24-CAVAN-raices-reales

October 13, 2017

El objeto de este ejercicio es mostrar como en ciertos problemas la posibilidad de visualizar lo resultados puede ayudar enormemente a resolverlos. Primero utilizamos gráficas simples de funciones, *plot*, que podemos dinamizar con *animate*. En la última parte construimos un gráfico más complejo, usando *matrix_plot*, que contiene la información que realmente nos interesa.

```
In [1]: var('a b')
```

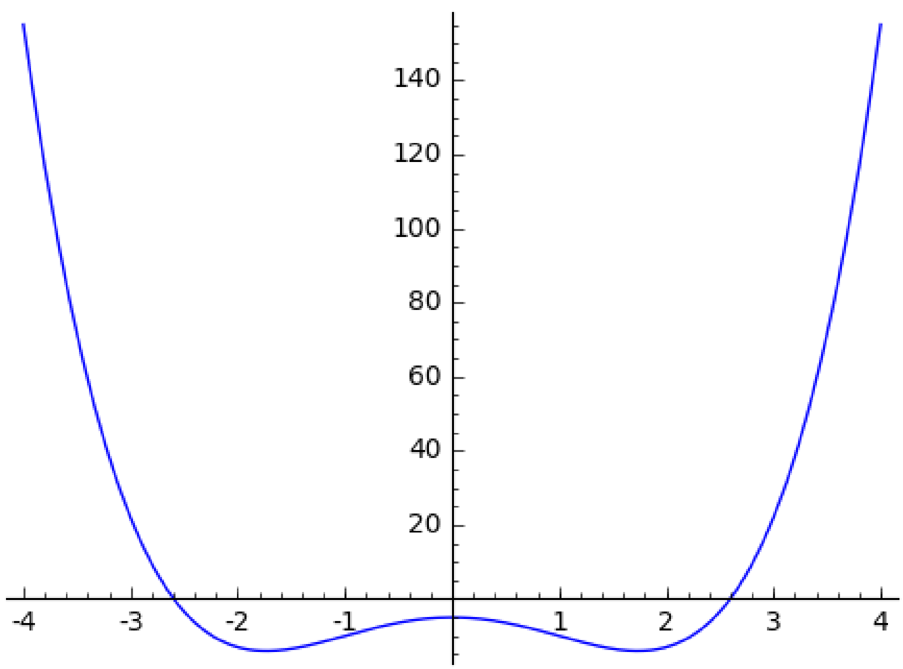
```
Out[1]: (a, b)
```

```
In [2]: p(x,a,b)=x^4-6*x^2+a*x+b
```

Caso $a = 0$ y b variando.

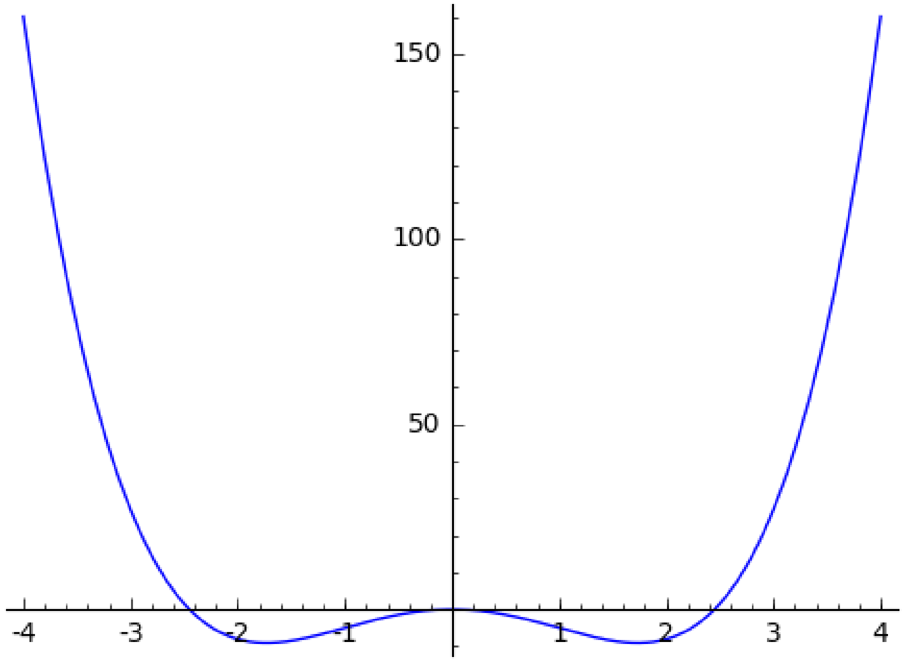
```
In [3]: plot(p(a=0,b=-5),-4,4,figsize=5)
```

```
Out[3]:
```



