Ejercicio Raices

John Jairo Gonzalez Martinez
Daniel Esteban Tibaquira Galindo
Karen Sofia Coral Godoy
Stiven Gonzalez Olaya

5 de Septiembre 2020

1 Metodo de Newton-Raphson

El método de Newton-Raphson busca encontrar la raíz (intercepto con el eje X) en una función. Este método utiliza la derivada de una función para encontrar la pendiente en un punto dado. Una vez que la encuentra, sustituye el valor inicial de x por un valor dado por la fórmula:

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

De esta forma se obtiene un valor de x con el que se repetirá este mismo proceso hasta dar con la raíz.

Data: $f \in C^2[a,b]$,X_0,tol y maxItr
Result: Si la iteración converge, una aproximaciónxnde un cero defen[a,b]y unaestimación del error.
initialization;
k = 0
$x_k = x_0$
do
  dx= $f(x_i)/f'(x_k)$
  x$_($k + 1$) = x_k - dx$
  x$_k = x_($k + 1$)$
  k = k + 1
while $dx \leq tol \parallel k \leq maxItr$;
return $x_k y dx$

Algorithm 1: Algoritmo de Newton-Raphson

1.1 Condiciones para aplicar el método

Este método es uno de los mas utilizados para localizar raíces ya que en general es muy eficiente, para aplicar el método de Newton- Raphson se requiere que las funciones sean diferenciables (derivables), y por tanto, continuas. Adicionalmente, es necesario proporcionar un valor inicial de x preferiblemente que se encuentre cercano al de la raíz y que no esté en un punto de inflexión de la función, porque esto provocaría que la tangente sea cero, y, por consiguiente, provocaría que la formula Newton-Raphson se indetermine. Sin embargo, este valor puede ser cualquier valor ya que el método convergirá a la raíz más cercana. El método no puede ser utilizado para los casos en que la derivada sea cero $f'(x) = 0$, y La eficiencia del método depende del valor inicial elegido.

Por otra parte es importante tener en cuenta que el método de Newton converge si se cumplen las siguientes tres condiciones para la función $f(x)$:

i. Existen dos puntos a y b en los que $f(a)f(b) < 0$.
ii. $f''(x)$ no cambia de signo en [a,b].
iii. Las tangentes a $f(x)$ en a y b cortan al eje de abscisas en [a,b].

Si no se cumplen estas condiciones, el método puede todavía converger, aunque no garantiza nada. En particular, cuando $f'(x)$ se hace muy pequeña en el intervalo comprendido entre el punto de partida y la raíz, el punto dado por la iteración siguiente tiene un valor muy grande, puesto que la tangente es casi paralela al eje de abcisas, y en estos casos el método diverge.

1.2 Explicación geometrica del algoritmo

Partiendo de una x inicial (x_0) de f, entonces x_1 es la intersección de la recta tangente a f en x_0, con el eje X, donde x_1 representa una aproximacion a la raiz.

Cuando se ha calculado la aproximación x_n, la siguiente aproximación x_{n+1} se obtiene hallando la intersección con el eje X de la recta tangente en el punto $(x_n, f(x_n))$, resolviendo $y = 0$. Reiterando este procedimiento se obtiene una sucesión de aproximaciones, que converge a la solución.

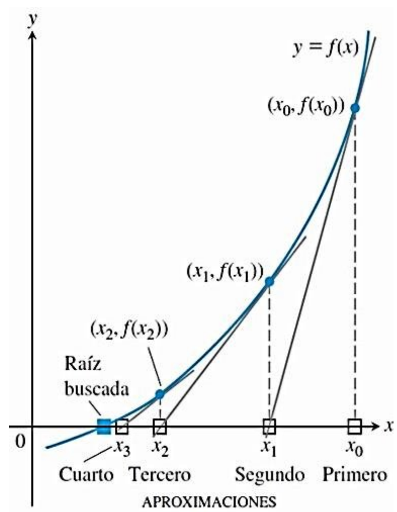

Figura 1: Descripción gráfica del método

1.3 Diagrama de flujo del algoritmo

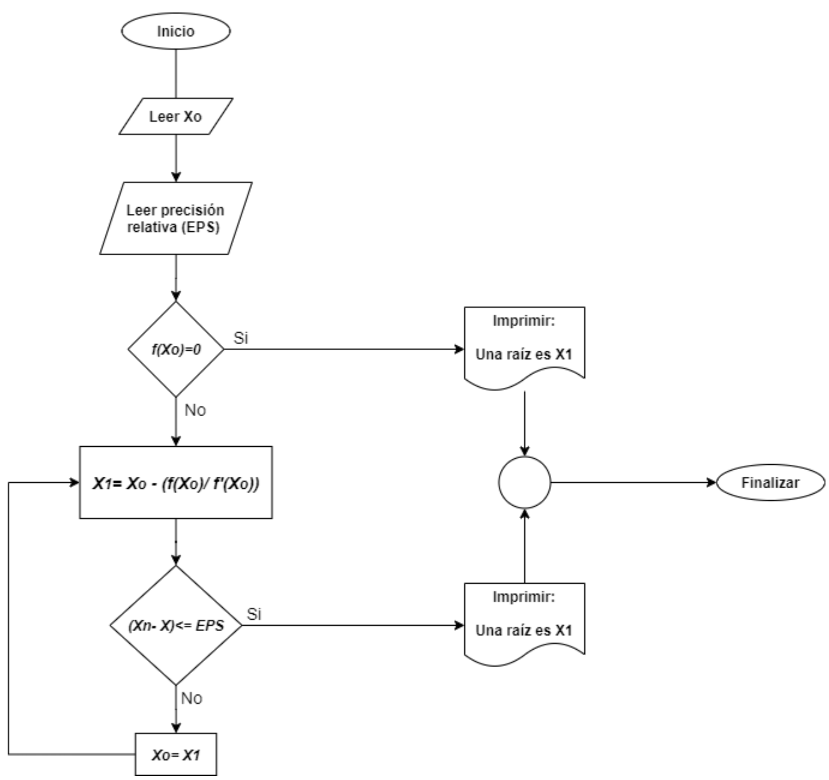

Figura 2: Diagrama de flujo de algoritmo de Newton

1.4 ¿Cuales son las raices?

