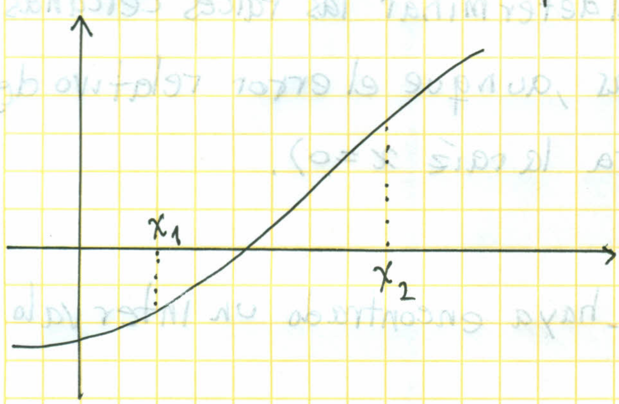
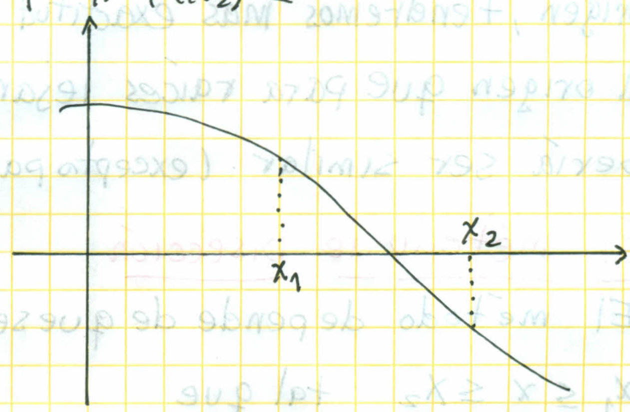
Raíces reales de una función

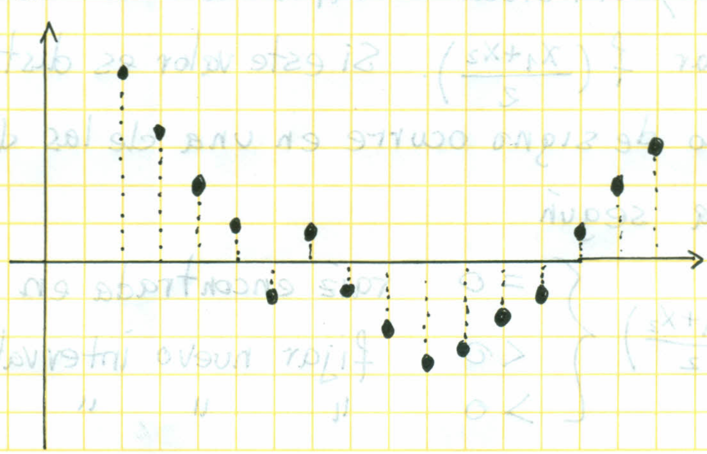
Las raíces

Desde el punto de vista matemático, las raíces de una función $y = f(x)$ son los valores de la variable independiente $x_1, x_2, \ldots, x_n$ tales que $f(x_i) = 0$ para $i = 1, \ldots, n$; pero desde el punto de vista computacional existe sólo una cantidad finita de números y puede que ninguno de estos produzca exactamente 0 para $f(x)$. Por esto, para encontrar raíces usando una computadora, buscaremos pares de valores muy juntos entre sí de manera que $f(x_1)$ y $f(x_2)$ tengan signos opuestos o, raramente, $f(x_1)$ o $f(x_2)$ sea exactamente 0. Nótese que $f(x_1)\, f(x_2) \leq 0$.





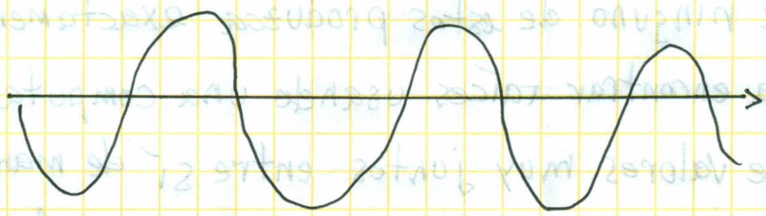
Debido a los efectos aleatorios del redondeo, muchas veces se tiene algo similar a



que se puede confundir con una secuencia de raíces. Para minimizar este tipo de efectos es necesario:

* Usar las técnicas de evaluación de funciones
* Estimar el tamaño del error de redondeo

Cuando tenemos una función con infinitas raíces entonces nos debemos contentar con encontrar las raíces en un intervalo finito.



