

# **Métodos Numéricos**

## **Actividad Final - 1er Cohorte - Grupo A:**

Método de la regla falsa

### **Presentado por:**

Sebastián Ricardo Cárdenas

Fabián Alberto Sánchez Ruiz

### **Presentado a:**

JUAN MANUEL MUSKUS MUSKUS

**Universidad de Córdoba**

**Facultad de Ingeniería**

**Montería – 2020**

```

%-----Actividad Final - 1er Cohorte - Grupo A-----
%-----Métodos Numéricos-----
%
% El siguiente script fue realizado para la actividad final del 1er
% corte de la materia de métodos numéricos. Sirve para hallar la raíz
% de una función entre un intervalo y con un margen de error, con el
% método de la regla falsa. La función, el intervalo y el margen de
% error son introducidos por el usuario. Todos los puntos de la
% actividad fueron realizados con el siguiente código.
%
% Grupo conformado por:
%     -Sebastián Ricardo Cárdenas
%     -Fabián Alberto Sánchez Ruiz
%
%-----
%-----

clc
clear all

%Pedimos la función
w = input("Ingrese su funcion: ", "S");
%Convertimos en función
fuc = inline(w);
%pedimos los datos //
disp("Ingrese el intervalo: ");
a = input("Ingrese el primer valor del intervalo: ");
b = input("Ingrese el segundo valor del intervalo: ");
e_usuario = input("ingrese el error inicial: ");
%//
%Evalúamos los puntos en la función
fa = fuc(a);
fb = fuc(b);
%Variable iteradora y Xr primero
ite = 1;
xr=0;

%Evalúamos
if (fa*fb<0)
    fprintf("i      Xn-1      Xn      f(Xn)      |Ea|\n");
    while(true)
        %Hallamos la xr y evaluamos xr
        xr = b - ((fb)*(a-b)/(fa - fb));
        fxr = fuc(xr);
        %verificamos que iteración es.
        if (ite < 2)
            %hacemos el cambio de coordenadas
            if (fa*fxr<0)
                b = xr;
                xp = b;
            else
                if (fa*fxr>0)
                    a = xr;
                    xp = a;
                end
            end
            %añadimos a la tabla de datos y sumamos 1na iteración

```

```

        fprintf("%d    %0.5f    %0.5f    %0.5f    ---- \n",ite, xp, xr, fxr);
        ite = ite + 1;
    else
        %Empezamos a calcular los errores
        Ea = abs(((xr-xp)/xr))*100;
        %Verificamos si cumplimos con la condición del error del usuario
        if(Ea<=e_usuario)
            break;
        else
            %Seguimos partiendo el intervalo
            if (fa*fxr<0)
                b = xr;
                xp = b;
            else
                if (fa*fxr>0)
                    a = xr;
                    xp = a;
                end
            end
        end
        %Añadimos a la tabla y sumamos una iteración
        fprintf("%d    %0.5f    %0.5f    %0.5f    %0.5f\n",ite, xp, xr, fxr, Ea);
        ite = ite + 1;
    end
    %pause(1);
end
fprintf("Finalmente tenemos la raiz: Xr = %0.7f",xr);
else
disp("No mijito, no se puede");
end

```

1. Usando el método de la regla falsa encuentre la raíz aproximada de:

$$f(x) = \sqrt{x} - \cos(x)$$

Iniciando en el intervalo  $[0, 1]$  y hasta que

$$|Ea| < 10^{-5}$$

```
Command Window
Ingrese su funcion: sqrt(x)-cos(x)
Ingrese el intervalo:
Ingrese el primer valor del intervalo: 0
Ingrese el segundo valor del intervalo: 1
fx ingrese el error inicial: 10^-5
```

