Aprendizaje automático

Práctica 1: Regresión

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 primera sesión práctica trabajaremos sobre la regresión. Esta sesión nos permitirá poner en práctica los conocimientos teóricos aportados por los cursos magistrales y comprender los diferentes ámbitos de aplicación de la regresión.

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 descenso de gradientes, teniendo en cuenta la normalización de los datos, el entrenamiento y la desnormalización de los pesos. Así, esta función nos devuelve el valor de los pesos para la regresión de nuestro modelo.

Algorithm 1 Descenso_gradiente(X, y, alpha, conv_obj)

$N \leftarrow num\_lin\_X$
$D \leftarrow num\_col\_X$
{Calculo de la media y de la varianza de los datos}
$mu \leftarrow [x1...xD]$
$var \leftarrow [x1...xD]$
for $i \leftarrow 1$ to D **do**
  $sum \leftarrow 0$
  $sum\_squared \leftarrow 0$
  for $j \leftarrow 1$ to N **do**
    $sum \leftarrow sum + X[j, i]$
    $sum\_squared \leftarrow sum\_squared + X[j, i]^2$
  end for
  $mu[i] = sum/N$
  $var[i] = sum\_squared - mu[i]^2$
end for
{Normalizacion de los datos}
for $i \leftarrow 2$ to D **do**
  for $j \leftarrow 1$ to N **do**
    $X[j, i] \leftarrow (X[j, i] - mu[i])/var[i]$
  end for
end for
{Aprendizaje}
$theta \leftarrow [x1...xD]'$
while $conv > conv\_obj$ **do**
  $[J, grad, Hess] \leftarrow CosteL2(theta, X, y)$
  $conv \leftarrow norm(grad)$
  $theta \leftarrow theta - alpha * grad$
end while
{Desnormalizacion de los pesos}
{Desnormalizacion de theta[1]}
for $i \leftarrow 2$ to D **do**
  $theta[1] \leftarrow theta[1] - theta[i] * mu[i]/var[i]$
end for
{Desnormalizacion de los autros pesos}
for $i \leftarrow 2$ to D **do**
  $theta[i] \leftarrow theta[i]/var[i]$
end for$=0$

Para comprobar la convergencia en el bucle while, varias opciones se pueden hacer. En particular, el objetivo de convergencia se puede considerar como un valor del coste de error. De lo contrario, lo que he decidido hacer es considerar el objetivo de convergencia como un umbral relacionado con el gradiente. En efecto, se sabe que el mínimo de la función de coste de error se obtiene cuando su derivada vale 0. Generalizando a D dimensiones, esto significa que el gradiente contiene valores muy cercanos a 0 para cada uno de sus componentes. Al usar el gradiente para obtener un escalar (cálculo de su norma), podemos lograr la convergencia de nuestro modelo y compararlo con el objetivo.

2. Memoria de la práctica

2.1. Cuestión 1

Al observar el precio de los apartamentos en función de su superficie, se observa que algunos datos son espurios. No obstante, son pocas. El coste cuadrático se ve muy afectado por los datos erróneos, por lo que descartaremos esta métrica. En cambio, la MAE, el MRE, el MedAE y el MedRE son pertinentes. En efecto, se expresan en euros o en porcentaje y se ven poco afectados por los datos espurios. Así, podemos creer las funciones asociadas para calcular estas métricas.

El MedAE y el MedRE tienen la ventaja de ser insensibles hasta en un 50 % de los datosespurios. Sin embargo, si hay pocos datos espurios (como en esta práctica), la mediana se aproxima a la media. Por eso, nos centraremos aquí en el estudio del MAE y del MRE, que son fácilmente interpretables.

2.2. Cuestión 2

En primer lugar, creamos un vector x que contiene x0 y x1. En efecto, no hay que olvidar añadir x0, de lo contrario el modelo no será correcto.

A continuación, utilizando la fórmula de la ecuación normal vista en curso, y su expresión en Matlab (theta = x \y), se puede deducir fácilmente el valor de los pesos en el vector theta.

Por lo tanto, con los datos de entrenamiento obtenemos que theta = [-3.4313e+04, 2.6082e+03], es decir que precio = -3.4313e+04 + 2.6082e+03 * superficie

Aquí puede ver los gráficos obtenidos para los datos de entrenamiento y los datos de test :

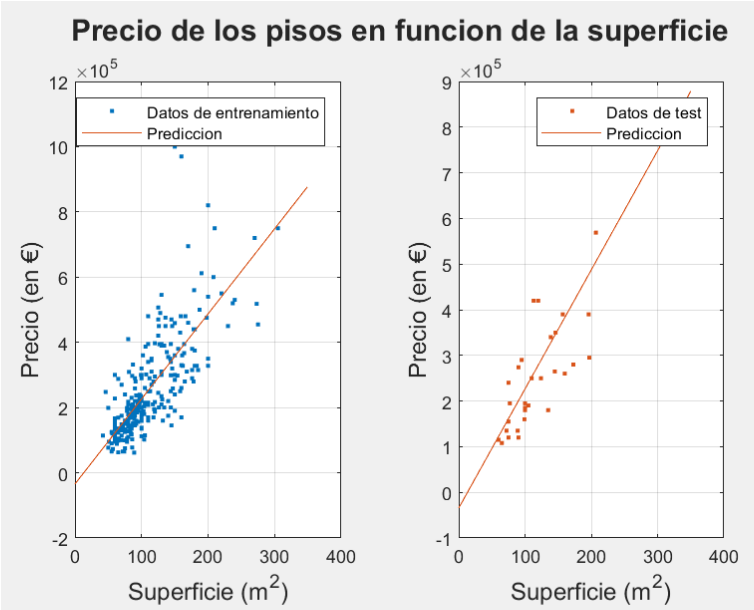

Figura 1: Predicción del precio en función de la superficie con la ecuación normal

Para evaluar nuestro modelo, calculamos las diferentes métricas de error según las fórmulas establecidas anteriormente.

