Aprendizaje automático

Práctica 2: Regularización

GAJAN Antoine (894825)

Escuela de
Ingeniería y Arquitectura
Universidad Zaragoza

Índice

Introducción

Como parte de la asignatura ”Aprendizaje automático”, los estudiantes se familiarizarán con los conceptos de inteligencia artificial. Más concretamente, durante el cuatrimestre se estudiarán las bases del aprendizaje supervisado y no supervisado.

En esta segunda sesión práctica trabajaremos sobre la regularización. Esta sesión nos permitirá poner en práctica los conocimientos teóricos aportados por los cursos magistrales y comprender como elegir el mejor modelo.

Esta memoria pondrá de relieve el proceso emprendido para responder a las preguntas, y producirá una reflexión personal sobre los resultados obtenidos.

1. Estudio previo

En un primer momento, utilizando las transparencias del curso, redactamos el algoritmo de K-Fold-Cross-Validation, teniendo en cuenta la expansión polinómica de los atributos. Así, esta función nos devuelve el mejor modelo con el valor de sus pesos. Este algoritmo tendrá en cuenta la expansión polinómica de los atributos y su normalización.

Algorithm 1 Kfoldcrossvalidation(K, X, y, modelos)

$best\_modelo \leftarrow modelos[1]$

$best\_errV \leftarrow inf$

{Para cada modelo}

for $modelo \in modelos$ **do**

  $err\_T \leftarrow 0$

  $err\_V \leftarrow 0$

{Para cada pliegue}

  for $fold \leftarrow 1$ `to` K **do**

    $[X\_train, y\_train, X\_val, y\_val] \leftarrow particion(fold, K, X, y)$

    $theta \leftarrow Learner(X\_train, y\_train)$

    $err_T \leftarrow err\_T + RMSE(theta, X\_train, y\_train)$

    $err\_V \leftarrow err\_V + RMSE(theta, X\_val, y\_val)$

  end for

{Medio de las errores}

  $err\_T \leftarrow err\_T/K$

  $err\_V \leftarrow err\_V/K$

{Actualizacion del mejor modelo}

  if $err\_V < best\_errV$ **then**

    $best\_errV \leftarrow err\_V$

    $best\_modelo \leftarrow modelo$

  end if

end for

{Aprender con el mejor modelo y todos los datos}

$X\_exp \leftarrow expandir(X, best\_modelo)$

$theta\_best\_modelo \leftarrow Learner(X\_exp, y)$

Como se ha estudiado en clase, K-Fold es una técnica de validación en la que dividimos los datos en k-subconjuntos y el método holdout se repite k-times donde cada uno de los k subconjuntos se utilizan como conjunto de validación y otros subconjuntos se utilizan para el propósito de entrenamiento. Luego se calcula el

error promedio de todos estas iteraciones k, que es más confiable en comparación con el método de aprendizaje estándar.

Podemos utilizar estos algoritmos para los dos objetivos siguientes: determinar el rango de una regresión polinómica y el valor del parámetro de regularización.

Para determinar el grado de una regresión polinómica, podemos proponer el siguiente algoritmo que, para los diferentes atributos, aplica el algoritmo K-Fold. Esto permitirá entonces, comparando los errores medios definidos anteriormente, determinar el mejor modelo, es decir, el que tiene el error de validación más bajo.

Algorithm 2 Grados(K, X, y)

$grados \leftarrow [1..,1]$
{Para cado atributo}
for $atribute \leftarrow 1$ to D **do**
  $best\_modelo \leftarrow Kfoldcrossvalidation(K, X, y, modelos)$
  $grados \leftarrow best\_modelo$
{Entrenamiento del mejor modelo con todos los datos}
  $X \leftarrow expandir(X, grados)$
  $theta \leftarrow Learner(X, y)$

Por último, se nos pidió que diseñáramos un algoritmo para encontrar el mejor valor de lambda en el caso de la regularización. De este modo, utilizando K-Fold Cross-Validation, podemos ofrecer el siguiente algoritmo (que elige entre valores entre 10^-5 y 10^5):

Algorithm 3 Regularizacion(K, X, y)

