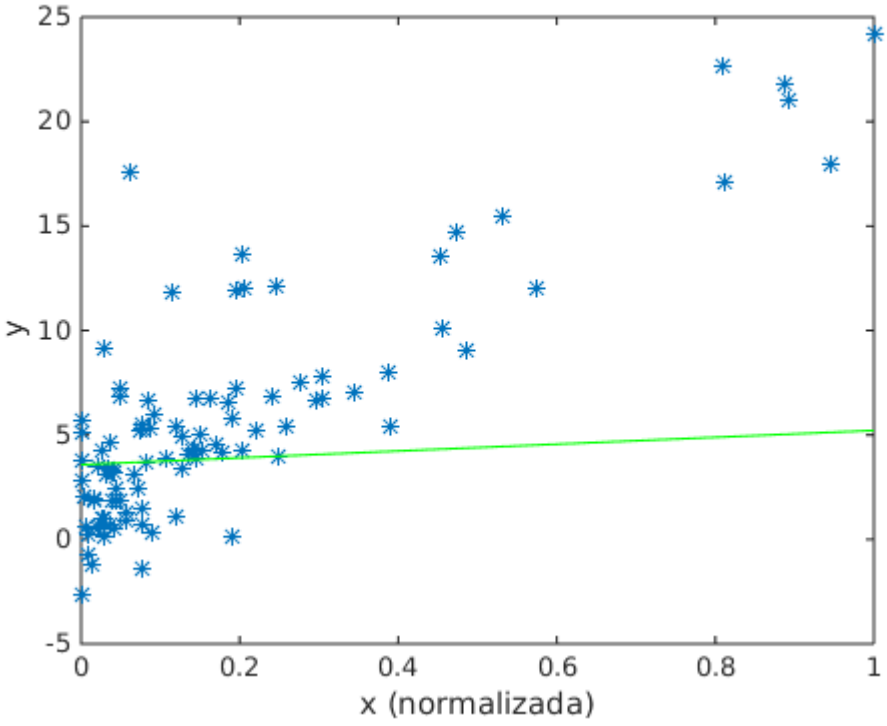
Primera Tarea:

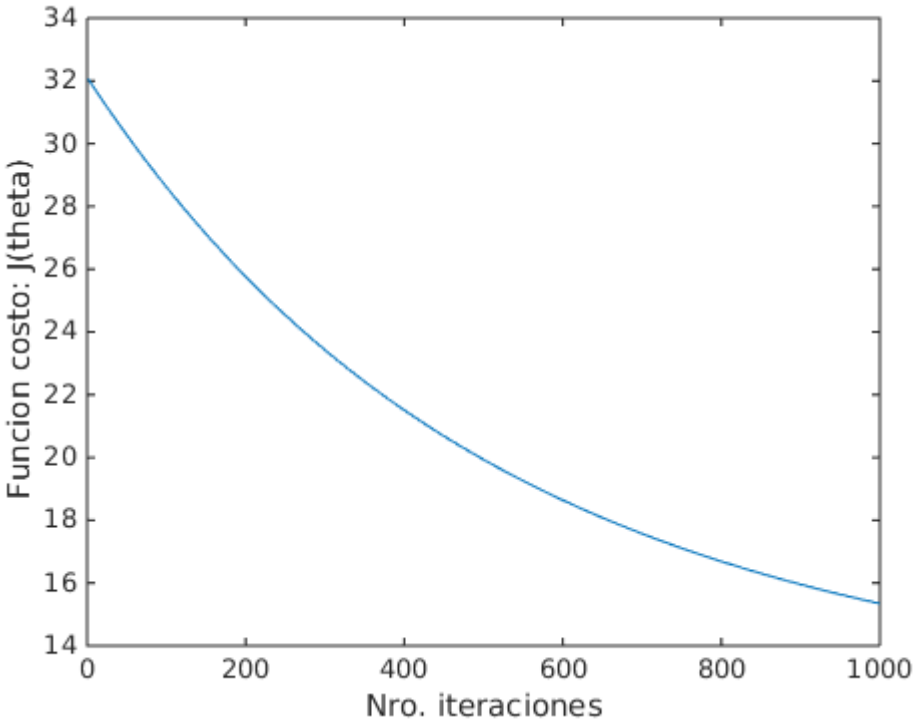
a) Crear una variable llamada 'historial_J' que almacene en cada iteración del algoritmo Gradient Descent el valor de la Función Costo.

Los archivos matlab modificados estan con el nombre: gradient_descent_one_var.m, linear_reg_script_one_var.m

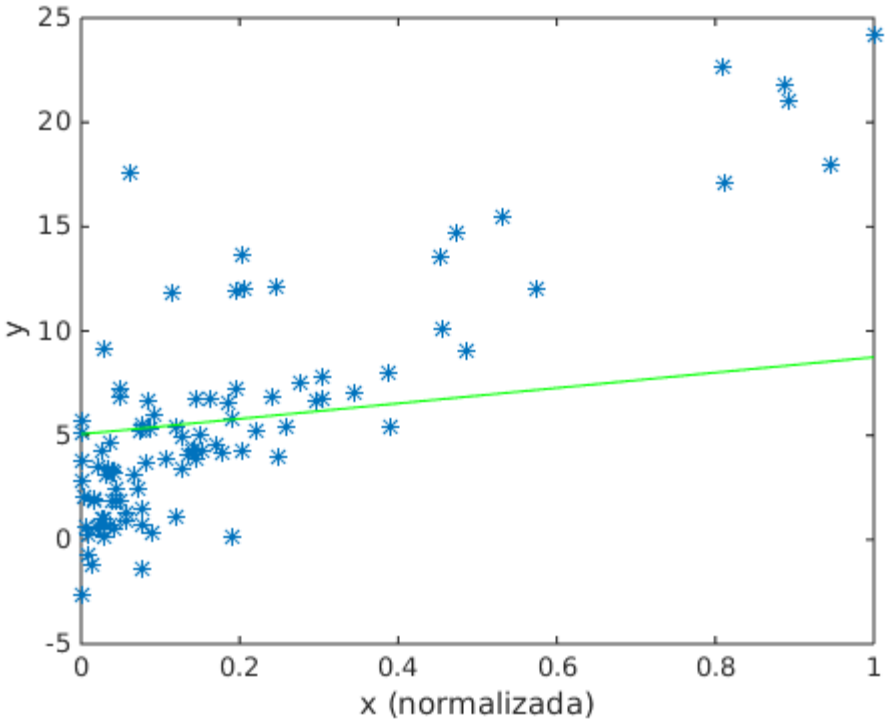
b) Para valores de alpha = 0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1, 1.3, 2, 3. Hacer un gráfico (iter, J) donde se plotee el valor de la Función Costo (J) en la i-ésima iteración (Iter). El máximo número de iteraciones será de 1000.

