

UNIDAD TEMÁTICA 3: Algoritmos Lineales

Trabajo de Aplicación 2

Caso de Estudio: Predicción del precio de venta de una casa, a partir de varios predictores

El dataset completo está disponible para descargar del repositorio de UCI:

<https://archive.ics.uci.edu/ml/machine-learning-databases/housing/>

Los objetivos de este **Trabajo de Aplicación** son:

1. Identificar cuáles atributos, entre los varios disponibles, son necesarios para predecir con exactitud la mediana de precios de una casa
2. Construir un modelo de regresión lineal múltiple para predecir la mediana de los precios utilizando los atributos más importantes
3. Evaluar la exactitud del modelo para predecir nuevos ejemplos

Debido a la naturaleza del enfoque de ajuste de funciones, una limitación importante que encontramos tiene que ver con la dimensionalidad. A medida que la cantidad de atributos o predictores crece, se reduce nuestra capacidad para obtener un buen modelo, pero además se agrega complejidad computacional y también se hace más difícil la interpretación del modelo.

Revisaremos aquí algunos métodos de selección de características – *“feature selection”* – que permitan reducir el número de predictores al mínimo posible sujeto a obtener un buen modelo.

Utilizando RapidMiner, veremos cómo realizar la preparación de los datos, la construcción del modelo y la validación. Finalmente revisaremos que se cumplan algunos requerimientos para asegurar que la regresión lineal se utiliza correctamente.

Ejercicio 1 (15 minutos + 3 de preguntas)

Descargar el dataset de UCI.

Analizar los atributos, y describir en un archivo de texto (para cada uno)

- su contexto y significado,
- tipos de datos y rangos
- distribuciones y outliers
- ¿cuál es la variable de salida?

