

11_1_ Tarea 1. Ejercicio predicción de ventas.

ENUNCIADO



Enunciado del ejercicio:

Predicción de Éxito de un Producto en el Mercado usando un Árbol de Decisiones

Una empresa quiere lanzar un nuevo producto al mercado. Antes de hacerlo, desea predecir si este producto será un éxito o un fracaso basándose en tres características:

1. Costo de producción
2. Precio de venta al público
3. Gasto en publicidad

Tienes acceso a un conjunto de datos con 1000 registros que contienen estos parámetros y el resultado de ventas de productos anteriores (Éxito/Fracaso).

```
import numpy as np
import pandas as pd
```

```
# Semilla para reproducibilidad
np.random.seed(42)
```

```
# Crear datos aleatorios para las características
```

```
costo_produccion = np.random.uniform(50, 200, 1000) # costos entre 50 y 200
```

```
precio_venta = costo_produccion + np.random.uniform(20, 80, 1000) # precio de venta mayor al costo de producción
```

```
gasto_publicidad = np.random.uniform(500, 5000, 1000) # gasto en publicidad entre 500 y 5000
```

```
# Suponemos que un producto es un éxito si el precio de venta es menos de 1.5 veces el costo y el gasto en publicidad es mayor a 2500
```

```
# Esto es solo para fines de ejemplo y no tiene una base real.
```





11_1_ Tarea 1. Ejercicio predicción de ventas.

ENUNCIADO

```
etiquetas = np.where((precio_venta < 1.5 * costo_produccion) & (gasto_publicidad > 2500), 'Éxito', 'Fracaso')
```

```
# Crear DataFrame con los datos
df = pd.DataFrame({
    'Costo_de_Produccion': costo_produccion,
    'Precio_de_Venta': precio_venta,
    'Gasto_en_Publicidad': gasto_publicidad,
    'Resultado': etiquetas
})

# Guardar el DataFrame en un archivo CSV
df.to_csv('datos_producto.csv', index=False)
```

El objetivo es configurar un árbol de decisiones para lograr la mejor precisión en las predicciones. Deberás ajustar los siguientes parámetros del modelo:

- criterion
- max_depth
- min_samples_split
- min_samples_leaf
- min_impurity_decrease
- random_state