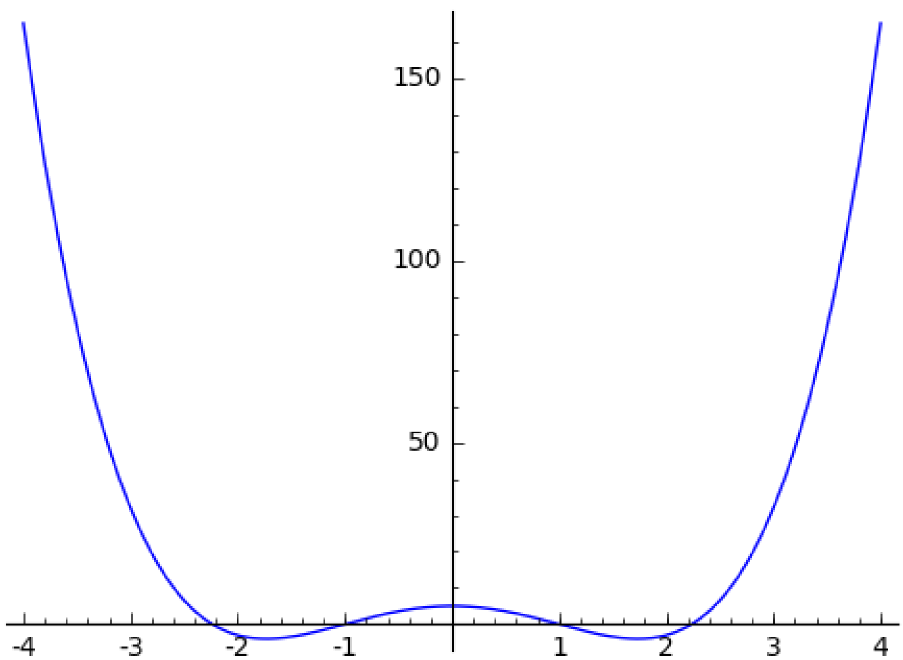
In [4]:
```
plot(p(a=0,b=0),-4,4,figsize=5)
```

Out[4]:



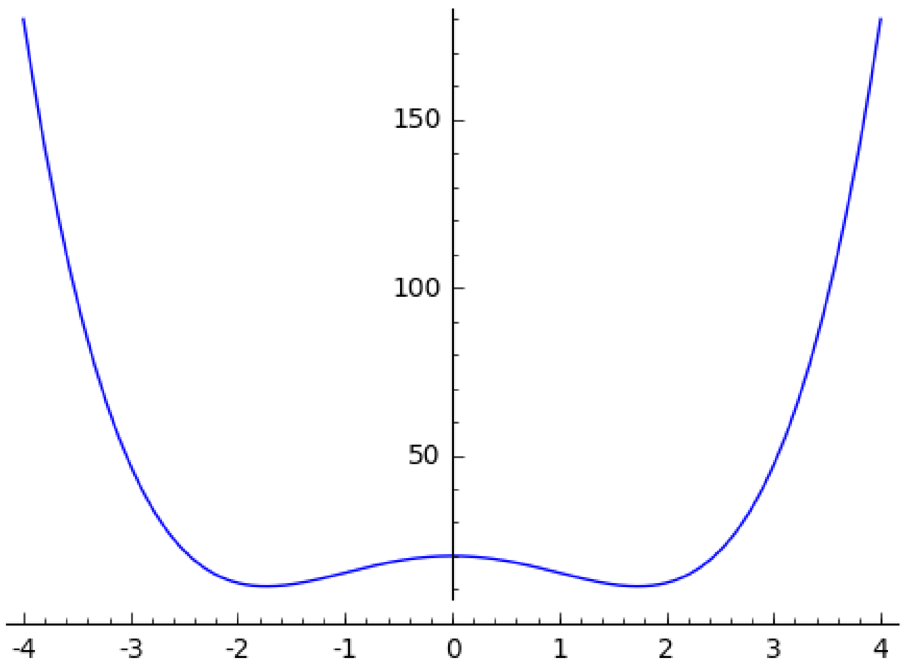
In [5]:
```
plot(p(a=0,b=5),-4,4,figsize=5)
```