En el ejercicio, se ingresó al script hecho en Matlab, la función  $\sqrt{x} - \cos(x)$  en el intervalo  $[0,1]$  con un margen de error menor a  $10^{-5}$  y podemos apreciar que converge a  $x_r = 0.641722$

```
Command Window
Ingrese su funcion: sqrt(x)-cos(x)
Ingrese el intervalo:
Ingrese el primer valor del intervalo: 0
Ingrese el segundo valor del intervalo: 1
ingrese el error inicial: 10^-5
i   Xn-1      Xn      f(Xn)      |Ea|
1   0.68507   0.68507   0.05332   ----
2   0.46933   0.46933   -0.20680   45.96977
3   0.61713   0.61713   -0.02997   23.95013
4   0.66368   0.66368   0.02693    7.01354
5   0.64902   0.64902   0.00894    2.25864
6   0.63897   0.63897   -0.00335    1.57165
7   0.64585   0.64585   0.00506    1.06523
8   0.64369   0.64369   0.00241    0.33660
9   0.64220   0.64220   0.00060    0.23113
10  0.64119   0.64119   -0.00065    0.15859
11  0.64188   0.64188   0.00021    0.10853
12  0.64166   0.64166   -0.00006    0.03419
13  0.64181   0.64181   0.00012    0.02342
14  0.64177   0.64177   0.00006    0.00738
15  0.64173   0.64173   0.00002    0.00505
16  0.64171   0.64171   -0.00000    0.00346
17  0.64173   0.64173   0.00002    0.00237
18  0.64172   0.64172   0.00001    0.00075
19  0.64172   0.64172   0.00001    0.00051
20  0.64172   0.64172   0.00000    0.00035
21  0.64172   0.64172   0.00000    0.00024
22  0.64171   0.64171   -0.00000    0.00016 |
23  0.64171   0.64171   0.00000    0.00011
24  0.64171   0.64171   0.00000    0.00004
25  0.64171   0.64171   0.00000    0.00002
26  0.64171   0.64171   0.00000    0.00002
27  0.64171   0.64171   0.00000    0.00001
fx Finalmente tenemos la raiz: Xr = 0.6417143>>
```

2. A que raíz de:

$$f(x) = (x + 2)(x + 1)2x(x - 1)3(x - 2)$$

¿Converge el método de la regla falsa cuando se aplica en los siguientes intervalos?

- a. [-3, 2.5]
- b. [-2.5, 3]
- c. [-1.75, 1.5]
- d. [-1.5, 1.75]

Procedemos a aplicar al script la función en cada uno de los intervalos para comprobar si en alguno de ellos se encuentra una raíz.

a.

```
Command Window
Ingrese su funcion: (x+2).*((x+1).^2).*x.*((x-1).^3).*(x-2)
Ingrese el intervalo:
Ingrese el primer valor del intervalo: -3
Ingrese el segundo valor del intervalo: 2.5
ingrese el error inicial: 10^-5
No hay raices
fx >>
```

b.

```
Command Window
Ingrese su funcion: (x+2).*((x+1).^2).*x.*((x-1).^3).*(x-2)
Ingrese el intervalo:
Ingrese el primer valor del intervalo: -2.5
Ingrese el segundo valor del intervalo: 3
ingrese el error inicial: 10^-5
No hay raices
fx >>
```

