APRENDIZAJE ESTADÍSTICO
Boletín 3: Árboles de Decisión

ANDRÉS CAMPOS CUIÑA

FECHA DE ENTREGA: 24/11/2021

ÍNDICE

1 EJERCICIO 1

Construye el árbol de clasificación (sin podar) mediante CART y utilizando como criterio la entropía. La condición de parada debe ser que los nodos hoja sean puros (todos los ejemplos son de la misma clase). En cada nodo del árbol se debe indicar:

- **La variable y su valor umbral.**
- **La entropía correspondiente.**
- **En los nodos hoja, la clase del nodo y los ejemplos que pertenecen al mismo.**

El árbol resultante es el siguiente:

[X1 < -0.5] -> Entropía 0.3182570841474064

[0] -> Miembros: [[-3, -1, -1, 0], [-2, 3, 1, 0], [-3, 5, 5, 0]]

[X2 < 0.5] -> Entropía 0.0

[0] -> Miembros: [[3, -2, 0, 0]]

[1] -> Miembros: [[4, 3, -1, 1], [1, 4, 0, 1]]

2 EJERCICIO 2

a. Una de las clases que implementa el algoritmo KNN en *scikit-learn* es sklearn.tree.DecisionTreeClassifier. Revisa los parámetros y métodos que tiene.

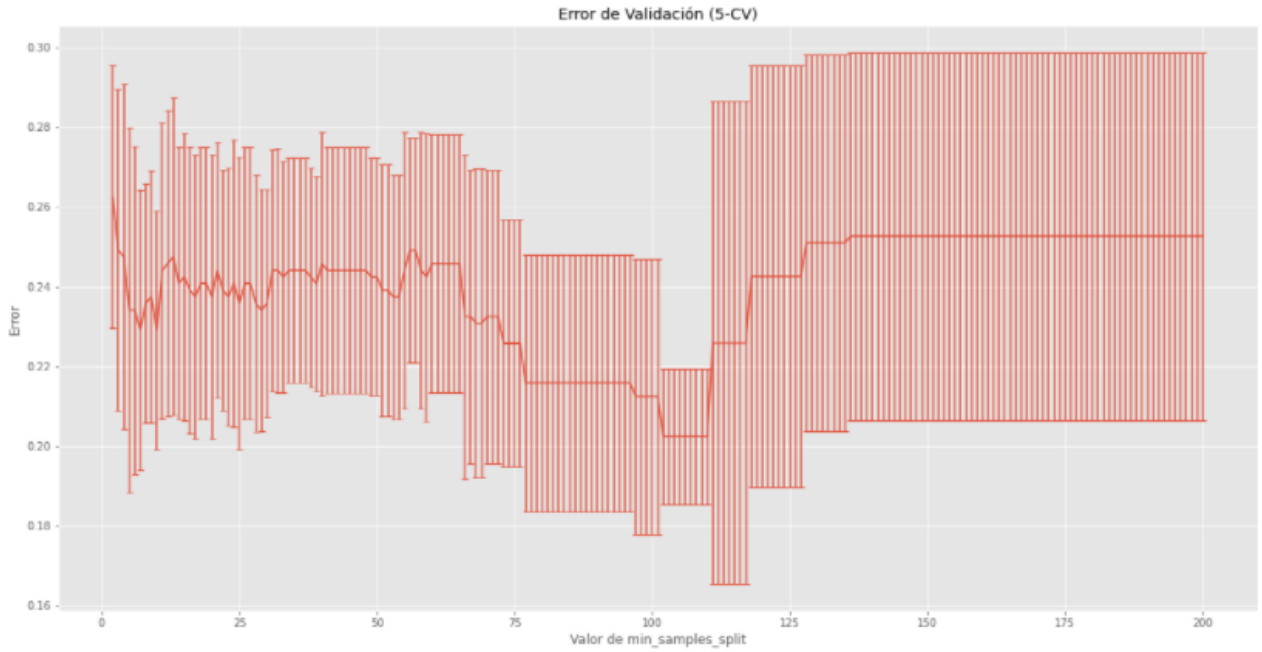
Revisado.

b. Divide los datos en entrenamiento (80%) y test (20%).

Hecho.

c. Realiza la experimentación con *DecisionTreeClassifier* usando los valores por defecto de los parámetros, excepto para criterion que debe tomar el valor 'entropy'. Además, utiliza como hiper-parámetro la variable min_samples_split (permitirá modificar el tamaño del árbol)

Muestra la gráfica del error de entrenamiento con validación cruzada (5-CV) frente al valor del hiper-parámetro. ¿Cuál es el menor error de validación cruzada, su desviación estándar y el valor del hiper-parámetro para el que se consigue? ¿Cuál es el valor del hiperparámetro si se aplicase la regla de una desviación estándar?