Out[5]:



```
In [6]: plot(p(a=0,b=20),-4,4,figsize=5)
```

Out[6]:



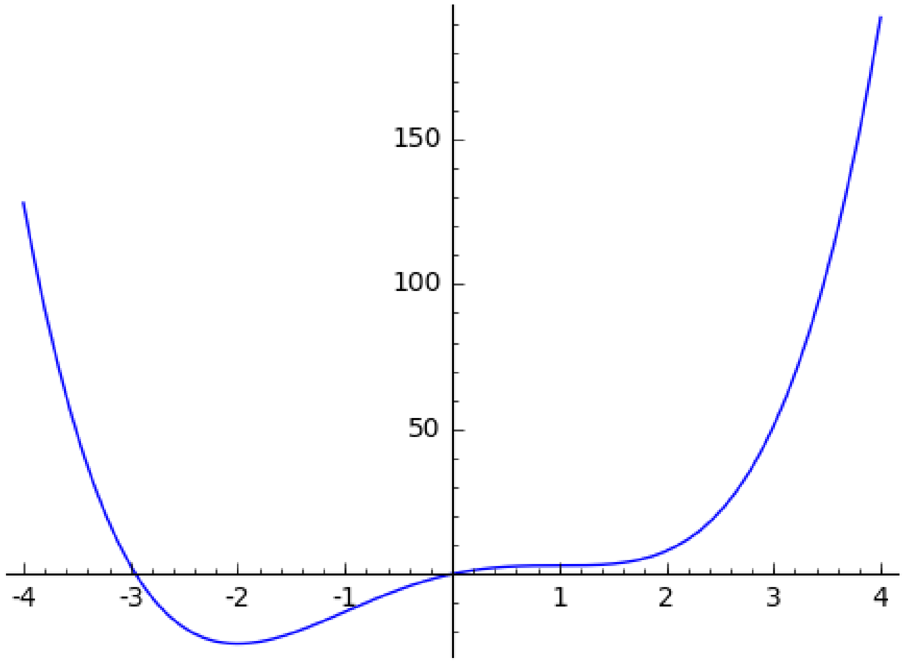

```
In [7]: A0 = [plot(p(a=0,b=k),(-4,4),ymin=-60,ymax=120,figsize=5) for k in srange(-40,40)]
        A = animate(A0)
        A.save('raices.gif')
```

Los gráficos animados permiten ver un poco mejor lo que está pasando, en este caso, bastante simple, el desplazamiento de la gráfica hacia arriba cuando b crece. Puedes ver una serie de ejemplos de animaciones en esta página.

Caso $a = 8$ y b variando

```
In [8]: plot(p(a=8,b=0),-4,4,figsize=5)
```

Out[8]:



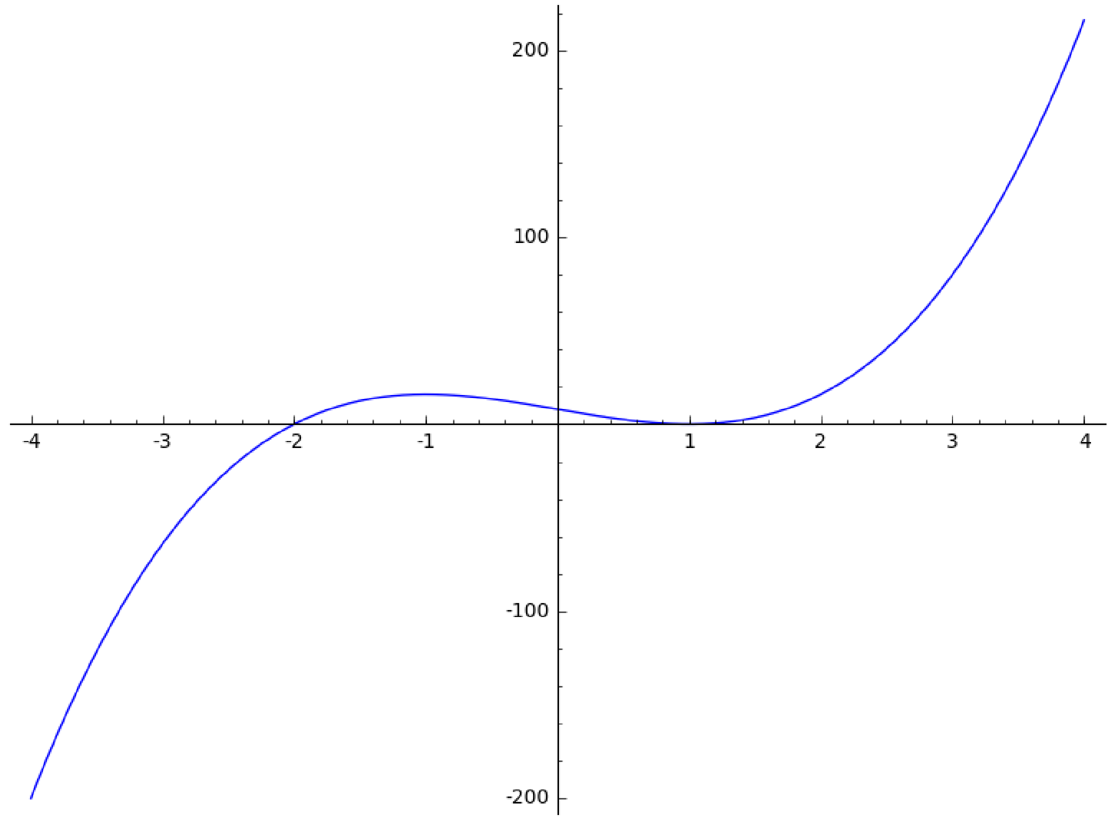
Cuando damos valores positivos a b la gráfica sube y cuando le damos valores negativos baja, pero, una vez fijamos a, siempre conserva la misma forma.

```
In [9]: q = diff(p(a=8),x);q
```

```
Out[9]: 4*x^3 - 12*x + 8
```

```
In [10]: plot(q,-4,4)
```

```
Out[10]:
```




```
In [11]: solve(q,x)

Out[11]: [x == -2, x == 1]

In [12]: factor(q)

Out[12]: 4*(x + 2)*(x - 1)^2
```

Es claro que la gráfica de q tiene un mínimo en $x = -2$ y en $x = 1$ se anulan la primera y la segunda derivadas y no es ni mínimo ni máximo. La gráfica es casi horizontal cerca de $x = 1$, pero aún así la función parece crecer. Para verlo calculamos 20 valores de la función separados por 0.1 unidades:

```
In [13]: [p(x=0.1*k,a=8,b=0) for k in srange(1,21)]

