

# Práctica 1

Regresión

**Javier Herrer Torres** (NIP: 776609)

Aprendizaje automático  
Grado en Ingeniería Informática



**Escuela de  
Ingeniería y Arquitectura  
Universidad Zaragoza**

Escuela de Ingeniería y Arquitectura  
Universidad de Zaragoza  
Curso 2020/2021

## Índice

<b>1. Objetivos</b>	<b>1</b>
<b>2. Metodología</b>	<b>1</b>
2.1. Estudio previo: Algoritmo de descenso de gradiente . . . . .	1
2.2. Ecuación normal . . . . .	2
2.3. Función de coste . . . . .	2
2.4. Regresión Robusta (Coste de Huber) . . . . .	2
<b>3. Resultados</b>	<b>3</b>
3.1. Regresión monovariante con ecuación normal . . . . .	3
3.2. Regresión multivariante con ecuación normal . . . . .	4
3.3. Regresión monovariante con descenso de gradiente . . . . .	5
3.4. Regresión multivariante con descenso de gradiente . . . . .	5
3.5. Regresión robusta con el coste de Huber . . . . .	6
<b>4. Conclusiones</b>	<b>7</b>

## 1. Objetivos

El objetivo es aplicar las técnicas de regresión lineal en casos reales y comparar los distintos algoritmos de resolución estudiados en clase.

## 2. Metodología

### 2.1. Estudio previo: Algoritmo de descenso de gradiente

Este algoritmo consiste en repetir la siguiente actualización de costes hasta que se observe convergencia:

$$\theta_{k+1} := \theta_k - \alpha g(\theta_k)$$

La implementación del mismo se encuentra en la función auxiliar `descenso_gradiente.m` realizada con la ayuda de la función `regresion` que aparece en la transparencia 28 del tema Regresión.

#### Inicialización y terminación del algoritmo

Será necesario determinar un valor para  $\alpha$  (factor de aprendizaje). Si es muy pequeño, el descenso de gradiente será lento. Por el contrario, si es muy grande el descenso de gradiente puede pasar de largo el mínimo. Puede fallar convergiendo, incluso divergiendo. La elección del factor de aprendizaje se ha realizado depurando el descenso de gradiente probando valores de  $\alpha : \{0,001, 0,01, 0,1, 1\}$  y observando si convergía más rápido.

Antes de comenzar el bucle, se ha realizado una primera iteración comprobando la  $|\Delta J|$ . El algoritmo terminará tras comprobar que ha convergido correctamente.

### Comprobación de la convergencia

Se ha empleado una variable auxiliar (`J_diff`) para obtener la diferencia de los costes, y se ha determinado un umbral  $|\Delta J| < 10^{-3}$  para la finalización del algoritmo.

### Escalado de atributos

Antes de iniciar el algoritmo, se han normalizado los atributos utilizando la función `normalizar.m` proporcionada en Moodle. Escalar facilita la convergencia del descenso de gradiente, sobre todo teniendo en cuenta los diferentes órdenes de magnitud de los atributos:  $x_1 = size(0 - 2000\text{feet}^2)$  y  $x_2 = number\_of\_bedrooms(1 - 5)$ . Por último, se ha procedido a desnormalizar los atributos mediante la función `desnormalizar.m` proporcionada en Moodle.

## 2.2. Ecuación normal

A diferencia del descenso de gradiente, es una solución analítica directa con coste cuadrático en la que no es necesario elegir  $\alpha$ .

$$(X^T X)\hat{\theta} = X^T y$$

En el mínimo se cumplirá:  $g(\hat{\theta}) = X^T X\hat{\theta} - X^T y = 0$

## 2.3. Función de coste

Para realizar el cálculo de la función de coste, se ha tenido en cuenta que la media de los residuos en una regresión lineal es siempre 0 porque se compensan los positivos y negativos así que tendría más sentido emplear una función de coste como la MSE (error cuadrático medio) o la RMSE (*Root-Mean-Square Error*). Para este cálculo se ha utilizado la función `RMSE.m` proporcionada en Moodle.

$$RMSE(\theta) = \sqrt{\frac{SSE(\theta)}{N}}$$

## 2.4. Regresión Robusta (Coste de Huber)

La regresión utilizando el coste de Huber se comporta mejor que el coste cuadrático ante casos atípicos. Con coste cuadrático, un error el doble de grande influye cuatro veces más, y los datos espurios pueden incluir demasiado en la solución (puntos palanca). Será necesario escoger  $\delta$  según el residuo razonable de los puntos buenos. Para el cálculo del coste de Huber en Matlab se ha empleado la función `CosteHuber.m` de las transparencias.

