



Juan Andres Martinez Amado

Parcial 1

Clase:

Análisis Numérico

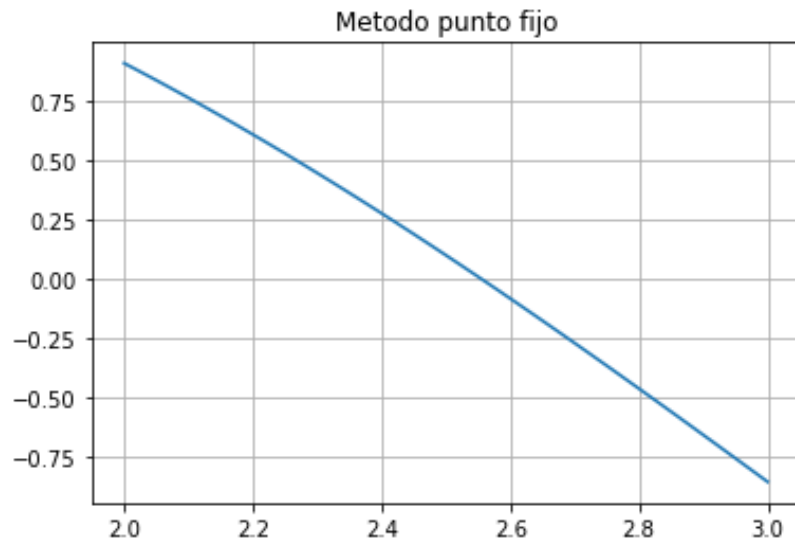
Profesora:

Eddy Herrera Daza

27de Agosto de 2021

3. Para cada una de las siguientes ecuaciones, determine un intervalo  $[a, b]$  en el que la iteración de **punto fijo** converge. Estime el número de iteraciones necesarias para obtener aproximaciones precisas dentro de  $10^{-5}$  y realice los cálculos. Implemente en R y/o Python
  - a.  $2 + \sin x - x = 0$
  - b.  $x - \cos x = 0$

Punto 3a)



Intervalo evaluado es de 2.0 a 3.0 de la función  $2 + \sin(x) - x$

Resultados punto fijo

```

Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\casta\OneDrive\Documents\Semestre\Analisis numerico\punto_fijo.py

1 import numpy as np
2
3 def fx(x):
4     return 2 + np.sin(x) - x
5
6 def gx(x):
7     return 2 + np.sin(x)
8
9
10 tolerancia = 10**-5
11 x1 = 0
12
13 error = np.abs(gx(x1) - x1)
14 i = 0
15
16 while (error > tolerancia and i < 200):
17     print(i, " xi = ", float('{:.5f}'.format(x1)), "f(x) = ", float('{:.5f}'.format(fx(x1))), " g(x1) = ",
18           float('{:.5f}'.format(gx(x1))), " error = ", float('{:.5f}'.format(error)))
19     error = np.abs(gx(x1) - x1)
20     x1 = gx(x1)
21     i = i + 1
22
23 print("El valor de x, tal que f(x) = 0 es : ", float('{:.5f}'.format(x1)), "numero de iteraciones ", i)
  
```

```

47 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
48 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
49 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
50 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
51 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
52 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
53 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
54 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
55 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
56 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
57 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
58 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
59 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
60 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
61 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
62 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
63 xi = 2.55448 f(x) = -0.00001 g(x1) = 2.55448 error = 0.00001
64 El valor de x, tal que f(x) = 0 es : 2.5542 numero de iteraciones 64
  
```

```
58 xi = 2.55421 f(x) = -2e-05 g(x1) = 2.55419 error = 3e-05
59 xi = 2.55419 f(x) = 2e-05 g(x1) = 2.5542 error = 2e-05
60 xi = 2.5542 f(x) = -1e-05 g(x1) = 2.55419 error = 2e-05
61 xi = 2.55419 f(x) = 1e-05 g(x1) = 2.5542 error = 1e-05
62 xi = 2.5542 f(x) = -1e-05 g(x1) = 2.55419 error = 1e-05
63 xi = 2.55419 f(x) = 1e-05 g(x1) = 2.5542 error = 1e-05
El valor de x, tal que f(x) = 0 es : 2.5542 numero de iteraciones 64
```