Out[13]: [0.740100000000000,
          1.36160000000000,
          1.86810000000000,
          2.26560000000000,
          2.56250000000000,
          2.76960000000000,
          2.90010000000000,
```

```
2.96960000000000,
2.99610000000000,
3.00000000000000,
3.00410000000000,
3.03360000000000,
3.11610000000000,
3.28160000000000,
3.56250000000000,
3.99360000000000,
4.61210000000000,
5.45760000000000,
6.57210000000000,
8.00000000000000]
```

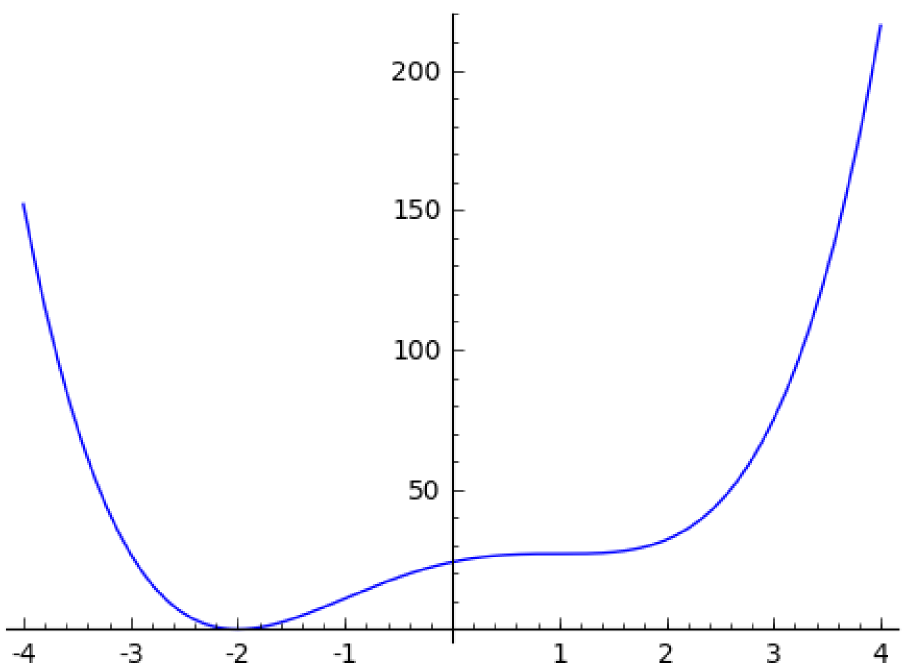
Calculamos el valor de b a partir del cual ya no hay soluciones reales de $p(x, 8, b)$. Debemos subir la gráfica tanto como el valor de la función en el mínimo:

```
In [14]: p(x=-2,a=8,b=0)
```

```
Out[14]: -24
```

```
In [15]: plot(p(a=8,b=24),-4,4,figsize=5)
```

```
Out[15]:
```