Además, gracias a la densidad de números que es mayor cerca al origen, tendremos más exactitud al determinar las raíces cercanas al origen que para raíces lejanas, aunque el error relativo debería ser similar (excepto para la raíz $x=0$).

El método de bisección

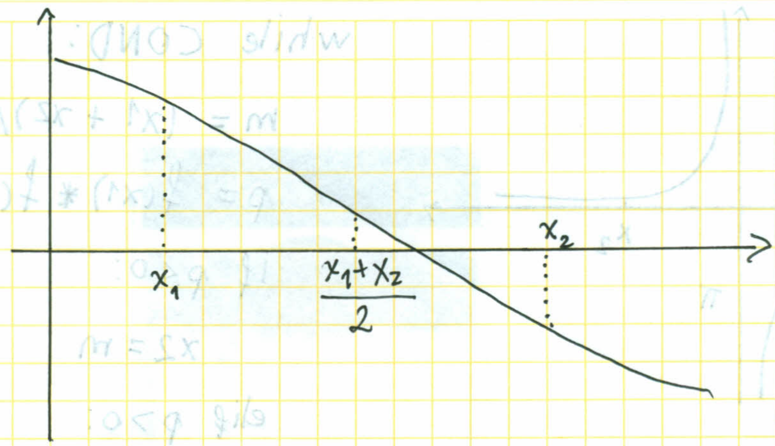
El método depende de que se haya encontrado un intervalo $x_1 \leq x \leq x_2$ tal que

$$f(x_1)f(x_2) < 0.$$

Una vez hecho esto, el método es capaz de reducir el tamaño del intervalo al evaluar $f\left(\frac{x_1+x_2}{2}\right)$. Si este valor es distinto de 0, entonces el cambio de signo ocurre en una de las dos partes. La decisión se toma según

$$f(x_1)f\left(\frac{x_1+x_2}{2}\right)\begin{cases} =0 & \text{raíz encontrada en } \frac{x_1+x_2}{2} \\ <0 & \text{fijar nuevo intervalo en } \left(x_1, \frac{x_1+x_2}{2}\right) \\ >0 & \text{〃 〃 〃 〃 } \left(\frac{x_1+x_2}{2}, x_2\right) \end{cases}$$

Tema: Fecha: /3/




```
while COND:
    m = (x1 + x2)/2.
    p = f(x1) * f(m)
    if p<0:
        x2 = m
    elif p>0:
        x1 = m
    else:
        break
return m
```

donde COND podría ser

* $|x1 - x2| > \varepsilon$
* $\left|\frac{x_1 - x_2}{x_1}\right| > \varepsilon$
* $|f(x_1) - f(x_2)| > \varepsilon$
* Repetir n veces, donde $10 \leq n \leq 40$ debería ser suficiente

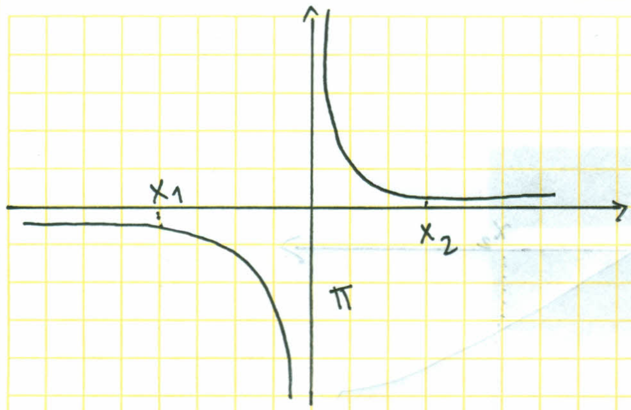
Advertencia: ε no debería ser demasiado pequeño.