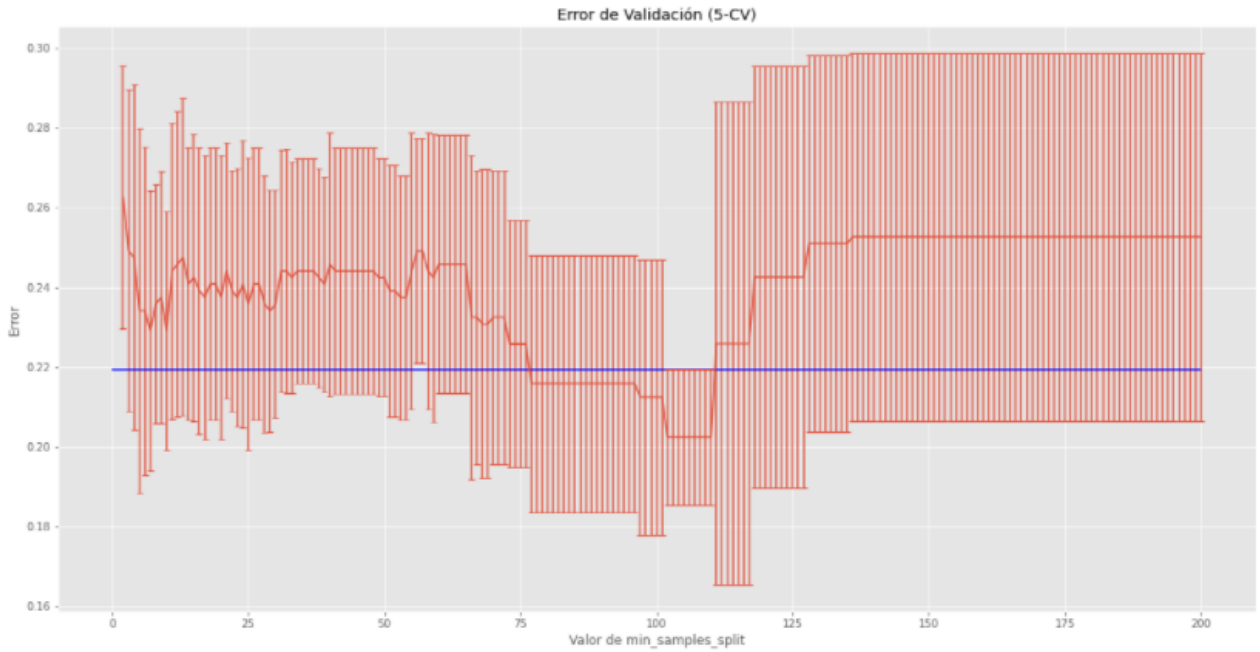
Menor error de validación cruzada, su desviación estándar y el valor del hiper-parámetro para el que se consigue:

	param_min_samples_split	mean_test_score	std_test_score	rank_test_score
108	110	0.202409	0.017044	1