En particular, habría que demostrar que:

La función $p(x,8,0)$, que vale 0 en $x = 0$, es creciente para $x \geq -2$ y por tanto no tiene ninguna raíz real con $x \geq 0$.

La función $p(x,8,0)$ tiene exactamente dos raíces reales, una en $x = 0$ y la otra en un x menor que -2.

Finalmente habría que demostrar que para $b \geq 24$ no hay raíces reales y para $b \leq 24$ hay dos raíces reales, en el caso $b = 24$ una raíz real doble en el punto $x = -2$.

Caso general

Queremos ahora estudiar la función

$$rr(a,b) := \#\{\text{raices reales de } p(x,a,b) = 0 \text{ contadas con su multiplicidad}\}.$$

Se trata de una función $rr(a,b) : \mathbb{R}^2 \to \{0,2,4\}$, que divide el plano real en regiones en las que el número de raíces reales es constante. Se trata de obtener, usando Sage, una idea lo más precisa posible del mapa que produce la función $rr(a,b)$. Por suerte, Sage dispone de una función que permite contar el número de raíces reales fácilmente. En este momento, no necesitamos saber cómo lo hace, pero si te interesa puedes consultar esta página de la Wikipedia. y esta otra.

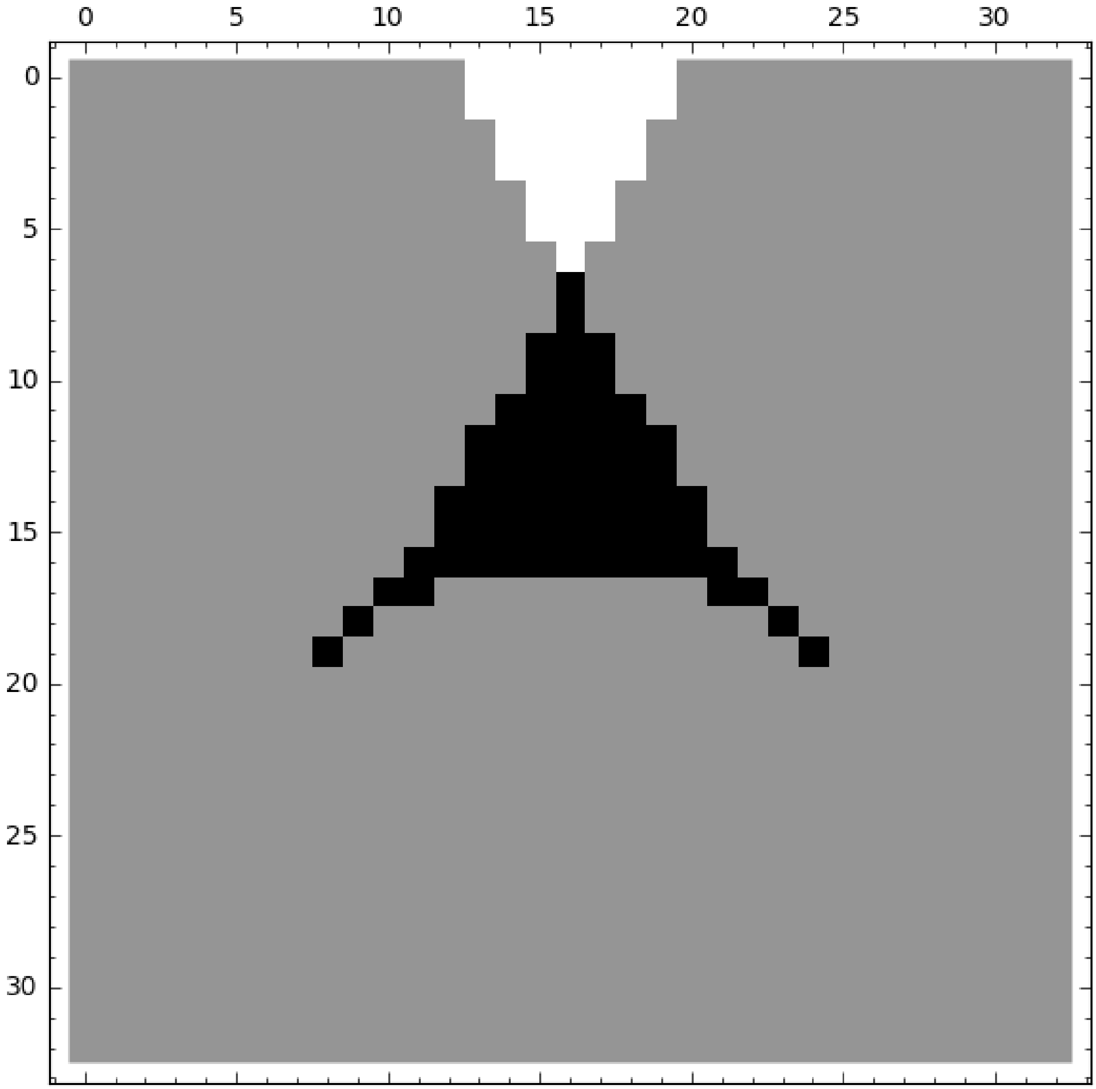
```
In [16]: from sage.rings.polynomial.real_roots import *

In [17]: R.<x>=QQ[] #Definimos el anillo de polinomios en una variable

In [18]: P1 = R(p(a=0,b=0));P1

Out[18]: x^4 - 6*x^2

In [19]: real_roots(P1)

Out[19]: [((-21/8, -9/4), 1), ((-47/256, 81/512), 2), ((9/4, 21/8), 1)]

In [20]: def numero_reales(P):
             return sum([item[1] for item in real_roots(P)])

In [21]: numero_reales(P1)

Out[21]: 4

In [22]: def generar_matriz(n):
             M = matrix(ZZ,2*n+1,2*n+1,(2*n+1)**2*[0])
             for A in srange(-n,n+1):
                 for B in srange(-n,n+1):
                     P = R(p(a=A,b=B))
                     M[-B+n,n+A]=numero_reales(P)
             return M
```

Hemos definido una función de Sage que produce una matriz que en el lugar ij tiene el número de raíces reales de $p(x,a,b)$ con $i = B - n$ y $j = A + n$. De esta forma se consigue que el lugar de la matriz correspondiente a $A = -n, B = n$ sea el 00.

Como la matriz tiene 3 posibles valores para sus entradas, $0,2,4$, *matrix_plot* utiliza tres tonos de gris: negro para el 0, gris para el 2 y blanco para el 4. La línea $a = 0$ es la vertical que divide al cuadrado en dos partes iguales, y, como sabemos pasa de 2 raíces reales para b negativo a cuatro y luego a cero.