$best\_errV \leftarrow inf$
$best\_lambda \leftarrow 10^-5$
{Para cado modelo posible con lambda}
for $lambda \leftarrow 10^-5$ to 10^5 with a step of x10 **do**
{Aprendizaje}
  $[err\_T, err\_V] \leftarrow Kfoldcrossvalidation(K, X, y, lambda)$
{Verificacion del mejor lambda}
  if $err\_V < best\_errV$ **then**
    $best\_errV \leftarrow err\_V$
    $best\_lambda \leftarrow lambda$
  end if
{Aprendizaje con todos los datos y el mejor modelo}
  $theta \leftarrow Learner(X, y, best\_lambda)$

2. Memoria de la práctica

2.1. Cuestión 1 : Selección de modelos mediante búsqueda heurística

Se trata de encontrar el mejor modelo a partir de una búsqueda heurística. Para ello, vamos a implementar el algoritmo de K fold cross validation y adaptarlo a nuestro caso concreto. Para ello, vamos a partir del hecho de que el mejor grado para cada atributo es 1. Para cada atributo, vamos a ver qué valor (entre 1 y 10), ayuda a minimizar la función de costo. Se actualizará entonces el grado del atributo i con el mejor grado encontrado. Se prosigue este funcionamiento para los demás atributos. Finalmente, entrenaremos el mejor modelo obtenido con el conjunto de datos. Nos aseguraremos de no olvidar dibujar las curvas de error para cada atributo, ya sea para los datos de entrenamiento o validación. A la vista del número de datos asociados a la fase de entrenamiento (545 datos), aplicaremos la K fold cross validación con K = 10. Esta opción parece ser la más relevante, ya que tenemos un número de datos ni bajo ni alto.

Así, para el atributo x1 (años del vehículo), se obtiene el dibujo de los errores siguientes:

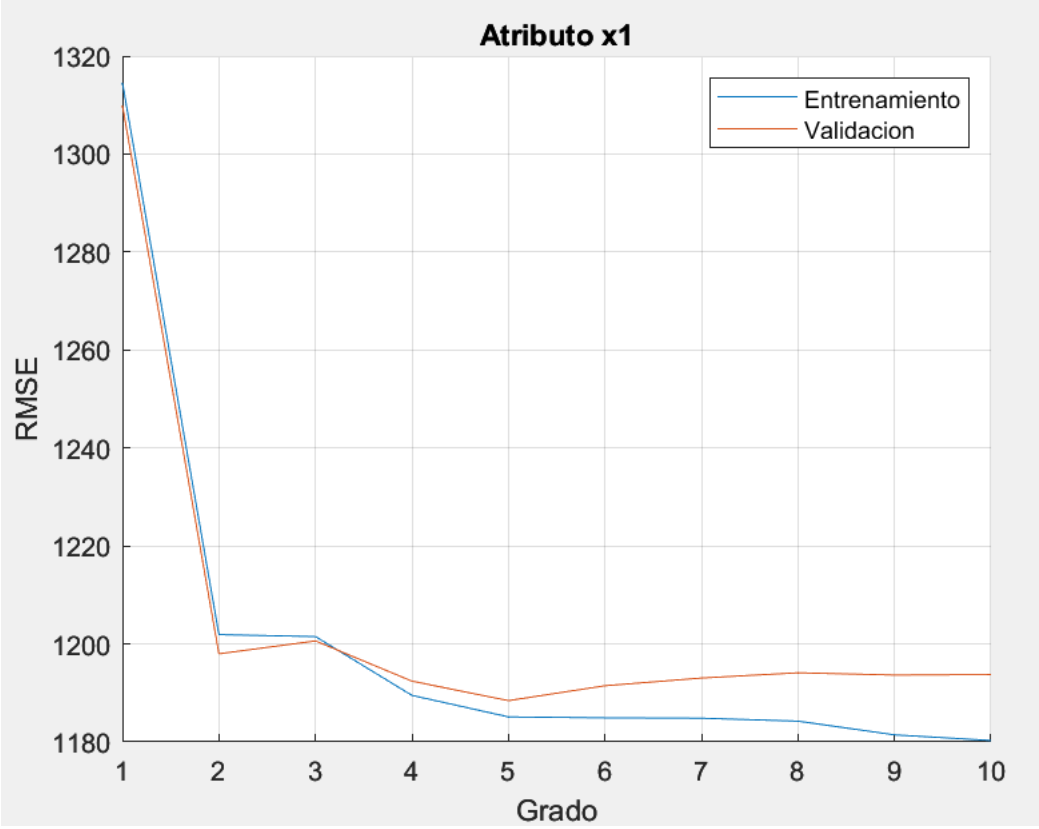

Figura 1: Dibujo de los errores para el atributo x1 en función del grado

Observamos que cuanto más alto es el grado (es decir, cuanto más complejo es el

modelo), menor es el error relacionado con los datos de entrenamiento. En cambio, al proponer un modelo más complejo (grado alto), se observa que el error relacionado con los datos de validación aumenta. Esto es problemático: estamos en el caso de sobreajuste. Con el fin de elegir el mejor modelo (es decir, el mejor valor del grado), tomamos el valor del grado para el que el error en los datos de validación es más bajo. En el caso de x1 (años del vehículo), se observa que se trata de grado = 5.

Como se explica en el algoritmo, ahora vamos a asumir que el grado de x1 es 5 y vamos a ver cuál es el mejor grado para x2 (kilometros del vehículo).

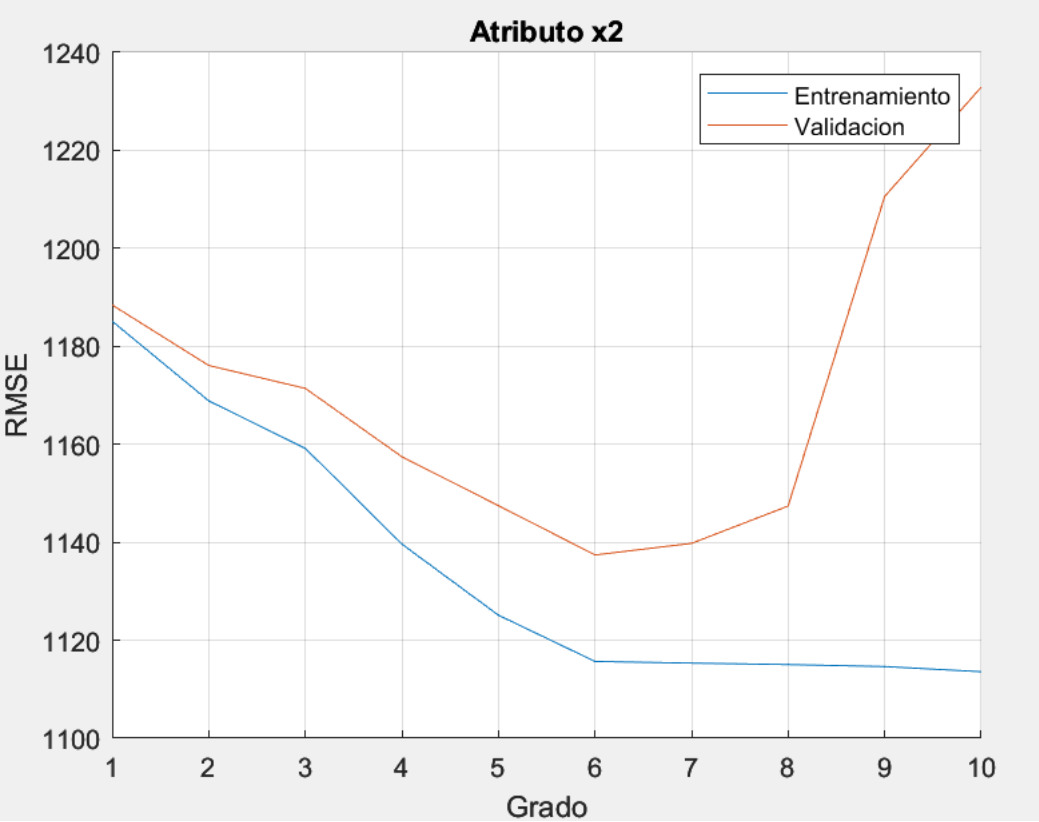

Figura 2: Dibujo de los errores para el atributo x2 en función del grado

Aquí se nota más claramente que grado = 6 es el valor que minimiza el error de validación. A partir de ahora se supondrá que el grado de x2 es 6.

