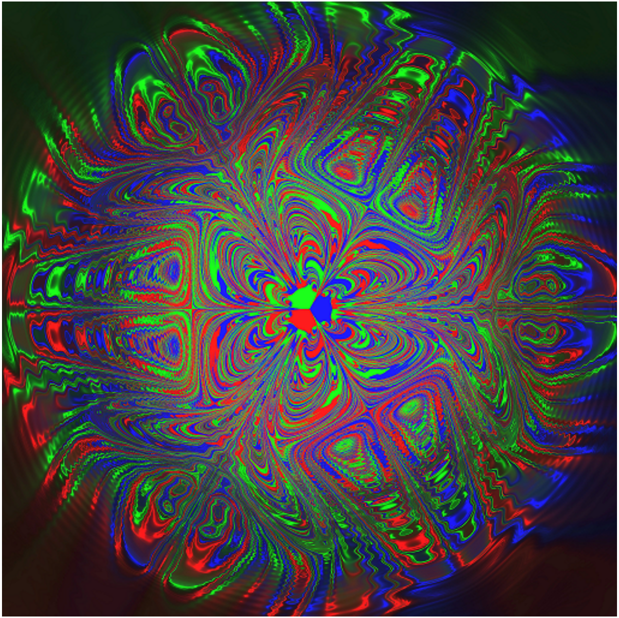
Ceros de ecuaciones

Juan Pablo Pinasco (jpinasco@dm.uba.ar//gmail.com)

Departamento de Matemática e IMAS,
FCEyN, UBA - CONICET

2020



Hoy:

- Ec no lineales
- bisec
- Newton Raph

Próxima:

- Convergencia NR: 71-76
- Punto Fijo: 76-79

Parte I

Ceros

1.- Problema

Dada $f : \mathbb{R} \to \mathbb{R}$, hallar r tal que $f(r) = 0$.

Veradero problema: no hay recetas ni fórmulas que permitan encontrar el cero. Ni siquiera para polinomios de grado 5 en adelante.

Existen diferentes métodos pero ninguno que sirva siempre, todos tienen algún detalle.

En general, se determina un intervalo donde haya una raíz, por ejemplo buscando a, b donde tenga distinto signo por prueba y error, después Bolzano garantiza que ahí hay un cero.

Este es un gran método, y es la demostración de Bolzano. Uno parte de $[a, b]$, con $f(a) > 0$ y $f(b) < 0$, divide el intervalo a la mitad y evalúa la f. Ahora, se queda con un nuevo intervalo según el signo de f en el punto medio.

def bizexion(f, a, b, epsilon)
  while $b - a >$epsilon:
    $c = 0{,}5 * (a + b)$
    if $f(c) >= 0$:
$\#$ si no saben meter f, en esa línea piden la cuenta explícita.
$\#$ por ejemplo: ”Mozo, la cuenta”.
$\#$ o si $f(x) = 3 * x * *7 - x + 10$, ponen if $3 * c * *7 - c + 10 >= 0$.
$\#$ recuerden import numpy as np, si usan np.e, np.sin, etc., hay
$\#$ muchas funciones en numpy pero hay que importar el módulo.
      $a = c$
    else:
      $b = c$
  return (a, b)

Si ponemos $epsilon = 0$ generamos una sucesión infinita de intervalos $[a_i, b_i]$, con $a_0 = a$, $b_0 = b$. Se cumple

$$a_1 \leq a_2 \leq \cdots \leq b,$$

$$b_1 \geq b_2 \leq \cdots \geq a,$$

$$b_i - a_i = \frac{b-a}{2^i}$$

Son sucesiones monótonas y acotadas, así que convergen, y lo hacen a lo mismo, un punto r tal que

$$a_i \leq r \leq b_i.$$

Como $0 \leq f(a_i) \to f(r)$, tenemos $0 \leq f(r)$.

Como $0 \geq f(g_i) \to f(r)$, tenemos $0 \geq f(r)$.

No queda otra que $f(r) = 0$.

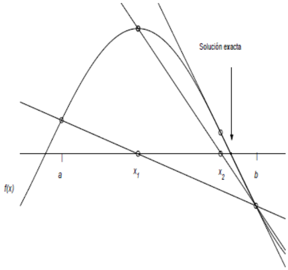
Teorema

Sea $f : [a, b] \to \mathbb{R}$ *continua,* $f(a)f(b) < 0$*. Entonces, el método de bisección genera una sucesión* $\{x_n\}_{n\geq 1}$ *que converge a una raíz* r *de* f*, y el error* $e_n = |x_n - r| \leq (b - a)/2^n$*.*

Dem: Por todo lo anterior, si llamamos $x_n = (a_{n-1} + b_{n-1})/2$, como la raíz está dentro del intervalo, tenemos

$$|r - x_n| \leq \frac{b_{n-1} - a_{n-1}}{2} = \frac{b - a}{2^n}$$

Este método es algo intermedio entre bisección y Newton-Raphson.