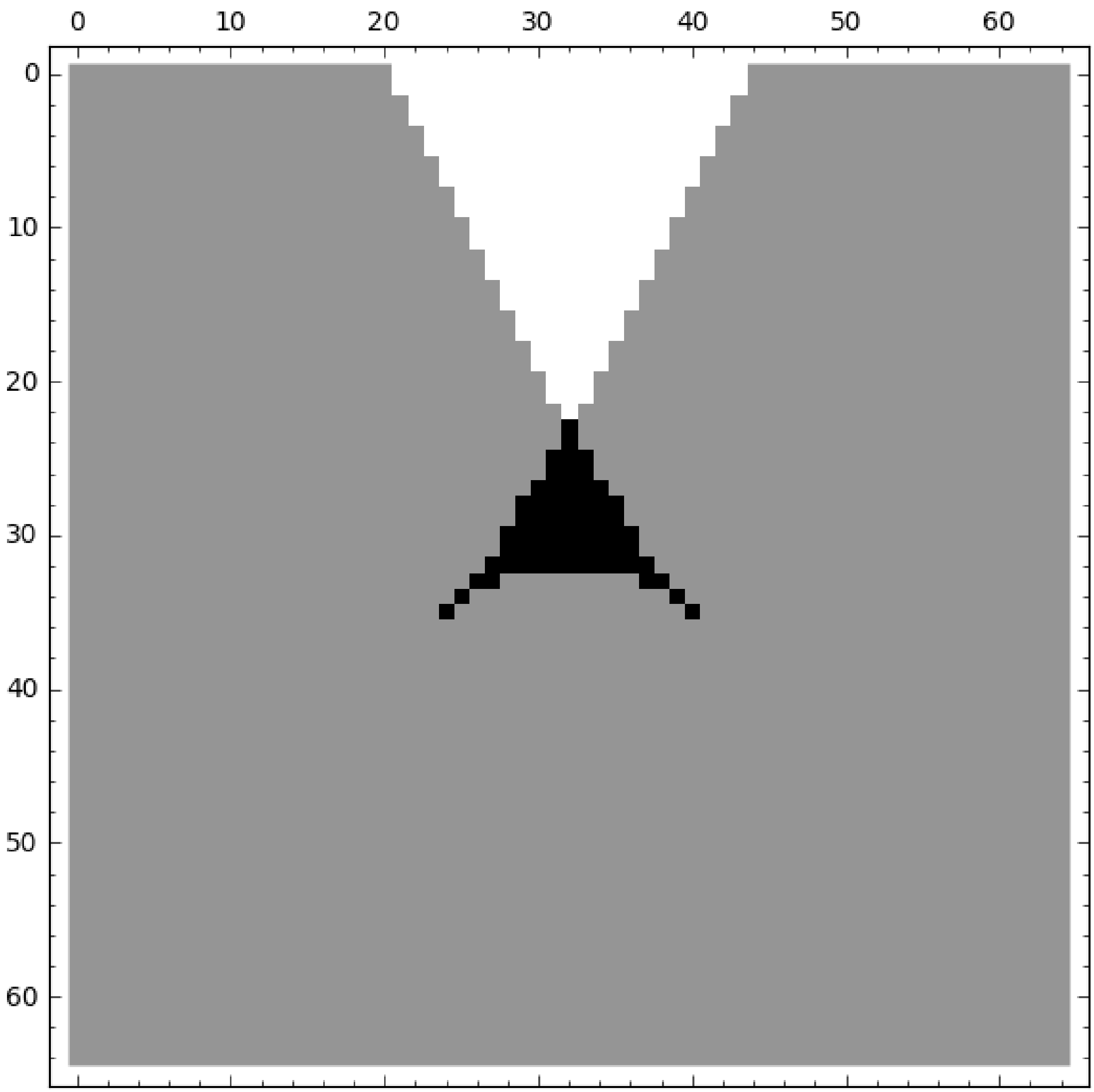
```
In [23]: matrix_plot(generar_matriz(16))
```

Out[23]:




```
In [24]: matrix_plot(generar_matriz(32))
```

Out[24]:



Comprobemos el caso $a = 8$:

Usando la función *numero_reales* podemos comprobar el caso $a = 8$, y, en particular vemos que es posible quen haya 4 raíces reales.

```
In [25]: M = generar_matriz(32)
```

```
In [26]: L=list(M.column(40));show(L)
```

```
[0,
 0,
 0,
 0,
 0,
 0,
```

```
0,
0,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
4,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
2,
```

```
 2,
 2,
 2,
 2,
 2,
 2,
 2,
 2,
 2,
 2,
 2]
```