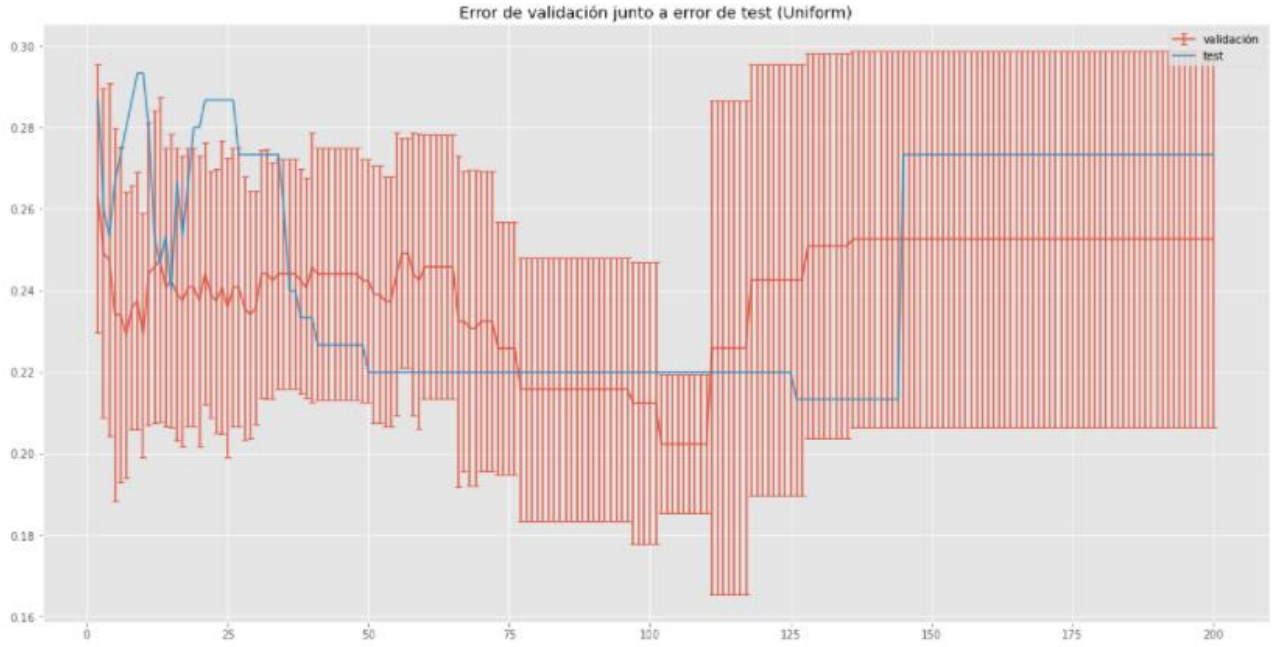
El valor del hiper-parámetro si se aplicase la regla de una desviación estándar:

	param_min_samples_split	mean_test_score	std_test_score	rank_test_score
108	110	0.202409	0.017044	1

La gráfica de la selección de este valor es la siguiente:



Muestra la gráfica del error de test frente al valor del hiper-parámetro, y valora si la gráfica del error de entrenamiento con validación cruzada ha hecho una buena estimación del error de test. ¿Cuál es el menor error de test y el valor del hiper-parámetro para el que se consigue? ¿Cuál es el error de test para el valor del hiper-parámetro seleccionado por la validación cruzada? ¿Cuál es el error de test para el valor del hiper-parámetro seleccionado por la validación cruzada mediante la regla de una desviación estándar?



El menor error de test y el valor del hiper-parámetro para el que se consigue:

	param_min_samples_split	mean_test_score	std_test_score	rank_test_score
142	144	0.213333	0.0	1

El error de test para el valor del hiper-parámetro seleccionado por la validación cruzada:

0.21999999999999997

El error de test para el valor del hiper-parámetro seleccionado por la validación cruzada mediante la regla de una desviación estándar:

0.21999999999999997

3 EJERCICIO 3

Dado el problema de regresión Energy Efficiency:

a. Una de las clases que implementa el algoritmo KNN en *scikit-learn* es *sklearn.tree.DecisionTreeRegressor*. Revisa los parámetros y métodos que tiene.

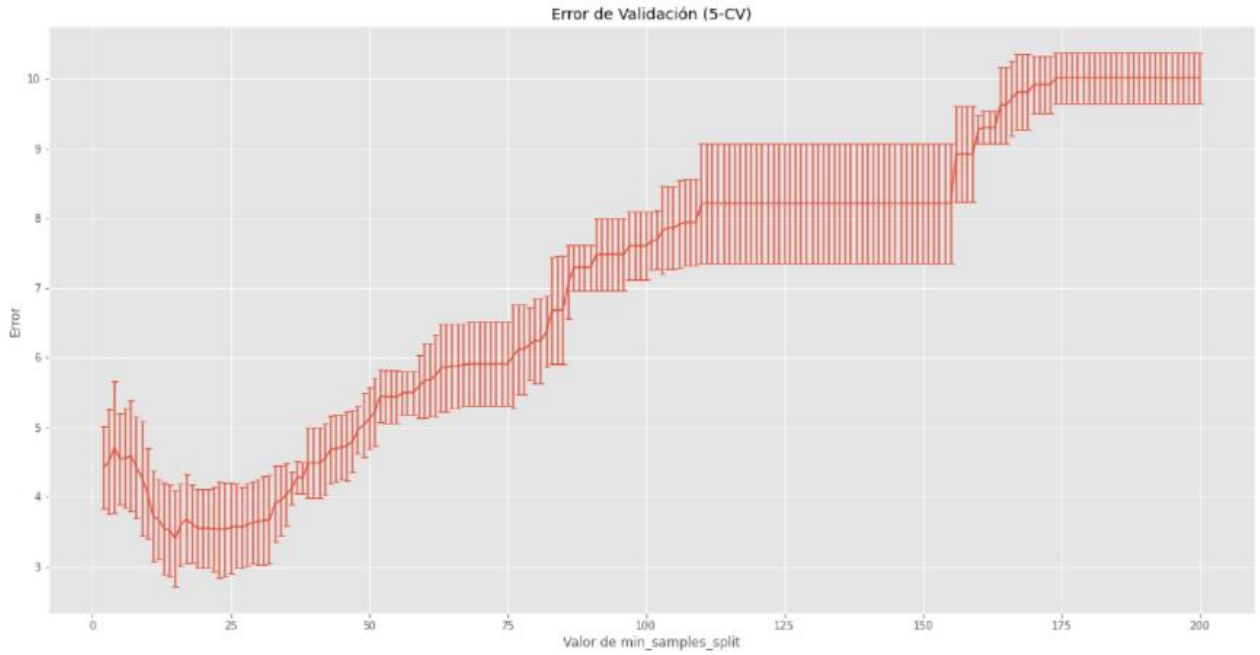
Revisado.

