***UNIVERSIDAD NACIONAL DE INGENIERIA***

***FACULTAD DE CIENCIAS***

***INTELIGENCIA ARTIFICIAL***



**INFORME 1**

**ALUMNO:**

DANIEL POZO G.

20141451C

**PROFESOR:**

ANTONIO MORAN

En este informe les mostraremos los resultados con el método directo para las regresiones cúbica y exponencial.

1. *Con el método directo:*
   1. **REGRESIÓN CUBICA**

clc

clear

%Regresion cubica

clear

a = 3;

b = 2;

c = -1;

d = 4;

x = 1:0.8:10;

x = x';

nx = length(x);

yb = a\*x.^3 + b\*x.^2 + c.\*x + d + 1.5\*rand(nx,1);

figure(1);

plot(x,yb,'bo');

a = 3;

b = -2;

c = 1;

d = 2;

niter = 1000;

Jold = 10^10;

sigma = 0.0000002;

for k = 1 : niter

aa(k,1) = a;

bb(k,1) = b;

cc(k,1) = c;

dd(k,1) = d;

Jnew = sum(a\*x.^3 + b\*x.^2 + c.\*x + d - yb).^2/nx;

dJ = sqrt(abs(Jnew - Jold)/Jold);

if(dJ < 0.01)

break;

end

dJda=(a\*sum(x.^6) + b\*sum(x.^5) + c\*sum(x.^4) + d\*sum(x.^3) - sum(x.^3.\*yb))/nx;

dJdb=(a\*sum(x.^5) + b\*sum(x.^4) + c\*sum(x.^3) + d\*sum(x.^2) - sum(x.^2.\*yb))/nx;

dJdc=(a\*sum(x.^4) + b\*sum(x.^3) + c\*sum(x.^2) + d\*sum(x) - sum(x.^2.\*yb))/nx;

dJdd=(a\*sum(x.^3) + b\*sum(x.^2) + c\*sum(x) + d\*nx - sum(yb))/nx;

%Metodo por descenso por gradiente:

a = a - sigma\*dJda;

b = b - sigma\*dJdb;

c = c - sigma\*dJdc;

d = d - sigma\*dJdd;

Jold = Jnew;

end

hold on

a

b

c

d

y = a\*x.^3 + b.\*x.^2+c.\*x + c;

plot(x,y);

figure(2);

subplot(4,1,1);

plot(1:length(aa),aa)

subplot(4,1,2);

plot(1:length(bb),bb);

subplot(4,1,3);

plot(1:length(cc),cc);

subplot(4,1,4);

plot(1:length(dd),dd);

**RESULTADOS**

Con los valores iniciales de las variables:

a = 3;

b = 2;

c = -1;

d = 4;

Tenemos los siguientes resultados:

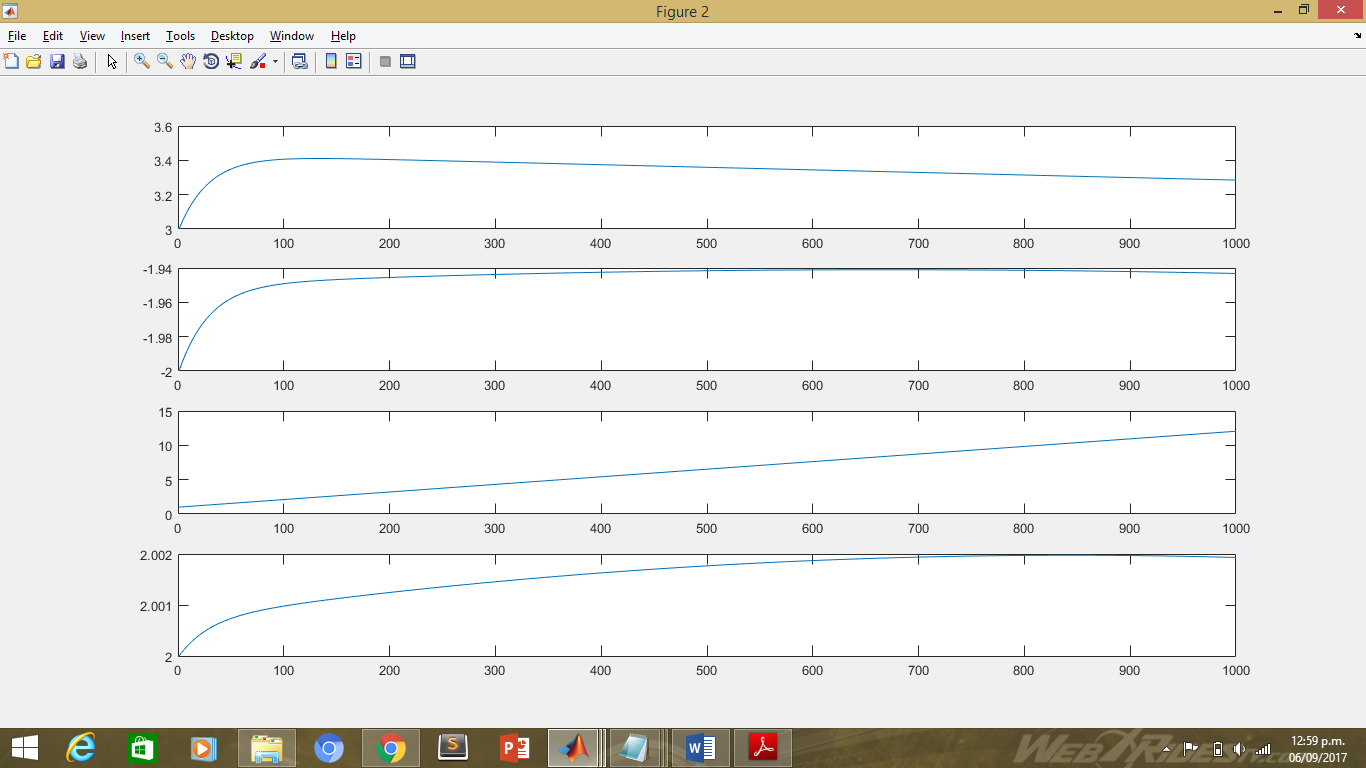
a = 3.2849

b = -1.9434

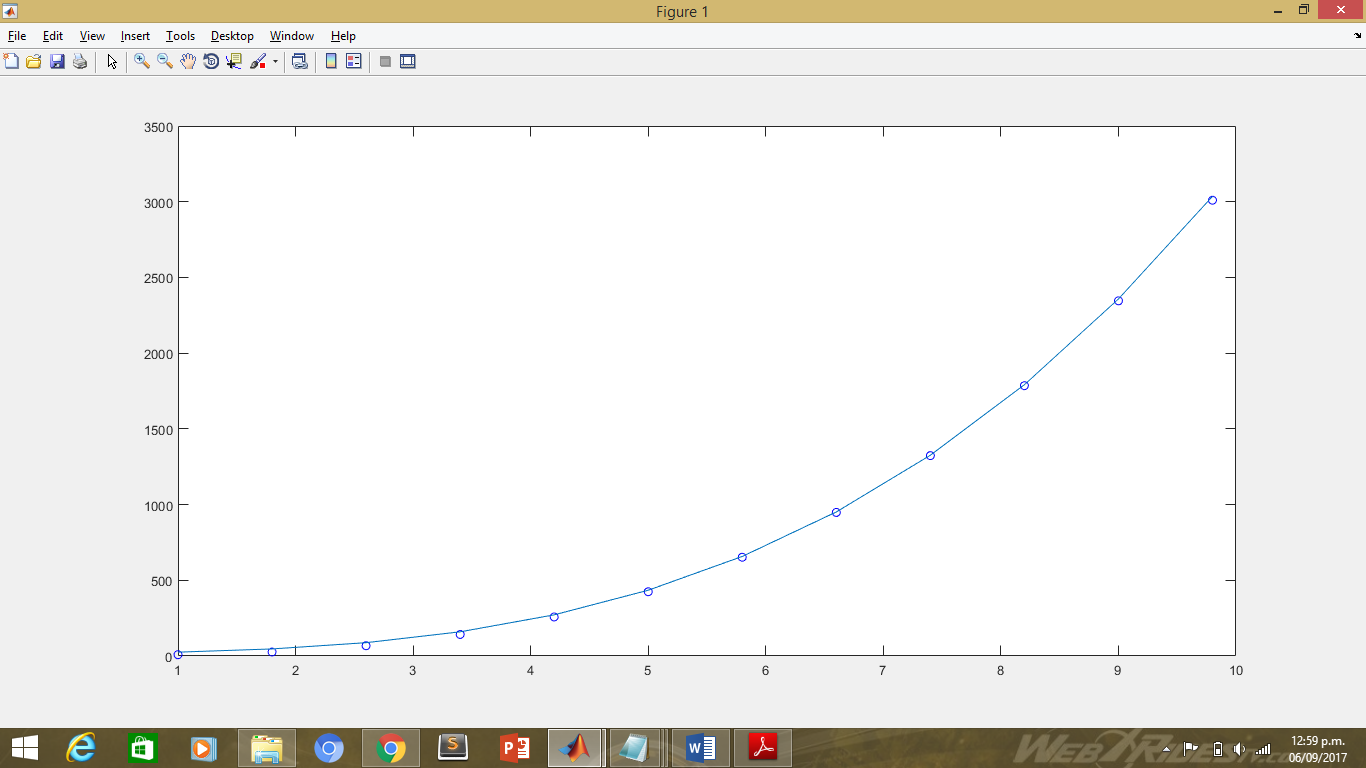
c = 12.0634

d = 2.0020

**GRÁFICAS**



**Figura 1**



**Figura 2**

* 1. **REGRESIÓN EXPONENCIAL**

clc

clear

%Regresion exponencial

clear

a = 1;