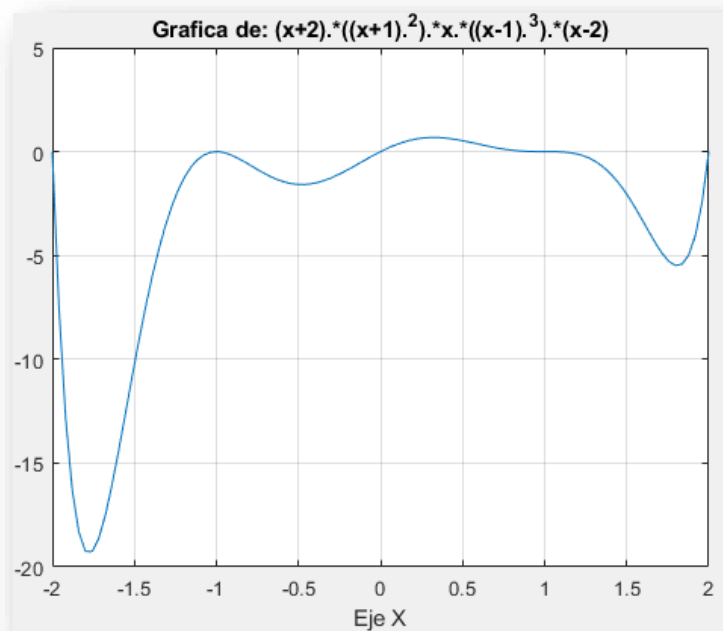
c.

```
Command Window
Ingrese su funcion: (x+2).*((x+1).^2).*x.*((x-1).^3).*(x-2)
Ingrese el intervalo:
Ingrese el primer valor del intervalo: -1.75
Ingrese el segundo valor del intervalo: 1.5
ingrese el error inicial: 10^-5
No hay raices
fx >>
```

d.

```
Command Window
Ingrese su funcion: (x+2).*((x+1).^2).*x.*((x-1).^3).*(x-2)
Ingrese el intervalo:
Ingrese el primer valor del intervalo: -1.5
Ingrese el segundo valor del intervalo: 1.75
ingrese el error inicial: 10^-5
No hay raices
fx >> |
```

Análisis: En ninguno de los casos el método converge hacia una raíz, ya que en los intervalos nos encontramos mas de 1 raíz, por lo cual, el método no llega a ninguna de estas.



3. Tanques del occidente, elabora boyas para depósitos de líquidos. Las boyas son cuerpos esféricos que tienen una gravedad específica de 0.6 y un radio de 0,5 cm. Se debe calcular la profundidad a la que se sumerge la boya cuando flotan en el líquido almacenado. La ecuación que da la profundidad de  $x$  a la cual la boya se encuentra sumergida bajo el líquido es:

$$f(x) = x^3 + 0.165x^2 - 3.993 * 10^{-4}$$

Análisis: Como  $x$  es la cantidad sumergida de la boya en el líquido, sabemos que parte desde 0, es decir  $x = 0$ , es el valor más bajo, ya que no es posible que la boya se sumerja con valores negativos, entonces empezamos a buscar la raíz en el intervalo de  $[0, \infty)$   
Comenzamos a buscar la raíz con un intervalo pequeño, puede ser  $[0, 10]$

Capturas:

Probamos con el intervalo  $[0,10]$

```
Command Window
11135 0.04364 0.04364 -0.00000 0.00896
11136 0.04365 0.04365 -0.00000 0.00896
11137 0.04365 0.04365 -0.00000 0.00896
11138 0.04366 0.04366 -0.00000 0.00896
11139 0.04366 0.04366 -0.00000 0.00896
11140 0.04366 0.04366 -0.00000 0.00896
11141 0.04367 0.04367 -0.00000 0.00896
11142 0.04367 0.04367 -0.00000 0.00896
11143 0.04368 0.04368 -0.00000 0.00895
11144 0.04368 0.04368 -0.00000 0.00895
11145 0.04368 0.04368 -0.00000 0.00895
11146 0.04369 0.04369 -0.00000 0.00895
11147 0.04369 0.04369 -0.00000 0.00895
11148 0.04370 0.04370 -0.00000 0.00895
11149 0.04370 0.04370 -0.00000 0.00895
11150 0.04370 0.04370 -0.00000 0.00895
11151 0.04371 0.04371 -0.00000 0.00895
11152 0.04371 0.04371 -0.00000 0.00895
11153 0.04372 0.04372 -0.00000 0.00895
11154 0.04372 0.04372 -0.00000 0.00895
11155 0.04372 0.04372 -0.00000 0.00894
11156 0.04373 0.04373 -0.00000 0.00894
11157 0.04373 0.04373 -0.00000 0.00894
11158 0.04373 0.04373 -0.00000 0.00894
11159 0.04374 0.04374 0.00000 0.00894
11160 0.04373 0.04373 -0.00000 0.00894
fx Finalmente tenemos la raíz: Xr = 0.0437348>>
```

