**PRACTICA 1: REGRESION LINEAL**

**Una variable:**

**Objetivo:**

Dado una conjunto de datos y para valores de x, encontrar los parámetros en la función lineal (hipótesis) h(x) para los cuales se minimice el valor de la función de coste J(θ0 ,θ1)

**Funciones:**

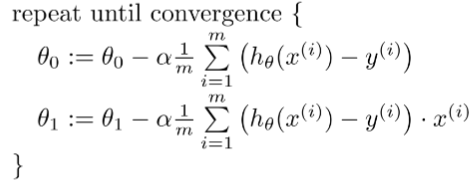
hipothesis.png

coste.png

**Método:**

Se utilizará un algoritmo para aplicar el método de descenso de gradiente para una variable, que irá actualizando los valores de θ0 y θ1 simultáneamente durante un número arbitrario de repeticiones.

**ALGORITMO:**



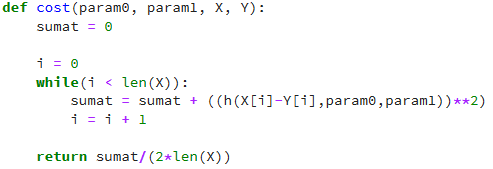
α = 0,015 1500 iteraciones

**Código:**

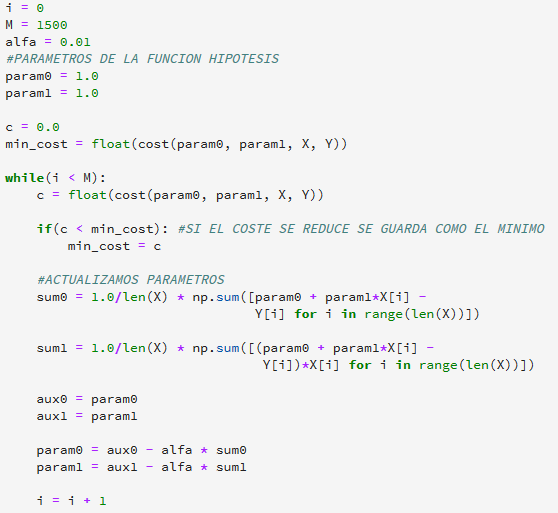
**Función hipótesis:**

**hipotesis python.png**

**Función coste:**

****

**Algoritmo de actualización de parámetros:**

****

**RESULTADOS DE EJECUCIÓN:**

Los valores finales de los parámetros son:

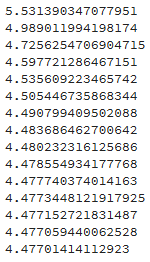
θ0 = -3,57

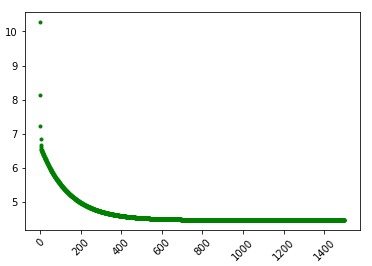
θ1 = 1,16

El valor de J(-3,57, 1,16) es 0.7143680244742163

**Evolución del coste:**

Valor de la función coste cada 100 iteraciones:

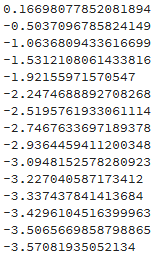
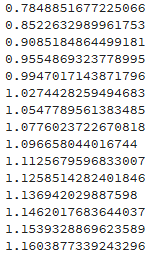
****

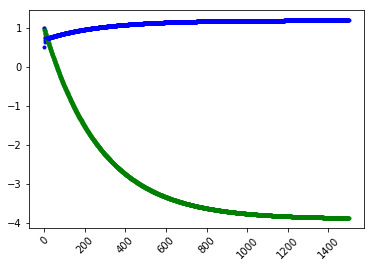
Representación gráfica de la evolución del coste a lo largo de las 1500 iteraciones:

**Evolución de los parámetros:**

Valor del coste cada 100 iteraciones :

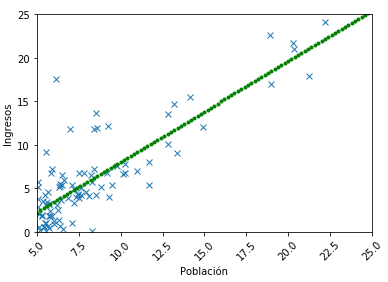
θ0 θ1



Representación gráfica de la evolución de los parámetros a lo largo de las 1500 iteraciones:

θ0

θ1

**GRAFICA FINAL**

Valores de hθ(x)

Datos de entrada

**Varias variables:**

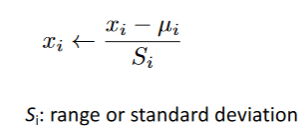
**Nuevos objetivos:**

En esta ocasión, se han modificado los algoritmos para funcionar con un número arbitrario de variables de entrada. En este caso concreto se utilizaran 2 variables de entrada.

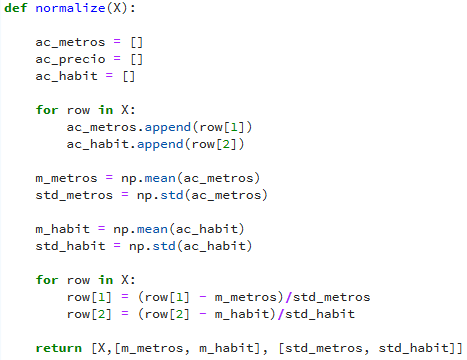
**Normalización de los datos:**

Para poder utilizar los datos de entrada, todos deben estar en la misma escala de magnitud. Mediante este proceso, todos los datos mantienen la proporción entre ellos, pero su valor se encontrará aproximadamente entre -1 y 1.