1.4.1 $f(x) = \cos^2(2x) - x^2$

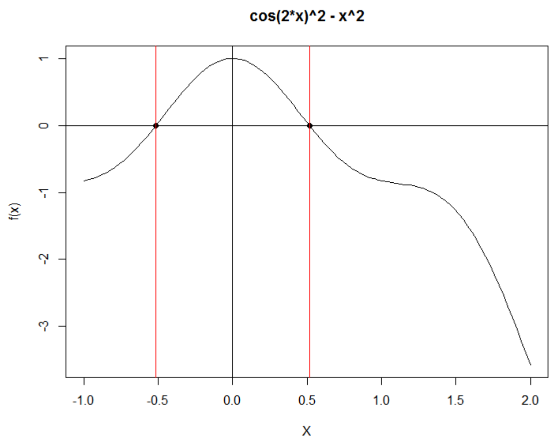

Figura 3: Grafica función $f(x) = \cos^2(2x) - x^2$

Valor raiz: Se encuentran 2 raices en esta función que son:

$-0.5149332646611294138010592584369123175782$

$0.5149332646611294138010592584369123175782$

Convergencia:
Se comporta de manera cuadrática, durante cada iteración el número de cifras significativas es el doble (aproximadamente) de la iteración anterior.

$\varepsilon = 10^{-8}$

Perdida de significancia:
Dado el valor conocido de la raíz, lo comparamos con el de una herramienta, tomando como $\varepsilon = 10^{-42}$ obtenemos que la raíz es $x \approx 0.5149332646611294138010592584369123175765$. La diferencia entre ambas entonces corresponde a: $-5.8735586198940741563087682423 5 \times 10^{-11}$. Analizando el comportamiento con una tolerancia más alta se comprenderá mejor cómo reacciona el algoritmo en términos de pérdida de significancia.

Número de iteraciones:
Con un X_0 de 1 y -1 son 5 iteraciones para cada una de las raices.

Grafica respecto a la tolerancia y al número de iteraciones

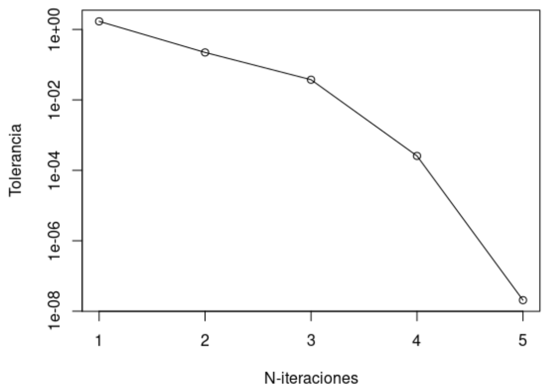

Figura 4: Grafica de numero de iteraciones con una tolerancia de $\varepsilon = 10^{-8}$

$\varepsilon = 10^{-16}$

Perdida de significancia:

Dado el valor conocido de la raíz, lo comparamos con el de una herramienta, tomando como $\varepsilon = 10^{-42}$ obtenemos que la raíz es $x \approx 0.51493326466112941380$ 10592584369123175765. La diferencia entre ambas entonces corresponde a: $4.13801059258436912317576 5 \times 10^{-16}$. Como sería de esperar la pérdida de significancia ahora es menor, como no se están obteniendo dígitos sin significancia aún, podemos ver que el algoritmo recupera más infomación con una tolerancia más alta, haciéndolo así más preciso.

Número de iteraciones:

Con un X_0 de 1 y -1 son 6 iteraciones para cada una de las raices.

Grafica respecto a la tolerancia y al número de iteraciones

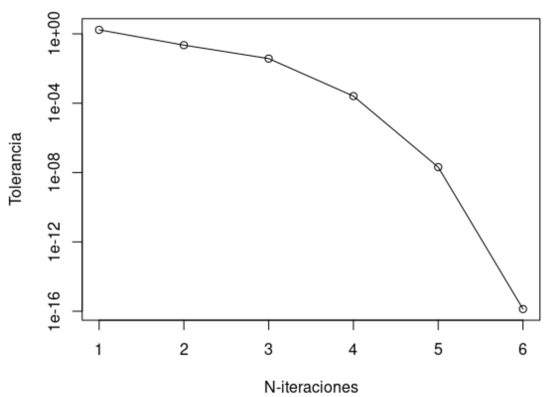

Figura 5: Grafica de numero de iteraciones con una tolerancia de $\varepsilon = 10^{-16}$

$\varepsilon = 10^{-32}$

Perdida de significancia:

Dado el valor conocido de la raíz, lo comparamos con el de una herramienta, tomando como $\varepsilon = 10^{-42}$ obtenemos que la raíz es $x \approx 0.51493326466112941380$ 10592584369123175765. La diferencia entre ambas entonces corresponde a: $1.33645853792943912317576 5 \times 10^{-16}$. Si bien comparando a la diferencia inmediamente anterior concluimos que el cambio es menos relevante, pero sigue siendo una mejora importante si comparamos el resultado utilizando el método con tolerancia $\varepsilon = 10^{-8}$.

Número de iteraciones:

Con un X_0 de 1 y -1 son 7 iteraciones para cada una de las raices.

Grafica respecto a la tolerancia y al número de iteraciones

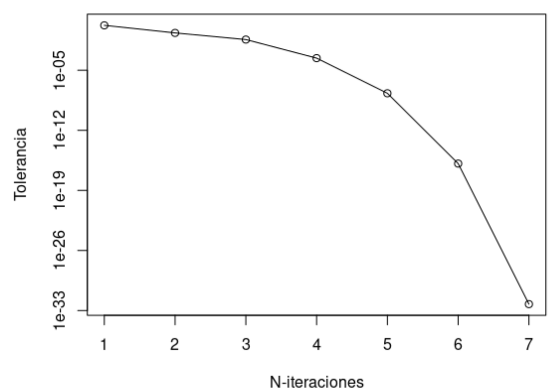

Figura 6: Grafica de numero de iteraciones con una tolerancia de $\varepsilon = 10^{-32}$

1.4.2 $f(x) = x \sin(x) - 1$ $[-1, 2]$

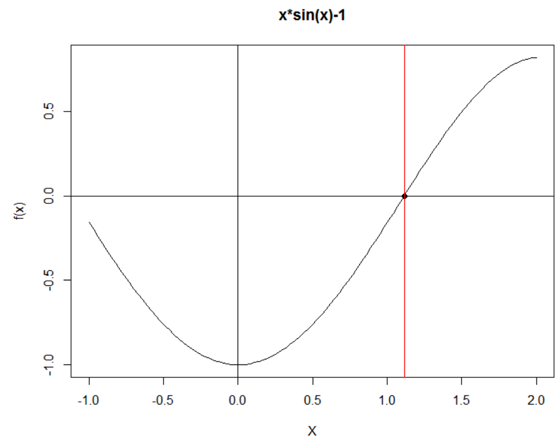

Figura 7: Grafica función $f(x) = x \sin(x) - 1$ $[-1, 2]$

Valor raiz: 1.114157140871930087300525178169203903956
Convergencia:
Se comporta de manera cuadrática, durante cada iteración el número de cifras significativas es el doble (aproximadamente) de la iteración anterior.

