

Computación

Aula Virtual: <https://famaf.aulavirtual.unc.edu.ar/course/view.php?id=747>

Resguardo tutoriales: <https://www.famaf.unc.edu.ar/~moreschi/docencia/Computacion/>

Tutorial Problema 6 de la Guía N° 4

Problema 6:

Grafique el error en la determinación de la raíces del ejercicio anterior en función del número de iteraciones para un máximo de 20 iteraciones. Tome como valor verdadero el obtenido por el método de Newton.

Tutorial:

- Guarde en el archivo p6.py las siguientes instrucciones:

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # Función cuya raíz queremos hallar
5 def f(x):
6     return 3*x - np.tan(x)
7
8 a = 0.1
9 b = 1.5
10 xx=[]
11 yy=[]
12
13 # Tolerancia para el intervalo que contiene la raíz
14 epsilon = 1.0e-6
15 #epsilon = 1.0e-12
16
17 intervalo = b - a
18 n = 0
19 while intervalo > epsilon:
20     n += 1
21     c = a + 0.5*intervalo
22     xx.append(c)
23     yy.append(f(c))
24     if f(c)*f(a) < 0:
25         b = c
26     else:
27         a = c
28
29     intervalo = b - a
30
31 # Aproximación a la raíz (punto medio del intervalo final)
32 c = a + 0.5*intervalo
33
34 print ("\nLa solución de '3x - tan(x)' es: %27.20e" % (c))
35 print ("\nEl programa realizó %d iteraciones de bisección\n" % (n))
```

```

36
37 x_f = 1.32419444957563481324e+00 # calculado con epsilon = 1.0e-12
38
39 x = np.linspace(0,np.pi,101)
40 y = 3*x - np.tan(x)
41
42 plt.figure( figsize=(10, 7.5) )
43 plt.title('Errores en las iteraciones para la Función "3x - tan(x)" ')
44 plt.xlabel('número de iteraciones')
45 plt.ylabel('errores')
46 plt.grid()
47 for i in range(len(xx)):
48     plt.semilogy(i,abs(xx[i]-x_f),'o',label=i)
49 plt.legend(loc="lower left")
50 plt.savefig('graficos/p6.png', dpi=100)
51 plt.show()

```

- Desde la terminal ejecute:

python3 p6.py

e interprete el resultado.

Alternativamente ejecute:

python3

y vaya agregando uno a uno los bloques del programa.

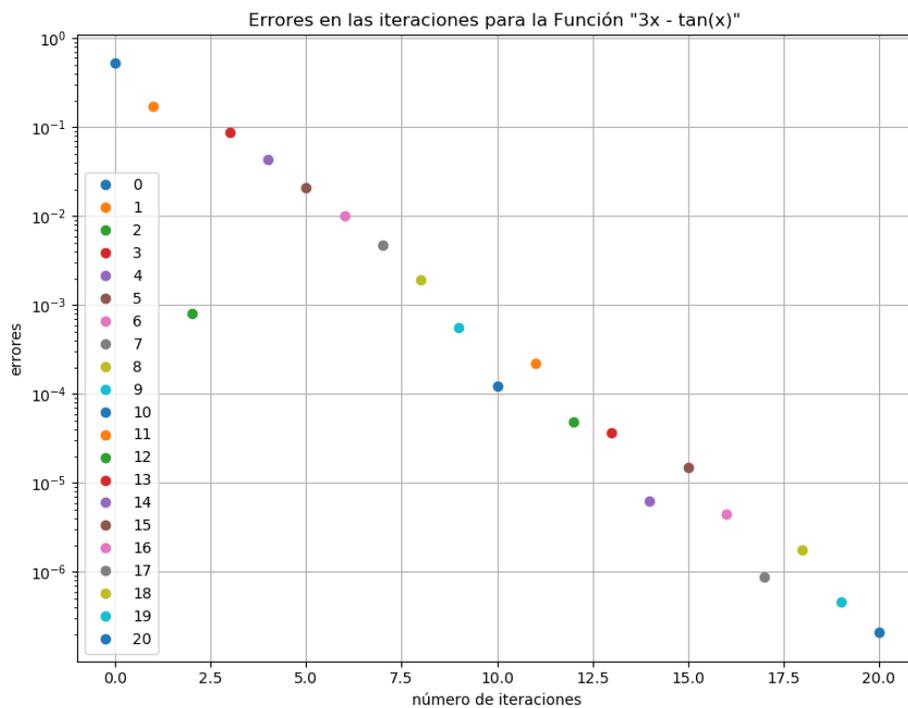


Figura 1