Figures now render in the Plots pane by default. To make them also appear inline in the

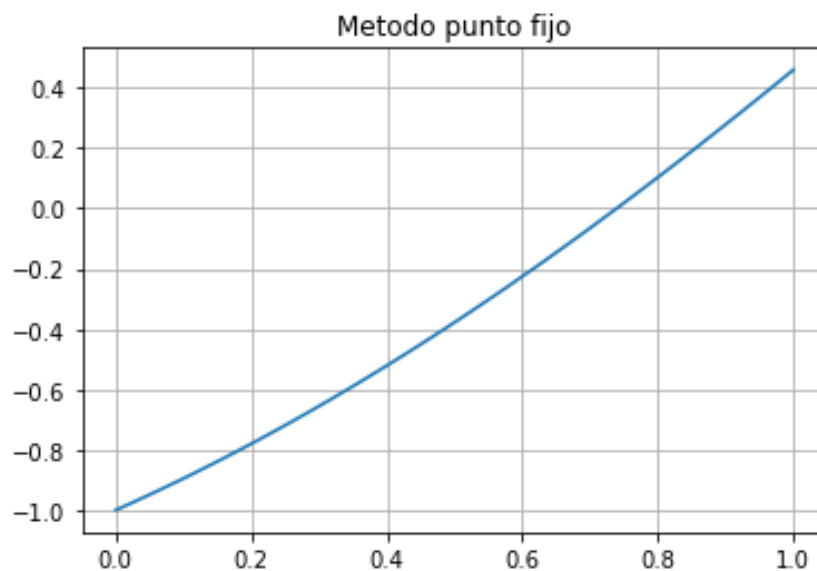
IPython console History

LSP Python: ready conda (Python 3.8.8) Line 24, Col 27 UTF-8 CRLF RW Mem 45%

Escritorio » ^ 🖨️ 🔊 ENG 8:16 a. m.