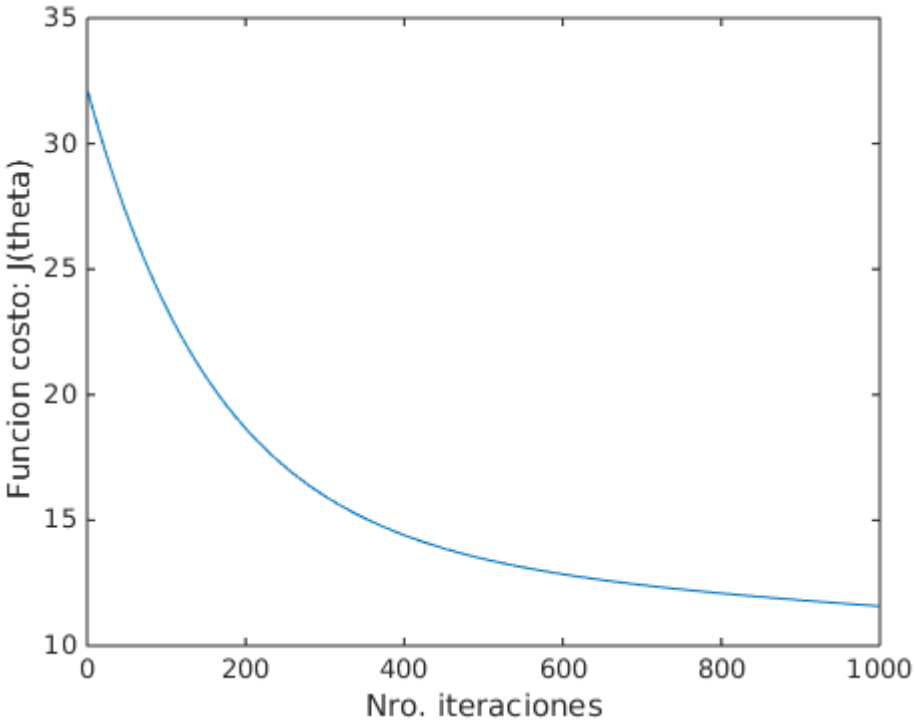
alpha=0.001



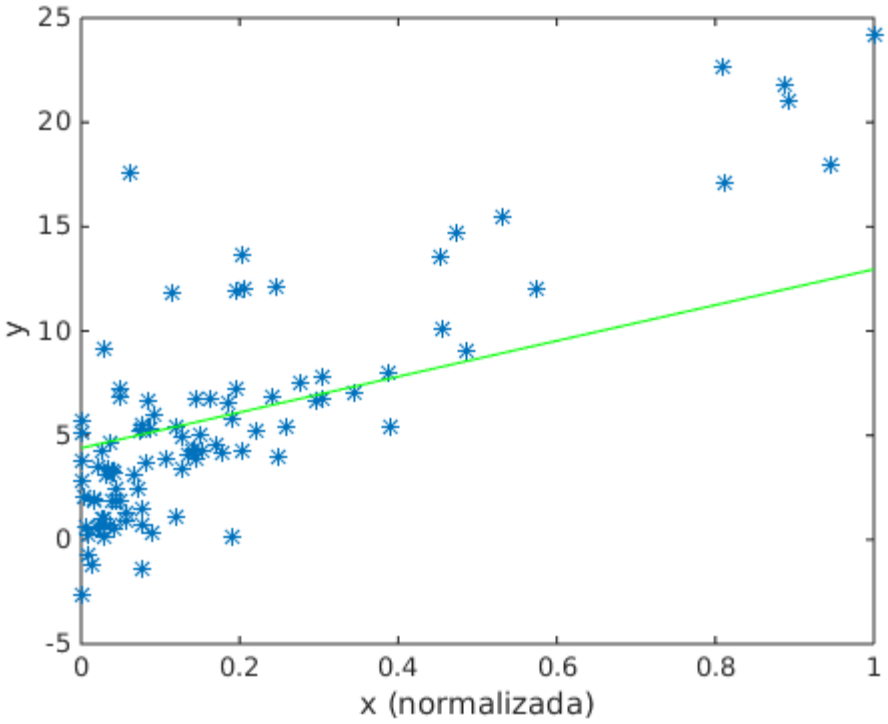

Funcion costo: J(theta)
Nro. iteraciones
34
32
30
28
26
24
22
20
18
16
14
0
200
400
600
800
1000

