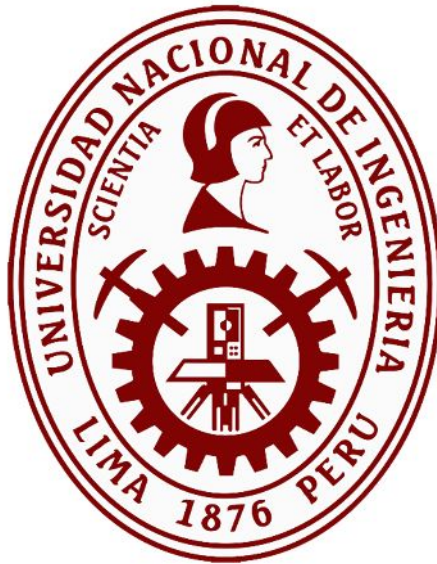


UNIVERSIDAD NACIONAL DE INGENIERIA



Practica Calificada 1

Regresión Lineal Cubica

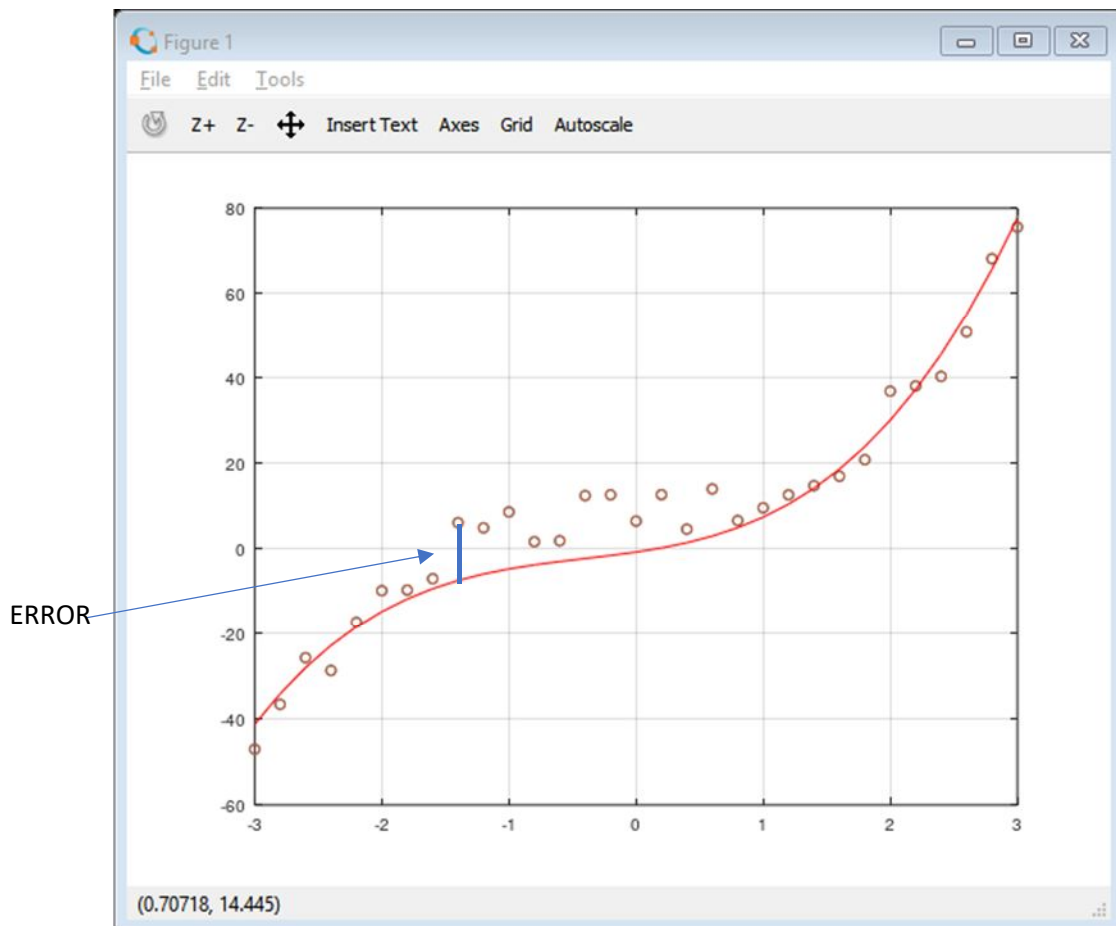
Lazaro Camasca Edson Nicks

Inteligencia Artificial

1. Regresión Iterativa

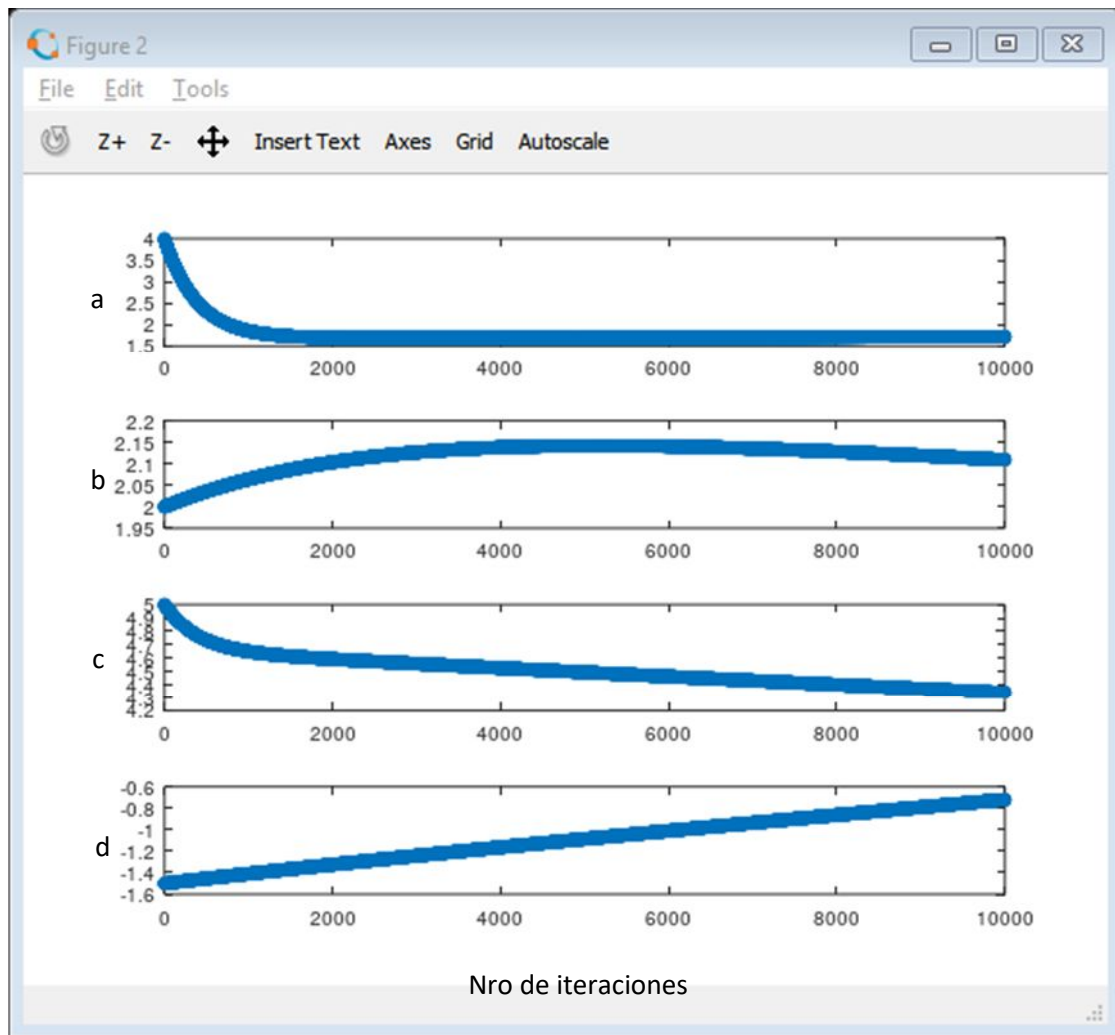
Al ejecutar el código, nos proporciona 3 figuras.

En la figura 1, se puede apreciar los puntos dispersores generados aleatoriamente, y una función cubica que ajuste a estos puntos, de tal manera que la distancia de error sea mínima.



En la figura 2, se puede ver como los coeficientes de la función cubica $4x^3+2x^2+5x-1.5$ en cada iteración se acercan a los coeficientes de $2x^3+1x^2+3x-1$

En el primer grafico se puede como 4 se acerca a 1.5 en 2000 iteraciones.