	MAE	MRE
Datos de entrenamiento	6.9534e+04	0.3009
Datos de test	6.3936e+04	0.2744

Cuadro 1: Métricas de errores del modelo de regresión monovariable con la ecuación normal

Estas métricas nos dicen que en promedio cometamos un error de 69 534 € en la estimación del precio. Aunque puede parecer un error alto para algunos, este error puede parecer más aceptable para MRE, que indica que se comete en promedio un error del 30 %. En otras palabras, en un apartamento de 100.000 €, lo estimaremos entre 70.000 y 130.000 €, lo cual es aceptable.

Sin embargo, el error podría mitigarse si consideráramos otro patrón de regresión. En efecto, en el caso real, se espera obtener valores estrictamente positivos para el precio, independientemente de la superficie. Ahora bien, se observa que nuestro modelo de regresión predice valores negativos cuando la superficie del piso es pequeña. Podríamos proponer un modelo ligeramente más complejo, con el fin de obtener una curva que predice únicamente valores positivos, lo que sería más realista.

Para terminar, el error cometido en los datos de entrenamiento y los datos de test es similar, se deduce que el modelo está convenientemente ajustado (sin sobreajuste ni sobajuste).

2.3. Cuestión 3

Con un razonamiento análogo al de la pregunta anterior y generalizando con 2 variables x1 y x2, obtenemos finalmente la ecuación del plano de predicción obtenida siguiente: precio = -1.2133e+04 + 3.0287e+03 * superficie - 1.8853e+04 * numero_de_habitaciones.

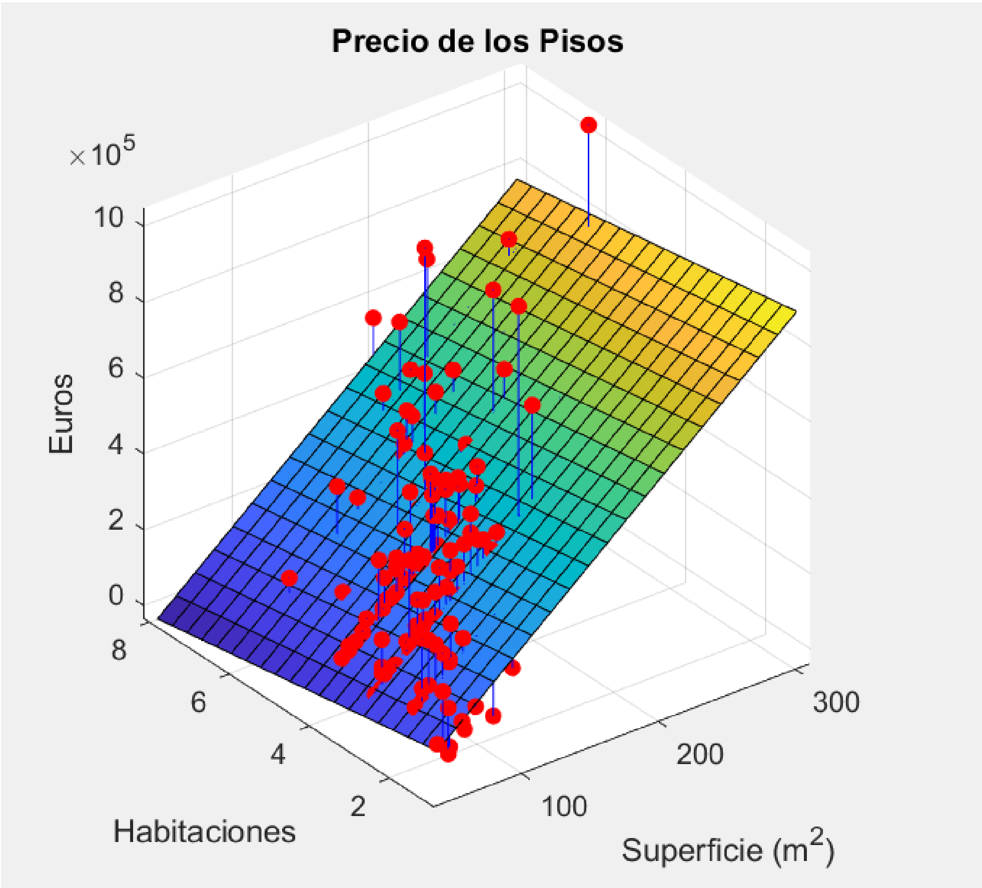

Figura 2: Predicción del precio en función de la superficie y el número de habitaciones con la ecuación normal