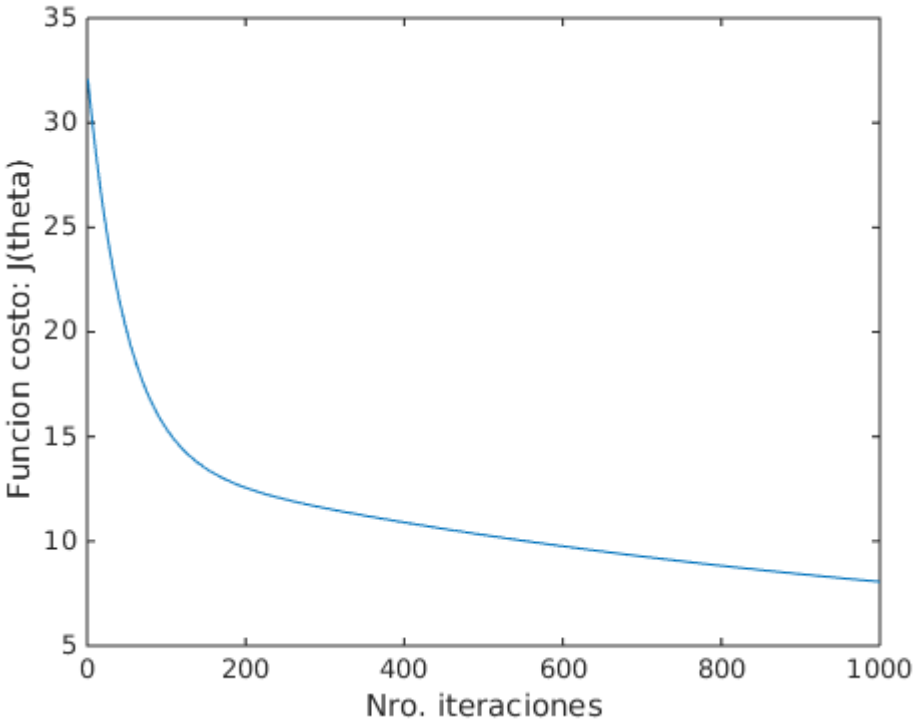
alpha=0.003


y
x (normalizada)
25
20
15
10
5
0
-5
0
0.2
0.4
0.6
0.8
1

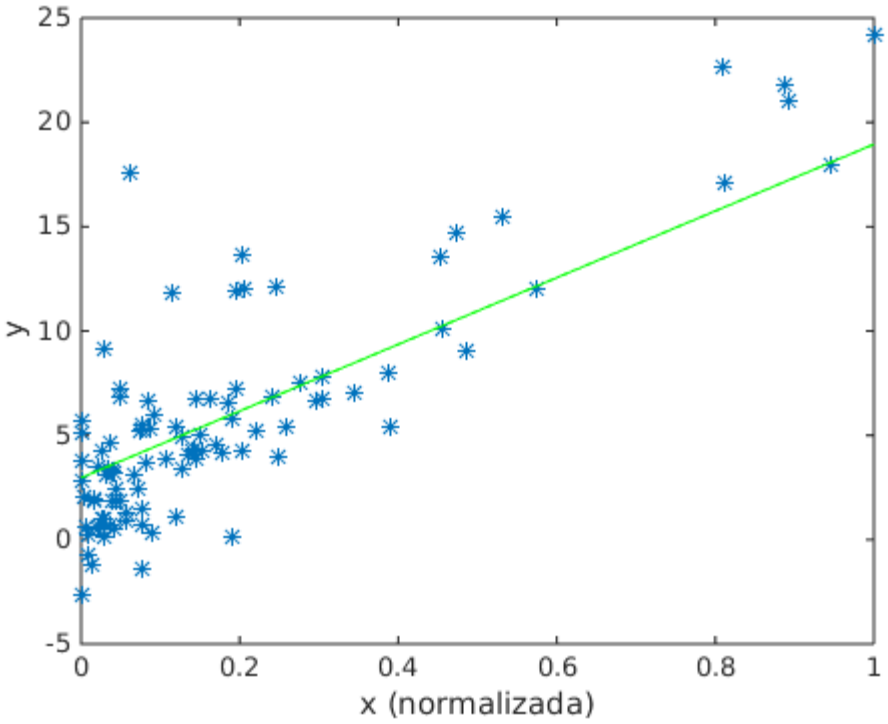

Funcion costo: J(theta)
Nro. iteraciones
35
30
25
20
15
10
0
200
400
600
800
1000