Probamos con el intervalo  $[0, 5]$

```
Command Window
2818 0.04338 0.04338 -0.00001 0.03533
2819 0.04340 0.04340 -0.00001 0.03532
2820 0.04341 0.04341 -0.00001 0.03531
2821 0.04343 0.04343 -0.00001 0.03529
2822 0.04344 0.04344 -0.00001 0.03528
2823 0.04346 0.04346 -0.00001 0.03527
2824 0.04347 0.04347 -0.00001 0.03526
2825 0.04349 0.04349 -0.00000 0.03524
2826 0.04350 0.04350 -0.00000 0.03523
2827 0.04352 0.04352 -0.00000 0.03522
2828 0.04354 0.04354 -0.00000 0.03521
2829 0.04355 0.04355 -0.00000 0.03519
2830 0.04357 0.04357 -0.00000 0.03518
2831 0.04358 0.04358 -0.00000 0.03517
2832 0.04360 0.04360 -0.00000 0.03516
2833 0.04361 0.04361 -0.00000 0.03514
2834 0.04363 0.04363 -0.00000 0.03513
2835 0.04364 0.04364 -0.00000 0.03512
2836 0.04366 0.04366 -0.00000 0.03511
2837 0.04367 0.04367 -0.00000 0.03509
2838 0.04369 0.04369 -0.00000 0.03508
2839 0.04370 0.04370 -0.00000 0.03507
2840 0.04372 0.04372 -0.00000 0.03506
2841 0.04373 0.04373 -0.00000 0.03504
2842 0.04375 0.04375 0.00000 0.03503
2843 0.04373 0.04373 -0.00000 0.03504
fx Finalmente tenemos la raiz: Xr = 0.0437345>>
```

Probamos con el intervalo  $[0, 2.5]$

```
Command Window
713 0.04237 0.04237 -0.00003 0.13906
714 0.04243 0.04243 -0.00003 0.13886
715 0.04249 0.04249 -0.00002 0.13867
716 0.04255 0.04255 -0.00002 0.13847
717 0.04260 0.04260 -0.00002 0.13828
718 0.04266 0.04266 -0.00002 0.13808
719 0.04272 0.04272 -0.00002 0.13789
720 0.04278 0.04278 -0.00002 0.13770
721 0.04284 0.04284 -0.00002 0.13750
722 0.04290 0.04290 -0.00002 0.13731
723 0.04296 0.04296 -0.00002 0.13712
724 0.04302 0.04302 -0.00001 0.13693
725 0.04308 0.04308 -0.00001 0.13674
726 0.04314 0.04314 -0.00001 0.13655
727 0.04319 0.04319 -0.00001 0.13636
728 0.04325 0.04325 -0.00001 0.13617
729 0.04331 0.04331 -0.00001 0.13598
730 0.04337 0.04337 -0.00001 0.13579
731 0.04343 0.04343 -0.00001 0.13561
732 0.04349 0.04349 -0.00000 0.13542
733 0.04355 0.04355 -0.00000 0.13523
734 0.04361 0.04361 -0.00000 0.13505
735 0.04367 0.04367 -0.00000 0.13486
736 0.04372 0.04372 -0.00000 0.13468
737 0.04378 0.04378 0.00000 0.13449
738 0.04372 0.04372 -0.00000 0.13467
fx Finalmente tenemos la raiz: Xr = 0.0437239>> |
```