Nótese que cada iteración reduce el intervalo a la mitad, así que después de 10 iteraciones el intervalo original se reduce en

$$2^{10} = 1024 \approx 10^3.$$

Algunos detalles

¿Qué pasa si la función tiene una singularidad en $x_1 \leq x \leq x_2$? Por ejemplo, consideremos la función $f(x) = \frac{1}{x-\pi}$ en el intervalo $1 \leq x \leq 4$.



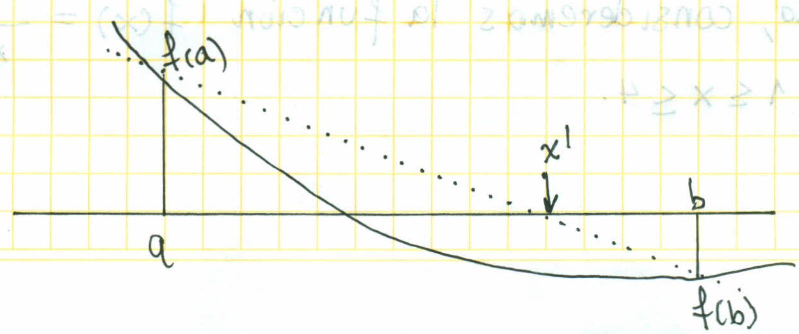

```
while COND:
    m = (x1 + x2)/2.
    p = f(x1) * f(m)
    if p<0:
        x2 = m
    elif p>0:
        x1 = m
    else:
        break
if abs(f(x1)) > 1./epsilon:
    return None
return m
```

¿Y si la función fuese $f(x) = \frac{1}{(x-\pi)^2}$?

≡ Los efectos del redondeo en el cálculo de los valores de la función no es serio pues el objetivo del método es encontrar intervalos en los que exista un cambio de signo, ya sea por la presencia de una raíz o por cualquier otro motivo.

El método de la falsa posición

Si existe un cambio de signo en $a \leq x \leq b$ y un extremo es largo y el otro corto, entonces la raíz probablemente se encuentra más cerca al extremo corto.



Para reducir el tamaño del intervalo se aproxima $f(x)$ con la línea

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

y se reduce el intervalo al reemplazar la raíz de la línea

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

por el extremo que tenga la función f con el mismo signo de $f(x')$

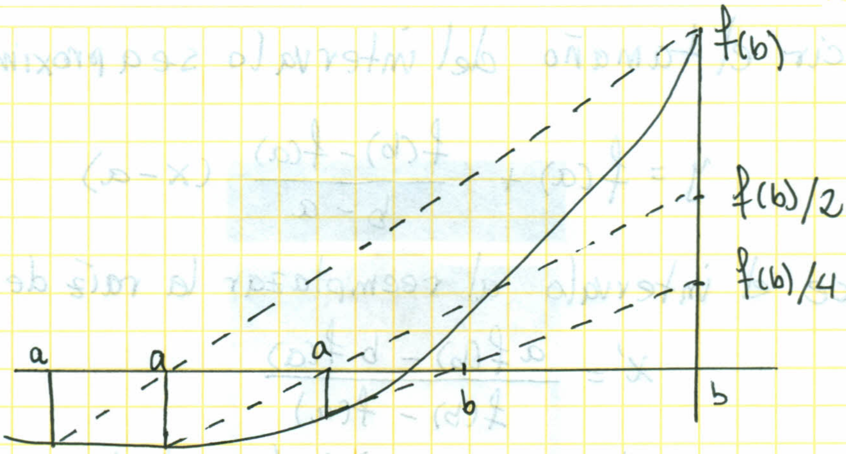
```
while abs(a-b) > 1e-5:
    x = (a*f(b) - b*f(a)) / (f(b) - f(a))
    p = f(a) * f(x)
    if p < 0:
        b = x
    elif p > 0:
        a = x
    else:
        break
return x
```

donde debemos recordar que $f(b) - f(a)$ es en realidad una suma.

Este método no es tan eficiente como el de bisección en términos de la reducción del intervalo, pues en este método ocurre en un solo lado por iteración.