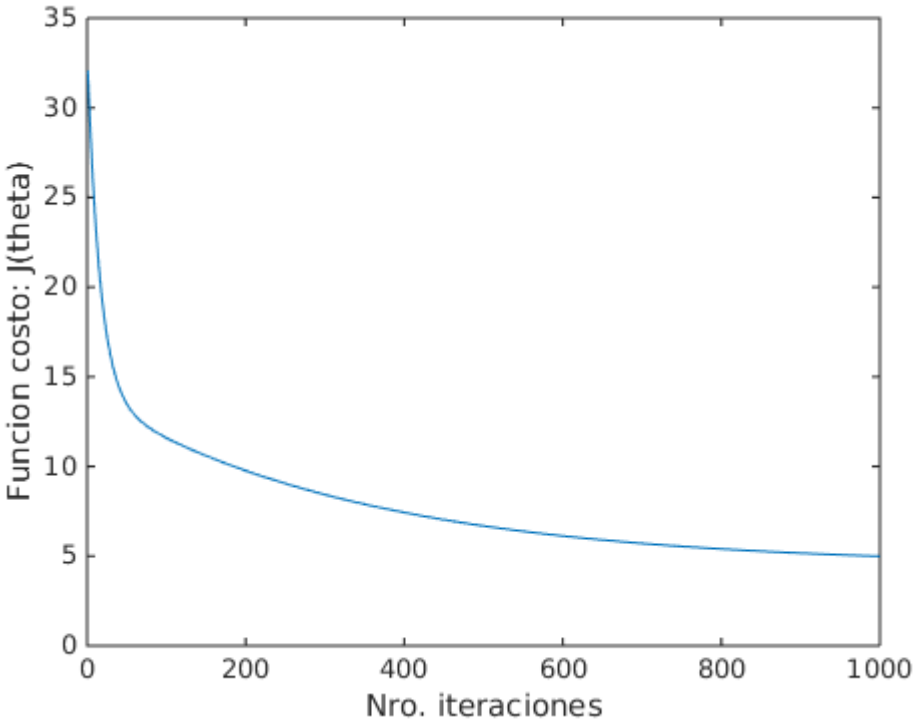
alpha = 0.01


y
x (normalizada)
25
20
15
10
5
0
-5
0
0.2
0.4
0.6
0.8
1

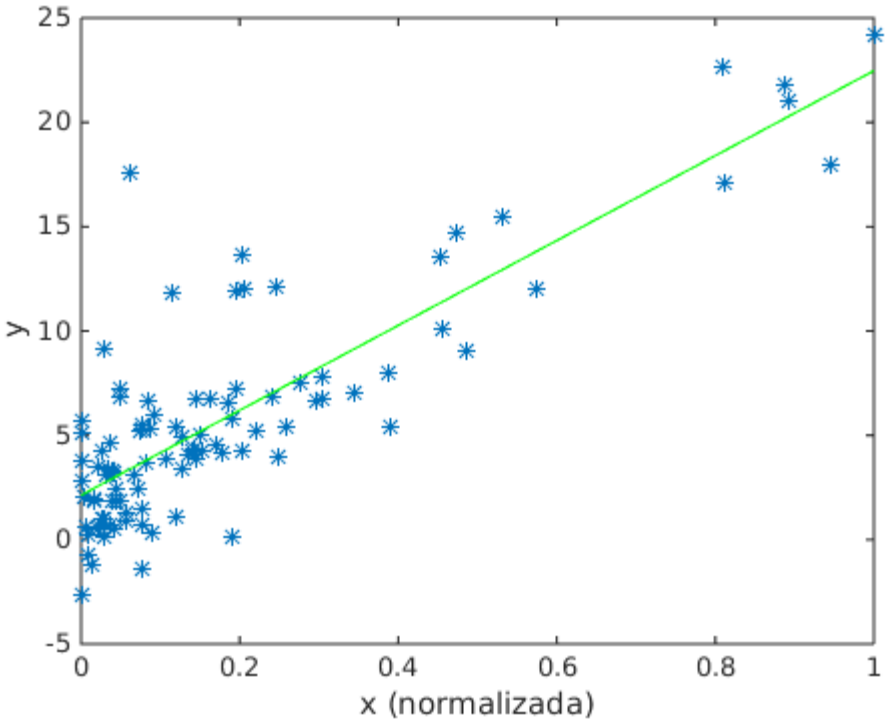

Funcion costo: J(theta)
Nro. iteraciones
35
30
25
20
15
10
5
0
200
400
600
800
1000