Así, podemos evaluar el modelo entrenado y calcular las métricas de error:

	MAE	MRE
Datos de entrenamiento	6.9089e+04	0.2980
Datos de test	5.5256e+04	0.2296

Cuadro 2: Métricas de errores del modelo de regresión multivariable con la ecuación normal

De nuevo, pueden extraerse conclusiones similares a las de la pregunta anterior. Aunque el error absoluto puede parecer grande, éste se debe al hecho de que el precio se cuenta en cientos de miles de euros. En cuanto al error relativo, vuelve a ser un promedio del 30 %, lo que parece aceptable. Sin embargo, sería posible mejorar el modelo basándose en nuestros conocimientos de la vida real, es decir, utilizar un modelo no lineal que pueda asemejarse a una curva para mejorar las predicciones.

2.4. Cuestión 4

2.4.1. Entrenamiento del modelo

Gracias al algoritmo escrito en preparación de la práctica, podemos traducir el algoritmo del descenso de gradientes a Matlab para obtener una función que lo calcula.

De este modo, podemos obtener, utilizando el paso de aprendizaje alfa elegido (alfa = 1) y el umbral de convergencia deseado (aquí 1), la ecuación siguiente: precio = -3.4313e+04 + 2.6082e+03 * superficie

Esto se traduce gráficamente en:

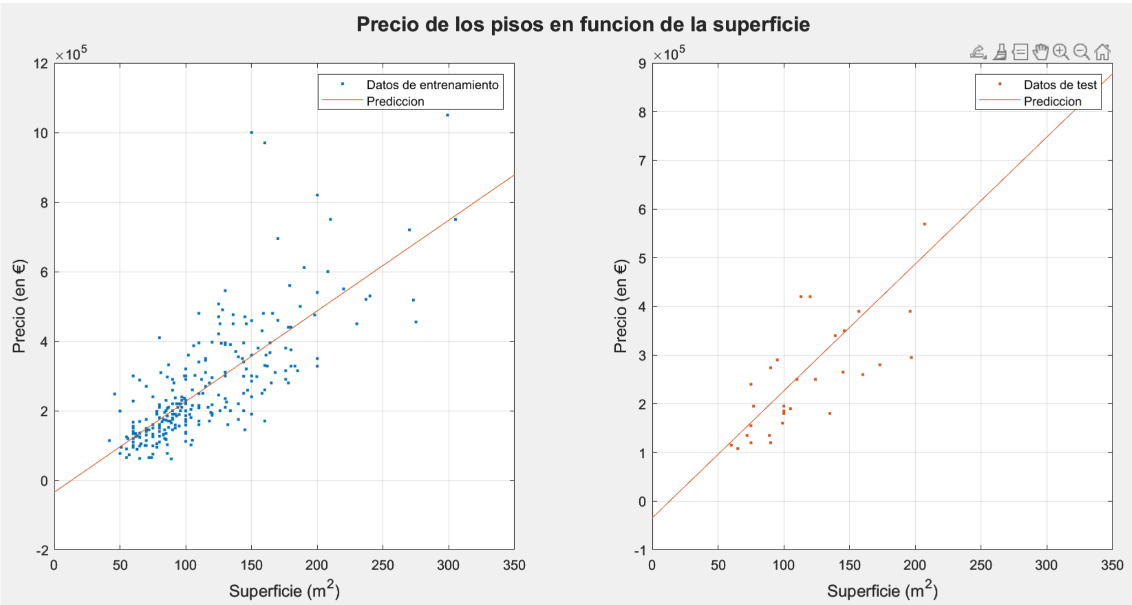

Figura 3: Predicción del precio en función de la superficie con descenso de gradiente