b. Divide los datos en entrenamiento (80%) y test (20%).

Hecho.

c. Realiza la experimentación con *DecisionTreeRegressor* usando como hiper-parámetro el valor de min_samples_split.

Muestra la gráfica del error de entrenamiento con validación cruzada (5-CV) frente al valor del hiper-parámetro. ¿Cuál es el menor error de validación cruzada, su desviación estándar y el valor del hiper-parámetro para el que se consigue? ¿Cuál es el valor del hiperparámetro si se aplicase la regla de una desviación estándar?



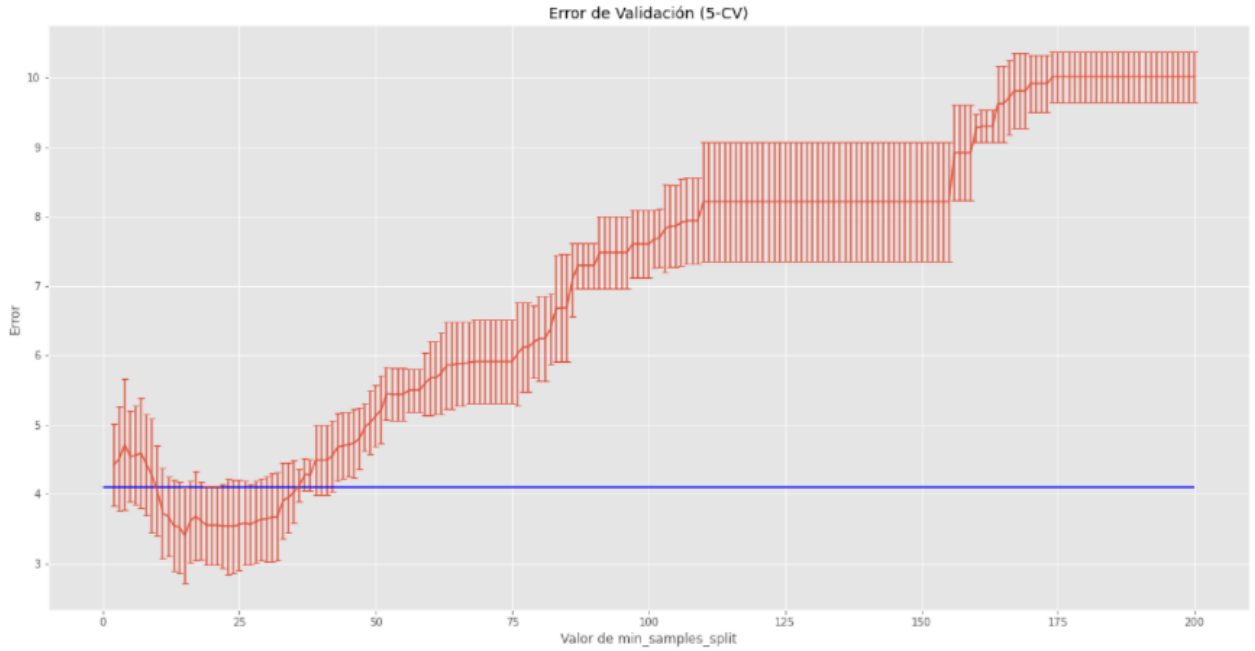
Menor error de validación cruzada, su desviación estándar y el valor del hiper-parámetro para el que se consigue:

	param_min_samples_split	mean_test_score	std_test_score	rank_test_score
13	15	3.400035	0.68836	1