El método modificado de la falsa posición

La debilidad del método de la falsa posición es el tiempo de ejecución. Para mejorar el método dividimos el valor conservado de f entre 2.




```
fa = f(a)
fb = f(b)
while abs(a-b) > ε:
    x = (a*fb - b*fa) / (fb - fa)
    p = fa * f(x)
    if p < 0:
        b = x
        fb = f(b)
        fa /= 2
    elif p>0:
        a = x
        fa = f(a)
        fb /= 2
    else:
        break
return (a, b)
```

Límites en la exactitud

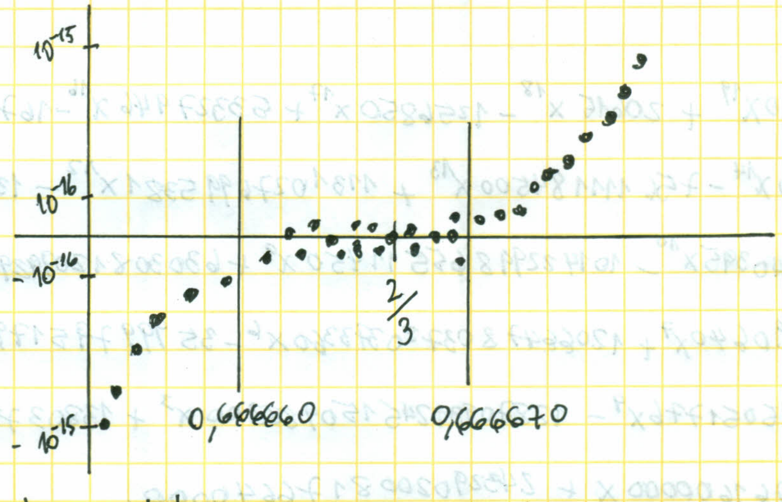
¿Qué límites tenemos en la exactitud de nuestras respuesta? La computadora típicamente usa 52 bits para la mantisa de los flotantes, que son más o menos 16 cifras decimales. ¿Esto implica que nuestras respuestas siempre serán correctas con 16 cifras significativas? NO. Considere el siguiente ejemplo:

Ejemplo Use el método de bisección para encontrar una raíz de $f(x) = x^3 - 2x^2 + \frac{4}{3}x - \frac{8}{27}$ con 6 cifras significativas.

Notemos que $x = 2/3$ es una raíz de f y que $f(0)f(1) < 0$

El método de bisección produce 0,6666564941; 0,6666717529 en 20 pasos y el mismo resultado en 200 pasos. Algo similar ocurre con el método de la falsa posición ¿qué sucede?

Lo que pasa es que existen varios valores a menos de 10^{-5} de distancia de 2/3 que producen 0 en la máquina y por lo tanto, desde el punto de vista de la computadora, son raíces de la función. Este es un problema de evaluación de funciones y no del método de bisección o algún otro método.



Para atacar este problema es necesario hacer un par de definiciones

(1) El error de entrada, que es el error que se comete en la entrada del algoritmo que, en este caso, es la función cuya raíz deseamos encontrar.

(2) El error de salida del algoritmo, que en este caso no es otra cosa que la solución numérica al problema de encontrar la raíz de $f(x)=0$.

En este caso particular, podemos ser más específicos. Imaginemos que tenemos una aproximación de la raíz verdadera (matemática) r. Sea nuestra aproximación x_a:

(1) El error de entrada es la constante que añadiríamos a f para hacer de x_a una raíz, esto es $|f(x_a)|$

(2) El error de salida es la diferencia entre la verdadera raíz r y x_a; es decir $|r-x_a|$.

Notemos que estos errores están directamente relacionados con los criterios usados para detener algunos algoritmos.

<u>El polinomio de Wilkinson</u>

El polinomio de Wilkinson es

$$W(x) = (x-1)(x-2)\cdots(x-20)$$

que es

$$\begin{aligned} W(x) = {} & x^{20} - 210x^{19} + 20615x^{18} - 1256850x^{17} + 53327946x^{16} - 1672280820x^{15} \\ & + 40171771630x^{14} - 756111184500x^{13} + 11310276995381x^{12} - 135585182899530x^{11} \\ & + 1307535010540395x^{10} - 10142299865511450x^{9} + 63030812099294896x^{8} - \\ & 311333643161390640x^{7} + 1206647803780373360x^{6} - 3599979517947607200x^{5} \\ & + 8037811822645051776x^{4} - 12870931245150988800x^{3} + 13803759753640704000x^{2} \\ & - 8752948036761600000x + 2432902008176640000. \end{aligned}$$

Obviamente, las raíces son $1, 2, \ldots, 20$. Si uno trata de resolver esto polinomio con un valor inicial cercano a 16 la respuesta no es 16, sino algo como $16{,}01468$.

