Newton-Raphson y Aitken Delta^2

Lida K. Guerrero L.
Juan M. Duarte L.

Pontificia Universidad Javeriana

Febrero 2021

1. Newton-Raphson

Es un método numérico conocido que se basa en la aproximación para encontrar raices, por medio de sucesiones iterativas usando la formula:

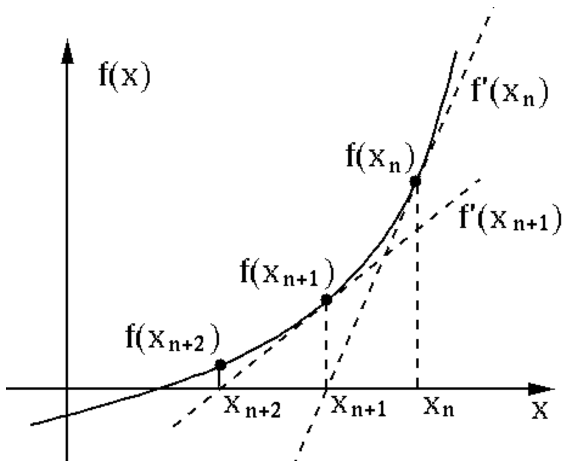
$g(x) = x - \frac{f(x)}{f'(x)}$

1.1. Condiciones

1. Para alcanzar la convergencia el valor inicial debe ser suficientemente cercano a la raiz a encontrar.
2. La función evaluada en x debe ser diferente de 0 :
   $f(x) \neq 0$
3. La derivada de la función evaluada, debe ser diferente de 0 :
   $f'(x) \neq 0$

1.2. Análisis geométrico

El método utiliza un valor de x inicial x_n que donde intercepta con la función, se realiza una recta tangente a ese punto, y se proyecta hasta encontrar el punto de corte de la tangente y el eje x, a este nuevo punto se le denomina $x_n{+}_1$ y se repite el proceso del trazo de la recta tangente hasta encontrar el nuevo punto de corte, este procedimiento se realiza hasta llevar al límite que tiende a cero.



[1]Imagen 1: Comportamiento geométrico

1.3. Diagrama de flujo

El algoritmo tiene definida una función y un valor inicial de x cercano a la raíz que se desea encontrar, se calcula la derivada de la función y se evalúan las condiciones que debe cumplir el método, si se cumplen procede a calcular en nuevo valor de x, x', el cual será utilizado para la siguiente iteraciòn, compara el valor de x con la tolerancia especificada, para volver a comenzar el procedimiento con el nuevo valor. Cuando una condición no se cumple, el algoritmo se detiene.

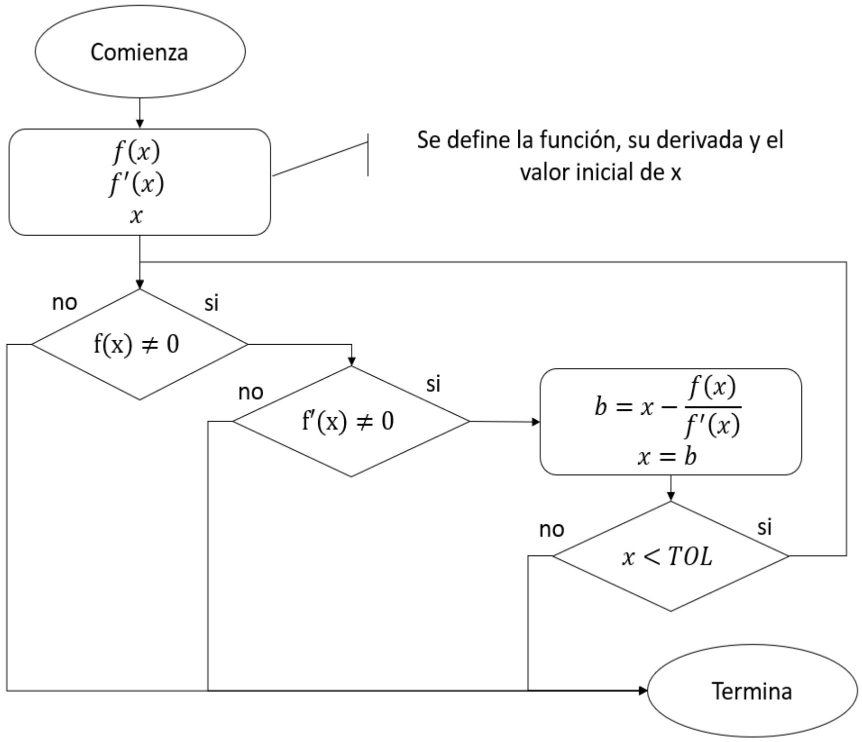


Imagen 2: Comportamiento del algoritmo

1.4. Comportamiento

La ecuación seleccionada para este ejemplo es $f(x) = cos^2(x) - x^2$, su derivada usando WolframAlpha nos da $f'(x) = -2(x + cos(x)sen(x))$. El método de Newton-Raphson nos pide que elijamos un X cercano a la raíz, se seleccionó $X = 1$ con el fin de analizar la raíz positiva de esta ecuación. Las raíces obtenidas usando la aplicación WolframAlpha son 0.739085 y -0.739085. Esta la usaremos para comparar nuestro resultado usando el método de Newton-Raphson.