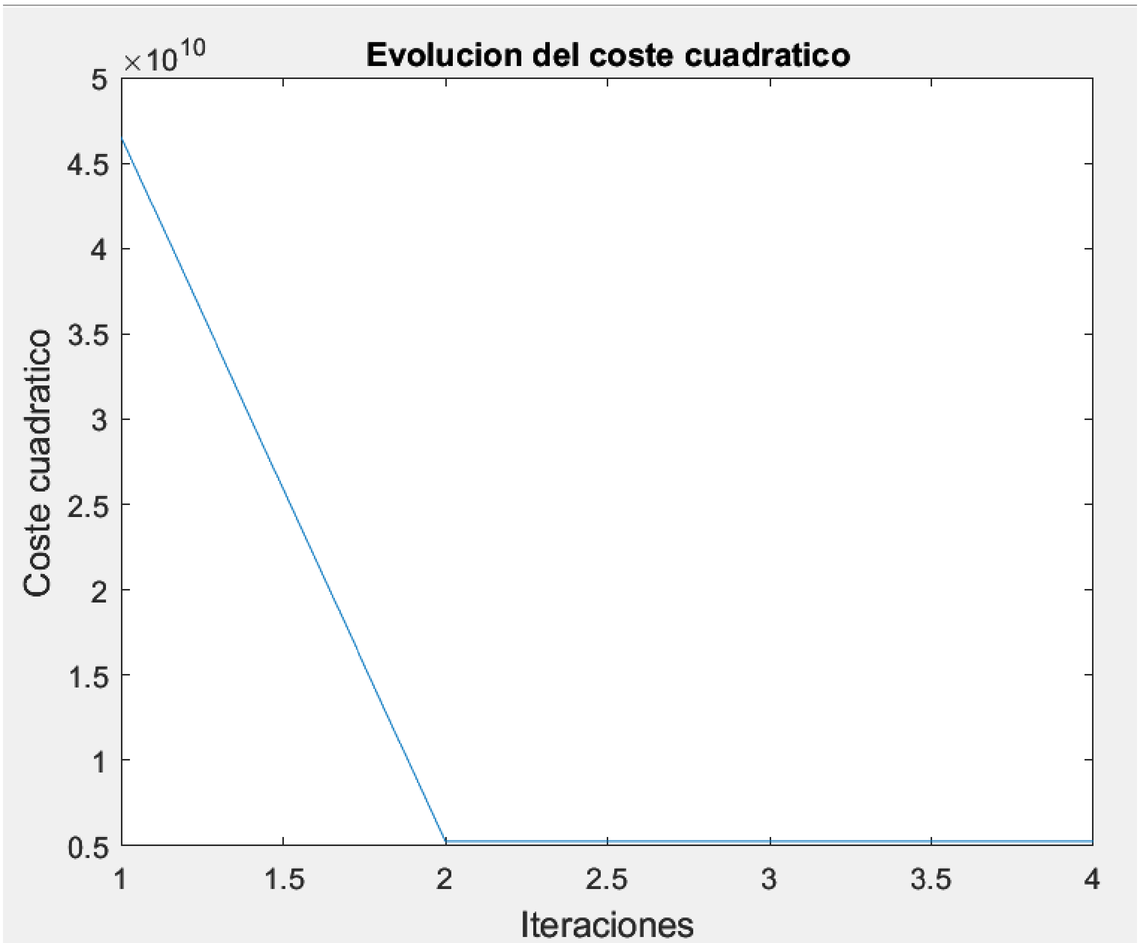

Figura 4: Evolución del coste cuadrático con descenso de gradiente (monovariable)

Luego, con este modelo entrenado, podemos calcular las diferentes métricas de errores para los datos de entrenamiento y de test. Estas últimas se resumen en el cuadro siguiente:

	MAE	MRE
Datos de entrenamiento	6.9535e+04	0.3009
Datos de test	6.3936e+04	0.2744

Cuadro 3: Métricas de errores del modelo de regresión monovariable con el descenso de gradiente

Podemos notar que tenemos exactamente la misma ecuación y por lo tanto, exactamente las mismas métricas de error. Esto se explica por dos visiones diferentes de las cosas. Con la ecuación normal, elegimos resolver matemáticamente el cálculo, calculando el inverso de X, mientras que con el descenso de gradiente, optamos por una búsqueda por ensayo y error, en el sentido de que, con cada iteración, nos acercábamos un poco más a la solución, sin recurrir al cálculo matricial.

2.4.2. Elección del paso de aprendizaje alfa

Para elegir alfa, primero miré la convergencia en función de diferentes alfa a mano. Luego, decidí diseñar un gráfico que resuma el número de iteraciones realizadas para converger con el valor alfa, como se muestra en la siguiente figura:

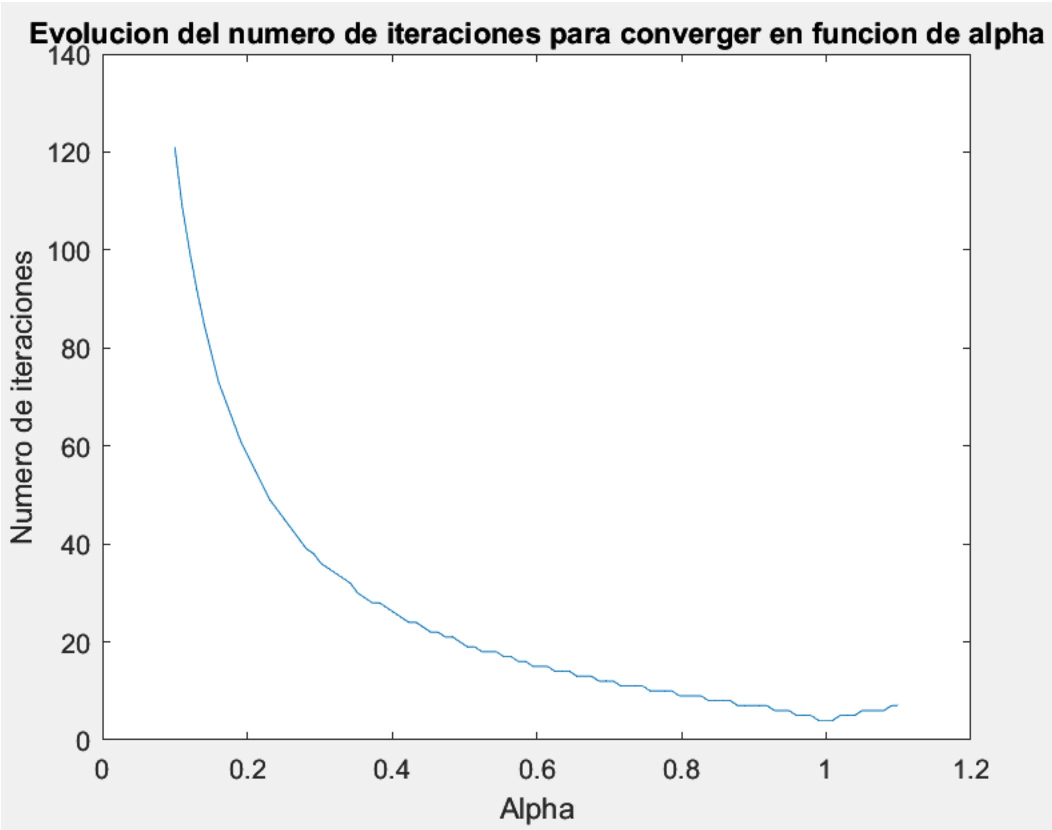

Figura 5: Número de iteraciones para converger con alfa

En el gráfico anterior, se puede observar que el mínimo se alcanza cuando alfa vale 1, lo que explica la elección de este alfa en esta práctica.

2.5. Cuestión 5

Utilizando de nuevo el descenso de gradientes y un razonamiento análogo al de la pregunta anterior, obtenemos la siguiente ecuación: precio = -1.2134e+04 + 3.0287e+03 * superficie - 1.8852e+04 * numero_de_habitaciones

Esto se traduce gráficamente en:

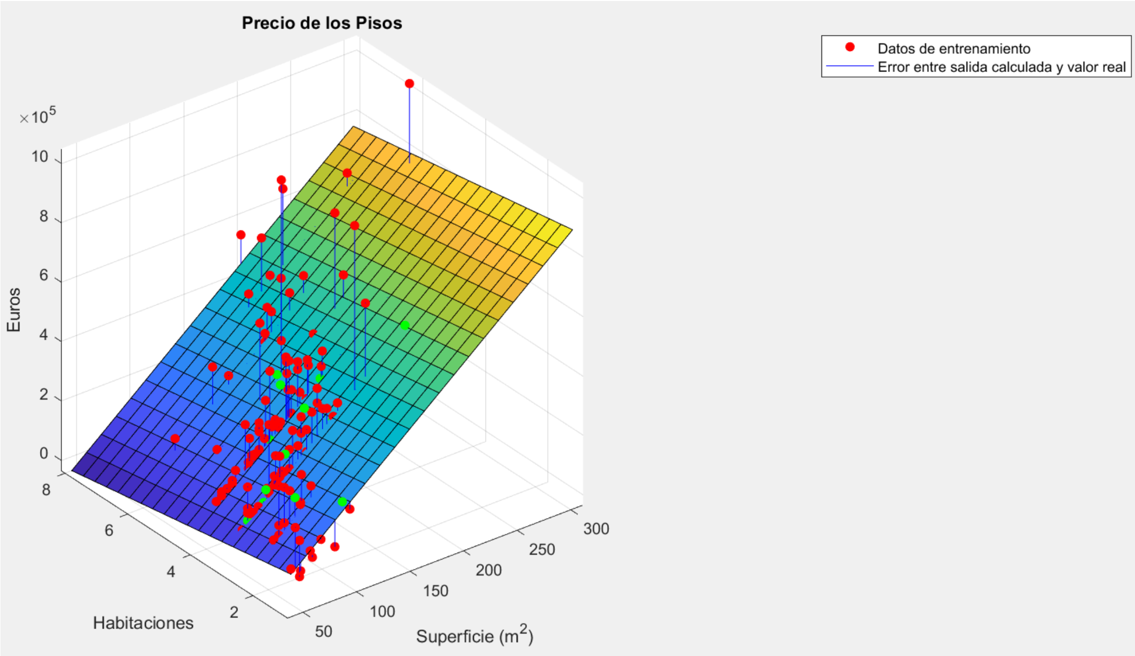

Figura 6: Predicción del precio en función de la superficie con descenso de gradiente