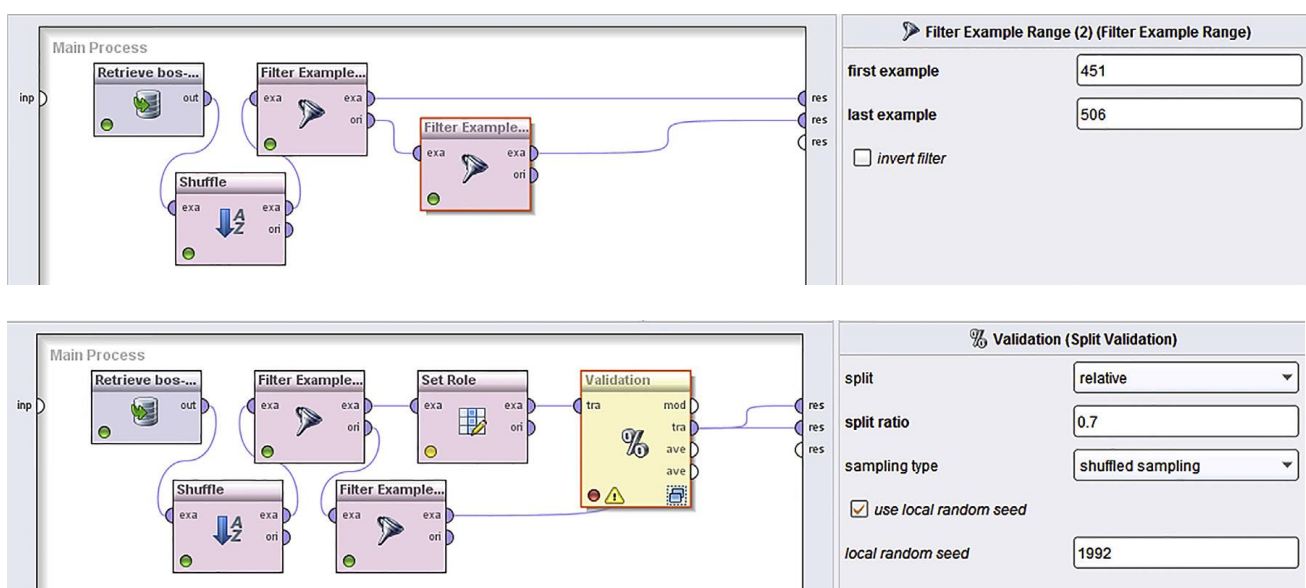
RESPONDER PREGUNTAS PROYECTADAS EN PANTALLA

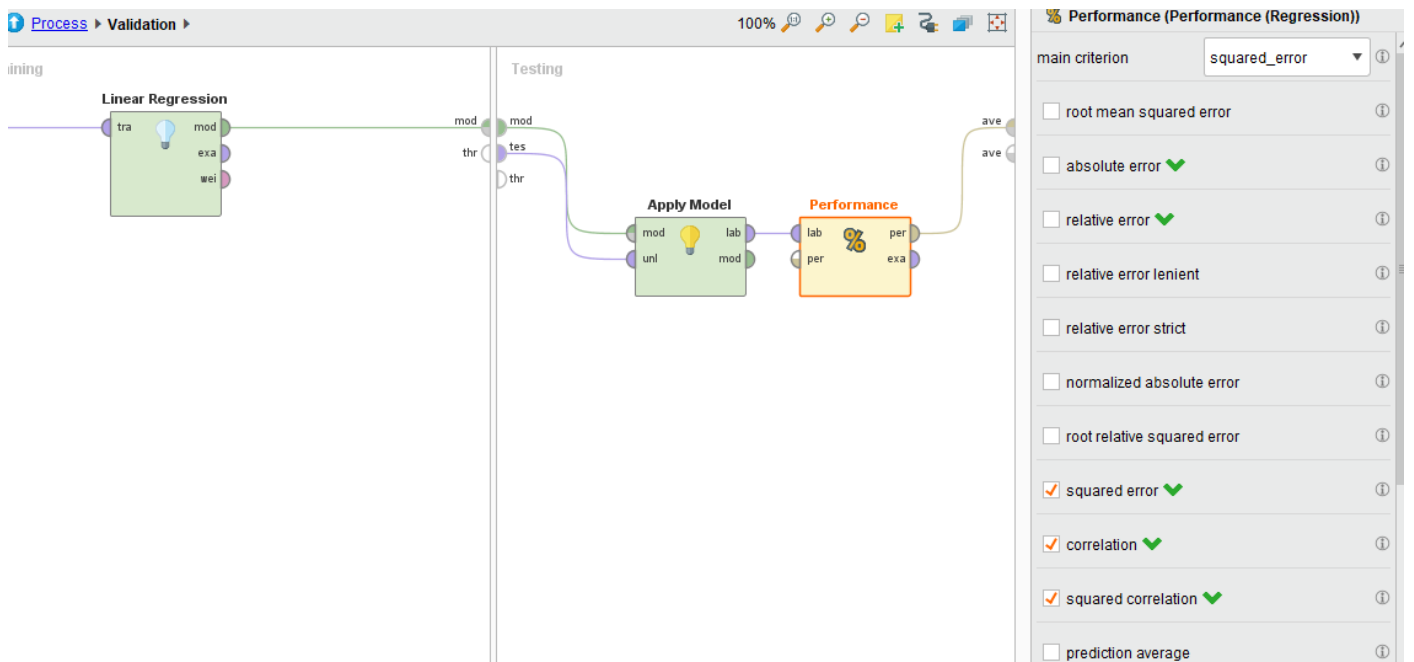
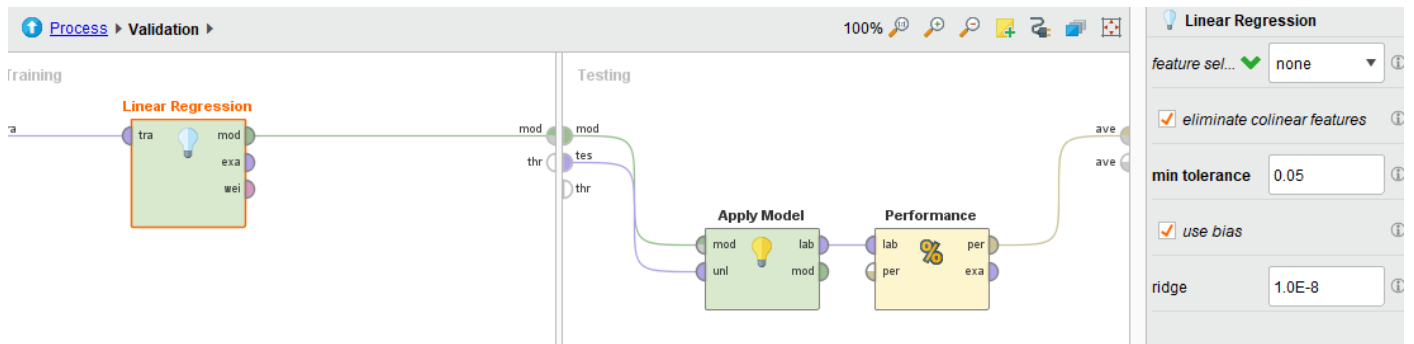
Ejercicio 2 (20 minutos + 5 de preguntas) – Preparación de los datos y construcción del modelo

