

Projecto 04 – Raízes de equações não lineares. Método da bissecção. Método de Newton-Raphson

Objectivo: Estudo de métodos iterativos para encontrar raízes de equações não lineares. Aulas teóricas 7 e 8.

Instruções: No início do ficheiro deve constar como comentário o nº e nome dos elementos do grupo. A análise dos resultados e o texto explicativo das instruções de código utilizadas, devem ser introduzidos como comentários no *ficheiro-py*.

Exercício

O número de indivíduos de uma população pode ser dado, num certo período de tempo t , pela função,

$$N(t) = N_0 \times e^{\lambda t} + \frac{v}{\lambda} (e^{\lambda t} - 1)$$

onde v representa o valor anual de imigração, λ a taxa anual de natalidade e N_0 o número de indivíduos no instante inicial. Sabe-se que uma dada população possui inicialmente um milhão de indivíduos, que 281×10^3 imigraram para a comunidade durante o primeiro ano e que no fim desse ano existem 1.780×10^6 de indivíduos. Pretende-se determinar a taxa de natalidade dessa população nesse ano.

Desenvolver um programa em *python* que execute as seguintes questões:

1. Localizar graficamente a posição da taxa anual de natalidade:
 - 1.1. Construir uma função para a calcular a raiz.
 - 1.2. Fazer o gráfico dessa função no intervalo $]0.01;1]$.
 - 1.3. Desenhar uma linha horizontal com ordenada = 0.
 - 1.4. Identificar a intersecção de ambas as linhas com uma circunferência a vermelho.
 - 1.5. Colocar o texto, título, e *labels* dos eixos na figura.

(apresentar um output gráfico em linha com o da figura 1)

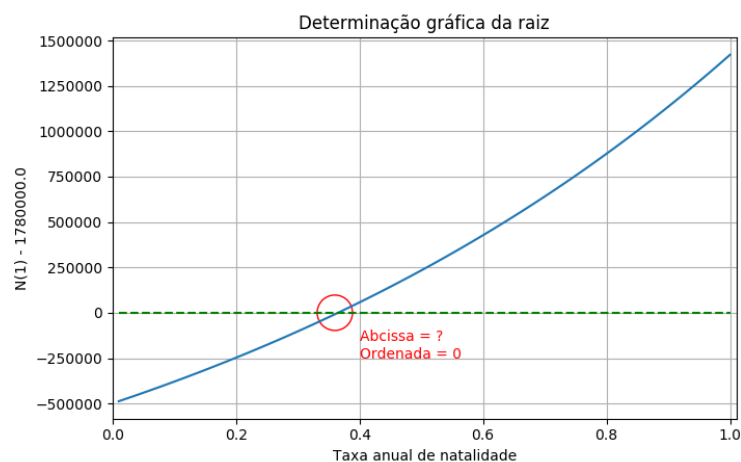


Figura 1

2. Determinar o valor da taxa anual de natalidade, pelo método da Bisseção, e apresentar um output gráfico em linha com os atributos gráficos e de legendagem da figura 2 (use o código disponibilizado nas aulas teóricas sempre que se justifique).

2.1. Determinar o menor intervalo $[a; b]$, dentro do qual esteja a raiz.

2.2. Construir a função *def bissec(a,b,tolerância)* que dê como output a raiz, o erro e o número de iterações.

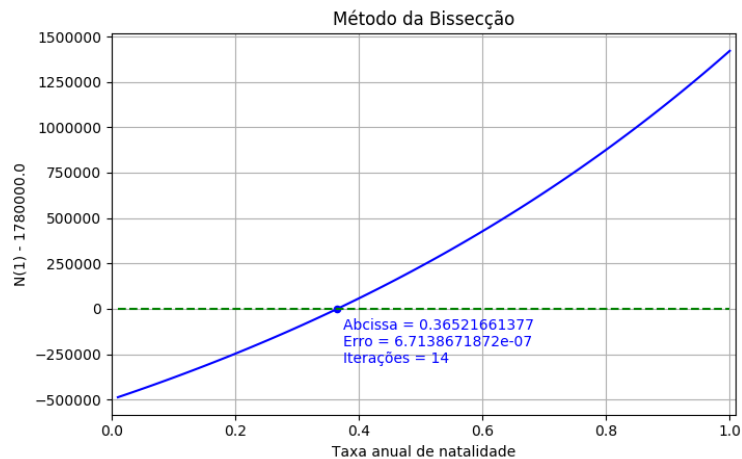


Figura 2

3. Resolver a questão 2 usando o método de Newton-Raphson, com um output gráfico em linha com o da figura 3. Use a biblioteca 'sympy' para derivar a equação da alínea 1.

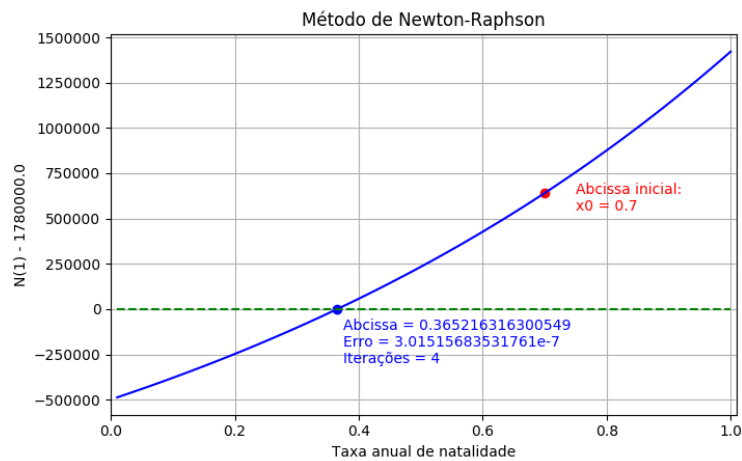


Figura 3