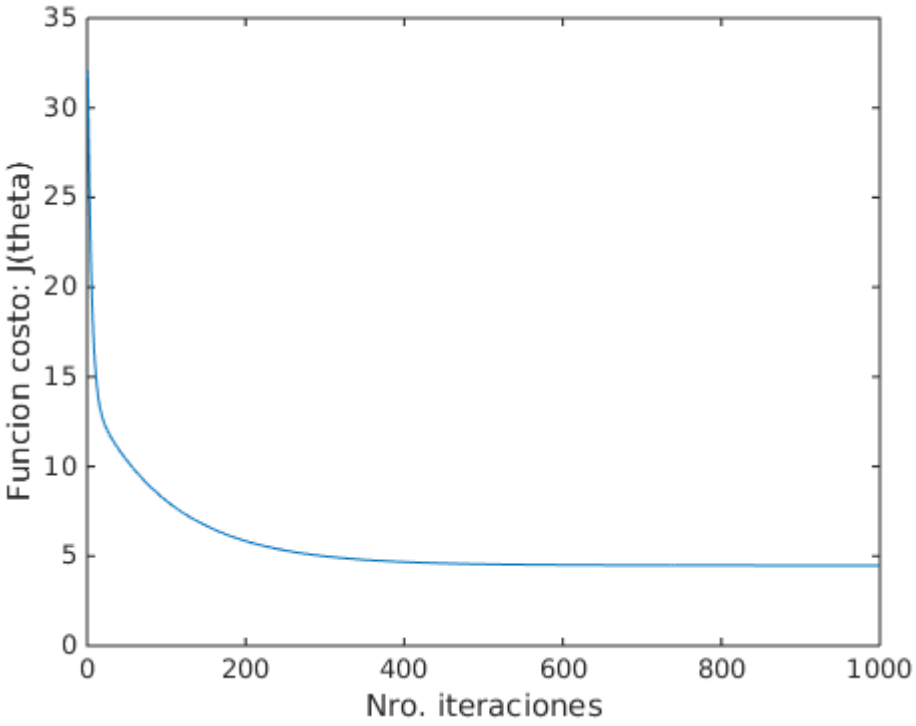
alpha = 0.03


y
x (normalizada)
25
20
15
10
5
0
-5
0
0.2
0.4
0.6
0.8
1

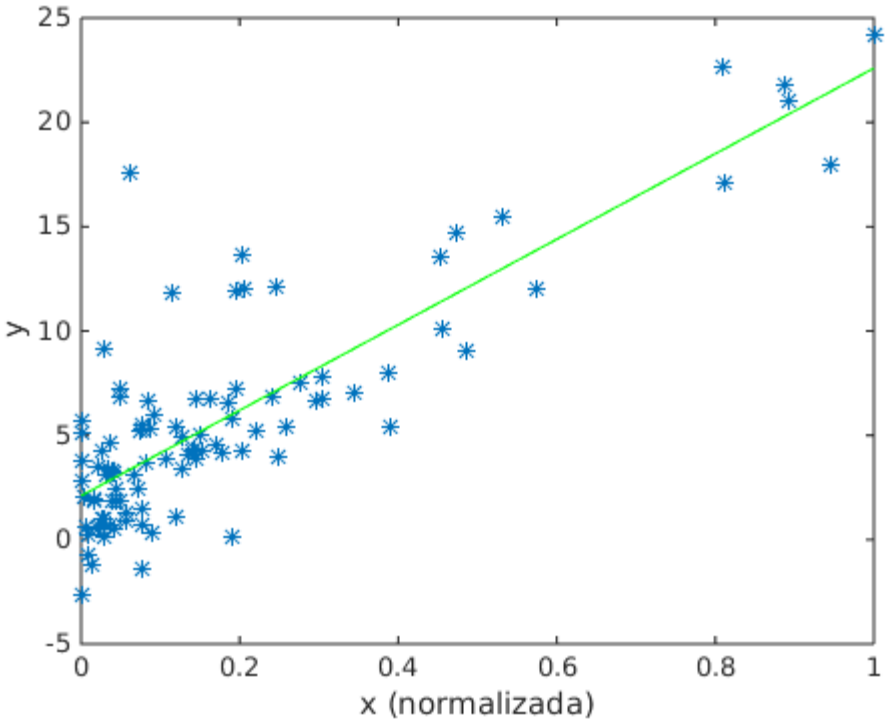

Funcion costo: J(theta)
Nro. iteraciones
0
5
10
15
20
25
30
35
0
200
400
600
800
1000

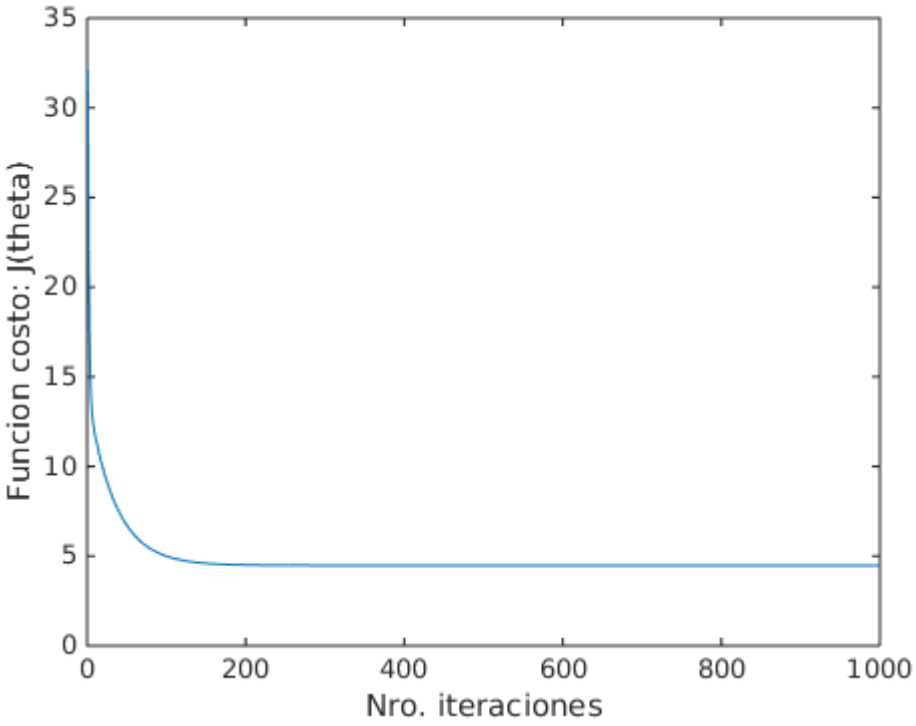
apha = 0.1


y
x (normalizada)
-5
0
5
10
15
20
25
0
0.2
0.4
0.6
0.8
1

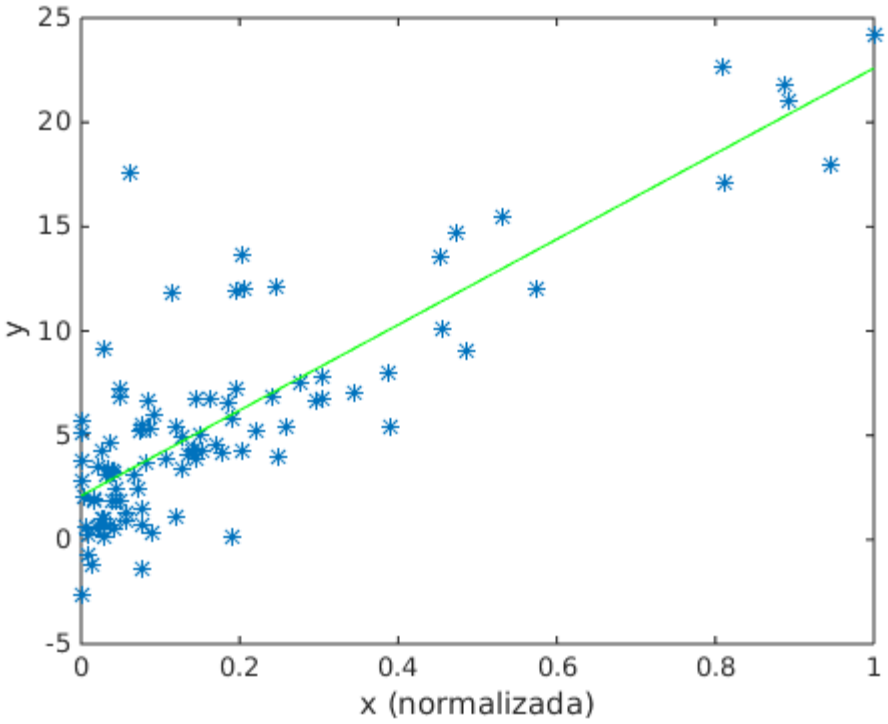

Funcion costo: J(theta)
Nro. iteraciones
0
5
10
15
20
25
30
35
0
200
400
600
800
1000