Ahora tenemos que el grado de x1 es 5 y el grado de x2 es 6. Podemos usar el mismo razonamiento que antes para determinar el grado de x3 (CV del vehículo).

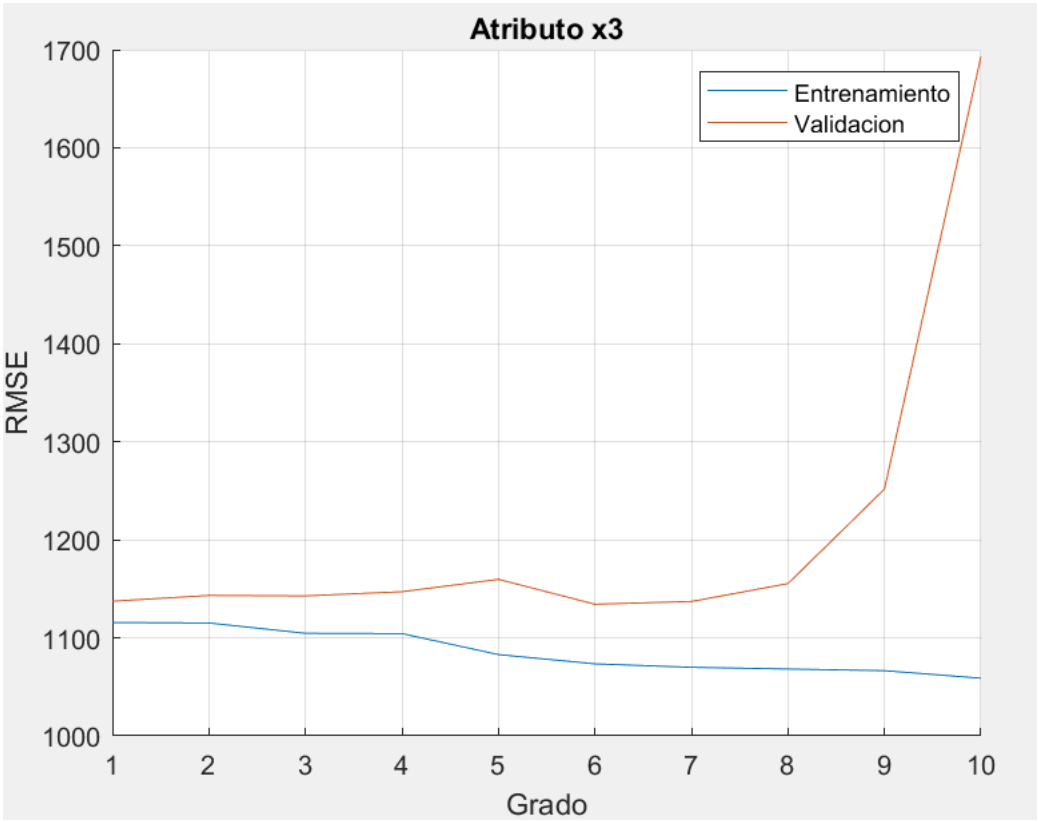

Figura 3: Dibujo de los errores para el atributo x3 en función del grado

Se deduce que el grado de x3 es 6, ya que es este valor el que minimiza el error con los datos de validación.

Con el mejor modelo entrenado en todos los datos, obtenemos los siguientes errores RMSE:

	RMSE
Datos de entrenamiento	1.0765e+03
Datos de test	1.0093e+03

Cuadro 1: Error con el modelo de una selección heurística

El error cometido, ya sea con datos de entrenamiento o pruebas, es del orden de los 1000 €. Este error puede parecer relativamente alto teniendo en cuenta el precio de los coches propuestos en los datos del ejercicio.