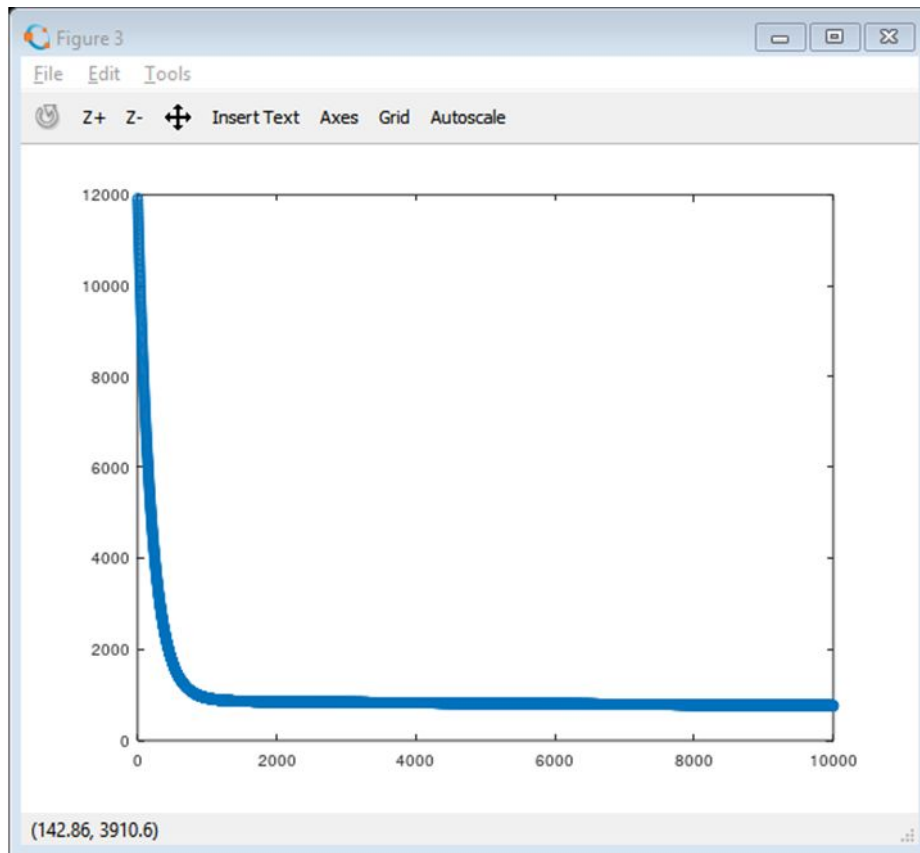


Al realizar todas las iteraciones necesarias sin sobrepasar el umbral, se obtuvo como el error máximo 1.34146, en las siguientes imágenes se puede apreciar, en la primera columna los valores exactos y en la segunda columna los resultados de la ejecución del programa.

```

>> |
ans =
    2.00000    1.71821
    1.00000    2.10916
    3.00000    4.34146
   -1.00000   -0.71954
  
```

En la siguiente figura se muestra como el Jacobiano tiende a su valor final en 2000 iteraciones, ya no era necesario realizar otras 8000 iteraciones.



