

METODO DE RAICES MULTIPLES

Objetivo del método

Encontrar una raíz a partir de un valor inicial, teniendo en cuenta una tolerancia y un número máximo de iteraciones. En este caso no es necesario tener un intervalo.

Generalidades

Para garantizar la convergencia del método de Newton-Raphson una de las condiciones que $f'(x) \neq 0$. Si se observa que al ejecutar Newton-Raphson $f'(x)$ se aproxima a cero. La rapidez del método disminuye y hay posibilidades de que haya una raíz múltiple.

El método de raíces múltiples es muy similar a método de Newton-Raphson con la particularidad de que en la estructura se debe de hallar la segunda deriva de la función a la cual se le está hallando la raíz. Esto permite tener una mayor seguridad y rapidez en la convergencia. Y se debe tomar en cuenta la expresión:

$$x_{n+1} = \frac{f(x_n)f'(x_n)}{f'(x_n)^2 - f(x_n)f''(x_n)}$$

Teniendo definida la expresión con la cual hallaremos la raíz.

- Se debe tomar el valor inicial "a"
- Calcular x_1 y continuar hasta x_n , hasta llegar a la aproximación de la raíz.

Pseudocodigo de Raices Múltiples

Leer a, b, n, delta, tole

Contador = 0

Valorfunción = delta + 1

Error = tole + 1

q = a

mientras contador < n & valorfunción > delta & error > tole & d2funcion(a)≠0

hacer

contador = contador + 1

tabla (contador, 1) = contador

$x = a - \frac{((funcion(a)) * (dfuncion(a)))}{((dfuncion(a))^2 - (funcion(a) * d2funcion(a)))}$;

%Ecuacion de Newton modificado

tabla (contador, 2) = x

y = función(x)

tabla (contador, 3) = y

valorfunción = valor absoluto (y)

error = valor absoluto (x - q)

tabla (contador, 4) = error

a=x

fin mientras

Código

```
function [ tabla ] = raicesmultiples (a, n, delta, tole)

format long

%Criterios de inicializacion

i=0;
funcionvalue=delta+1;
error= tole+1;

%Programa de Newton Modificado

    while i<n & funcionvalue>delta & error>tole & d2funcion(a)~=0

        i=i+1; %Numero de iteraciones a realizar
        tabla(i,1)=i;

            x= a - (((funcion(a))*(dfuncion(a)))/(((dfuncion(a))^2-
(funcion(a)*d2funcion(a)))); %Ecuacion de Newton modificado
            tabla(i,2)= x;

                y=funcion(x); %evaluar la variable x en la funcion
                tabla(i,3)=y;

                    funcionvalue=abs(y);
                    error= abs(x-a); % cálculo del error
                    %error=abs(x-a)/abs(x); %error relativo
                    %error=(abs(x-a)/abs(x))*100; %error porcentual
                    tabla(i,4)=error;

                        a=x; %conteo para poder calcular el error

                    endwhile

endfunction
```

