Tarea Método de Newton

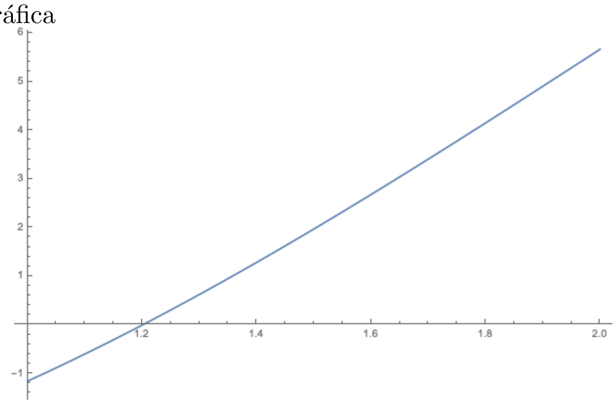
Angel Caceres Licona

June 3, 2020

1 Considerar la función $f(x) = x^2 - 4\cos(x), x \in \mathbb{R}$...

2 Graficar en el intervalo $(1, 2)$

Gráfica



3 Código del programa

```
from math import *

def newtonIterationFunction(x):
    return x - ((x**2 - 4*cos(x)) / (2*x+4*sin(x)))


def function(x):
    return x**2 - 4*cos(x)

x = 1
c = 1
xold = 0
fc = 1

for i in range(1000):
```

```
        print "Iteraciones: ",str(i),"Valor aproximado: ", str(x),
            "Intervalo", str(c)
        c = (x - xold)
        fc = function(x)
        xold = x
        x = newtonIterationFunction(x)
        if (abs(fc) < 0.000000005) :
            break
```

3.1 Usar el método de newton para localizar una aproximacion...

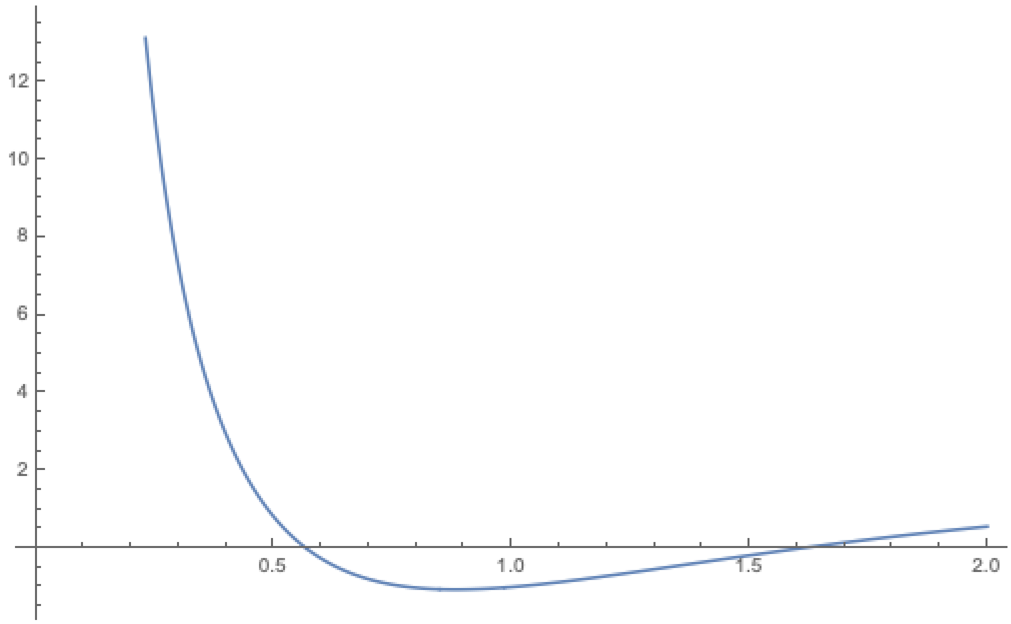
n	p_i	E_i	$f(p_i)$
0	1.21640595224	1	-1.16120922347
1	1.20159918212	0.2164059522393	0.0915686905446
2	1.20153830038	-0.0148067701169	0.000373428376449
3	1.20153829934	-6.08817408994e-05	6.38189479041e-09

3.2 Aplicar el método de bisección, secante y falsa posicion con la misma tolerancia...

Para biseccion salieron 28 iteraciones. Para secante salieron 8 iteraciones. Para secante salieron 7 iteraciones.

4 Considere la funcion $-8e^{1-x} + \frac{7}{x}$

4.1 Grafique en el intervalo $(0, 2)$



4.2 Con el programa obtengo los siguientes resultados

n	p_i	E_i	$f(p_i)$
0	0.55470744502	0.5	0.810229834399
1	0.567540459004	0.05470744502	0.131690221821
2	0.56813355775	0.0128330139839	0.00557743516944
3	0.568134762962	0.000593098746384	1.1287832983e-05
4	0.568134762967	1.20521197233e-06	4.64979166281e-11

4.3 Compare el resultado con biseccion, secante y punto fijo

Para biseccion obtuve 28 iteraciones Para secante obtuve 7 iteraciones Para falsa posicion obtuve 12 iteraciones

4.4 Ahora la segunda raiz

n	p_i	E_i	$f(p_i)$
0	1.60935506282	1.6	-0.01549308875229
1	1.60938106752	0.00935506281674	-4.2828011722e-05
2	1.60938106772	2.60047028984e-05	-3.35051986156e-10

4.5 Compare el resultado con biseccion, secante y punto fijo

Para biseccion obtuve 26 iteraciones Para secante obtuve 5 iteraciones Para falsa posicion obtuve 6 iteraciones