El código será adjuntado con este reporte, **RegresionIterativaCubica.m**

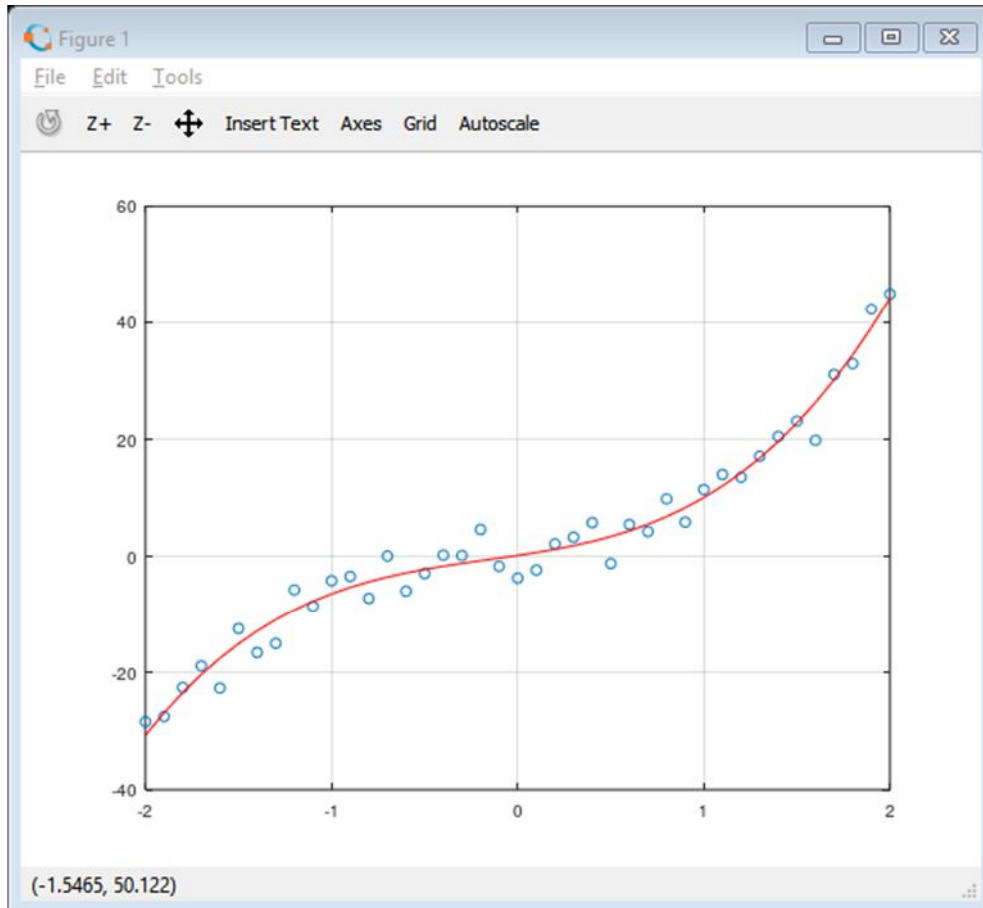
```

RegresionIterativaCubica.m
1  %-----
2  % Regresion Iterativa Cubica
3  % Lazaro Camasca Edson
4  % PC 1 - Inteligencia Artificial
5  %-----
6
7  clc
8  clear
9  close all;
10
11  an = 2;
12  bn = 1;
13  cn = 3;
14  dn = -1;
15  x_inf = -3; x_aum = 0.2; x_sup = 3;
16  x = x_inf:x_aum:x_sup; % Vector[inf:aum:sup]
17  x = x'; % Transpuesta de x
18  N = length(x);
19
20  error = 15*rand(N,1); %Cuando se aumenta el error, los datos son más disp
21  yb = an*x.^3 + bn*x.^2 + cn*x + dn + error;
22
23  figure(1);
24  plot(x, yb, 'o'); % Graficar la funcion y = f(x)
25  hold on;
26
27  %-----
28
29  % - 4 -

```

2. Regresión Exacta

En la figura 1 se puede apreciar los puntos generados aleatoriamente y la función que se ajusta a estos.



Los valores obtenidos por este método suelen tener mayor error que el iterativo.

```
Command Window
ans =

    4.00000    3.79704
    1.00000    0.59660
    3.00000    3.08883
   -4.00000    1.86177

>>
```

El código será adjuntado con este reporte, **RegresionExactaCubica.m**, mostramos una parte del código.

```
1 %-----
2 % Regresion Exacta Cubica
3 % Lazaro Camasca Edson
4 % PC 1 - Inteligencia Artificial
5 %-----
6
7 clc
8 clear
9 close all;
10
11 an = 4;
12 bn = 1;
13 cn = 3;
14 dn = -4;
15 x_inf = -2; x_aum = 0.1; x_sup = 2;
16 x = x_inf:x_aum:x_sup; % Vector[inf:aum:sup]
17 x = x'; % Transpuesta de x
18 N = length(x);
19
20 error = 10*rand(N,1); %Cuando se aumenta el error, los
21 yb = an*x.^3 + bn*x.^2 + cn*x + dn + error;
22
23 figure(1);
24 plot(x,yb,'o')
25
26 %--- Fila 1---
27 A11 = sum(x.^6);
28 A12 = sum(x.^5);
29 A13 = sum(x.^4);
30 A14 = sum(x.^3);
31
32 %--- Fila 2---
33 A21 = sum(x.^5);
```