En primer lugar, separaremos el dataset en un conjunto para entrenamiento y otro conjunto “no visto” de prueba.

Construiremos un modelo a partir del conjunto de entrenamiento y luego probaremos su rendimiento con el conjunto de prueba. Sigue los siguientes pasos:

1. Crear un nuevo proceso en RapidMiner
2. Importar – operador “**retrieve**” - el dataset “housing” descargado y analizar los datos (atributos, tipos de datos, etc). Observar que tiene **506** ejemplos. Aplicar “**set role**” para identificar la variable objetivo.
3. Aplicar el operador “**shuffle**” para randomizar el orden de los datos (así cuando separemos las dos particiones, éstas serán estadísticamente similares)
4. Utilizando (2 veces) el operador “**Filter Examples Range**” dividir el dataset en 2 conjuntos: el conjunto de entrenamiento con los ejemplos **1 – 450**, y el conjunto de test con los ejemplos **451-506**.
5. Conectar el dataset de entrenamiento a un subproceso “**Split Validation**”. En éste, dejar el estándar de partición 70/30. Observar que, si se desea ejecutar el modelo varias veces en las mismas condiciones, será conveniente establecer una semilla para la partición aleatoria.
6. En el subproceso interno de “**Split Validation**”, en el panel izquierdo insertar un operador “**Linear Regression**” y verificar que en el panel derecho se encuentre el operador “**Apply Model**” seguido de “**Performance(Regression)**”.
7. En el operador “**Performance (Regression)**” seleccionar los parámetros “**squared error**”, “**correlation**,” y “**squared correlation**”
8. En los parámetros del operador “**Linear Regression**” seleccionar “**none**” para “**feature selection**”. Observar las otras opciones disponibles. Dejar los otros parámetros por defecto (“**eliminate colinear features**” y “**use bias**” chequeados).





9. RESPONDER PREGUNTAS PROYECTADAS EN PANTALLA

Ejercicio 3 (20 minutos + 5 de preguntas) – Ejecución e interpretación

Ejecutar el modelo con un breakpoint luego del operador “Linear Regression”

1. Observar la información de salida de Linear Regression:

- En “data”:
 - Coeficientes del modelo, errores, y “code” (ordenar por este campo)
 - RapidMiner asigna cuatro estrellas a todo factor que sea altamente significativo

Attribute	Coefficient	Std. Error	Std. Coefficient	Tolerance	t-Stat	p-Value	Code ↑
INDUS	-0.023	0.076	-0.017	0.692	-0.302	0.763	
AGE	0.011	0.017	0.034	0.814	0.675	0.500	
ZN	0.032	0.017	0.081	0.883	1.855	0.065	*
CHAS	2.038	1.109	0.054	0.996	1.837	0.067	*
CRIM	-0.102	0.046	-0.086	0.854	-2.211	0.028	**
TAX	-0.011	0.005	-0.193	0.729	-2.411	0.016	**
B	0.008	0.004	0.068	0.892	2.069	0.039	**
NOX	-15.697	4.839	-0.192	0.830	-3.244	0.001	***
RAD	0.266	0.081	0.238	0.781	3.286	0.001	***
RM	3.916	0.492	0.304	0.594	7.961	0.000	****
DIS	-1.327	0.243	-0.305	0.880	-5.465	0.000	****
PTRATIO	-0.852	0.164	-0.196	0.785	-5.201	0.000	****
LSTAT	-0.610	0.065	-0.451	0.479	-9.329	0	****
(Intercept)	33.529	6.034	?	?	5.557	0.000	****