$\varepsilon = 10^{-8}$
Perdida de significancia:
Dado el valor conocido de la raíz, lo comparamos con el de una herramienta, tomando como $\varepsilon = 10^{-42}$ obtenemos que la raíz es $x \approx 1.11415714087193008730052517816920390395$. La diferencia entre ambas entonces corresponde a: $4.087193008730052517816920390395 \times 10^{-8}$. Analizando el comportamiento con una tolerancia más alta se comprenderá mejor cómo reacciona el algoritmo en términos de pérdida de significancia.
Número de iteraciones:
Con un X_0 de 1 son 3 iteraciones para hallar la raiz
Grafica respecto a la tolerancia y al número de iteraciones

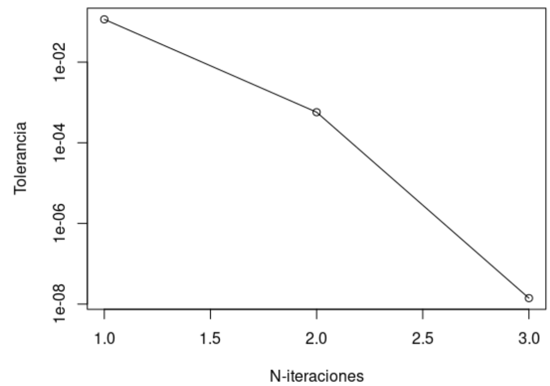

Figura 8: Grafica de numero de iteraciones con una tolerancia de $\varepsilon = 10^{-8}$

$\varepsilon = 10^{-16}$
Perdida de significancia:
Dado el valor conocido de la raíz, lo comparamos con el de una herramienta, tomando como $\varepsilon = 10^{-42}$ obtenemos que la raíz es $x \approx 1.11415714087193008730052517816920390395$. La diferencia entre ambas entonces corresponde a: $8.730052517816920390395000000000000000000 \times 10^{-17}$. De nuevo encontramos una relación entre la tolerancia y las cifras significativas, pues en este caso la diferencia es cerca de dos veces menor que utilizando la tolerancia $\varepsilon = 10^{-8}$
Número de iteraciones:
Con un X_0 de 1 son 4 iteraciones para hallar la raiz
Grafica respecto a la tolerancia y al número de iteraciones

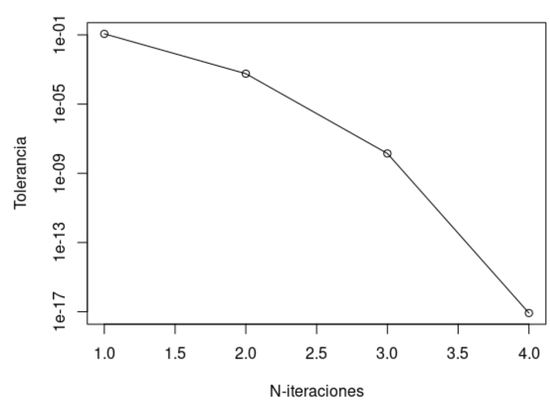

Figura 9: Grafica de numero de iteraciones con una tolerancia de $\varepsilon = 10^{-16}$

$\varepsilon = 10^{-32}$

Perdida de significancia:

Dado el valor conocido de la raíz, lo comparamos con el de una herramienta, tomando como $\varepsilon = 10^{-42}$ obtenemos que la raíz es $x \approx 1.11415714087193008730052517816920390395$. La diferencia entre ambas entonces corresponde a: $8.37837192182220390395000000000000000000 \times 10^{-18}$. Esta función reacciona de manera diferente que la primera, pues vemos que la pérdida de significancia continua disminuyendo, lo cual nos puede hacer pensar que si utilizamos una tolerancia más grande esta continuará en este patrón.

Número de iteraciones:

Con un X_0 de 1 son 5 iteraciones para hallar la raiz

Grafica respecto a la tolerancia y al número de iteraciones

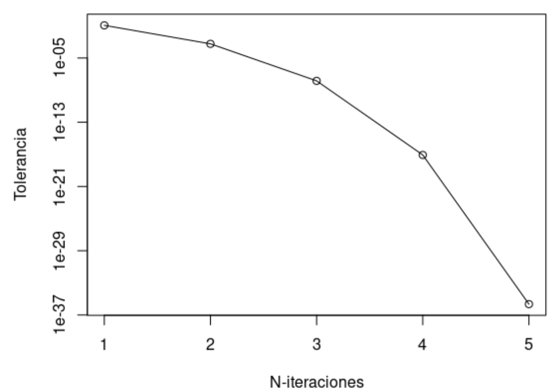

Figura 10: Grafica de numero de iteraciones con una tolerancia de $\varepsilon = 10^{-32}$