Nuevamente, este tipo de problemas no se deben al algoritmo de resolución

Sensibilidad para encontrar raíces

Un problema se denomina "sensible" si pequeños errores en la entrada producen grandes errores en la salida. Consideremos el problema de hallar una raíz r. de $f(x)=0$. Imaginemos que cometemos un pequeño error en la entrada ($\varepsilon g(x)$, para ε pequeño) que produzca un error Δr en la salida:

$$f(r+\Delta r) + \varepsilon g(r+\Delta r) = 0,$$

que expandiendo en serie de Taylor resulta en

$$f(r) + \Delta r f'(r) + \varepsilon g(r) + \varepsilon \Delta r g'(r) \approx 0.$$

así que

$$\Delta r \approx \frac{-\overset{0}{\cancel{f(r)}} - \varepsilon g(r)}{f'(r) + \varepsilon g'(r)} \approx -\varepsilon \frac{g(r)}{f'(r)}$$

pues asumimos que $\varepsilon << f'(r)$, además de que $f'(r) \neq 0$.

Ejemplo Estime la raíz más grande de $P(x) = (x-1)(x-2)(x-3)(x-4)(x-5)(x-6)$ asumiendo un error en la entrada de $10^{-6}x^7$.

Si no utilizamos el error, la raíz más grande es 6. ¿Qué pasa si añadimos este error relativamente pequeño? la formula nos dice

$$\Delta r \approx -\varepsilon \frac{6^7}{5!} = -2332{,}8\ \varepsilon$$

es decir que la raíz la estimaremos como 6,0023328. Usando el método de bisección obtenemos 6,0023268.

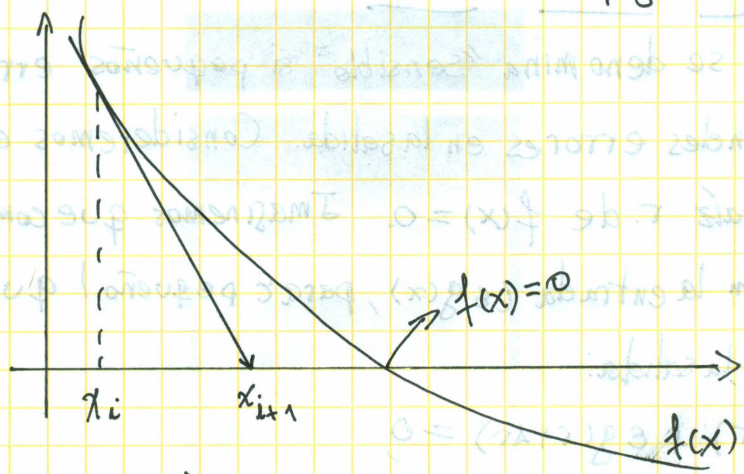
Esto es un ejemplo de cómo los errores en las entradas se propagan en las salidas

Podemos hacer la siguiente definición

$$\text{factor de incremento del error} = \frac{\text{error en la salida (relativo)}}{\text{error en la entrada (relativo)}}$$

El método de Newton

La esencia del método está ilustrada en la figura



Para encontrar la raíz $f(x)=0$, se inicia con una estimación x_0 y se utiliza la tangente de f en x_0 para mejorar la estima de la raíz y se repite el proceso. La ecuación de la tangente es

$$y = f(x_i) + f'(x_i)(x - x_i)$$

que se intersecta con el eje x cuando $y=0$, de manera que

$$x_{i+1} = x = x_i - \frac{f(x_i)}{f'(x_i)}$$

Ejemplo Use el método de Newton para encontrar la raíz de $x^3+x-1=0$. R. $x_{i+1} = \dfrac{2x_i^3+1}{3x_i^2+1}$