Usando una tolerancia de 10^{-8} nuestro programa itero 3 dándonos los siguientes resultados:

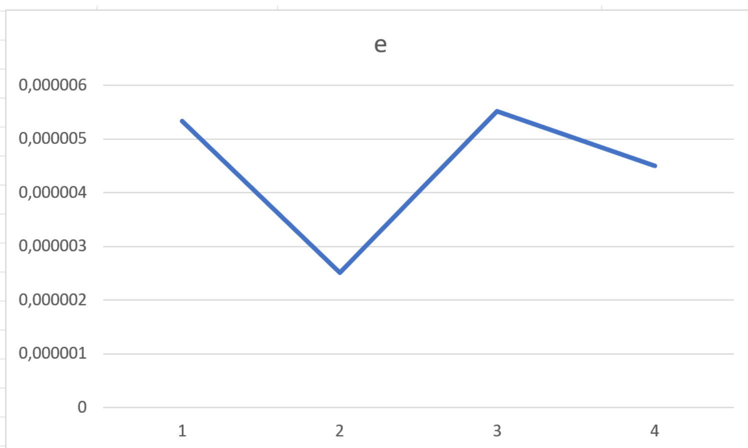
Iteraciones	Resultados
Iteración 1	0.756617
Iteración 2	0.739216
Iteración 3	0.7390851

Cuadro 1: Resultados.

La raíz obtenida es 0.7390851. Al comparar con el resultado de Wolfram

Alpha, tenemos que el error relativo es de $1{,}35 * 10^{-5}\,\%$ lo cual consideramos que es muy bueno para tan solo 3 iteraciones.

Utilizando la relación gráfica de e_{i-1} y e_i y la definición 2.5 del enunciado, podemos deducir que la función utilizada no converge linealmente ya que no se estabiliza en un valor S , por el contrario sus valores son diferentes por iteración y eso tambien se puede observar en la gráfica:
(Los valores de error son calculados con datos de GeoGebra y nuestro algoritmo.)



X	X*	error
0,756617	0,7566170403	5,32634E-06
0,739216	0,7392159814	2,51618E-06
0,7390851	0,7390851408	5,52034E-06
0,7390851	0,7390851332	4,49204E-06

Cuadro 2: Resultados.

En la definición 2.6, al aumentar el valor de r observamos que el limite tiende a infinito, entre mayor sea el valor de r, la cifra será mas grande, se muestra el resultado de:
$\frac{|x_{i+1}-x^*|}{|x_i-x^*|}$, variando los valores de r.

r=1	r=2	r=3
431787,6005	1,07143E+13	2,65864E+20
7036,634414	3,78314E+11	2,03394E+19
1	24509803,96	6,0073E+14
22261600,42	6,7053E+14	2,01967E+22

Cuadro 3: Resultados.

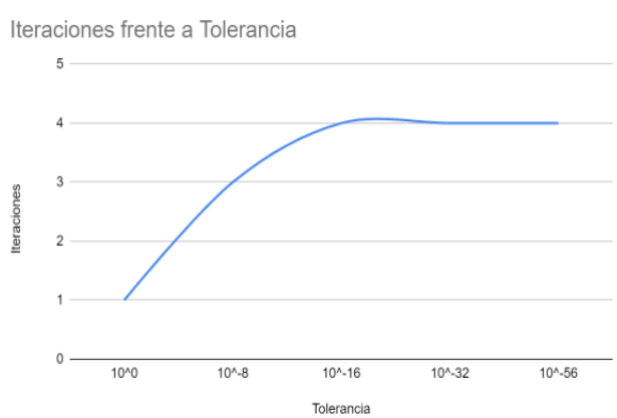
Revisando nuestro programa encontramos pérdida de significancia ya después de la tercera iteración ya que la cuarta nos daba el mismo resultado. Esto se debe a la falta de manejo en las cifras decimales el programa solo nos dio 7 cifras decimales para poder trabajar. Más adelante actualizando el programa usando librerías que nos dieron la oportunidad de un mejor manejo de estas. Utilizando Geogebra tenemos ahora que la raíz positiva es de 0.7390851332143 el cual es una aproximación mayor dándonos el error relativo de $4{,}49 * 10^{-6}\,\%$ el cual es de magnitud mayor al anterior haciendo nuestro programa mejor que el resultado obtenido el WolframAlpha.Para solucionar este problema solo fue necesario incrementar usar librerías para un mejor manejo de números muy pequeños.

