

Ejercicio 1: Predicción del precio de compra de jugadores de fútbol

Contexto: Eres un analista de datos en un club de fútbol y deseas predecir el precio de compra de los jugadores en función de su edad y goles anotados en la última temporada.

Código para Google Colab

```
# Importar las librerías necesarias

import numpy as np

import matplotlib.pyplot as plt

from sklearn.linear_model import LinearRegression

from sklearn.model_selection import train_test_split


# Generar datos aleatorios para jugadores de fútbol

np.random.seed(42) # Para reproducibilidad

n_jugadores = 100


# Edad (18 a 35 años)

edades = np.random.randint(18, 36, n_jugadores)


# Goles anotados en la última temporada (0 a 30)

goles = np.random.randint(0, 31, n_jugadores)


# Precio de compra (en millones de euros), generado a partir de una función lineal
con ruido

precios = 2 * edades + 3 * goles + np.random.normal(0, 10, n_jugadores) # Precio
realista


# Crear matriz de características

X = np.column_stack((edades, goles))

y = precios
```

```
# Dividir los datos en entrenamiento y prueba

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)


# Crear y entrenar el modelo de regresión lineal

model = LinearRegression()

model.fit(X_train, y_train)


# Realizar predicciones

y_pred = model.predict(X_test)


# Mostrar resultados

print("Predicciones de precios:", y_pred)

print("Valores reales de precios:", y_test)


# Visualizar los resultados (gráfico 2D de edades vs. precios)

plt.scatter(X[:, 0], y, color='blue', label='Datos reales')

plt.plot(X[:, 0], model.predict(X), color='red', label='Predicciones')

plt.xlabel('Edad (años)')

plt.ylabel('Precio de compra (millones de euros)')

plt.title('Predicción del precio de compra de jugadores de fútbol')

plt.legend()

plt.show()
```

Ejercicio 2: Predicción del precio de autos según su marca y motor

Contexto: Eres un analista de datos en una concesionaria de autos y deseas predecir el precio de compra de un auto en función de su cilindrada del motor (en litros) y la marca (convertida a una variable numérica).

Código para Google Colab

```
# Importar las librerías necesarias

import numpy as np

import matplotlib.pyplot as plt

from sklearn.linear_model import LinearRegression

from sklearn.model_selection import train_test_split


# Generar datos aleatorios para autos

np.random.seed(42) # Para reproducibilidad

n_autos = 100


# Marca del auto (0 = Toyota, 1 = Ford, 2 = BMW)

marcas = np.random.randint(0, 3, n_autos)


# Cilindrada del motor (1.0 a 5.0 litros)

cilindrada = np.round(np.random.uniform(1.0, 5.0, n_autos), 1)


# Precio de compra (en miles de euros), generado a partir de una función lineal
con ruido

precios_autos = 20 * marcas + 15 * cilindrada + np.random.normal(0, 5, n_autos)

# Precio realista


# Crear matriz de características

X_autos = np.column_stack((marcas, cilindrada))
```

```
y_autos = precios_autos
```

```
# Dividir los datos en entrenamiento y prueba
```

```
X_train_autos, X_test_autos, y_train_autos, y_test_autos = train_test_split(X_autos,  
y_autos, test_size=0.2, random_state=42)
```

```
# Crear y entrenar el modelo de regresión lineal
```

```
model_autos = LinearRegression()
```

```
model_autos.fit(X_train_autos, y_train_autos)
```

```
# Realizar predicciones
```

```
y_pred_autos = model_autos.predict(X_test_autos)
```

```
# Mostrar resultados
```

```
print("Predicciones de precios de autos:", y_pred_autos)
```

```
print("Valores reales de precios de autos:", y_test_autos)
```

```
# Visualizar los resultados (gráfico 2D de cilindrada vs. precios)
```

```
plt.scatter(X_autos[:, 1], y_autos, color='blue', label='Datos reales')
```

```
plt.plot(X_autos[:, 1], model_autos.predict(X_autos), color='red',  
label='Predicciones')
```

```
plt.xlabel('Cilindrada del motor (litros)')
```

```
plt.ylabel('Precio de compra (miles de euros)')
```

```
plt.title('Predicción del precio de autos')
```

```
plt.legend()
```

```
plt.show()
```