(x='top', data=artists_billboard, kind="count")
sns.catplot(x='artist_type', data=artists_billboard, kind="count")
sns.catplot(x='top', data=artists_billboard, hue='artist_type', kind="count")
```

5. Preparamos los datos para el modelo:

```
X = artists_billboard.drop(['top'], axis=1)
y = artists_billboard['top']

print(X.shape)
```

6. Entrenamos un modelo de Árbol de Decisión:

```
model = DecisionTreeClassifier()
model.fit(X, y)
```

7. Hacemos predicciones y evaluamos el modelo:

```
predictions = model.predict(X)

print(predictions[:5])
print("Accuracy:", accuracy_score(y, predictions))
```

8. Validación cruzada del modelo:

```
kfold = KFold(n_splits=10, random_state=7, shuffle=True)
cv_results = cross_val_score(model, X, y, cv=kfold, scoring='accuracy')

print("Decision Tree: %f (%f)" % (cv_results.mean(), cv_results.std()))
```

9. Visualizamos el árbol de decisión:

```
export_graphviz(model, out_file="tree.dot", feature_names=X.columns,
class_names=['No Top', 'Top'], filled=True)
check_call(['dot', '-Tpng', 'tree.dot', '-o', 'tree.png'])
display(PImage("tree.png"))
```

10. Hacemos una predicción con nuevos datos:

```
X_new = pd.DataFrame({'feature1': [valor], 'feature2': [valor],
..., 'featureN': [valor]})

model.predict(X_new)
```

Resultados:

A continuación, el árlbol de decisión que se generó con el código

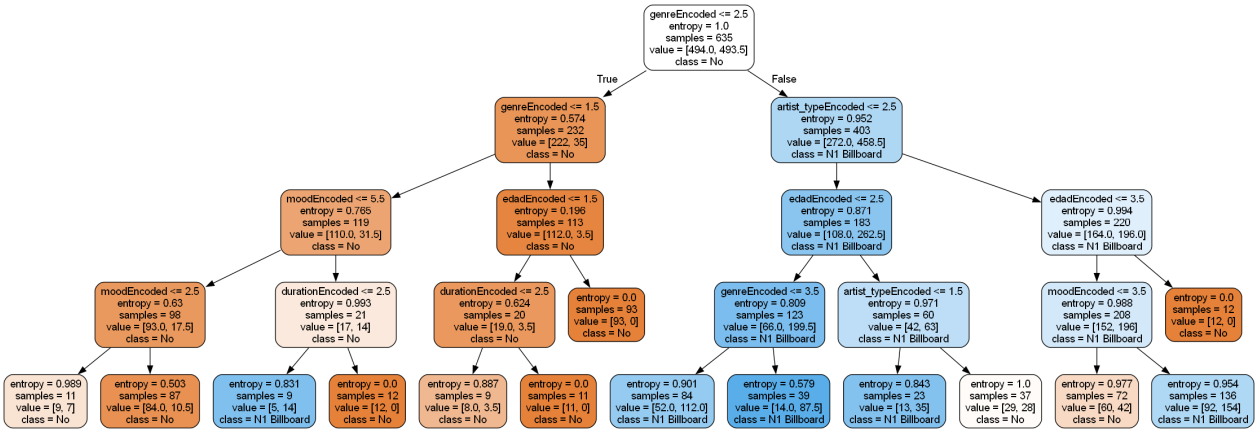

Figure 2: Árbol de Decisión generado.

Ahora vamos a poner a prueba nuestro arbol. Predicción de Canciones al Billboard 100. Probabilidad de que Havana llegué al top 1:

```
x_test = pd.DataFrame(columns=('top','moodEncoded', 'tempoEncoded',
'genreEncoded','artist_typeEncoded','edadEncoded','durationEncoded'))
x_test.loc[0] = (1,5,2,4,1,0,3)
y_pred = decision_tree.predict(x_test.drop(['top'], axis = 1))
print("Prediccion: " + str(y_pred))
y_proba = decision_tree.predict_proba(x_test.drop(['top'], axis = 1))
print("Probabilidad de Acierto: " + str(round(y_proba[0][y_pred][0]* 100, 2))+"%")
Output:
86.21%
Por lo tanto, la canción Havana de Camila Cabello, tiene probabilidades
muy altas de alcazar el top 1 con su canción.
```

Conclusión:

En esta avtividad hemos aprendido como hacer paso a paso un árbol de decisiones con un conjunto de datos. Desde la revisión y preprocesamiento de los datos hasta la validación del modelo, hemos visto cómo influye el tamaño y balance del conjunto de datos en la precisión obtenida. Además, creo que este análisis muestra la importancia de contar con conjuntos de datos más representativos y explorar ajustes en la configuración del modelo para mejorar su desempeño. Sin duda, este proceso nos ha brindado una visión práctica de cómo funcionan los algoritmos de aprendizaje automático en la clasificación de datos del mundo real.