



UNIVERSIDAD AUTÓNOMA DE QUERÉTARO
FACULTAD DE INGENIERÍA

REPORTE DE ALGORITMOS

FALSA POSICION

Nombre	Expediente
Zuñiga Fragoso Diego Joel	317684

Asignatura: Método Numéricos 2023-2

Docente: Vargas Vázquez Damián



I. Antecedentes teóricos

El método de la Falsa Posición, también conocido como método de interpolación lineal, es un algoritmo numérico utilizado para encontrar raíces de funciones continuas en un intervalo dado. Aquí están algunos antecedentes teóricos clave:

1. Interpolación Lineal:

El método de la Falsa Posición utiliza la interpolación lineal para aproximar la ubicación de la raíz. En lugar de dividir el intervalo a la mitad, como en el método de bisección, este método utiliza una línea secante para estimar la posición de la raíz.

2. Teorema del Valor Intermedio:

Al igual que otros métodos para encontrar raíces, el método de la Falsa Posición se basa en el teorema del valor intermedio. Este teorema establece que si una función es continua en un intervalo y cambia de signo en ese intervalo, entonces debe haber al menos una raíz en ese intervalo.

3. Condiciones de Convergencia

La convergencia del método de la Falsa Posición está condicionada a la continuidad de la función en el intervalo dado y a la existencia de una raíz en ese intervalo. Además, se requiere que la función cambie de signo en los extremos del intervalo.

4. Selección de Puntos Iniciales:

La eficacia del método depende en gran medida de la elección adecuada de los puntos iniciales. Estos puntos deben garantizar que la función cambie de signo en el intervalo inicial.

5. Iteraciones:

El método de la Falsa Posición utiliza una fórmula iterativa para actualizar los extremos del intervalo en cada paso. La elección de la nueva aproximación se realiza considerando la intersección de la línea secante con el eje x .



6. Ventajas y Limitaciones:

Este método puede ser más rápido que la bisección en términos de convergencia, pero puede ser más propenso a oscilaciones y no siempre converge. La elección de los puntos iniciales es crítica para su éxito.

II. Algoritmos y sus resultados

Cada algoritmo esta seccionado e incluye descripciones de lo que sucede. Además de contar con capturas de sus resultados

Falsa Posición

```
%% Ingreso de datos y declaracion de variables
syms x;

% Entrada de datos
fx = input('Ingrese la función f(x): ');
xl = input('Ingrese el valor inferior: ');
xu = input('Ingrese el valor superior: ');

% Entrada de error estimado e iteraciones
errorest = input('Ingrese el error estimado: ');
imax = input('Número de iteraciones: ');

disp('-----');

% Para la gráfica
xio = xl - 0.5;
xuo = xu + 0.5;

% Variables para la gráfica
vxr = [];
vfxr = [];

% Evaluación de fxl y fxu
fxl = subs(fx, x, xl);
fxu = subs(fx, x, xu);

met = fxl * fxu;

%% Comprobacion de compatibilidad
if met >= 0
    error('No cumple con las condiciones para llevarse a cabo.');
```

end

```
%% Bucle principal de Falsa Poscion

erroraprox = 100;
```

```
i = 0;
xro = 0;

while (erroraprox > errorest) && (i < imax)
    i = i + 1;
    xr = xu - (fxu * (xl - xu)) / (fxl - fxu);
    erroraprox = (abs((xr - xro) / xr)) * 100;
    xro = xr;

    % Mostrar información de la iteración
    fprintf('Iteración %i: \n', i)
    fprintf('xl = %2.6f   xu = %2.6f   xr = %2.6f \n', xl, xu, xr)
    fprintf('Ea = %2.6f \n', erroraprox)
    disp('-----');

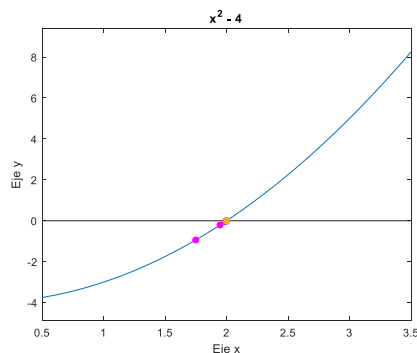
    % Evaluar fxr y almacenar valores para la gráfica
    fxr = subs(fx, x, xr);
    vxr(i) = xr;
    vfxr(i) = fxr;

    % Actualizar xl y xu
    if (fxl * fxr) < 0
        xu = xr;
    elseif (fxl * fxr) > 0
        xl = xr;
    end

    % Actualizar fxl y fxu
    fxl = subs(fx, x, xl);
    fxu = subs(fx, x, xu);
end

%% Graficado
ezplot(fx, [xio, xuo])
hold on
plot(vxr, vfxr, 'm*', 'LineWidth', 2)
hold on
stem(xr, fxr, 'filled')
xlabel('Eje x');
ylabel('Eje y');
```

Resultado





```
>> Falsa_posicion
Ingrese la función f(x): x^2 - 4
Ingrese el valor inferior: 1
Ingrese el valor superior: 3
Ingrese el error estimado: 0.01
Número de iteraciones: 15

-----

Iteración 1:
xl = 1.000000    xu = 3.000000    xr = 1.750000
Ea = 100.000000

-----

Iteración 2:
xl = 1.750000    xu = 3.000000    xr = 1.947368
Ea = 10.135135

-----

Iteración 3:
xl = 1.947368    xu = 3.000000    xr = 1.989362
Ea = 2.110892

-----

Iteración 4:
xl = 1.989362    xu = 3.000000    xr = 1.997868
Ea = 0.425759

-----

Iteración 5:
xl = 1.997868    xu = 3.000000    xr = 1.999573
Ea = 0.085297

-----

Iteración 6:
xl = 1.999573    xu = 3.000000    xr = 1.999915
Ea = 0.017065

-----

Iteración 7:
xl = 1.999915    xu = 3.000000    xr = 1.999983
Ea = 0.003413

-----
```

III. Conclusiones

En conclusión, el método de la Falsa Posición se revela como una herramienta interesante y versátil en la búsqueda de raíces de funciones continuas. Al aprovechar la interpolación lineal, este método ofrece una alternativa dinámica a la bisección, buscando acelerar la convergencia hacia la solución deseada.