Probamos con el intervalo  $[0, 1]$

```
Command Window
1905  0.04374  0.04374  -0.00000  0.00014
1906  0.04374  0.04374  -0.00000  0.00014
1907  0.04374  0.04374  -0.00000  0.00014
1908  0.04374  0.04374  -0.00000  0.00014
1909  0.04374  0.04374  -0.00000  0.00014
1910  0.04374  0.04374  -0.00000  0.00014
1911  0.04374  0.04374  -0.00000  0.00014
1912  0.04374  0.04374  -0.00000  0.00014
1913  0.04374  0.04374  -0.00000  0.00014
1914  0.04374  0.04374  -0.00000  0.00014
1915  0.04374  0.04374  -0.00000  0.00014
1916  0.04374  0.04374  -0.00000  0.00014
1917  0.04374  0.04374  -0.00000  0.00014
1918  0.04374  0.04374  -0.00000  0.00014
1919  0.04374  0.04374  -0.00000  0.00014
1920  0.04374  0.04374  -0.00000  0.00014
1921  0.04374  0.04374  -0.00000  0.00014
1922  0.04374  0.04374  -0.00000  0.00014
1923  0.04374  0.04374  -0.00000  0.00014
1924  0.04374  0.04374  -0.00000  0.00014
1925  0.04374  0.04374  -0.00000  0.00014
1926  0.04374  0.04374  -0.00000  0.00014
1927  0.04374  0.04374  -0.00000  0.00014
1928  0.04374  0.04374  -0.00000  0.00014
1929  0.04374  0.04374  0.00000  0.00014
1930  0.04374  0.04374  -0.00000  0.00014
fx Finalmente tenemos la raíz: Xr = 0.0437371>>
```

Probamos con el intervalo  $[0, 0.1]$

```
Command Window
Ingrese su funcion: x.^3 + 0.165.*x.^2 - 3.993 .*10^-4
Ingrese el intervalo:
Ingrese el primer valor del intervalo: 0
Ingrese el segundo valor del intervalo: 0.1
Ingrese el error inicial: 10^-4
i   Xn-1   Xn       f(Xn)   |Ea|
1   0.01507 0.01507  -0.00036  ---- |
2   0.02787 0.02787  -0.00025  45.92609
3   0.03873 0.03873  -0.00009  28.06065
4   0.04797 0.04797  0.00009  19.24575
5   0.04013 0.04013  -0.00007  19.53975
6   0.04131 0.04131  -0.00005  2.86003
7   0.04231 0.04231  -0.00003  2.37148
8   0.04316 0.04316  -0.00001  1.97438
9   0.04389 0.04389  0.00000  1.64922
10  0.04327 0.04327  -0.00001  1.42062
11  0.04336 0.04336  -0.00001  0.21360
12  0.04344 0.04344  -0.00001  0.18109
13  0.04351 0.04351  -0.00000  0.15356
14  0.04357 0.04357  -0.00000  0.13026
15  0.04361 0.04361  -0.00000  0.11051
16  0.04366 0.04366  -0.00000  0.09377
17  0.04369 0.04369  -0.00000  0.07958
18  0.04372 0.04372  -0.00000  0.06754
19  0.04374 0.04374  0.00000  0.05733
20  0.04372 0.04372  -0.00000  0.04872
21  0.04373 0.04373  -0.00000  0.00734
22  0.04373 0.04373  -0.00000  0.00623
23  0.04373 0.04373  -0.00000  0.00529
24  0.04373 0.04373  -0.00000  0.00450
25  0.04374 0.04374  -0.00000  0.00382
26  0.04374 0.04374  -0.00000  0.00324
27  0.04374 0.04374  0.00000  0.00275
28  0.04374 0.04374  0.00000  0.00234
29  0.04374 0.04374  -0.00000  0.00035
fx Finalmente tenemos la raíz: Xr = 0.0437370>>
```

Observaciones: Se aprecia que el método converge hacia la raíz  $x_r = 0.043737$