$$J_H(\theta) = \sum_{i=1}^N L_H(r^{(i)}, \delta) = \sum_{|r^{(i)}| \leq \delta} r^{(i)^2}/2 + \sum_{|r^{(i)}| > \delta} \delta|r^{(i)}| - \delta^2/2$$

### 3. Resultados

#### 3.1. Regresión monovariable con ecuación normal

##### Ecuación de predicción obtenida

Se ha obtenido la siguiente ecuación de predicción:

$$\hat{y} = h_{\theta}(x_1) = -34313,39 + 2608,21x_1$$

, donde  $x_1$  corresponde al atributo de la superficie ( $m^2$ ). En la figura 1 se encuentra dicha ecuación dibujada en una gráfica junto con los puntos de entrenamiento.

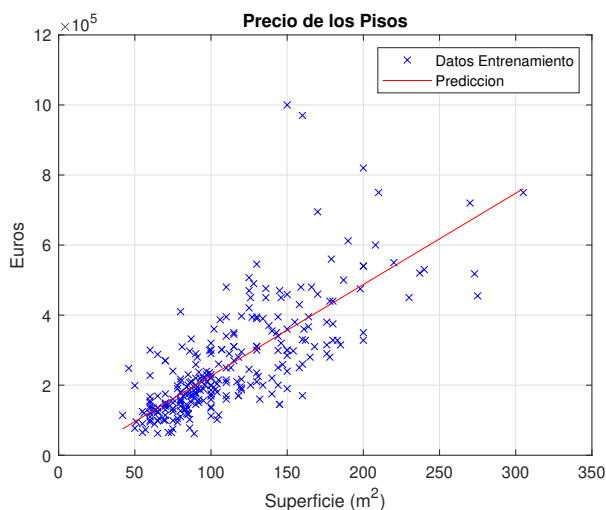


Figura 1: Recta de regresión monovariable con ecuación normal

##### Comparación de residuos

Posteriormente, se han comparado los residuos obtenidos con los puntos de entrenamiento y los de test. Se han obtenido los siguientes valores:

$$RMSE_{train} = 102739,71$$

$$RMSE_{test} = 80628,54$$

Se puede apreciar que el  $RMSE_{train} > RMSE_{test}$ . Esto puede deberse a una mayor presencia de datos atípicos en los datos de entrenamiento que en los de validación.

### 3.2. Regresión multivariable con ecuación normal

#### Ecuación de predicción obtenida

Se ha obtenido la siguiente ecuación de predicción:

$$\hat{y} = h_{\theta}(x_1, x_2) = -12132,91 + 3028,74x_1 - 18852,83x_2$$

, donde  $x_1$  corresponde al atributo de la superficie ( $m^2$ ) y  $x_2$  al número de habitaciones. En la figura 2 se encuentra dibujada en una gráfica junto con los puntos de entrenamiento.

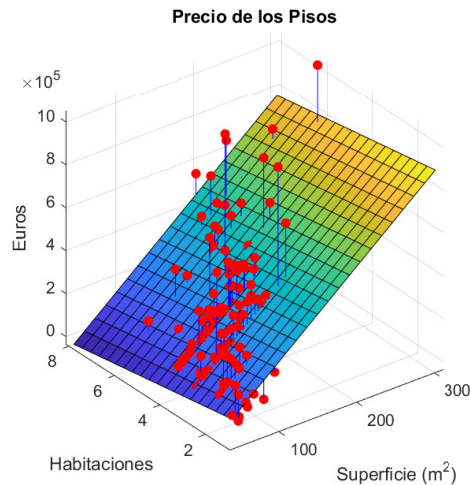


Figura 2: Recta de regresión multivariable con ecuación normal

#### Comparación de residuos

Posteriormente, se han comparado los residuos obtenidos con los puntos de entrenamiento y los de test. Se han obtenido los siguientes valores:

$$RMSE_{train} = 101314,34$$

$$RMSE_{test} = 74740,53$$

Se puede apreciar que el  $RMSE_{train} > RMSE_{test}$ . Esto puede deberse a una mayor presencia de datos atípicos en los datos de entrenamiento que en los de validación.