1.4.3 $f(x) = x^3 2x^2 + \frac{4}{3}x - \frac{8}{27}$

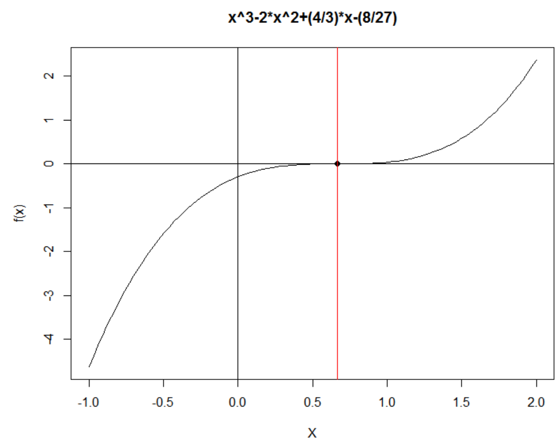

Figura 11: Grafica función $f(x) = x^3 2x^2 + \frac{4}{3}x - \frac{8}{27}$

Valor raiz: 0.66666987095209829807330182122489041198620
Convergencia:
Se comporta de manera cuadrática, durante cada iteración el número de cifras significativas es el doble (aproximadamente) de la iteración anterior.
$\varepsilon = 10^{-8}$
Perdida de significancia:
Dado el valor conocido de la raíz, lo comparamos con el de una herramienta, tomando como $\varepsilon = 10^{-42}$ obtenemos que la raíz es $x \approx 0.66666666666666666666 66666666666666666667$. La diferencia entre ambas entonces corresponde a: $-4.473333333333333333333333333333333333 \times 10^{-6}$. Analizando el comportamiento con una tolerancia más alta se comprenderá mejor cómo reacciona el algoritmo en términos de pérdida de significancia.
Número de iteraciones:
Con un X_0 de 1 son 31 iteraciones para hallar la raiz
Grafica respecto a la tolerancia y al número de iteraciones

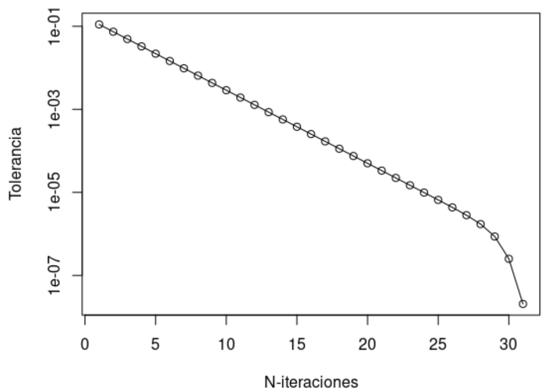

Figura 12: Grafica de numero de iteraciones con una tolerancia de $\varepsilon = 10^{-8}$

$\varepsilon = 10^{-16}$

Perdida de significancia:

Dado el valor conocido de la raíz, lo comparamos con el de una herramienta, tomando como $\varepsilon = 10^{-42}$ obtenemos que la raíz es $x \approx 0.66666666666666666666 6666666666666666667$. La diferencia entre ambas entonces corresponde a: $-3.20428542333333333333333333333000000 \times 10^{-6}$. Comparando este resultado antes los de las funciones anteriores es claro que no se está obteniendo una significancia tan relevante, sin embargo también hay que tomar en cuenta que incluso con tolerancia $\varepsilon = 10^{-8}$ el error es menor en magnitud comparado a los ejercicios anteriores.

Número de iteraciones:

Con un X_0 de 1 son 34 iteraciones para hallar la raiz

Grafica respecto a la tolerancia y al número de iteraciones

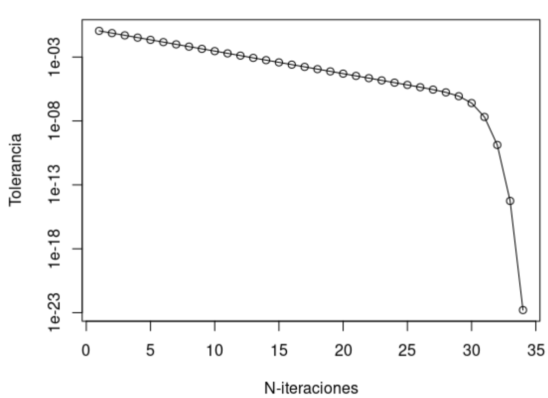

Figura 13: Grafica de numero de iteraciones con una tolerancia de $\varepsilon = 10^{-16}$

$\varepsilon = 10^{-32}$
Perdida de significancia:
Dado el valor conocido de la raíz, lo comparamos con el de una herramienta, tomando como $\varepsilon = 10^{-42}$ obtenemos que la raíz es $x \approx 0.66666666666666666666 6666666666666666667$. La diferencia entre ambas entonces corresponde a: $-3.2042854316314066351545583333333333000000 \times 10^{-6}$. Como en el caso de la primera función, la pérdida de significacia no cambia de forma importante.
Número de iteraciones:
Con un X_0 de 1 son 35 iteraciones para hallar la raiz
Grafica respecto a la tolerancia y al número de iteraciones

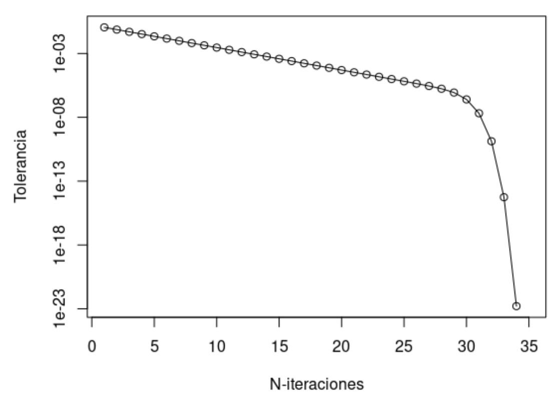

Figura 14: Grafica de numero de iteraciones con una tolerancia de $\varepsilon = 10^{-32}$

1.5 Como se puede solucionar el problema de significancia

Existen diferentes estrategias que buscan reducir este problema, por ejempo sumar números de magnitud pequeña similar antes de sumar números de más grande magnitud. Existe un problema de pérdida de significancia cuando se se substraen números de similar magnitud, esto ocurre cuando se hace la conversión de decimal a binario. Otro escenario que causa alta pérdida de significancia es en la división y la multiplicación, se genera desbodarmiento, para contrarrestar esta situación es recomendado definir una secuencia de operaciones que no modifique el resultado. Es posible también convertir productos en sumas logaritmicas, esto permite conservar cifras significantes y evitar el desbordamiento. A través de las herramientas y paquetes proveidos por el entorno y lenguaje en el que se esté implementando, es posible asignar diferentes tipos de variables con el fin de lograr un manejor adecuado de la pérdida de significancia. Por ejemplo en Python es posible hacer uso del tipo de variable *Decimal* la cual permite almacenar un mayor número de bits que el tipo *Float*. De la misma manera, al momento de indicar la solución es posible establecer el número de dígitos que se desea analizar. Una forma mediante la que se puede estudiar la precisión del resultado (y la pérdida de significancia) es utilizando herramientas de medición, como el paquete *Metrics* de R, conociendo el valor esperado y el resultado es posible encontrar datos representativos sobre cómo se comporta el resultado frente a lo esperado.