Vemos que aparece un 4 en la lista.

```
In [27]: [(B,numero_reales(R(p(a=8,b=B)))) for B in srange(-32,32)]
```

```
Out[27]: [(-32, 2),
          (-31, 2),
          (-30, 2),
          (-29, 2),
          (-28, 2),
          (-27, 2),
          (-26, 2),
          (-25, 2),
          (-24, 2),
          (-23, 2),
          (-22, 2),
          (-21, 2),
          (-20, 2),
          (-19, 2),
          (-18, 2),
          (-17, 2),
          (-16, 2),
          (-15, 2),
          (-14, 2),
          (-13, 2),
          (-12, 2),
          (-11, 2),
          (-10, 2),
          (-9, 2),
          (-8, 2),
          (-7, 2),
          (-6, 2),
          (-5, 2),
          (-4, 2),
          (-3, 4),
          (-2, 2),
```

```
 (-1, 2),
 (0, 2),
 (1, 2),
 (2, 2),
 (3, 2),
 (4, 2),
 (5, 2),
 (6, 2),
 (7, 2),
 (8, 2),
 (9, 2),
 (10, 2),
 (11, 2),
 (12, 2),
 (13, 2),
 (14, 2),
 (15, 2),
 (16, 2),
 (17, 2),
 (18, 2),
 (19, 2),
 (20, 2),
 (21, 2),
 (22, 2),
 (23, 2),
 (24, 2),
 (25, 0),
 (26, 0),
 (27, 0),
 (28, 0),
 (29, 0),
 (30, 0),
 (31, 0)]
```

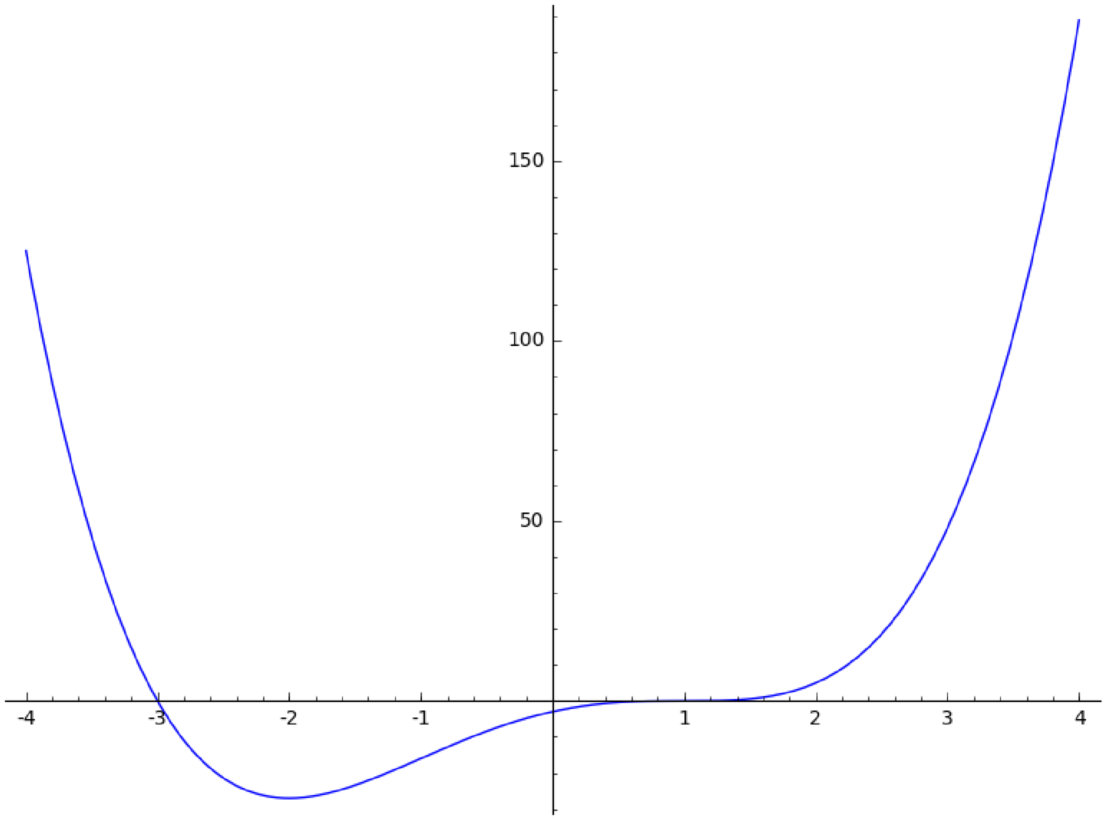
Y que el cuatro corresponde al valor $b = -3$.

```
In [28]: R(p(a=8,b=-3)).factor()
```

```
Out[28]: (x + 3) * (x - 1)^3
```

```
In [29]: plot(p(a=8,b=-3),-4,4)
```

```
Out[29]:
```



£CONTINUARÁ...?

```
In [ ]:
```