



Actividad el algoritmo de árbol de decisión (Tiempo estimado: 90 [minutos]) Actividad individual.

Objetivo de aprendizaje:

Programar un algoritmo de predicción en múltiples variables en Python con el uso de las librerías de Scikit_Learn para crear un modelo de árbol de decisión.

Habilidades:

Razonamiento lógico y sistémico.

Instrucciones:

1. Revisa las **secciones 1 a 5** del siguiente link:
<https://www.kaggle.com/learn/intro-to-machine-learning>
2. Responde correctamente las siguientes preguntas guía:
 - a) Capturing patterns from data is called FITTING
 - b) The data used to fit the model is called TRAINING DATA
 - c) After the model has been fit, you can apply it to new data to PREDICT
 - d) You predict the price of any house by tracing through FEATURES
 - e) The point at the bottom where we make a prediction is called LEAF
 - f) Explica lo que representa cada fila de la siguiente tabla (revisa la sección 2)

COUNT: filas sin valores no-faltates

MEAN: la media

STD: desviación estándar

MIN: mínimo valor de cada columna

?: los valores se ordenan de forma ascendente

MAX: máximo valor de columna

	Rooms	Bathroom	Landsize	Latitude	Longitude
count	60.000000	60.000000	60.000000	60.000000	60.000000
mean	2.716667	1.566667	251.133333	-37.777957	144.939105
std	0.783120	0.620734	244.073028	0.048900	0.054444
min	1.000000	1.000000	0.000000	-37.848100	144.867900
25%	2.000000	1.000000	123.000000	-37.808125	144.878975
50%	3.000000	1.500000	165.500000	-37.801550	144.952150
75%	3.000000	2.000000	266.750000	-37.723775	144.995400
max	6.000000	3.000000	1063.000000	-37.716400	145.000400

- g) The column we want to predict, which is called PREDICTION TARGET
- h) The columns that are inputted into our model (and later used to make predictions) are called FEATURES
- i) The steps to building and using a model are: DEFINE, FIT, PREDICT, EVALUATE
- j) Observa cómo es la predicción de los precios tomando en cuenta cinco registros. NOTA: En realidad no son las primeras cinco en orden de aparición puesto que se excluyeron los registros que tienen celdas vacías en el archivo melb_data.csv ¿cuál sería el MAE para los datos predichos? 1035000. 1465000. 1600000. 1876000. 1636000



- k) ¿Cómo se define el Error Medio Absoluto (MAE *Mean Absolute Error*)
WITH THE MAE METRIC, WE TAKE THE ABSOLUTE VALUE OF EACH ERROR. THIS
CONVERTS EACH ERROR TO A POSITIVE NUMBER. WE THEN TAKE THE AVERAGE OF
THOSE ABSOLUTE ERRORS.
- l) ¿Cuál es el MAE de tu entrenamiento con cinco registros? 666.6666666666666
- m) Después de dividir el conjunto de datos en las variables de entrenamiento y validación
(train_X, val_X, train_y, val_y), así como después de entrenar nuevamente, cuál
es el MAE que obtienes? 330966.6666666667

3. En Google Colaboratory codifica el script que se adjunta a este documento.
4. En un repositorio en GitHub agrega el script y en los comentarios las preguntas y respuestas.



Recomendaciones al facilitador:	Recursos y materiales necesarios:
<ol style="list-style-type: none">1. Descarga el archivo melb_data.csv NOTA: el archivo original que puedes descargar en el minicurso, tiene más de 5000 registros; Google Colaboratory no permite cargar archivos con demasiados registros, es por ello que se redujo a los primeros 100.2. El documento en el enlace https://www.kaggle.com/learn/intro-to-machine-learning es un minicurso, el cual pretende hacer una breve introducción al aprendizaje de máquinas. Es importante que leas con detenimiento las secciones 1 a 5 para poder entender el script.	<p>Cuenta de correo en Gmail Registro en Kaggle.com</p>



```
[12] import pandas as pd
      from sklearn.tree import DecisionTreeRegressor
      from sklearn.metrics import mean_absolute_error
      from sklearn.model_selection import train_test_split
```

```
[13] from google.colab import files
      uploaded = files.upload()
```

📁 Seleccionar archivos | melb_data.csv

- **melb_data.csv**(text/csv) - 13839 bytes, last modified: n/a - 100% done
Saving melb_data.csv to melb_data.csv

```
[14] melbourne_data = pd.read_csv('melb_data.csv')
      melbourne_data.columns
```

📁 Index(['Suburb', 'Address', 'Rooms', 'Type', 'Price', 'Method', 'SellerG',
'Date', 'Distance', 'Postcode', 'Bedroom2', 'Bathroom', 'Car',
'Landsize', 'BuildingArea', 'YearBuilt', 'CouncilArea', 'Latitude',
'Longitude', 'Regionname', 'Propertycount'],
dtype='object')

```
▶ melbourne_data = melbourne_data.dropna(axis=0)
  y=melbourne_data.Price
  melbourne_features = ['Rooms', 'Bathroom', 'Landsize', 'Latitude', 'Longitude']
  X = melbourne_data[melbourne_features]
  X.describe()
```

```
[ ] X.head()
```

```
[ ]
  melbourne_model = DecisionTreeRegressor(random_state=1)
  melbourne_model.fit(X, y)
  print("Making predictions for the following 5 houses:")
  print(X.head())
  print("The predictions are")
  print(melbourne_model.predict(X.head()))
```

```
[ ] predicted_home_prices = melbourne_model.predict(X)
  mean_absolute_error(y, predicted_home_prices)
```

```
▶ train_X, val_X, train_y, val_y = train_test_split(X, y, random_state = 0)
  melbourne_model = DecisionTreeRegressor()
  melbourne_model.fit(train_X, train_y)
  val_predictions = melbourne_model.predict(val_X)
  print(mean_absolute_error(val_y, val_predictions))
```





UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO

Instalación Configuración y Comunicación de Sistemas Operativos

RÚBRICA DE EVALUACIÓN DE LA INFOGRAFÍA				
PUNTO A EVALUAR	MUY BIEN 10	BIEN 8	REGULAR 5	PUNTAJE OBTENIDO
Información vertida en el documento	La información que se vierte en el documento es veraz y está en el contexto correcto.	La información que se vierte en el documento es veraz pero no está en el contexto correcto.	La información que se vierte en el documento no es veraz y no está en el contexto correcto.	
			TOTAL	