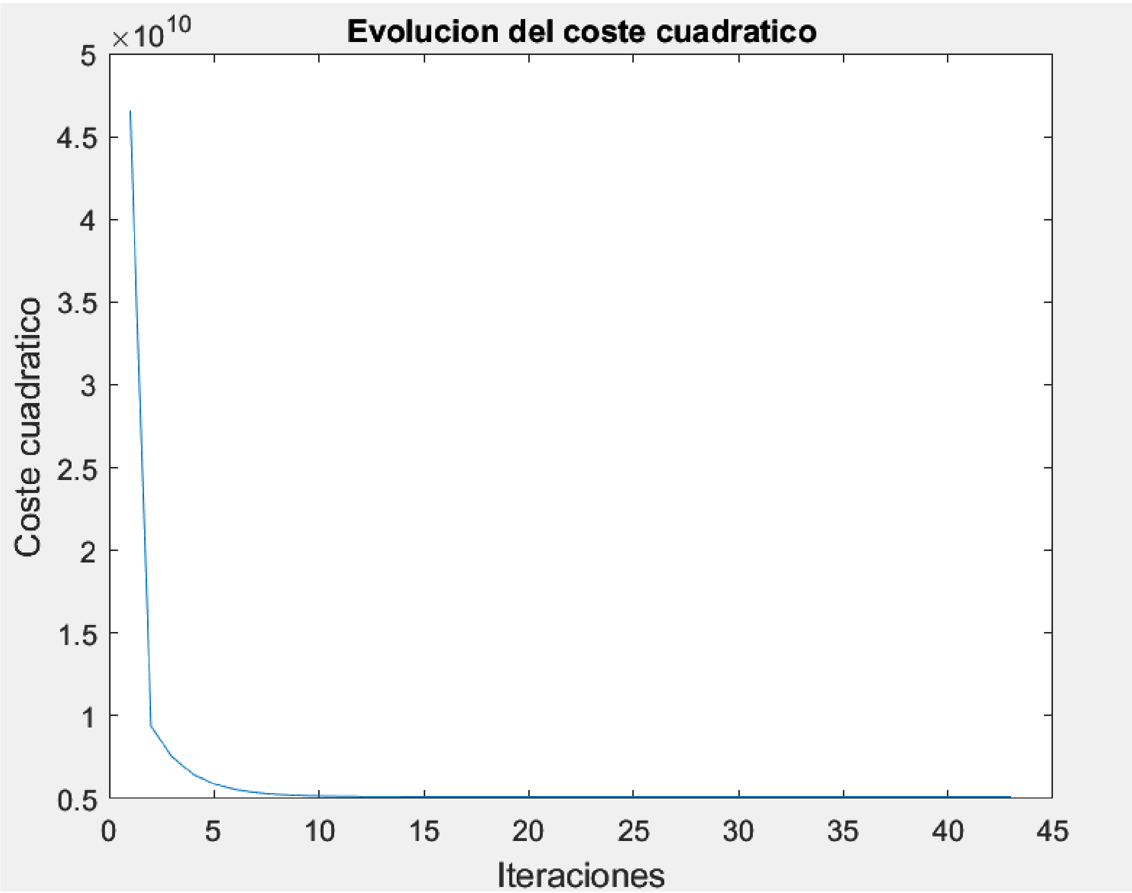

Figura 7: Evolución del coste cuadratico con descenso de gradiente (multivariable)

Al evaluar el modelo, se obtiene:

	MAE	MRE
Datos de entrenamiento	6.9089e+04	0.2980
Datos de test	5.5257e+04	0.2296

Cuadro 4: Métricas de errores del modelo de regresión multivariable con el descenso de gradiente

De nuevo, obtenemos resultados idénticos entre los dos métodos (descenso de gradiente y ecuación normal). También se observa que el MRE asociado a los datos de prueba es ligeramente inferior al asociado al de los datos de entrenamiento. Esto se debe a que los datos de test presentan menos puntos espurios.

2.6. Cuestión 6

2.6.1. Entrenamiento del modelo

Ahora utilizamos el coste de Huber. Esta función se ha definido en el curso y la utilizaremos aquí. Reemplazamos el costo cuadrático por el costo de Huber en el descenso de gradiente.

La función de Huber depende de delta, debemos elegir sabiamente delta. Después de varias pruebas a mano, elegí usar delta = 4e+04 (esta elección se explicará en la siguiente parte). Esto permite lograr la convergencia en un tiempo razonable.

De este modo, podemos obtener, utilizando el paso de aprendizaje alfa elegido (alfa = 1) la ecuación siguiente: precio = -1.1259e+04 + 2.985e+03 * superficie - 1.8167e+04 * numero_de_habitaciones

Esto se traduce gráficamente en:

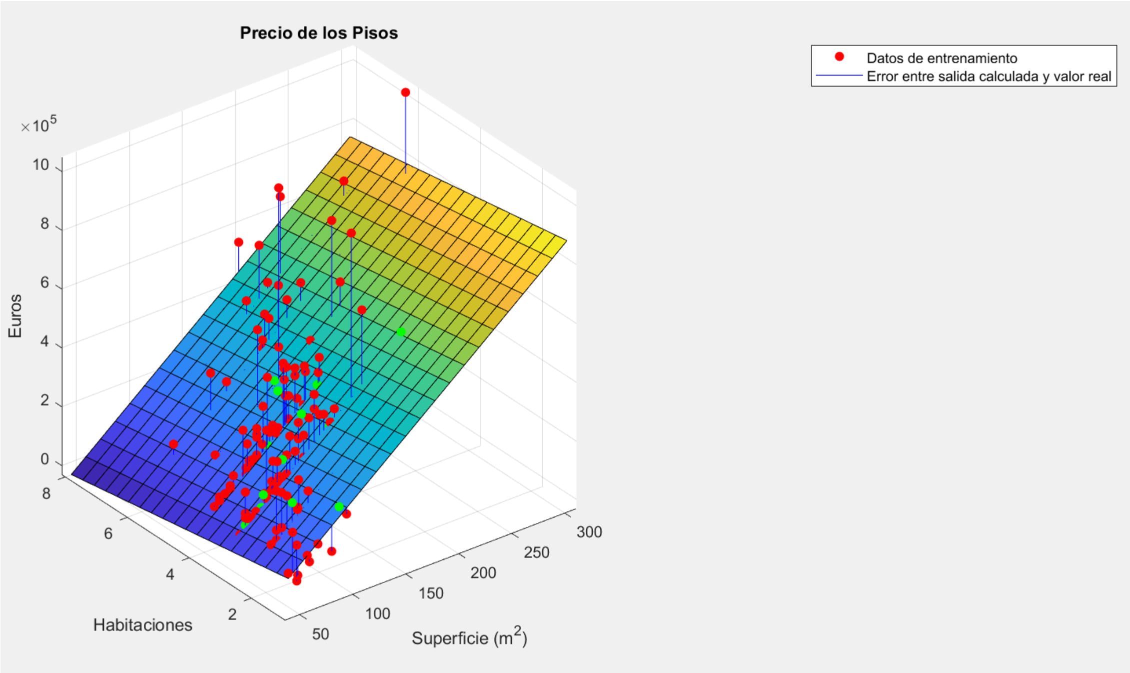

Figura 8: Predicción del precio con la regresión robusta

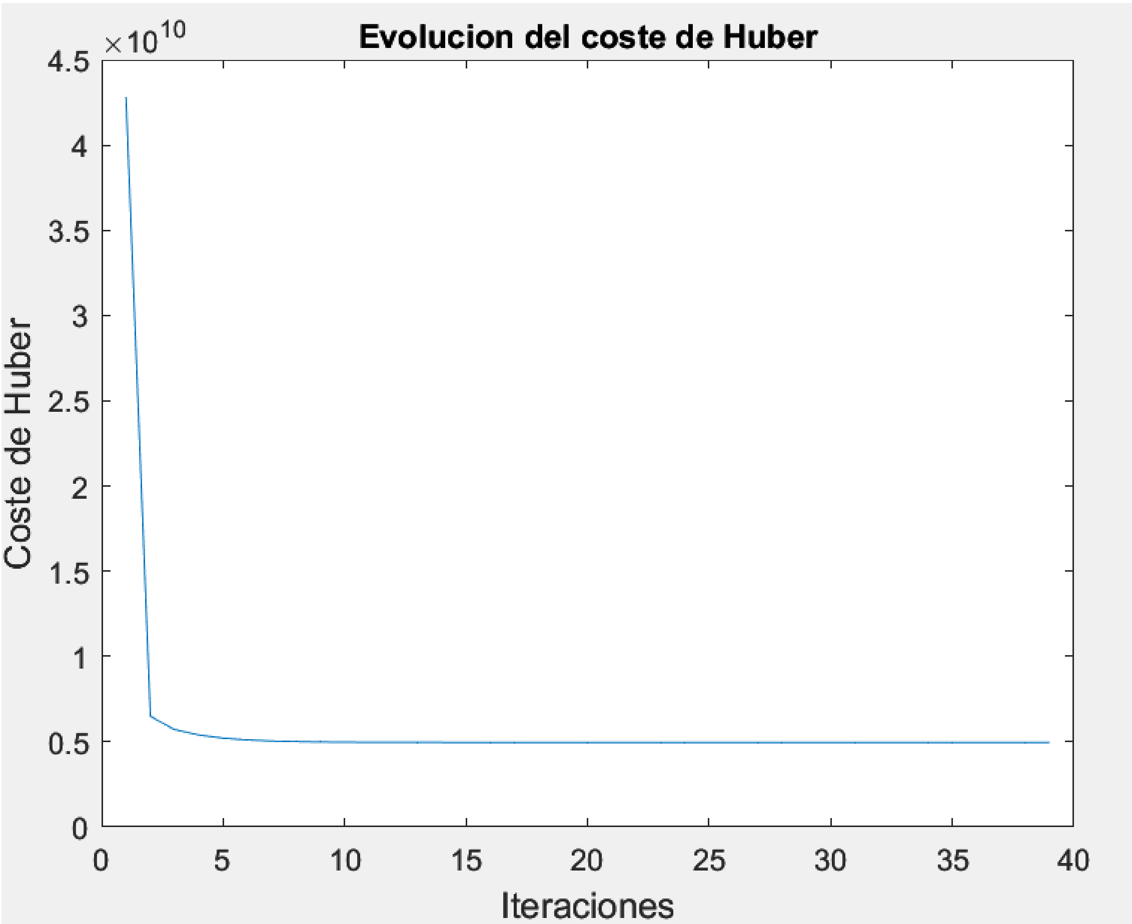

Figura 9: Evolución de la función coste de Huber con la regresión robusta

El coste de Huber nos permite obtener un resultado ligeramente diferente de los obtenidos con la ecuación normal y el descenso de gradientes. Esto se debe a que el coste de Huber es una expresión diferente cuando los residuos calculados son superiores al umbral delta.

Ahora podemos evaluar el modelo para ver si los resultados son mejores (menor error) o no.

	MAE	MRE
Datos de entrenamiento	6.8848e+04	0.2958
Datos de test	5.5041e+04	0.2280

Cuadro 5: Métricas de errores del modelo de regresión robusta

El error cometido es del mismo orden de magnitud que con las mediciones anteriores. Aquí, la diferencia no es significativa, ya que hay muy pocos datos espurios.