1.6 ¿Qué pasa con el método cuándo hay más de una raiz?

Dada la naturaleza del método, dependerá de la primera aproximación, sea indicada por el usuario o indicada por el algoritmo. Cerca a una raíz múltiple la convergencia de este método deja de ser cuadrática y comienza a ser lineal. Esto es beneficio pues si la multiplicidad de una raíz es conocida se puede utilizar la siguiente ecuación para obtener un número de iteraciones menor.

$$x_{i+1} = x_i - m\frac{f(x_i)}{f'(x_i)} \tag{2}$$

1.7 ¿Qué pasa cuando la función es periódica, par o impar, estas características influyen?

Es influyente, pues conociendo la sección anterior es posible entender que si la función es periódica la eficiencia del método mejora, pues al dejar de converger de forma cuadrática (cercano a la solución el número de cifras significativas es aproximadamente el doble que el anterior), se pueden conseguir cambios substanciales en cuanto a complejidad del algoritmo, rendimiento y tiempo de ejecución.

1.8 Gráfica de la relación entre ϵ_{i+1} y ϵ_i

1.8.1 $f(x) = \cos^2(2x) - x^2$

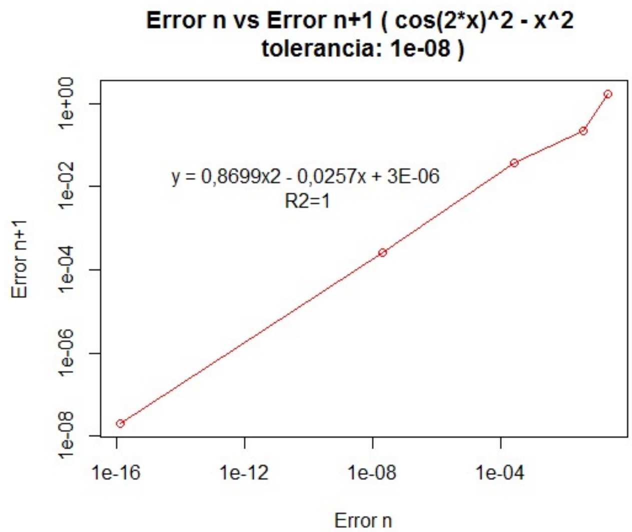

Figura 15: Gráfica de la relación entre ϵ_{i+1} y ϵ_i con una tolerancia de $\varepsilon = 10^{-8}$

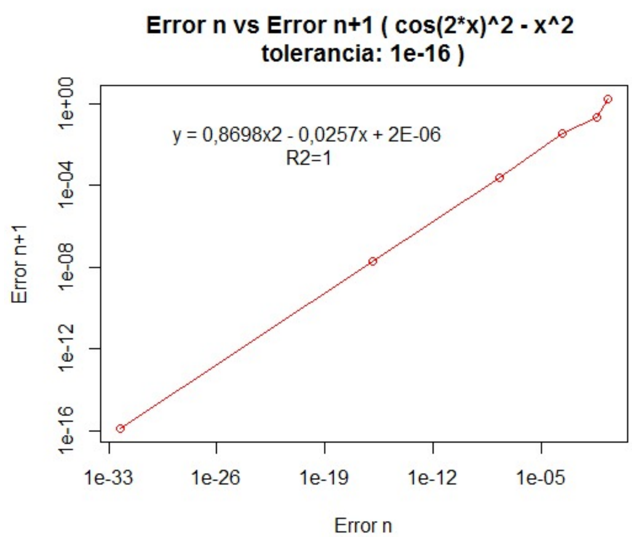

Figura 16: Gráfica de la relación entre ϵ_{i+1} y ϵ_i con una tolerancia de $\varepsilon = 10^{-16}$

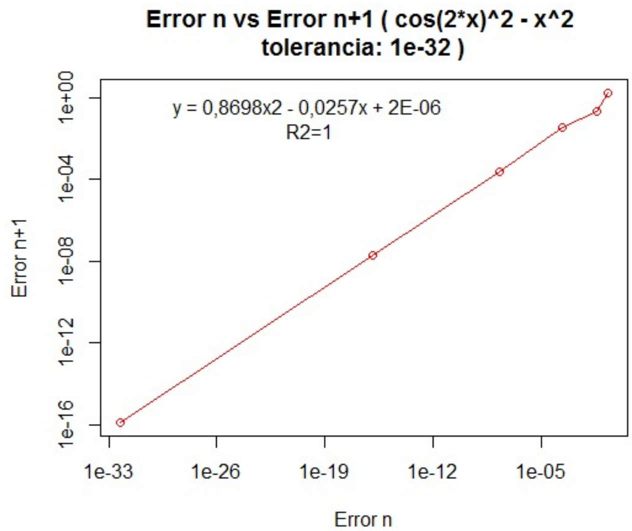

Figura 17: Gráfica de la relación entre ϵ_{i+1} y ϵ_i con una tolerancia de $\varepsilon = 10^{-32}$

1.8.2 $f(x) = x\sin(x) - 1$ $[-1, 2]$

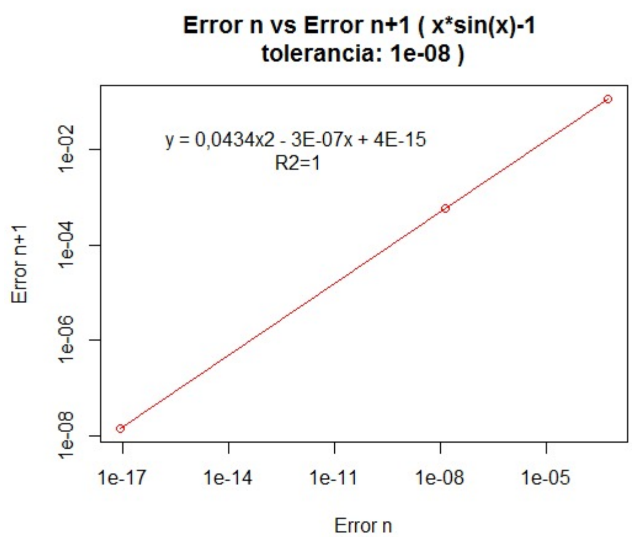

Figura 18: Gráfica de la relación entre ϵ_{i+1} y ϵ_i con una tolerancia de $\varepsilon = 10^{-8}$