b = 2;

x = -6:0.2:6;

x = x';

nx = length(x);

y = b.\*exp(x.\*a) + 1.5\*rand(nx,1);

lny = x.\*a + rand(nx,1) + log(b) + 1.5\*rand(nx,1);

figure(1)

plot(x,y,'bo')

aa = [sum(x.^2) sum(x)

sum(x) nx];

bb = [sum(x.\*lny)

sum(lny)];

cc = inv(aa)\*bb;

cc

a = cc(1,1);

b = cc(2,1);

hold on

y = b.\*exp(x.\*cc(1,1));

plot(x,y)

**RESULTADOS**

Con los valores iniciales de las siguientes variables:

a = 1

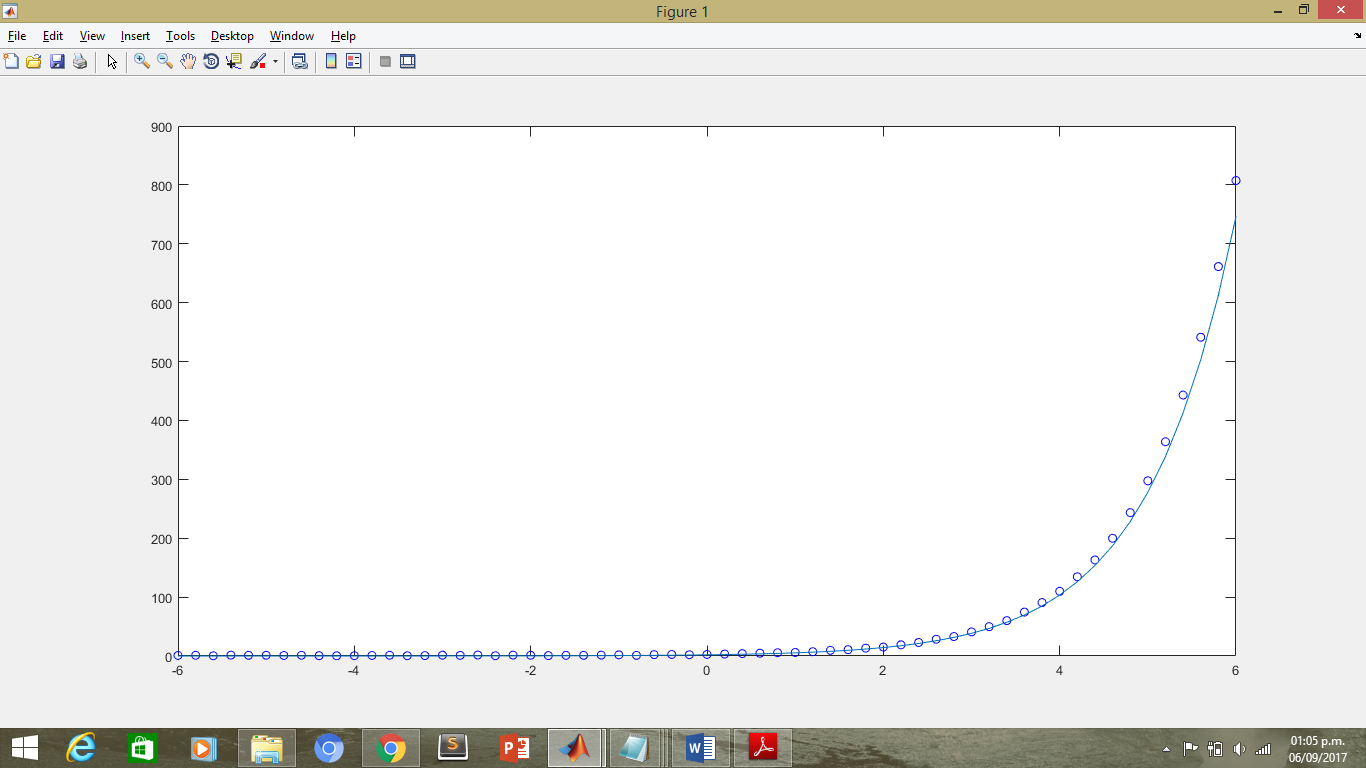
b = 2

Obtenemos los siguientes resultados:

cc = 0.9826

1.9525

**GRÁFICA**



**Figura 3**