### Comparación de resultados

Finalmente, se han comparado los resultados obtenidos en el apartado 3.1 respecto a 4 muestras en la que el atributo  $x_1 = 100m^2$ , y el atributo  $x_2 \in \{2, 3, 4, 5\}$  habitaciones. Para la regresión monovariable se ha obtenido la predicción:

$$\hat{y} = h_{\theta}(100) = 226507,56$$

Este modelo únicamente toma en cuenta el atributo  $x_1$  por lo que no sería necesario comprobar el resultado modificando  $x_2$  ya que siempre obtendríamos la misma predicción. Para la regresión multivariable se han obtenido los siguientes resultados para una superficie constante de  $x_1 = 100m^2$ :

$$\hat{y} = h_{\theta}(100, 2) = 253035,87$$

$$\hat{y} = h_{\theta}(100, 3) = 234183,04$$

$$\hat{y} = h_{\theta}(100, 4) = 215330,21$$

$$\hat{y} = h_{\theta}(100, 5) = 196477,38$$

Se observa por lo tanto que el precio disminuye a medida que aumenta el número de habitaciones para una superficie constante. Esto permite deducir que el modelo de regresión multivariable es mejor que el monovariable ya que se obtienen unos resultados más exactos al tener en cuenta el atributo  $x_2$ .

### 3.3. Regresión monovariable con descenso de gradiente

#### Evolución del coste

Se ha realizado el descenso de gradiente tal y como se ha expuesto en el punto 2.1, seleccionando un factor de aprendizaje  $\alpha = 0,001$ . Se ha implementado una función auxiliar `descenso_gradiente.m`, que también dibuja la gráfica de evolución del coste para verificar la convergencia (ver figura 3).

#### Ecuación de predicción obtenida

Se ha obtenido la siguiente ecuación de predicción:

$$\hat{y} = h_{\theta}(x_1) = -34313,39 + 2608,21x_1$$

, donde  $x_1$  corresponde al atributo de la superficie ( $m^2$ ). Se puede apreciar que se han obtenido exactamente los mismos resultados que en el apartado 3.1. Esto se explicará en Conclusiones.

### 3.4. Regresión multivariable con descenso de gradiente

#### Evolución del coste

Se ha realizado el descenso de gradiente tal y como se ha expuesto en el punto 2.1, seleccionando un factor de aprendizaje  $\alpha = 0,003$ . Se ha implementado una función auxiliar `descenso_gradiente.m`, que también dibuja la gráfica de evolución del coste para verificar la convergencia (ver figura 4).

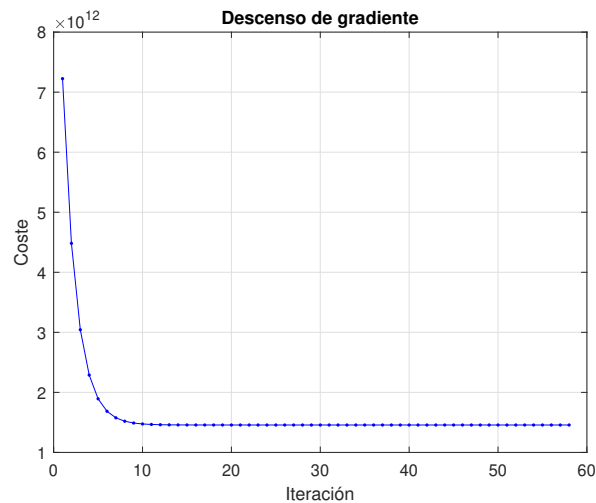


Figura 3: Regresión monovariable. Evolución del coste con descenso de gradiente

### Ecuación de predicción obtenida

Se ha obtenido la siguiente ecuación de predicción:

$$\hat{y} = h_{\theta}(x_1, x_2) = -12132,91 + 3028,74x_1 + -18852,83x_2$$

, donde  $x_1$  corresponde al atributo de la superficie ( $m^2$ ) y  $x_2$  al número de habitaciones. Se puede apreciar que se han obtenido exactamente los mismos resultados que en el apartado 3.2. Esto se explicará en [Conclusiones](#).

## 3.5. Regresión robusta con el coste de Huber

### Evolución del coste

Se ha repetido la regresión multivariable con descenso de gradiente del punto 3.4 utilizando regresión robusta con el coste de Huber (explicado en el punto 2.4). Se ha seleccionado un factor de aprendizaje  $\alpha = 0,003$  y  $\delta = 50000$  como residuo razonable de los puntos buenos. Se ha implementado una función auxiliar `descenso_gradiente_huber.m` que emplea a su vez la función `CosteHuber.m` para calcular el coste, y también dibuja la gráfica de evolución del coste para verificar la convergencia (ver figura 5).

