

PRÁCTICA N° ACT0302

Entrenamiento del modelo sin tratar

- **Módulo: PIA**
- **Unidad de trabajo: Entrenamiento del modelo sin tratar**
- **Nombre y apellidos: Alvaro Lucio-Villegas de Cea**



Índice

Enunciado:	3
Resolución:	3

Enunciado:

Ahora que hemos visto cómo entrenar el modelo una vez limpiado los datos, vamos a comparar qué hubiera ocurrido si hubiéramos realizado los entrenamientos sin procesado de datos.

En esta práctica se ha de realizar el entrenamiento del modelo sin haber realizado una limpieza de datos y se compararán los resultados con los obtenidos en la versión que se ha entrenado con los datos limpios.

Resolución:

En este caso lo único que vamos a realizar es la conversión de la columna churn a números para poder introducir los datos en el modelo y que funcione correctamente el entrenamiento.

```
Preparación de datos para modelo. Para ello separaremos los datos en 20/80

from sklearn.model_selection import train_test_split

Y=df["Churn"]
Y=Y.apply(lambda x: 1 if x=="Yes" else 0)

X=df.drop(columns=["Churn"])
X_train, X_test, y_train, y_test = train_test_split(pd.get_dummies(X), Y, test_size=0.2, random_state=0)
```

[14] ✓ 1.5s

Entrenamiento en este caso al ser valores en su mayoría son binarios el mejor modelo sería una regresión Logística.

```
from sklearn.linear_model import LogisticRegression

# Distintos tipos de solver {'lbfgs', 'liblinear', 'newton-cg', 'newton-cholesky', 'sag', 'saga'}

model = LogisticRegression(solver='lbfgs', max_iter=1000, random_state=0)
model.fit(X_train, y_train)

model.score(X_test, y_test)
```

[15] ✓ 1m 9.7s

... 0.7963094393186657

La puntuación del modelo es de 0.7963094393186657 sin limpieza de datos.

Como podemos observar en el resultado, es inferior al score de la anterior actividad. Ya que la limpieza de datos es muy importante para la calidad del entrenamiento del modelo.

Ya que al haber muchos más datos y menos tratados, el modelo lo tiene muy complicado para sacar un resultado al existir tanto ruido en el análisis.