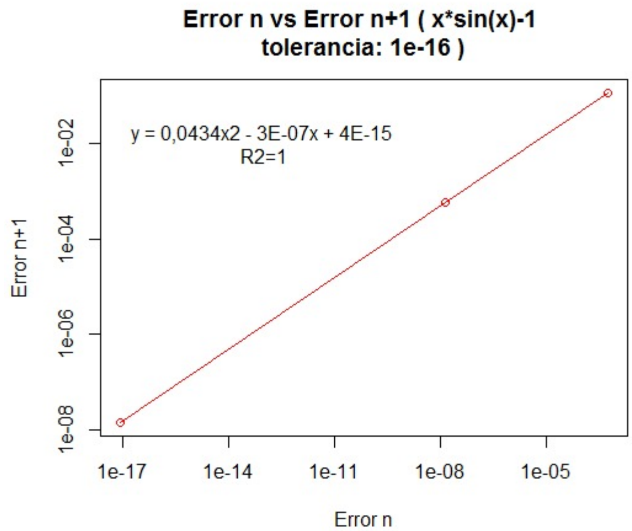

Figura 19: Gráfica de la relación entre ϵ_{i+1} y ϵ_i con una tolerancia de $\varepsilon = 10^{-16}$

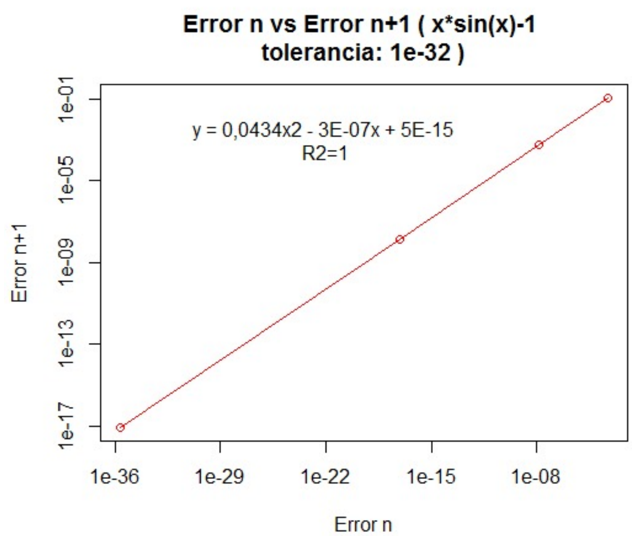

Figura 20: Gráfica de la relación entre ϵ_{i+1} y ϵ_i con una tolerancia de $\varepsilon = 10^{-32}$

1.8.3 $f(x) = x^3 2x^2 + \frac{4}{3}x - \frac{8}{27}$

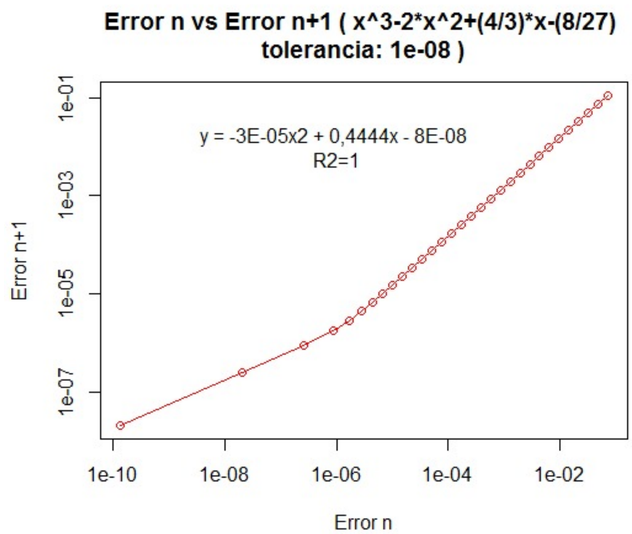

Figura 21: Gráfica de la relación entre ϵ_{i+1} y ϵ_i con una tolerancia de $\varepsilon = 10^{-8}$

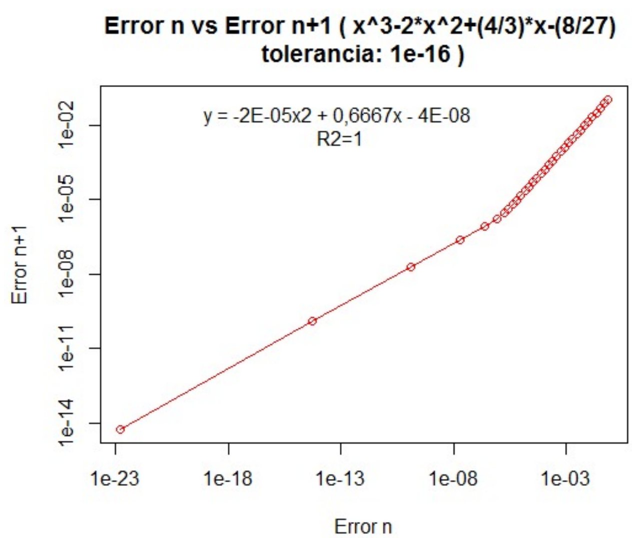

Figura 22: Gráfica de la relación entre ϵ_{i+1} y ϵ_i con una tolerancia de $\varepsilon = 10^{-16}$

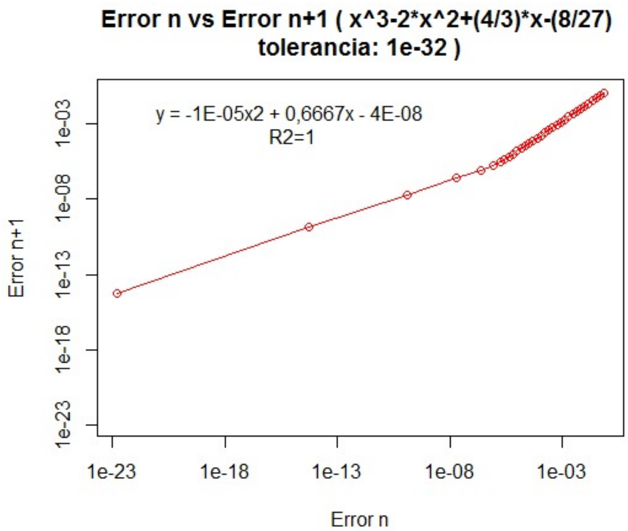

Figura 23: Gráfica de la relación entre ϵ_{i+1} y ϵ_i con una tolerancia de $\varepsilon = 10^{-32}$

1.9 Comportamiento respecto al método de bisección

El método de Newton-Raphson respecto al método de bisección es más preciso y tiene menos casos de error. Así mismo, el error iterativo disminuye rápidamente con cada iteración.

