Encontrando raíces

Paula Janeth Moreno García

Octubre 2018

1 Introducción

Se hablará de dos métodos para encontrar raíces reales de una función. Siguiendo en cada subsección con el código que le pertenece a cada uno. El primero será el método de bisección, en el cual se toma un intervalo y se aproxima lo más posible a una raíz. El segundo es el método de Newton-Rhapson.

2 Método de Bisección

El método de bisección se utiliza para encontrar raíces de una función. Este método es aplicable para cuando se tiene la función $f(x) = 0$, siendo x una variable real y $f(x)$ una función continua definida en un intervalo $[a, b]$. Este es un intervalo en el que se cree que una de las raíces se encuentra. Se define ahora a la raíz como c. Siendo ésta expresada como $c = \frac{(a+b)}{2}$. Evaluamos ahora a la nueva raíz en $f(x) \to f(c)$.
Si $f(a) * f(n) < 0$ tomamos ahora el intervalo con b=n, y repetimos el proceso.
Si $f(b) * f(n) < 0$ hacemos lo mismo en el intervalo cambiando a=n, y repetimos el proceso.

2.1 Programa

```
PROGRAM BISECCION
IMPLICIT NONE
REAL:: a,b,c,error,f  !a y b son el intervalo en el
 que se cree que la raíz exite, c es la raíz resultante,
error se refiere a qué tan alejado se está de la raíz real y f es la función
error=1.0e-06 !
WRITE(,)"Escribe un intervalo a y b en los que se encuentre la ra?z"
!El usuario ya debería tener una idea de dónde se encuentra al menos una de las raíces
10 READ(,) a,b !Se lee el intervalo escogido
15 IF (f(a)*f(b) .lt. 0) THEN !Si cuando la función evaluada en a
 multiplicada por la función evaluada en b es menor que cero,
entonces podemos obtener c.
c=(a+b)/2.0
```

```
ELSE !Si f(a)*f(b) es mayor que cero, entonces se le pide al lector
que escoja un nuevo intervalo, pues la raíz no se encuentra ahí
WRITE(,)"Intenta con otro valor de a y b"
GO TO  10 !Volver a leer los valores de a y b
END IF !termina la condición
IF (f(a)*f(c) .lt. 0) THEN !comienza nueva condicion,
si f(a)*f(c) es menor que cero entonces el nuevo intervalo es de a,c
b=c !se cambia c por b
ELSE !Si f(a)*f(c) es mayor que cero, entonces se cambia a por c
a=c
END IF
IF (abs(b-a) .gt. error) GO TO 15
WRITE(,)"La ra?z es",c !Escribir la raíz
END PROGRAM
real function f(x) !se define la función real de la
que queremos saber sus raíces
implicit none
real::x !variable real "x"
f=3*x+sin(x)-exp(x) !función
end function
```

3 Método de Newton-Raphson

Este método es uno de los mas utilizados para localizar raíces ya que en general es muy eficiente y siempre converge para una función polinomial.

Se requiere que las funciones sean diferenciables, y por tanto, continuas, para poder aplicar este método.

Se debe partir de un valor inicial para la raíz: xi , este puede ser cualquier valor, el método convergirá a la raíz mas cercana.

Si se extiende una tangente desde el punto b , el punto donde esta tangente cruza al eje x representa una aproximación mejorada de la raíz.

Para mejorar la aproximación se utiliza la siguiente iteración, a mayor número de itraciones, mayor la aproximación:

$$x_1 = x_0 - \frac{f(x)}{f'(x)} \tag{1}$$

x_1 es nuestra nueva aproximación a la raíz, x_0 será sustituida por x_1 en nuestra nueva aproximación. Este proceso se llevará a cabo el número de veces necesario hasta llegar al margen de error deseado.

3.1 Programa

```
Program NewtonRap

IMPLICIT NONE
```

```
!Declaramos las siguientes variables
!x: será una mejor a proximación a la raiz
!x1: un punto en el eje x cercano a la raiz
!error: corresponde a un porcentaje deseado para la exactitud de la raiz
!f: es para la función que se desea aproximar su raiz
!f1: es la derivada de f

REAL:: x,x1,error,f,f1

!Pedimo al usario ingrese nuestro punto cercano a la raiz x1 y lo leemos
WRITE(*,*)'Ingresa un punto cerca de la raiz'
READ(*,*) x1

!Usaremos un loop para llevar acabo la operación las veces necesarias
!El loop será deteneido por un if en donde se muestra que el error debe de ser
!menor a un porcentaje ya señalado
DO
   !Se evalua la función y su derivada en x1
   f=x1**3-x1-2
   f1=3*x1**2-1
   !Aquí tenemos el método de Newton que nos da una mejor aproximación a
   !nuestra raiz con la siguiente operación
   x=x1-(f/f1)
   !Definimos el error
   error= abs((x-x1)/x)*100
   x1=x
   !Si este if cumple con lo señalado entonces se detendrá el loop, de otra
   !forma sería infinito
   if(error<0.0000001)exit
END DO

!Imprimimos en pantanlla el valor de x correspondiente a la raiz
PRINT*,'La raiz es:',x

END PROGRAM NewtonRap
```

4 Conclusión

Para concluir, podemos decir que los dos métodos son muy útiles para encontrar raíces de forma manual y fácil.