Algebra Linear Computacional - Lista 5.2

Mayara Aragão

Tarefa 1)

Derivação numérica

In [1]:
```
import numpy as np
```

Desenvolvi uma rotina que recebe como ultimo parametro de entrada a variavel $regra$, que seleciona o método de derivação numérica desejado:

In [2]:
```
def derivada(x, delta, regra):
    if regra == 1:
        df = (f(x+delta) - f(x-delta))/(2*delta) # Diferença central
    elif regra == 2:
        df = (f(x+delta)-f(x))/delta # Passo à frente
    else:
        df = (f(x) - f(x-delta))/delta # Passo atrás
    return df
```

Tarefa 2)

Procedimento de interpolação de Richard

In [3]:
```
def derivada_richard(x, delta, regra, p):
    df1 = derivada(x, delta, regra)
    delta2 = delta/2
    df2 = derivada(x, delta2, regra)
    q = delta/delta2
    a0 = df1 + (df1 - df2)/(q**(-p) -1)
    return a0
```

Tarefa 3)

Utilizando os programas acima, calcular as derivadas seguintes funções:

i.

$$f(x) = x^3 + \frac{1}{e^x}$$
$$x = 3$$

Defindo a função:

In [4]:
```
def f(x):
    return x**3 + 1/np.exp(x)
```

Utilizarei para o cálculo das derivadas numéricas, $\Delta x = 0.001$

In [5]:
```
x=3
delta = 0.001
```

Diferença central

In [6]:
```
d = derivada(x, delta, 1)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  3  é: 26.95021392332997
```

Passo à frente

In [7]:
```
d = derivada(x, delta, 2)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  3  é: 26.959238816868236
```

Passo atrás

In [8]:
```
d = derivada(x, delta, 3)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  3  é: 26.9411890297917
```

Interpolação de Richard

Diferença central

Utilizando $p = 1$ inicialmente:

In [9]:
```
d = derivada_richard(x, delta, 1, 1)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  3  é: 26.950212435798093
```

Para $p = 2$:

In [10]:
```
d = derivada_richard(x, delta, 1, 2)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  3  é: 26.950212931642053
```

Passo à frente

Para $p = 1$:

In [11]:
```
d = derivada_richard(x, delta, 2, 1)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  3  é: 26.950212435796317
```

Para $p = 2$:

In [12]:
```
d = derivada_richard(x, delta, 2, 2)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  3  é: 26.953221229486957
```

Passo atrás:

Para $p = 1$:

In [13]:
```
d = derivada_richard(x, delta, 3, 1)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  3  é: 26.95021243579987
```

Para $p = 2$:

In [14]:
```
d = derivada_richard(x, delta, 3, 2)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  3  é: 26.947204633797146
```

Valor analítico da função dada no ponto $x = 3$ é dada por $\frac{df(x)}{dx} = 26.9502129316$. Este valor foi encontrado na interpolação de richard com $p = 2$ pelo método da diferença central, mas os outros métodos também tiveram bons resultados. O método que obteve pior resultado, em ambos os casos foi o do passo para trás, no caso normal e para richard com $p = 2$

ii.

$$f(x) = x^{\frac{1}{3}} + ln(x)$$
$$x = 2$$

Defindo a função:

In [15]:
```
def f(x):
    return x**(1/3)+np.log(x)
```

In [16]:
```
x=2
delta = 0.001
```

Diferença central

In [17]:
```
d = derivada(x, delta, 1)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  2  é: 0.7099868930373621
```

Passo à frente

In [18]:
```
d = derivada(x, delta, 2)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  2  é: 0.709826895211707
```

Passo atrás

In [19]:
```
d = derivada(x, delta, 3)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  2  é: 0.7101468908630171
```

Interpolação de Richard

Diferença central

Utilizando $p = 1$ inicialmente:

In [20]:
```
d = derivada_richard(x, delta, 1, 1)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  2  é: 0.7099868159550216
```

Agora para $p = 2$:

In [21]:
```
d = derivada_richard(x, delta, 1, 2)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  2  é: 0.7099868416491351
```

Passo à frente

Para $p = 1$:

In [22]:
```
d = derivada_richard(x, delta, 2, 1)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  2  é: 0.7099868159696765
```

Para $p = 2$:

In [23]:
```
d = derivada_richard(x, delta, 2, 2)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  2  é: 0.7099335090503534
```

Passo atrás

Para $p = 1$:

In [24]:
```
d = derivada_richard(x, delta, 3, 1)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  2  é: 0.7099868159403666
```

Para $p = 2$:

In [25]:
```
d = derivada_richard(x, delta, 3, 2)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  2  é: 0.7100401742479168
```

Valor analítico da função dada no ponto $x = 2$ é dada por $\frac{df(x)}{dx} = 0.70998684164$. Este valor também foi encontrado na interpolação de richard com $p = 2$ pelo método da diferença central, mas os outros métodos também estão próxímos ao valor analítico considerando uma tolerância de pelo menos 10^{-4}, com excessão do método do passo atrás que obteve piores resultados considerando $p = 2$.

iii.

$$f(x) = 1 - exp\left(-\left(\frac{x}{5}\right)^2\right)$$
$$x = 6$$

In [26]:
```
def f(x):
    return 1-np.exp(-x**2/25.0)
```

In [27]:
```
x=6
delta = 0.001
```

Diferença central

In [28]:
```
d = derivada(x, delta, 1)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  6  é: 0.11372532398545854
```

Passo à frente

In [29]:
```
d = derivada(x, delta, 2)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  6  é: 0.11370750701844567
```

Passo atrás

In [30]:
```
d = derivada(x, delta, 3)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  6  é: 0.1137431409524714
```

Interpolação de Richard

Diferença central

Utilizando $p = 1$ inicialmente:

In [31]:
```
d = derivada_richard(x, delta, 1, 1)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  6  é: 0.11372532425835136
```

Utilizando $p = 2$:

In [32]:
```
d = derivada_richard(x, delta, 1, 2)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  6  é: 0.11372532416738708
```

Passo à frente

Para $p = 1$:

In [33]:
```
d = derivada_richard(x, delta, 2, 1)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  6  é: 0.11372532425835136
```

Para $p = 2$:

In [34]:
```
d = derivada_richard(x, delta, 2, 2)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  6  é: 0.1137193851783828
```

Passo atrás

Para $p = 1$:

In [35]:
```
d = derivada_richard(x, delta, 3, 1)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  6  é: 0.11372532425835136
```

Para $p = 2$:

In [36]:
```python
d = derivada_richard(x, delta, 3, 2)
print('A derivada no ponto ', x, ' é:', d)
```

```
A derivada no ponto  6  é: 0.11373126315639137
```

O valor analítico da função dada no ponto $x = 6$ é dada por $\frac{df(x)}{dx} = 0.1137253241$. Este valor também foi encontrado na interpolação de richard com $p = 2$ pelo método da diferença central, mas os outros métodos também estão próxímos ao valor analítico. Concluindo então que todos são eficientes, embora a interpolação de richard com $p = 2$ tenha se aproximado mais ao valor desejado.