A diferencia del método de bisección, este algoritmo, dado que es un método abierto, no necesita un intervalo para empezar el algoritmo y no tiene requisitos específicos para asegurar el funcionamiento correcto, por tanto la convergencia no está garantizada por un teorema de convergencia global como es en el caso del método de bisección. Así, es necesario partir de una aproximación inicial próxima a la raíz buscada para que el método converja.

1.9.1 $f(x) = \cos^2(2x) - x^2$

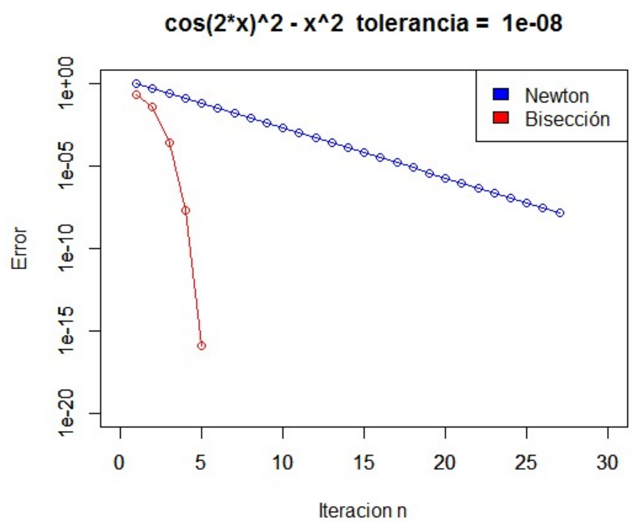

Figura 24: Gráfica de comparacion metodo de Newton-Bisección con una tolerancia de $\varepsilon = 10^{-8}$

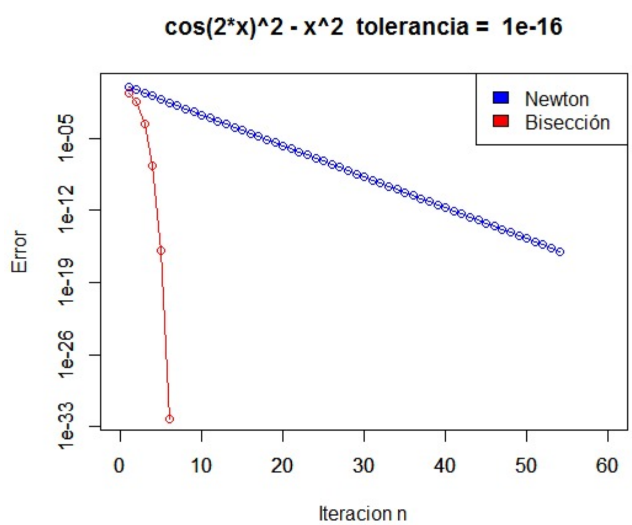

Figura 25: Gráfica de comparacion metodo de Newton-Bisección con una tolerancia de $\varepsilon = 10^{-16}$

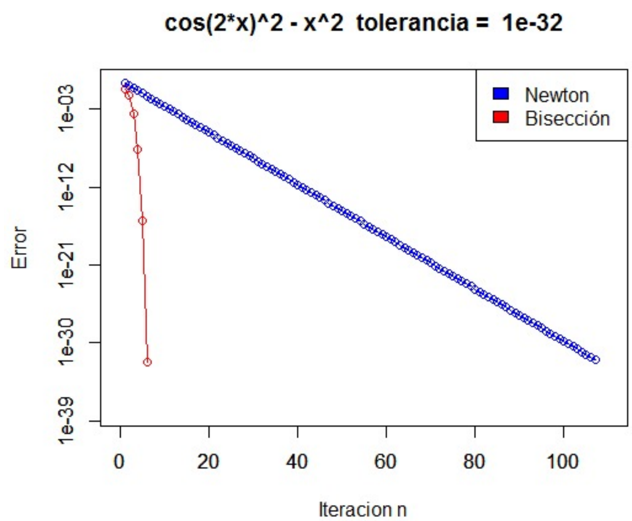

Figura 26: Gráfica de comparacion metodo de Newton-Bisección con una tolerancia de $\varepsilon = 10^{-32}$

1.9.2 $f(x) = x\sin(x) - 1$ $[-1, 2]$

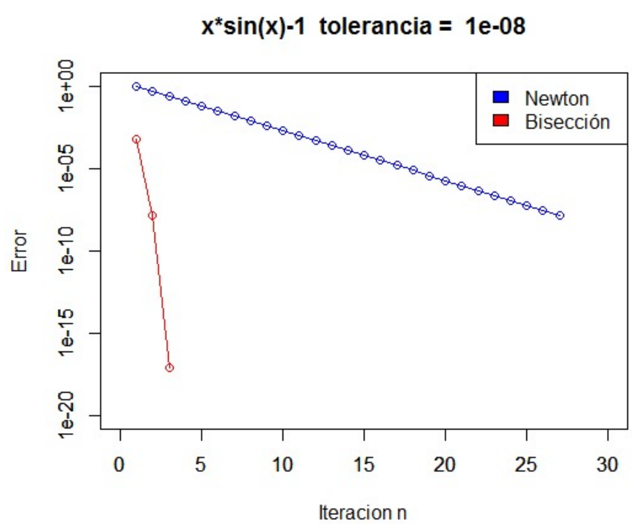

Figura 27: Gráfica de comparacion metodo de Newton-Bisección con una tolerancia de $\varepsilon = 10^{-8}$