2.6.2. Selección del parámetro delta

Intenté elegir un delta que optimizara el tiempo de búsqueda. Para ello, primero elegí alfa de la misma manera que en la cuestión 4 (alfa = 1). Luego dibujé el gráfico del número de iteraciones antes de la convergencia en función del parámetro delta, de la siguiente manera:

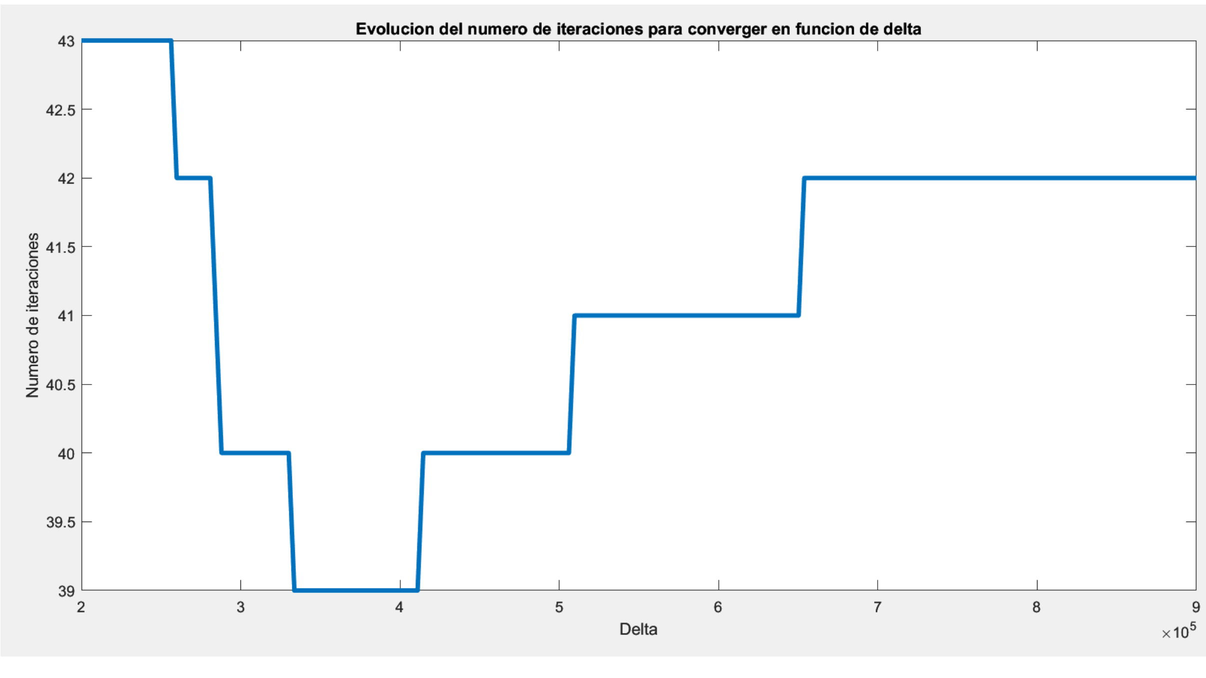

Figura 10: Evolución de la convergencia en función de delta

Se obtiene así que 4e+05 es el "mejor"valor, ya que es en este número que el número de iteraciones es el más bajo. Sin embargo, se observa que el número de iteraciones mínimo con la función coste de Huber es 10 veces mayor que el del descenso de gradiente con la función cuadrática.

De ello se deduce que la función coste de Huber tiene la ventaja de no ser sensible a los datos extremos, pero que, en cambio, su tiempo de aprendizaje es más largo. Por lo tanto, es necesario ver si hay datos extremos antes de comenzar a hacer una regresión para elegir sabiamente el modelo adecuado.

2.7. Cuestión 7

La regresión robusta produce resultados tan precisos como la regresión lineal, pero es mucho menos sensible a la presencia de valores extremos. Por lo tanto, este método de regresión me parece adecuado si tenemos muchos datos extremos. En esta práctica, tenemos relativamente pocos datos extremos, lo que significa que la diferencia entre la regresión lineal y la sólida es muy pequeña. Concluyo de esta práctica los siguientes usos de los diferentes métodos de regresión :

- Regresión lineal con ecuación normal: se utiliza cuando se tienen pocos atributos porque el cálculo de la inversa de la matriz X podría resultar costoso si no.
- Regresión lineal con gradiente descendente: se utiliza cuando se tienen pocos datos extremos, pero tiene la desventaja de tener que elegir un alfa juiciosamente. Pudo ser utilizada en el caso general con cualquier función de coste derivable y con un número muy grande de atributos.
- Regresión robusta: se utiliza cuando se tienen datos extremos que pueden impactar fuertemente la regresión. Sin embargo, existe la desventaja de elegir un delta adecuado. La presencia de este hiperparámetro hace que la función sea más compleja y, por lo tanto, más complicada de optimizar, ya que requiere más tiempo de entrenamiento.

Conclusión

Así, esta práctica ha permitido poner de relieve los diferentes usos de los métodos de regresión lineal.

Estos diferentes métodos de regresión podrán movilizarse en las prácticas siguientes y en la continuación de nuestro curso universitario.

Índice de figuras

Índice de cuadros