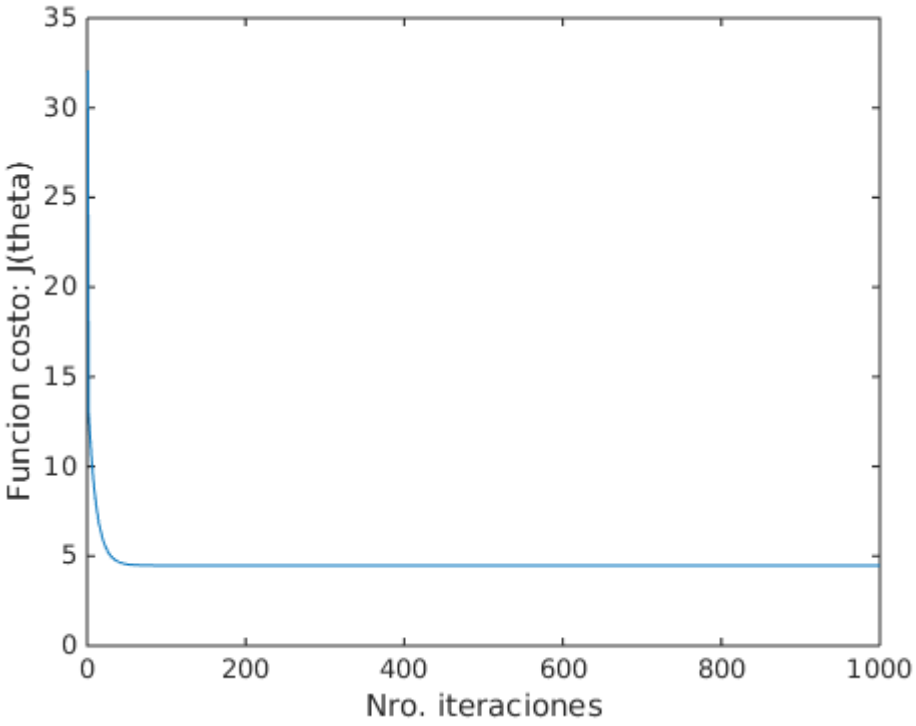
alpha = 0.3


y
x (normalizada)
-5
0
5
10
15
20
25
0
0.2
0.4
0.6
0.8
1

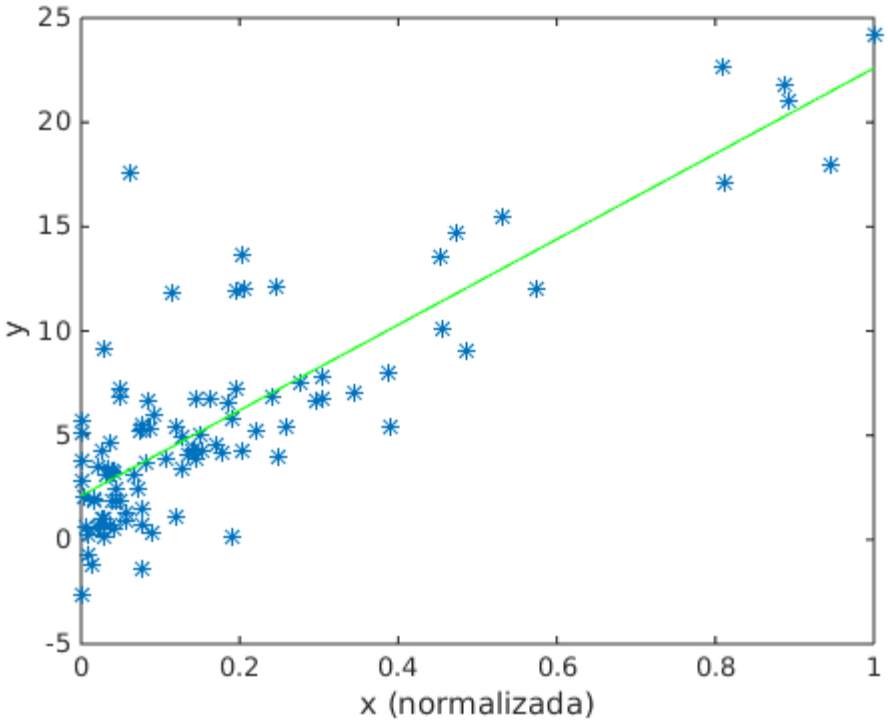

Funcion costo: J(theta)
Nro. iteraciones
0
5
10
15
20
25
30
35
0
200
400
600
800
1000