- En “Description”, observar el modelo en sí mismo (coeficientes de cada predictor)

```
LinearRegression
- 0.102 * CRIM
+ 0.032 * ZN
- 0.023 * INDUS
+ 2.038 * CHAS
- 15.697 * NOX
+ 3.916 * RM
+ 0.011 * AGE
- 1.327 * DIS
+ 0.266 * RAD
- 0.011 * TAX
- 0.852 * PTRATIO
+ 0.008 * B
- 0.610 * LSTAT
+ 33.529
```

2. Tomar nota de los resultados obtenidos para cada predictor.

- ¿cuáles no parecen ser muy significativos?
 - Probar con las opciones disponibles para el parámetro “feature selection”.
 - Utilizar “greedy” y volver a generar el modelo
- Tomar nota de los predictores que se han eliminado del modelo, y observar los coeficientes resultantes

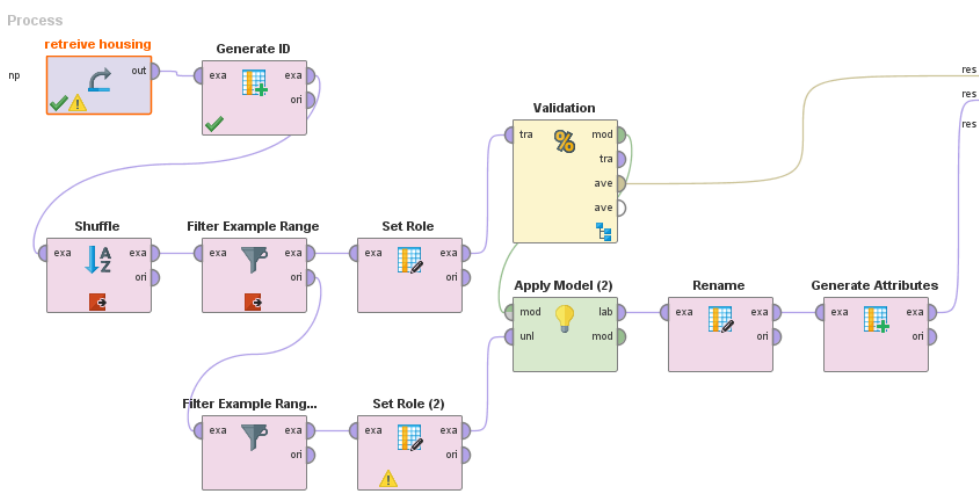
3. Operador **PERFORMANCE**:


- Utilizar “squared correlation” – este es el indicador R^2 visto en clase
 - Tomar nota de los valores obtenidos sin y con feature selection
- Observar también los valores del error medio cuadrático
- RESPONDER PREGUNTAS PROYECTADAS EN PANTALLA**


Ejercicio 4 (20 minutos + 5 de preguntas) - Aplicación sobre datos “no vistos”


Luego del 2do operador “Filter Example Range” necesitamos un 2do “set role”: al atributo “MEDV” lo asignamos como rol objetivo “prediction”.





1. Agregar otro operador “*Apply Model*” y conectar la salida del “*Set Role*” a su puerto de entrada “*unl*”.
2. Conectar la salida “*mod*” del proceso de Validación a la entrada “*mod*” (modelo) del nuevo operador “*Apply Model*”
3. Así hemos cambiado el atributo MEDV de los 56 ejemplos “no vistos” a una “*prediction*”. Cuando apliquemos el modelo a este conjunto de ejemplos, podremos comparar los valores de predicción (MEDV) con los valores originales de MEDV (que existen en nuestro dataset) para probar qué tan bien se comporta nuestro modelo con datos nuevos.
4. La diferencia entre la predicción (MEDV) y MEDV se llama “residuo”. Para visualizar los residuos, primero cambiamos el nombre de *MEDV (prediction)* a “*predictedMEDV*”, y luego podemos utilizar “*Generate Attributes*”, para calcular los residuos (observar cómo se define este nuevo atributo)
5. Observar las estadísticas de los residuos. ¿Qué se destaca?



 Edit Parameter List: function descriptions

 Edit Parameter List: **function descriptions**
List of functions to generate.

attribute name	function expressions
residuals	(predictedMEDV-MEDV) 

 Add Entry  Remove Entry  Apply  Cancel