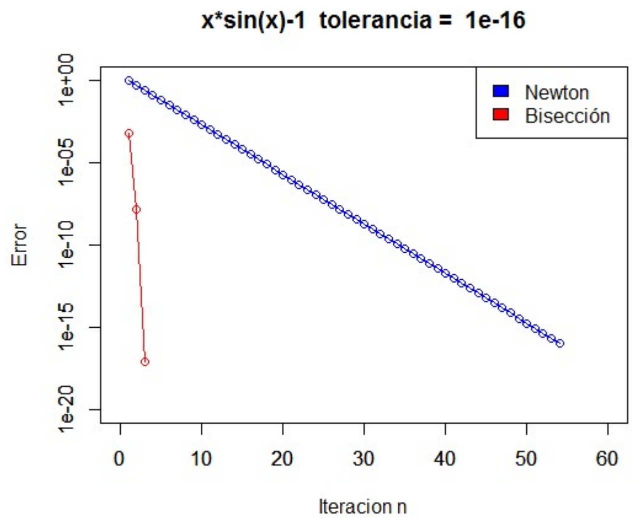

Figura 28: Gráfica de comparacion metodo de Newton-Bisección con una tolerancia de $\varepsilon = 10^{-16}$

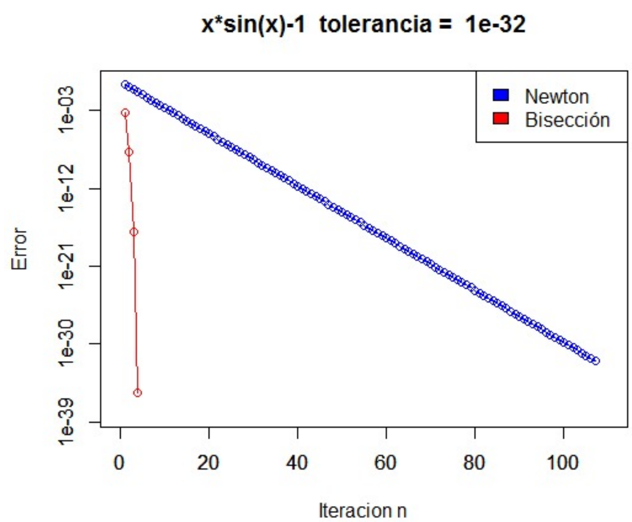

Figura 29: Gráfica de comparacion metodo de Newton-Bisección con una tolerancia de $\varepsilon = 10^{-32}$

1.9.3 $f(x) = x^3 2x^2 + \frac{4}{3}x - \frac{8}{27}$

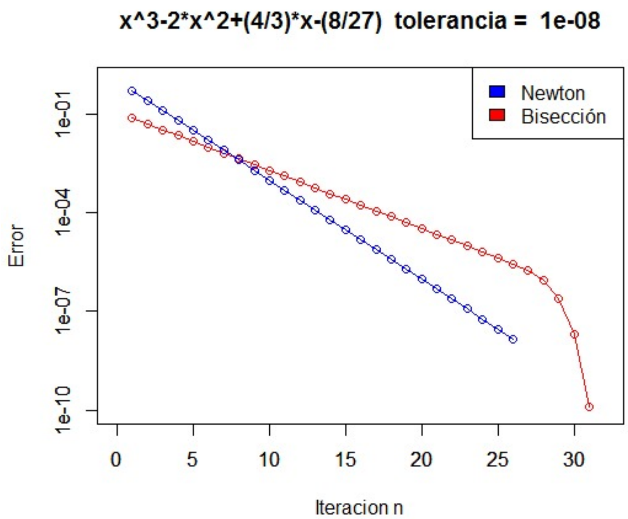

Figura 30: Gráfica de comparacion metodo de Newton-Bisección con una tolerancia de $\varepsilon = 10^{-8}$

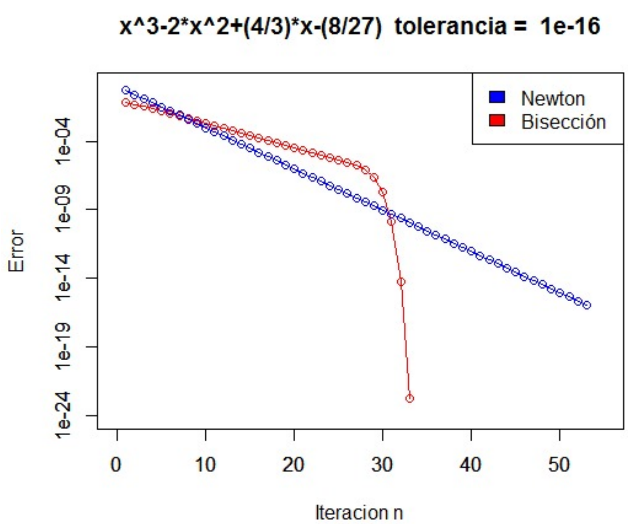

Figura 31: Gráfica de comparacion metodo de Newton-Bisección con una tolerancia de $\varepsilon = 10^{-16}$

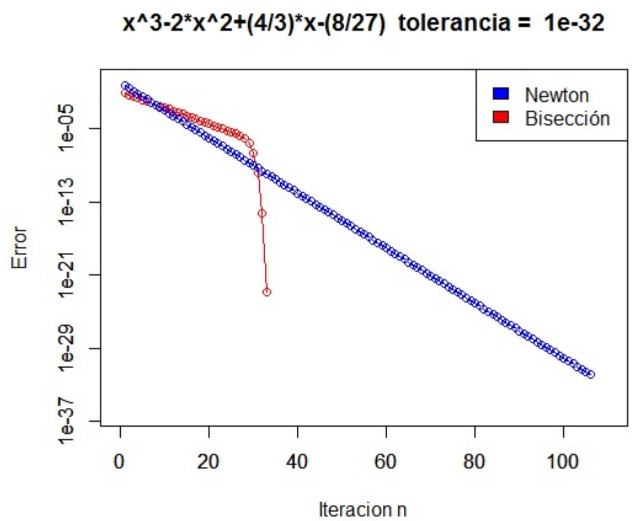

Figura 32: Gráfica de comparacion metodo de Newton-Bisección con una tolerancia de $\varepsilon = 10^{-32}$