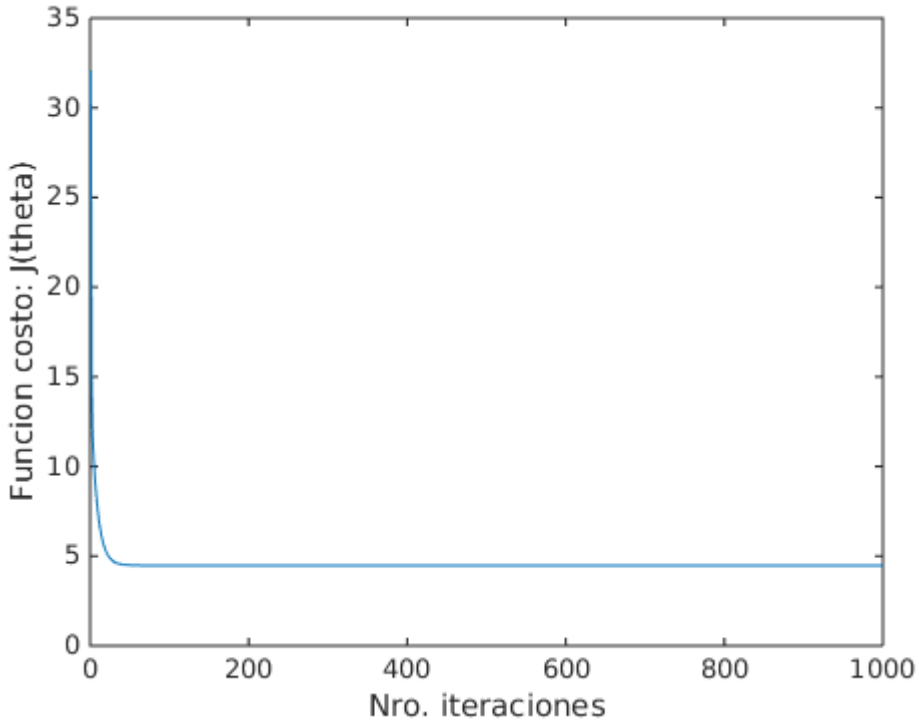
alpha = 1


y
x (normalizada)
-5
0
5
10
15
20
25
0
0.2
0.4
0.6
0.8
1

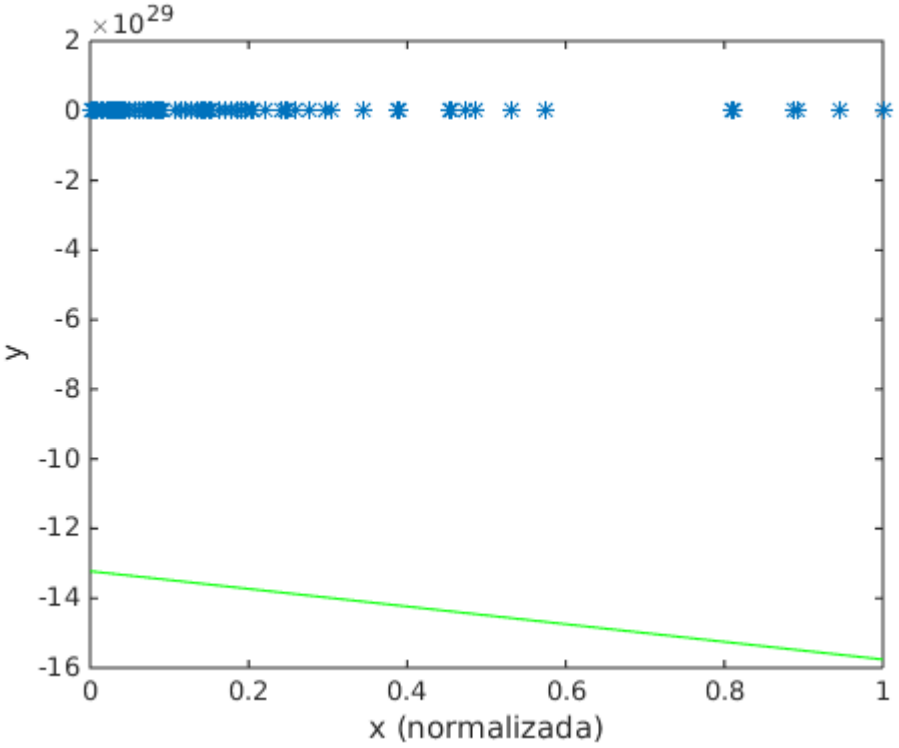

Funcion costo: J(theta)
Nro. iteraciones
0
5
10
15
20
25
30
35
0
200
400
600
800
1000