La raíz aproximada es 2.5542 con 64 iteraciones con  $10^{-5}$  de tolerancia

3b)  $x - \cos(x)$



Intervalo evaluado de 0 a 1 de la función  $x - \cos(x)$

```
Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:/Users/casta/OneDrive/Documents/Semestre/Analisis numerico/punto_fijo.py
1 import numpy as np
2 import matplotlib.pyplot as plt
3 def fx(x):
4     return x - np.cos(x)
5
6 def gx(x):
7     return np.cos(x)
8
9
10 tolerancia = 10**-5
11 x1 = 0
12 error = np.abs(gx(x1) - x1)
13 i = 0
14
15 while (error > tolerancia and i<200):
16     print(i, " xi = ", float('{:.5f}'.format(x1)), "f(x) = ", float('{:.5f}'.format(fx(x1))), " g(x1) = ",
17           if i > 0:
18               error = np.abs(gx(x1)-x1)
19               x1 = gx(x1)
20               i = i + 1
21
22 print("El valor de x, tal que f(x) = 0 es : ", float('{:.5f}'.format(x1)), "numero de iteraciones ", i)
23
24 x = np.linspace(0, 1, 100)
25 plt.title("Metodo punto fijo")
26 plt.plot(x,fx(x), label = "f(x)")
27
28 plt.grid()
29 plt.show()

In [12]: runfile('C:/Users/casta/OneDrive/Documents/Semestre/Analisis numerico/
punto_fijo.py', wdir='C:/Users/casta/OneDrive/Documents/Semestre/Analisis numerico')
0 xi = 0.0 f(x) = -1.0 g(x1) = 1.0 error = 1.0
1 xi = 1.0 f(x) = 0.4597 g(x1) = 0.5403 error = 1.0
2 xi = 0.5403 f(x) = -0.31725 g(x1) = 0.85755 error = 0.4597
3 xi = 0.85755 f(x) = 0.20326 g(x1) = 0.65429 error = 0.31725
4 xi = 0.65429 f(x) = -0.13919 g(x1) = 0.79348 error = 0.20326
5 xi = 0.79348 f(x) = 0.09211 g(x1) = 0.70137 error = 0.13919
6 xi = 0.70137 f(x) = -0.06259 g(x1) = 0.76396 error = 0.09211
7 xi = 0.76396 f(x) = 0.04186 g(x1) = 0.7221 error = 0.06259
8 xi = 0.7221 f(x) = -0.02832 g(x1) = 0.75042 error = 0.04186
9 xi = 0.75042 f(x) = 0.01901 g(x1) = 0.7314 error = 0.02832
10 xi = 0.7314 f(x) = -0.01283 g(x1) = 0.74424 error = 0.01901
11 xi = 0.74424 f(x) = 0.00863 g(x1) = 0.7356 error = 0.01283
12 xi = 0.7356 f(x) = -0.00582 g(x1) = 0.74143 error = 0.00863
13 xi = 0.74143 f(x) = 0.00392 g(x1) = 0.73751 error = 0.00582
14 xi = 0.73751 f(x) = -0.00264 g(x1) = 0.74015 error = 0.00392
15 xi = 0.74015 f(x) = 0.00178 g(x1) = 0.73837 error = 0.00264
16 xi = 0.73837 f(x) = -0.0012 g(x1) = 0.73957 error = 0.00178
17 xi = 0.73957 f(x) = 0.00081 g(x1) = 0.73876 error = 0.0012
18 xi = 0.73876 f(x) = -0.00054 g(x1) = 0.7393 error = 0.00081
19 xi = 0.7393 f(x) = 0.00037 g(x1) = 0.73904 error = 0.00054
20 xi = 0.73904 f(x) = -0.00025 g(x1) = 0.73918 error = 0.00037
21 xi = 0.73918 f(x) = 0.00017 g(x1) = 0.73902 error = 0.00025
22 xi = 0.73902 f(x) = -0.00011 g(x1) = 0.73913 error = 0.00017
23 xi = 0.73913 f(x) = 8e-05 g(x1) = 0.73905 error = 0.00011
24 xi = 0.73905 f(x) = -5e-05 g(x1) = 0.73911 error = 8e-05
25 xi = 0.73911 f(x) = 3e-05 g(x1) = 0.73907 error = 5e-05
26 xi = 0.73907 f(x) = -2e-05 g(x1) = 0.73909 error = 3e-05
27 xi = 0.73909 f(x) = 2e-05 g(x1) = 0.73908 error = 2e-05
28 xi = 0.73908 f(x) = -1e-05 g(x1) = 0.73909 error = 2e-05
29 xi = 0.73909 f(x) = 1e-05 g(x1) = 0.73908 error = 1e-05
El valor de x, tal que f(x) = 0 es : 0.73908 numero de iteraciones 30

In [13]:
```

```
25 xi = 0.73911 f(x) = 3e-05 g(x1) = 0.73907 error = 5e-05
26 xi = 0.73907 f(x) = -2e-05 g(x1) = 0.73909 error = 3e-05
27 xi = 0.73909 f(x) = 2e-05 g(x1) = 0.73908 error = 2e-05
28 xi = 0.73908 f(x) = -1e-05 g(x1) = 0.73909 error = 2e-05
29 xi = 0.73909 f(x) = 1e-05 g(x1) = 0.73908 error = 1e-05
El valor de x, tal que f(x) = 0 es : 0.73908 numero de iteraciones 30

In [13]:
```

IPython console History

LSP Python: ready conda (Python 3.8.8) Line 25, Col 31 UTF-8 CRLF RW Mem 45%

Escritorio 8:22 a.m.

La raíz aproximada es 0.73908 con numero de iteraciones 30 y tolerancia  $10^{-5}$