Dados a, b y f con $f(a)f(b) < 0$, definimos la recta

$$y - f(a) = \frac{f(b) - f(a)}{b - a}(x - a)$$

que tiene un cero

$$\begin{aligned} x_1 &= a - \frac{f(a)}{f(b) - f(a)}(b - a) \\ &= \frac{af(b) - bf(a)}{f(b) - f(a)} \end{aligned}$$

Evaluando $f(x_1)$, redefinimos a o b, e iteramos. Esta sucesión converge a la raíz (pero la longitud del intervalo no necesariamente tiende a cero).

En general es mejor que bisección, pero a veces tarda más (en los casos en los cuales Newton-Raphson falla)

Ventaja: no hay cancelaciones catastróficas que nos hagan dividir por cero.

En este método se comienza con un punto x_0, se utiliza la recta tangente a f en el punto para definir un nuevo punto x_1, y se continúa iterando.

Se basa en la esperanza de que la recta tangente se parezca a la función y corte al eje en el mismo lugar.

Hay muchos casos donde falla, depende del x_0 inicial, de que la raíz sea simple, que f' no se anule, etc.

A cambio, sirve para sistemas de ecuaciones, o para buscar raíces complejas.

Necesitamos f derivable. Conociendo x_n, definimos x_{n+1} como el cero de la recta tangente,

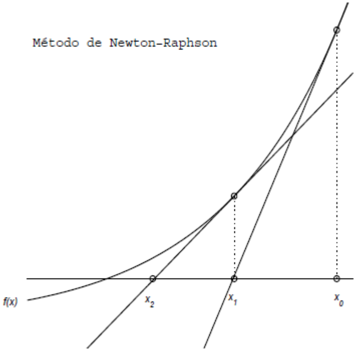
$$f(x_n) + f'(x_n)(x_{n+1} - x_n) = 0$$

Es decir,

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Por razones obvias, conviene que f' no se anule cerca de la raíz.

Aunque $f' \neq 0$, si toma valores demasiado chicos, el cociente puede irse a infinito.



Teorema

Sea $f \in C^1(\mathbb{R})$*, y* $\{x_n\}_{n\geq 1}$ *una sucesión construída con este método. Si* $x_n \to r$*, entonces* $f(r) = 0$*.*

Dem: Tenemos $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$ Entonces,

$$f(x_n) = f'(x_n)(x_n - x_{n+1}) \to 0 \qquad n \to \infty.$$

* * *

Para garantizar la convergencia y acotar el error necesitamos unas cuantas hipótesis y una definición:

Def: un cero r de f se dice *simple* si $f'(r) \neq 0$.

Teorema

Sea $f \in C^2([a,b])$, *y* r *un cero simple de* f. *Sea* $I = [r-\alpha, r+\alpha]$, *y sean* δ, M *positivos tales que para todo* $x \in I$,

- $|f'(x)| \geq \delta$,
- $|f''(x)| \leq M$.

Sea $\{x_n\}_{n\geq 1}$ *una sucesión construida con este método, y* $e_n = x_n - r$.

Entonces existe $\varepsilon > 0$ *tal que* $I_\varepsilon = [r-\varepsilon, r+\varepsilon] \subset I$, *y se tiene* $|e_n| \to 0$ *si* $x_0 \in I_\varepsilon$.

Más aún,

$$|e_{n+1}| \leq \frac{1}{2}\frac{M}{\delta}|e_n|^2.$$

Un caso importante:

Teorema

Si $f \in C^2(R)$, con $f'' > 0$ (convexa), cambia de signo, y f no alcanza su mínimo en x_0, entonces el método de Newton Raphson comenzando en x_0 converge.

En el primer teorema conseguimos

$$e_{n+1} = Ce_n^2.$$

Si f es C^2, y la raíz es simple, sale de hacer Taylor y acotar f' y f''. Lo importante es que esto dice que la convergencia es *cuadrática*, y en cada iteración duplicamos el número de cifras correctas.

La contra es que necesitamos empezar cerca de la raíz, y no tenemos cómo saber qué tan cerca estamos.

Si la función tiene k raíces, podemos clasificar cada x_0 en $k+1$ grupos: k corresponden a los puntos que convergen a la respectiva raíz, y el último son aquellos x_0 para los cuales el algoritmo no converge a ninguna.

En dimensión 2 (o mayor), con $F : \mathbb{R}^2 \to \mathbb{R}^2$, o en los complejos, tenemos

$$z_{k+1} = z_k - DF(z_k)^{-1} F(z_k)$$

La teoría es igual, pero hay fenómenos lindos.