**ALGORITMO:**

****

**Código:**

****

Nota: esta función devuelve no sólo los datos de entrada normalizados, si no también los valores de la media y la desviación estándar. Esto es para poder normalizar nuevos datos de entrada, en caso de que queramos hacer predicciones.

**FORMULAS MODIFICADAS:**

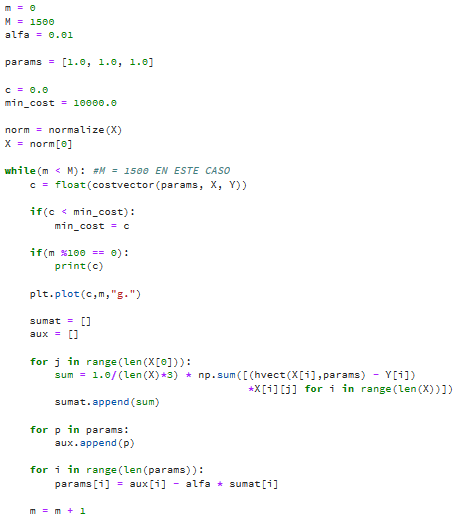
**Función hipótesis:**

**codigo hipotesis vectorial.png**

**descenso de gradiante vectorizado.pngFunción de coste:**

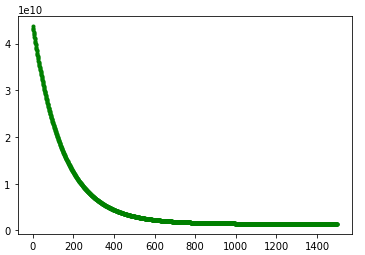
codigo funcion coste vector.png

**Actualización de parámetros:**

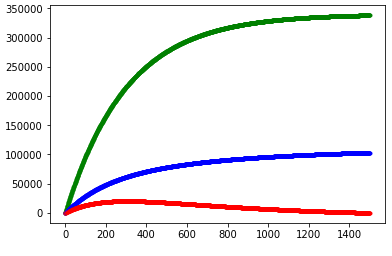
****

**RESULTADO DE EJECUCION:**

Evolución de los valores de la función de coste a lo largo de 1500 iteraciones:

****

Evolución de los tres parámetros:



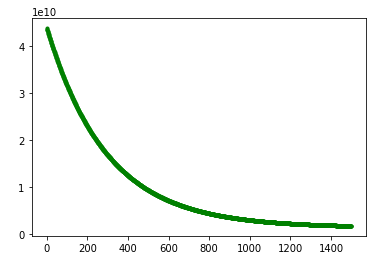
θ0

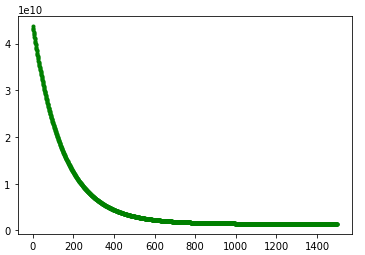
θ1

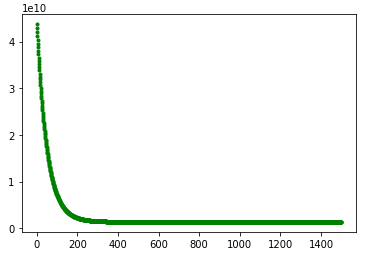
θ2

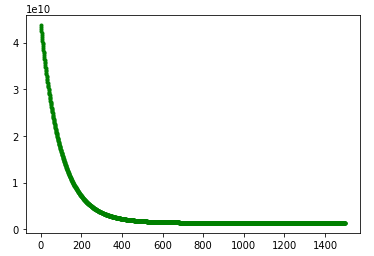
**VARIANDO LA TASA DE APRENDIZAJE:**

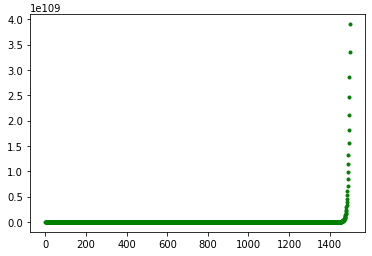
Comprobamos la evolución de la función para diferentes valores de α:

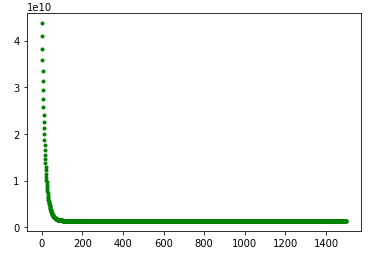
**α = 0.01: α = 0.005:**

****

**α = 0.015: α = 0.03:**

****

**α = 0.1: α = 4:**

****

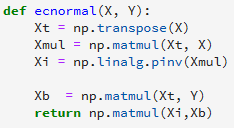
**METODO DE LA ECUACIÓN NORMAL:**

Para calcular los parámetros siguiendo el método de la ecuación normal, no hace falta normalizar los parámetros.

**Ecuación:**

**ecuacion normal.png**

**Implementación:**

****

**Diferencia en el resultado:**

Para un piso de 1650 m y 3 habitaciones, se predicen los siguientes precios

* Descenso del gradiante**:** 293365.776802749
* Ecuación normal: 293081.46433498873