Nuestro programa dependía de una tolerancia, esta era la que alteraba el número de iteraciones. A menor tolerancia el programa debía intentar iterar mas con el fin de culplir la condicion. Pudimos observar como cambiando la tolerancia de 10^{-8} a 10^{-16} nuestro programa inmediatamente le agregaba una iteración más y así en adelante.

Notamos que nuestro programa, antes de usar las librerias que nos permiten usar más decimales, convergia rápidamente a un número con 7 decimales, 0.7390851, el cual era el más cercano posible representado con esta cantidad de decimales.

Cuando existen más de una raíz, nuestro método tiene la necesidad de un dato X el cual debe ser cercano a una de estas, y se prohíbe que la recta tangente en ese X sea paralela al eje X. Gracias a este X nuestro programa se orienta y elige la raíz más cercana a este usando el método de Newton-Raphson.

Si la función llega a ser par o impar, se aplica lo mismo mencionado anteriormente y dependiendo de el X que se eligió se tomará la raíz más cercana. Si se llegara a elegir un punto medio entre estos dos la X' nos da hasta qué punto se acerca y la condición nos impide que esta recta tangente sea paralela al eje X, si lo es es necesario cambiar el X, esto es una precondición del método y no depende de la implementación del código.



Los métodos de Newton-Raphson y el Bisección son usados para encontrar raíces, ambos métodos cuentan con precondiciones muy similares. El método de Bisección trabaja dividiendo el intervalo a la mitad y acercándose a la raíz de esa forma. A diferencia del método de Newton-Raphson que inicia con un X cercano a la raíz y usa ese punto encontrando la tangente y cruzando con el eje X encontrando así el siguiente punto y repitiendo el algoritmo con el fin de acercarse a la raíz. Los comportamientos, por lo anteriormente mencionado, son diferentes pero su finalidad es la misma. El método de Newton-Raphson se considera más rápido en llegar a la raíz pero su complejidad es mayor.

El método de Taylor se usa para encontrar un valor en un punto, en este caso nos interesa las raíces. Este método usa el polinomio de Taylor y varias derivadas para hallar estos puntos. El método de Newton-Raphson nos da su aproximación sin la necesidad de derivar más de una vez la cual se nos hace más simple. La representación gráfica del método de Newton-Raphson, mostrado en la tabla, nos facilita el entendimiento del algoritmo.Taylor por el otro lado es más complejo usando múltiples integrales, series de Maclaurin y funciones comunes, como es la exponencial, logarítmica, geométrica, binomial y otras. Todo esto con el fin de encontrar el valor en un punto. La ventaja que tiene Taylor es que puede elegir cualquier punto que se desee a diferencia de Newton-Raphson que es diseñado para hallar las raíces.

2. Aitken delta^2

Es un método utilizado para acelerar la convergencia de un secuencia, por medio de sucesiones, el método fue desarrollado por Alexander Aitken y su fórmula es:
$x_{n+2} = x_{n+2} - \frac{(x_{n+2}-x_{n+1})^2}{x_{n+2}-2x_{n+1}+x_n}$
Para demostrar la eficiencia de Aitken, podemos utilizar la serie infinita de letbinz para calcular pi:
$\frac{\pi}{4} = \sum_{n=0}^{\infty} \frac{-1^n}{2n+1}$

n	sumatoria	Error Aprox.
0	1	-
1	0,666666	50,00 %
2	0,866666	23,07 %
3	0,723809	19,73 %
4	0,834920	13,30 %
5	0,744011	12,21 %
.	.	.
.	.	.
.	.	.
63	0,781492	1,007 %
64	0,789244	0,982 %

Cuadro 4: Tabla Letbinz.[2]

Podemos observar que para obtener un error aproximadomenor al 1 %
necesitamos 64 iteraciones. Aplicando el metodo de Aitken, podemos obtener un error aproximado menor de 1 % con menos iteraciones (Ver cuadros 1 y 2), por lo que se concluye que es un método eficiente.

iteración	aceleración	Error Aprox.
1	0,791666	-
2	0,783333	1,06 %
3	0,786309	0,37 %

Cuadro 5: Tabla Aitken.[2]

Referencias

[1] fourier.eng, "Newton-raphson method (univariate)," accessed: 2021-02-15. [Online]. Available: http://fourier.eng.hmc.edu/e176/lectures/NM/node20.html

[2] H. Negrete, "Aceleración de aitken," accessed: 2021-02-15. [Online]. Available: https://www.youtube.com/watch?v=lGsptXG15ik