2.2. Cuestión 2 : Selección de modelos mediante búsqueda exhaustiva (grid search)

En esta pregunta queremos determinar el mejor modelo polinómico mediante una investigación exhaustiva. Para ello, se crea en un primer momento un cuadro con

el conjunto de las posibilidades. Para ello, utilizaremos las siguientes instrucciones, que permiten generar todas las permutaciones posibles de 3 valores comprendidos entre 1 y 10:

```
[A, B, C] = ndgrid(1:10, 1:10, 1:10);
all_possibilities = [A(:), B(:), C(:)];
```

Luego utilizaremos la K fold cross validation para encontrar entre todos estos modelos ($10^3 = 1000$) cuál es el mejor modelo, es decir, el que minimiza el error en los datos de validación. El resultado es el siguiente:

```
best_modelo = [5  6  6]
```

Se obtiene entonces que el mejor modelo es aquel para el cual el rango asociado a x1 es 5, y los rangos asociados a x2 y x3 valen 6, como con la selección heurística. Así que volvemos a cometer los mismos errores:

	RMSE
Datos de entrenamiento	1.0765e+03
Datos de test	1.0093e+03

Cuadro 2: Error con el modelo de una selección exhaustiva

Observamos que los errores relacionados con los datos de entrenamiento y pruebas son muy similares. Esto nos permite deducir que nuestro modelo está correctamente ajustado. En efecto, si el error de los datos de entrenamiento fuera mayor que el de los datos de pruebas, estaríamos en el caso de uno subajuste, y en el otro caso, en un caso de sobreajuste.

2.3. Cuestión 3 : Regularización

En este pregunta, vamos a programar la búsqueda de un modelo de regresión polinómica de grado 10 para los tres atributos, con regularización.

Esta cuestión es ligeramente más compleja que las anteriores. En efecto, en un primer momento hay que comprender que uno tiene un único modelo (todos los grados valen 10), y que vamos a ver qué valor de lambda permite minimizar el error de validación como se explicó anteriormente.

Para poder resolver este problema reutilizando el código previamente establecido, vamos a añadir a la función de aprendizaje llamada Learner, un atributo correspondiente a lambda. Si el usuario no proporciona lambda (parámetro opcional), entonces se supondrá que se trabajará con lambda = 0. De lo contrario, se trabajará con el valor proporcionado. La ecuación normal cambia ligeramente en el caso en que

lambda es diferente de 0. Así, se encuentra con la fórmula establecida en el precio, según la cual:

```
H = X'*X + lambda*diag([0  ones(1,D)]);
theta = H \ (X' * y)
```

Gracias a esta adaptación del código, podemos escribir fácilmente la función de regularización utilizando la propuesta como estudio de preparación de práctica. Así obtenemos el gráfico siguiente de errores:

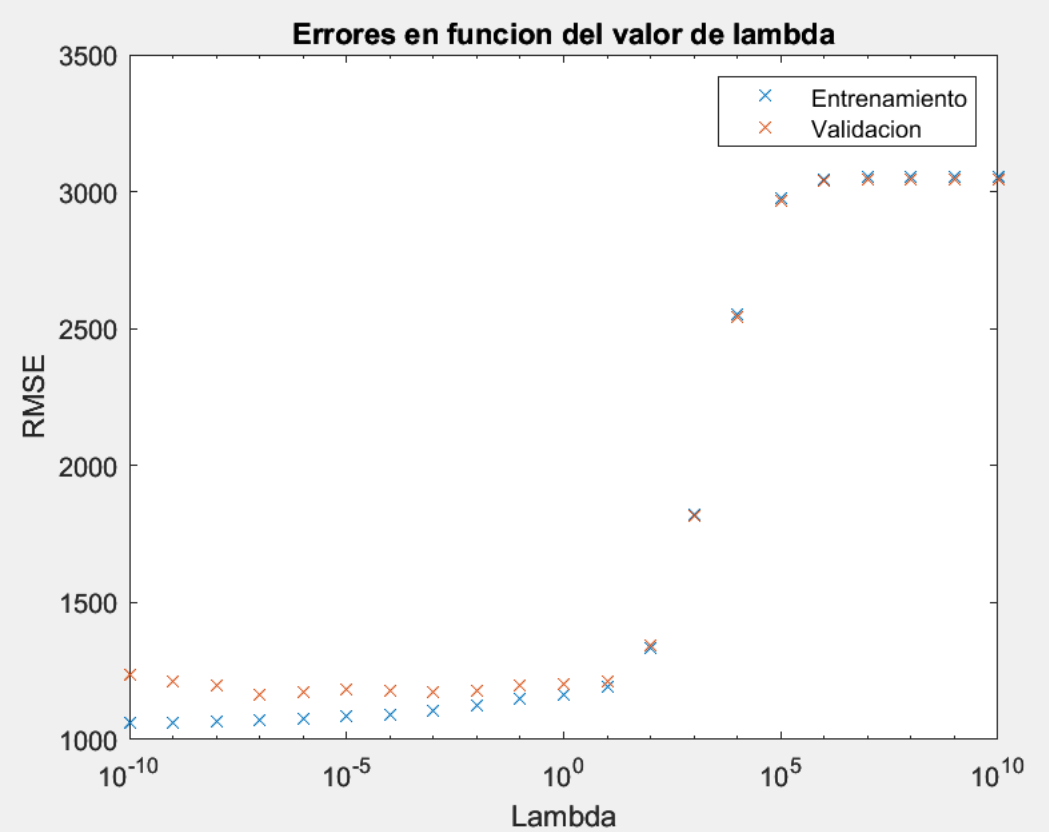

Figura 4: Dibujo de los errores en función de lambda

Encontramos que el RMSE mínimo para los datos de validación se alcanza cuando lambda $= 10e - 7$. Podemos recordar que un valor bajo de lambda permite no penalizar demasiado a los modelos complejos costosos de calcular (es decir, a los grados altos). Podemos recordar que un valor bajo de lambda permite no penalizar demasiado a los modelos complejos costosos de calcular (es decir, a los grados altos). Cuando lambda es alto, se favorecen los modelos complejos y costosos.

Como resultado del aprendizaje de todos los datos del modelo con lambda $= 10^{-}7$, hemos cometido los siguientes errores:

	RMSE
Datos de entrenamiento	1.0722e+03
Datos de test	1.0071e+03

Cuadro 3: Error con el modelo de la regularizacion

Observamos que el error cometido con la regularización es ligeramente menor que con los métodos de selección heurística y exhaustiva. El error es del orden de 1000 €, lo que es bastante alto teniendo en cuenta los precios de los coches ofrecidos en los datos de este ejercicio.

2.4. Cuestión 4 : Resumen de las conclusiones

Obtenemos el siguiente cuadro resumen :

	Datos de entrenamiento	Datos de test
Selección heurística	1.0765e+03	1.0093e+03
Selección exhaustiva	1.0765e+03	1.0093e+03
Regularización	1.0722e+03	1.0071e+03

Cuadro 4: Resumen de los errores RMSE con los diferentes modelos

La selección heurística permitió obtener la misma solución que la selección exhaustiva. Sin embargo, la selección heurística tiene la ventaja de hacer menos iteraciones, ya que hay menos modelos para probar. De hecho, mientras que probamos 30 modelos con la selección heurística, probamos 1000 con la selección exhaustiva. En caso de que no haya muchas combinaciones posibles, se pueden realizar ambos modos de selección. En caso de que haya una multitud de combinaciones posibles, es mejor elegir la selección heurística.

La regularización nos ha permitido determinar cuál es el mejor lambda en este ejercicio. Hemos notado que es bastante débil, lo que significa que no ha penalizado a los modelos con un alto nivel de complejidad. Con la regularización, los resultados obtenidos son ligeramente mejores que los obtenidos con los métodos de selección mencionados anteriormente. De ello se deduce que la regularización es un método adecuado que da buenos resultados cuando el número de atributos es elevado y cada atributo influye en el resultado.

Conclusión

Esta práctica nos ha permitido comprender mejor los diferentes métodos de selección de modelos y regularización, así como sus ventajas y desventajas. Estos conocimientos podrán aplicarse en las próximas prácticas y en casos concretos en las empresas. En particular, podremos diseñar modelos adecuados, ajustados y limitar el sobreajuste.

Índice de figuras

Índice de cuadros