### Ecuación de predicción obtenida

Se ha obtenido la siguiente ecuación de predicción:

$$\hat{y} = h_{\theta}(x_1, x_2) = -12571,18 + 2765,04x_1 - 14629,85x_2$$

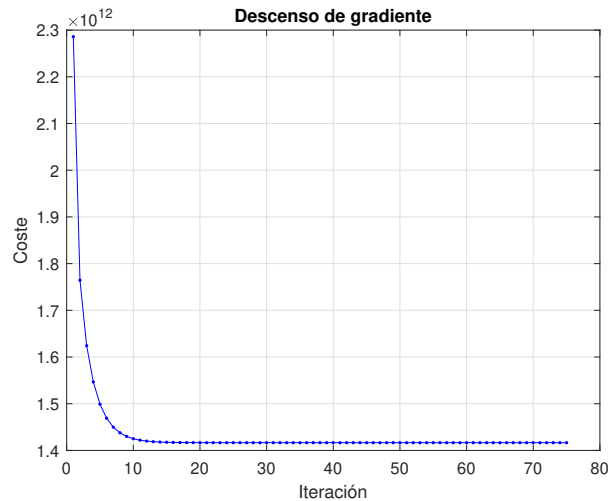


Figura 4: Regresión multivariable. Evolución del coste con descenso de gradiente

, donde  $x_1$  corresponde al atributo de la superficie ( $m^2$ ) y  $x_2$  al número de habitaciones. En la figura 6 se encuentra dibujada en una gráfica junto con los puntos de entrenamiento.

### Comparación de residuos

Posteriormente, se han comparado los residuos obtenidos con los puntos de entrenamiento y los de test. Se han obtenido los siguientes valores:

$$RMSE_{train} = 102738,50$$

$$RMSE_{test} = 70521,19$$

Se puede apreciar que el  $RMSE_{test}$  se ha visto reducido en un 5,65 % respecto al calculado en el apartado 3.2.

## 4. Conclusiones

Comparando los resultados obtenidos de los algoritmos de descenso de gradiente y ecuación normal, se puede observar que se han obtenido las mismas ecuaciones de predicción aunque por distintos métodos. Esto se debe a que los parámetros  $\alpha$  y el umbral  $|\Delta J|$  empleados en el descenso de gradiente implican una gran precisión.

Para este caso en específico, en el que la D no es muy grande, el algoritmo de ecuación normal es mejor ya que se obtiene una solución analítica directa con un coste computacional aceptable. Sin embargo, el descenso de gradiente es una solución iterativa en la que es necesario escoger un factor de aprendizaje



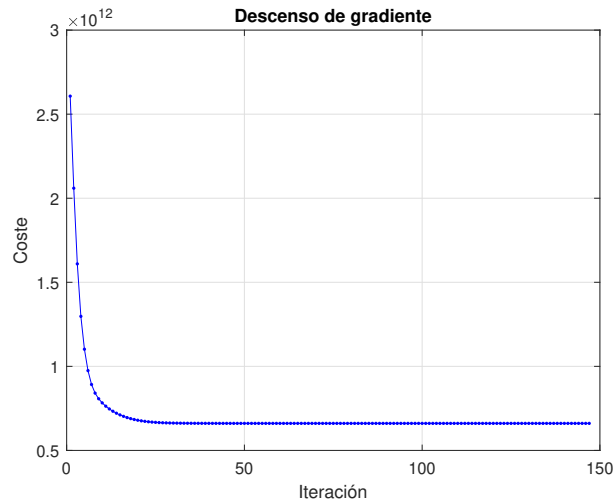


Figura 5: Coste de Huber. Evolución del coste con descenso de gradiente

mediante depuración. Además, aunque se han obtenido los mismos resultados, en el caso general el descenso de gradiente puede converger a un mínimo local.

Por último, se ha comprobado que se cumple lo expuesto en el apartado 2.4 ya que al reducir el impacto de los datos atípicos (puntos palanca), se ha reducido también el  $RMSE_{test}$ .

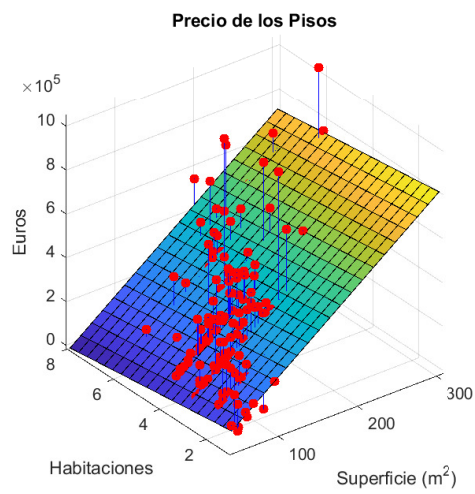


Figura 6: Coste de Huber. Recta de regresión multivariable