Semana 11

Andres Julián Alaix

Regresión no lineal

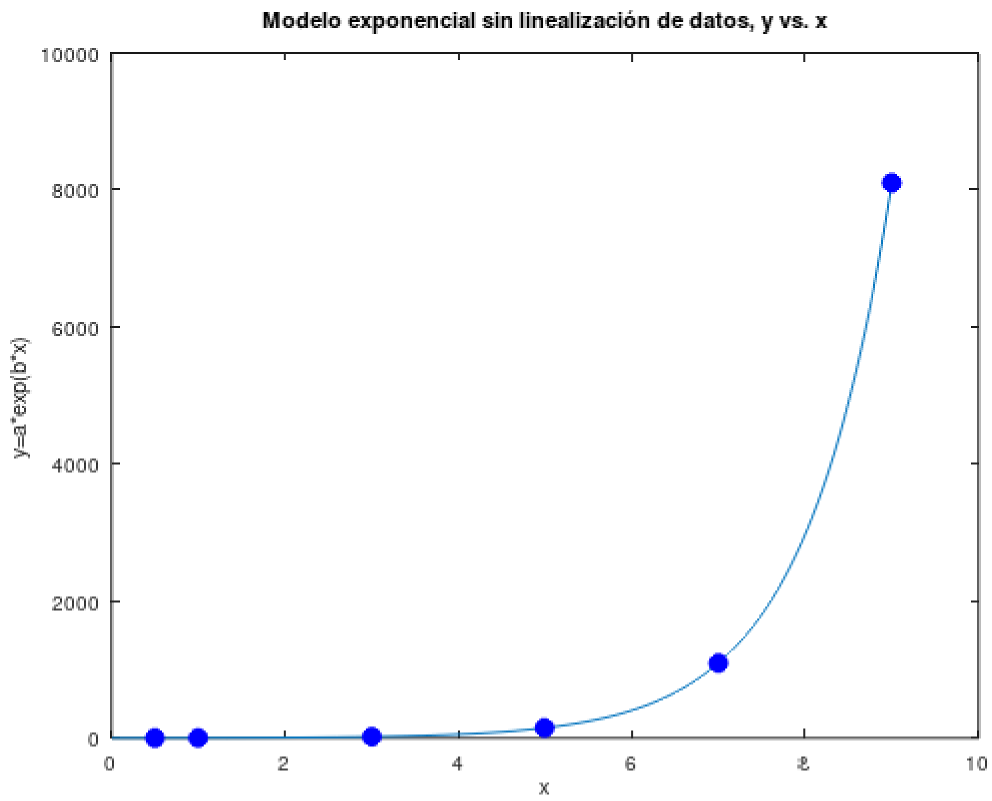
```
function varargout = mtl_reg_sim_nonlinearwithout()
% Se definen los arreglos de entrada X y Y. También se da un v
X=[0.5,1,3,5,7,9];
Y=[1.648,2.71,20.08,148.4,1096,8103];
initial_value_b=0.1;
% Fin de los datos de entrada
clc

disp(sprintf('\n\nNOTA: Este código muestra el uso de Octave 
disp(sprintf('el procedimiento para hacer la regresión de un 
disp(sprintf('sin linealización de los datos, es decir de la 
disp(sprintf('\n\n******************************************
disp(sprintf('Usando la suma de los residuales cuadrados y mi
disp(sprintf('a (a) y (b) obtenemos dos ecuaciones no lineale
disp(sprintf('Afortunadamente, las dos ecuaciones no lineales
disp(sprintf('una ecuación no lineal en (b). Una vez solucion
disp(sprintf('\nLos arreglos de entrada X y Y son:'))
X
Y

disp(sprintf('\n\n************** Exponencial ************** E
disp(sprintf('Datos (x1,y1),(x2,y2),...,(xn,yn), mejor ajuste

n=length(X);
z=zeros(1,n);

% Hallamos el valor de (b) primero.
b_value=fzero(@(b)(f(b,X,Y,n)),initial_value_b);
```

```
% Llamar fzero con una función anónima de un argumento que ca
% Calculamos el valor de (a)
sum_yex=0;
sum_e2x=0;
for i=1:n
    sum_yex=sum_yex+Y(i)*exp(b_value*X(i));
    sum_e2x=sum_e2x+exp(2*b_value*X(i));
end
a_value=sum_yex/sum_e2x;
% Presentamos los valores de (a) y (b)
a_value
b_value
disp(sprintf('\nEl modelo sin la linealización de datos se de

disp(sprintf('b=%5g',b_value))
disp(sprintf('\b*x)                (4)'))



% Dibujamos la figura de x*exp(b*x) y los puntos de datos.
xp=(0:0.001:max(X));
yp=zeros(1,length(xp));

for i=1:length(xp)
    yp(i)=a_value*exp(b_value*xp(i));
end
plot(xp,yp)
title('Modelo exponencial sin linealización de datos, y vs. x
xlabel('x')
ylabel('y=a*exp(b*x)')
hold on
plot(X,Y,'bo','MarkerFaceColor','b')
hold off
end

% Usando fzero se halla la raíz de la ecuación
function eqn1 = f(b,X,Y,n)
sum_yxex=0;
sum_yex=0;
```

```
sum_ex2=0;
sum_xex2=0;
for i=1:n
    sum_yxex=sum_yxex+Y(i)*X(i)*exp(b*X(i));
    sum_yex=sum_yex+Y(i)*exp(b*X(i));
    sum_xex2=sum_xex2+X(i)*exp(2*b*X(i));
    sum_ex2=sum_ex2+exp(2*b*X(i));
end
eqn1 = sum_yxex-sum_yex/sum_ex2*sum_xex2;

end
```

```
******************************************************* Introduction ****************************************************
*
Usando la suma de los residuales cuadrados y minimizando con respecto
a (a) y (b) obtenemos dos ecuaciones no lineales simultáneas.
Afortunadamente, las dos ecuaciones no lineales simultáneas pueden ser reducidas a
una ecuación no lineal en (b). Una vez solucionada, (a) puede ser hallada directamente.

Los arreglos de entrada X y Y son:
X =

   0.5000   1.0000   3.0000   5.0000   7.0000   9.0000

Y =

   1.6480e+00   2.7100e+00   2.0080e+01   1.4840e+02   1.0960e+03   8.1030e+03



************** Exponencial ************** Exponencial *************** Exponencial *********************
Datos (x1,y1),(x2,y2),...,(xn,yn), mejor ajuste de y=a*exp(b*x) para los datos.
```

```
a_value = 0.9976
b_value = 1.0003

El modelo sin la linealización de datos se describe como
              y=0.997609
b=1.00026
*x)               (4)
```



Conclusiones

- El código ajusta un conjunto de datos a un modelo exponencial de la forma

$$= a * exp(b * x)$$

- Se hace uso del método `fzero`, lo que hace que se encuentre el paramento `b` mas optimo, para calcular `a`
- El ajuste exponencial es importante para validar la efectividad del metodo