Al igual que los problemas de clasificación vistos en artículos anteriores, los árboles de decisión son una herramienta muy intuitiva para atacar los problemas supervisados de regresión. En este tipo de problemas, el objetivo será predecir el valor de una variable continua a partir de un modelo generado con predictores mayormente numéricos.

Los árboles de decisión están dentro del grupo de Algoritmos No Lineales, los cuales hacen menos suposiciones acerca de la estructura de los datos y/o la naturaleza del problema. También cuentan con la ventaja de una mayor interpretabilidad del modelo generado. Un árbol de decisión se representa frecuentemente como un árbol binario donde cada nodo representa una condición a evaluar con respecto a un predictor en particular, y cada hoja alberga un posible valor de la variable de salida. Esta representación permite una rápida evaluación de un nuevo ejemplo para poder obtener la predicción con respecto al mismo.

Por tanto, la complejidad de los árboles de decisión radica en la configuración de los parámetros del modelo, los cuales permiten generar distintas versiones del árbol en base a los valores escogidos.

* Criterio
* Profundidad máxima
* Confianza
* Ganancia mínima
* Mínimo tamaño de hoja
* Otros

## Contexto

Para este ejercicio se utilizará un Dataset provisto por la Univeridad de California Irvine (UCI) de precios de inmuebles bajo el nombre de distribución “Housing”. Este es un Dataset clásico utilizado en problemas de regresión, donde se intenta predecir la mediana de casas ocupadas por los dueños.

# Datos

El mismo cuenta con 12 predictores numéricos continuos y 1 binomial. Los atributos del dataset son los siguientes:

* CRIM – Proporción de crimen per cápita
* ZN – Proporción de terreno residencial ubicado en solares mayors a 25.000 metros cuadrados.
* INDUS – Proporción de acres de negocios no-retail por pueblo
* CHAS – Contacto con el río Charles { Sí, No}
* NOX – Concentración de óxido nítrico
* RM – Número promedio de habitacione
* AGE – Proporcion de unidades ocupadas por dueños que fueron construidas antes de 1940
* DIS – Distancias ponderadas a 5 Centros de empleo de Boston
* RAD – Índice de accesibilidad a autopistas periféricas
* TAX – Proporción de impuestos del valor total por cada 10000 USD
* PTRATIO – Proporción de Profesor y Estudiantes en pueblo
* B - Proporción de gente negra
* LSTAT – Porcentaje de estatus más bajo de la población

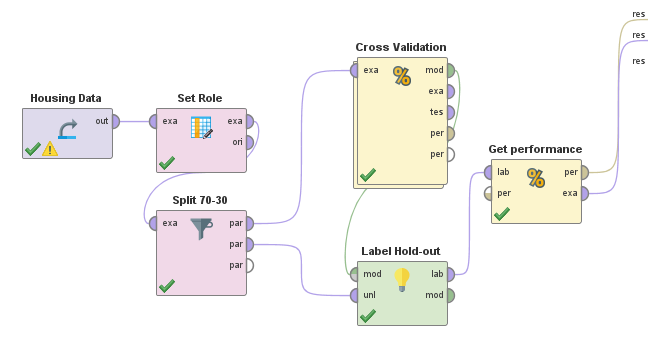
El Dataset comprende 506 valores únicos y se encuentra totalmente completo, es decir ningún predictor contiene valores faltantes.

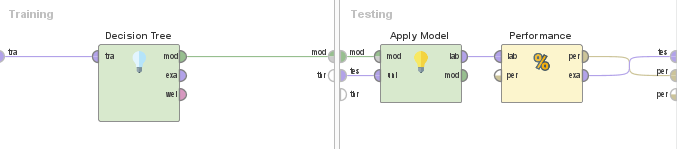
## Modelo

Tras analizar la completitud y buen estado de los datos, se debe implementar el modelo y analizar sus resultados. Para ello, haré un flujo nuevo en Rapidminer que utilice el operador de Árbol de decisión. A partir del Dataset “Housing”, generaré dos conjuntos según la regla 70/30. El primero será destinado a entrenamiento y el segundo a testing del modelo.

Además, utilizaré en esta ocasión un K-Fold Cross Validation para obtener un resultado más realista con respecto a este Dataset que es de un tamaño reducido. Este operador nos permitirá dividir el Dataset en K particiones o “Folds”, entrenar K modelos distintos, reservando cada una de esas particiones para test en cada iteración.

El flujo en Rapidminer se debería ver de la siguiente manera:





Al ejecutar este modelo, Rapidminer obtenemos como resultados como la visualización del árbol generado que podrá darnos más información sobre el ajuste de los parámetros. También el vector de Performance indicando los valores de Raíz cuadrada del error medio (RMSE), el error absoluto y el error medio de predicción.

Se analizaron distintos valores de hyperparámetros para el árbol de decisión de regresión para obtener distintos resultados.

A continuación, se registran los mejores valores encontrados por método de tanteo y observación.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Profundidad | Mejora mínima | Tamaño de hoja mínimo | Alternativas de prepruning | RMSE | Error absoluto | Error relativo |
| 10 | 0.01 | 5 | 10 | 4.795 | 3.009 +/- 3.734 | 14.11% +/- 14.90% |