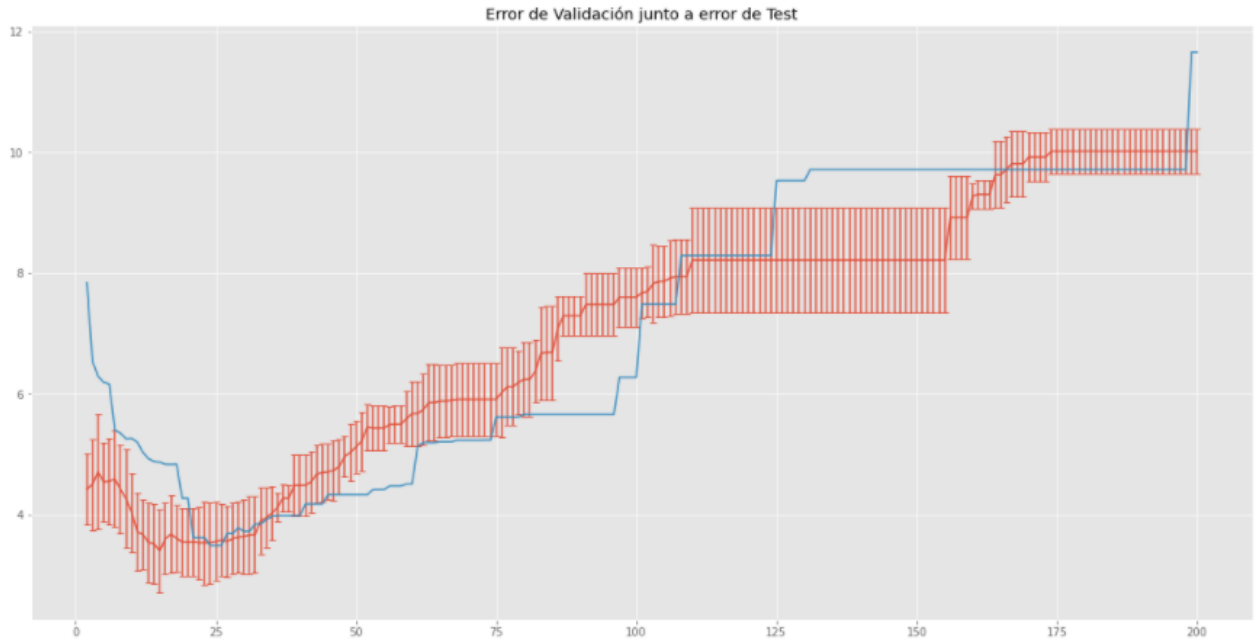
El valor del hiper-parámetro si se aplicase la regla de una desviación estándar:

	param_min_samples_split	mean_test_score	std_test_score	rank_test_score
33	35	4.024361	0.449845	25

La gráfica de la selección de este valor es la siguiente:



Muestra la gráfica del error de test frente al valor del hiper-parámetro, y valora si la gráfica del error de entrenamiento con validación cruzada ha hecho una buena estimación del error de test. ¿Cuál es el menor error de test y el valor del hiper-parámetro para el que se consigue? ¿Cuál es el error de test para el valor del hiper-parámetro seleccionado por la validación cruzada? ¿Cuál es el error de test para el valor del hiper-parámetro seleccionado por la validación cruzada mediante la regla de una desviación estándar?



El menor error de test y el valor del hiper-parámetro para el que se consigue:

	param_min_samples_split	mean_test_score	std_test_score	rank_test_score
24	26	3.492052	0.0	1

El error de test para el valor del hiper-parámetro seleccionado por la validación cruzada:

4.870069961576666

El error de test para el valor del hiper-parámetro seleccionado por la validación cruzada mediante la regla de una desviación estándar:

3.9709026558994247

4 EJERCICIO 4

¿Crees que sería de interés aplicar un método de selección de variables (Forward stepwise selection, etc.) junto con el algoritmo CART?

En mi opinión, aplicar algún método de selección de variables como el *forward stepwise selection* podría ser de interés ya que nos permitiría eliminar aquellas variables que no sean muy significativas, dejándonos con un modelo más sencillo (más general) y por lo tanto también más fácil sería interpretar el árbol de selección resultante.

Además, también nos podría permitir ahorrarnos tiempo en el entrenamiento del árbol de decisiones al no tener que comprobar los cortes de aquellas variables que no sean consideradas como significativas tras ejecutar la selección de variables.

Por otra parte, la selección de variables también requiere de un tiempo de cómputo a mayores. Es por esto que cuando estemos ante un número muy alto de variables sí que sea efectivo ejecutar un método para disminuir la dimensión de nuestro problema, para ahorrar algo de tiempo de entrenamiento.