clc

p=[2 -3 2 -1];

roots(p)

ans =

1.0000

0.2500 + 0.6614i

0.2500 - 0.6614i

x=-3:0.01:3;

y=polyval(p,x);

plot(x,y)

grid on

d= 3 + 4i

d =

3.0000 + 4.0000i

real(d)

ans =

3

imag(d)

ans =

4

r=roots(p)

r =

1.0000

0.2500 + 0.6614i

0.2500 - 0.6614i

r(imag(r)==0) %solo las raíces reales

ans =

1.0000

%grafica el polinomio y su derivada

p1=polyder(p)

p1 =

6 -6 2

y1=polyval(p1,x);

plot(x,y,x,y1)

plot(x,y,'r',x,y1,'k')

q=[1 0 -10]; %Hallar las intersecciones del pol. p y q

p

p =

2 -3 2 -1

roots(p-q)

{??? Error using ==> minus

Matrix dimensions must agree.

}

q=[0 q]

q =

0 1 0 -10

roots(p-q)

ans =

1.5301 + 1.3796i

1.5301 - 1.3796i

-1.0602

%En general:

lp=length(p);

lq=length(q);

q=[zeros(1,lp-lq) q];

roots(p-q)

ans =

1.5301 + 1.3796i

1.5301 - 1.3796i

-1.0602

%Hallar el punto mínimo del polinomio w

w=[1 -2 3];

yw=polyval(w,x);

plot(x,yw)

w=[1 -2 3];

w1=polyder(w);

w2=polyder(w1);

pc=roots(w1);

pminimos=pc(polyval(w2,pc)>0) %criterio de segunda derivada

pminimos =

1

p

p =

2 -3 2 -1

w

w =

1 -2 3

t=conv(p,w) %producto de polinomios

t =

2 -7 14 -14 8 -3

k=deconv(t,p) %division de polinomios

k =

1 -2 3

q=deconv([1 -1 0 1],[1 0 -1])

q =

1 -1

[q r]=deconv([1 -1 0 1],[1 0 -1])

q =

1 -1

r =

0 0 1 0

%Generar un polinomio con las raíces dadas en r:

r=[ 1 3 2]

r =

1 3 2

p=poly(r)

p =

1 -6 11 -6

roots(p)

ans =

3.0000

2.0000

1.0000

%Ajuste de curvas:

x=[1 3 4 4];

y=[1 2 3 4];

plot(x,y,'\*')

axis([0 5 0 5]) %Escala de los ejes

p=polyfit(x,y,1) %Obtiene una recta

p =

0.8333 0.0000

x0=0:0.01:5;

y0=polyval(p,x0);

plot(x,y,'\*',x0,y0)

q=polyfit(x,y,2); %Obtiene un polinomio de grado 2 que se ajusta a x,y

q =

0.3333 -0.8333 1.5000

y1=polyval(q,x0);

plot(x,y,'\*',x0,y0,x0,y1)