Convergencia del método de Newton

Sea f una función con derivada segunda. Si $f'(r)\neq 0$ entonces el método de Newton converge localmente a r de manera cuadrática:

$$\lim_{i\to\infty} \frac{e_{i+1}}{e_i^2} = \frac{1}{2}\frac{f''(r)}{f'(r)} \qquad \text{con } e_i = x_i - r \qquad (*)$$

La demostración es sencilla usando el teorema de Taylor con resto

$$f(r) = 0 = f(x_i) + (r-x_i)f'(x_i) + \frac{(r-x_i)^2}{2} f''(c_i)$$

con c_i entre x_i y r. Reordenando

$$x_i - \underbrace{\frac{f(x_i)}{f'(x_i)}} - r = \frac{(r-x_i)^2}{2}\frac{f''(c_i)}{f'(x_i)}$$

$$\underbrace{x_{i+1} - r}_{e_{i+1}} = \frac{1}{2} e_i^2 \frac{f''(c_i)}{f'(x_i)}$$

de donde $\frac{e_{i+1}}{e_i^2} = \frac{1}{2}\frac{f''(c_i)}{f'(x_i)}$. Tomando $i\to\infty$ se obtiene (*), asumiendo que x_i converge a r (y c_i también, pues está entre x_i y r). ¿Cómo demostramos que x_i tiende a r? piense en el teorema del valor medio con x_i y r como extremos.

La relación (*) nos permite escribir que

$$|e_{i+1}| \approx \frac{1}{2}\left|\frac{f''(r)}{f'(r)}\right| e_i^2$$

para aproximar el error por iteración, asumiendo que $f'(r)\neq 0$.

¿Qué sucede si $f'(r)=0$? tome en cuenta el caso en que $f(x)=x^2$. Para este caso $x_{i+1}=\frac{x_i}{2}$ que converge linealmente.

Si asumimos que tenemos una función f $m+1$ veces diferenciable en un intervalo $[a,b]$ con multiplicidad m en la raíz r, entonces

$$\lim_{i\to\infty}\frac{e_{i+1}}{e_i} = \frac{m-1}{m}$$

El método modificado de Newton

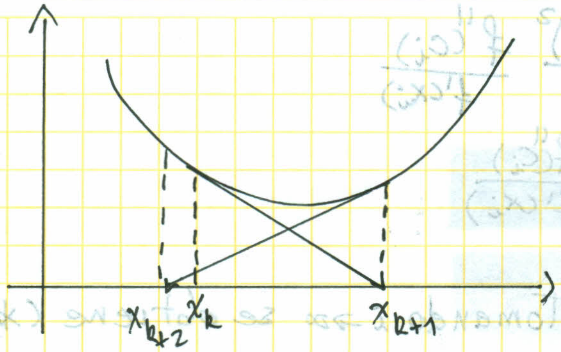
Si f es una función con derivada $(m+1)$-ésima en $[a,b]$ y tiene una raíz r con multiplicidad $m>1$ entonces

$$x_{i+1} = x_i - m\frac{f(x_i)}{f'(x_i)}$$

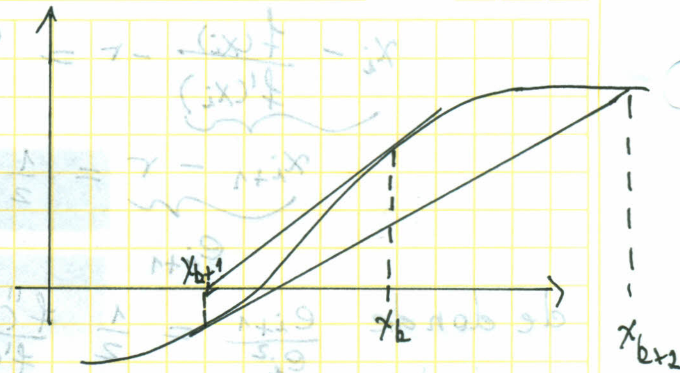
converge localmente a r de manera cuadrática.

Limitaciones del método de Newton

Tema: Fecha: / /



Mínimo local



Punto de inflexión