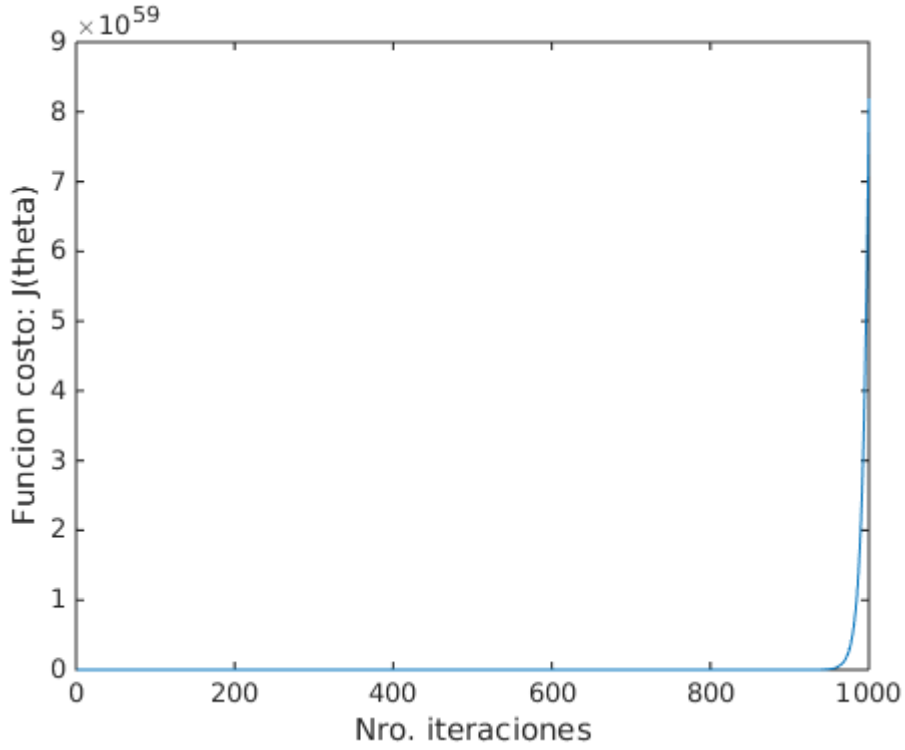
alpha = 1.3


y
x (normalizada)
-5
0
5
10
15
20
25
0
0.2
0.4
0.6
0.8
1

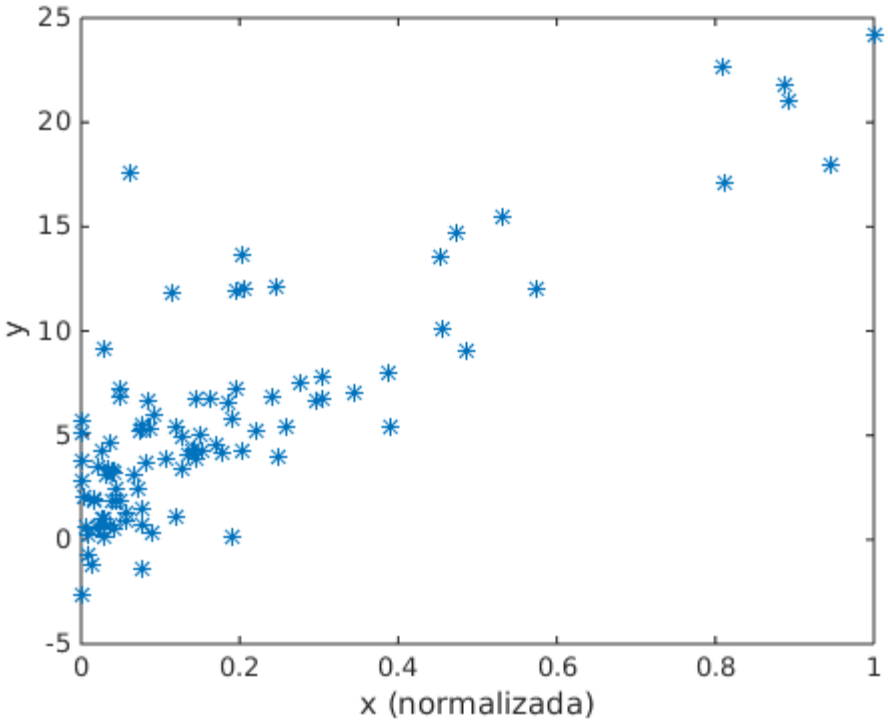

Funcion costo: J(theta)
Nro. iteraciones
0
5
10
15
20
25
30
35
0
200
400
600
800
1000

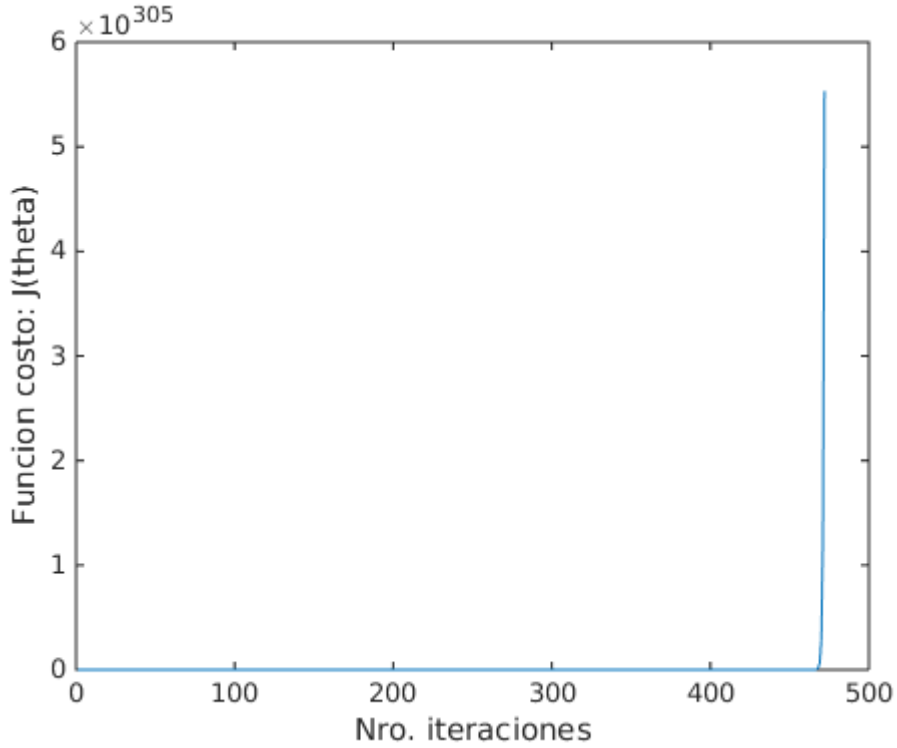
alpha = 2


×10^29
y
x (normalizada)
2
0
-2
-4
-6
-8
-10
-12
-14
-16
0
0.2
0.4
0.6
0.8
1


×10^59
9
8
7
6
5
4
3
2
1
0
Funcion costo: J(theta)
0
200
400
600
800
1000
Nro. iteraciones

alpha = 3


25
20
15
10
5
0
-5
y
0
0.2
0.4
0.6
0.8
1
x (normalizada)



Pregunta: Cuál valor de alpha cree usted que consigue converger más rápido?

Al parecer cuando alpha = 1, o que se acerque por arriba o por abajo, por ejemplo: alpha = 1.3 (se acerca por arriba)