Introdução à Computação em Física

DIFERENCIAÇÃO NUMÉRICA

PROF. WALBER

Refs.:
Computational Physics, R. Landau, M. J. Paez, C. C. Bordelanu
Elementary Numerical Analysis, An algorithmic Approach, S. D. Conte. & C. de Boor

Cálculo de derivadas:

Em muitos problemas de Física há a necessidade de se calcular as derivadas de uma dada função f(x). Nesse contexto, iremos estudar alguns métodos númericos que podem ser utilizados para se obter de maneira aproximada a derivada de uma dada função.

Antes, vale lembrar que por definição:

$$\frac{df(x)}{dx} = f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

Computador pode encontrar problemas, uma vez que h é bem pequeno

N. Piskounov, Cálculo diferencial e integral (Vol.1)

Cálculo de derivadas:

O ponto de partida dos métodos que veremos é a chamada série de Taylor. Nesse contexto, podemos aproximar a função f(x+h) em torno de x como uma série contendo derivadas sucessivas:

$$f(x+h) = f(x) + \frac{h}{1!}f'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(x) + \ldots$$

N. Piskounov, Cálculo diferencial e integral (Vol.1)

Método derivada a direita (forward difference -fd):

Isolando a derivada primeira na série de Taylor:

$$f(x+h) = f(x) + \frac{h}{1!}f'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(x) + \dots$$

$$\Downarrow$$

$$f'(x) = \frac{1}{h}\left(f(x+h) - f(x) - \frac{h^2}{2!}f''(x) - \frac{h^3}{3!}f'''(x) - \dots\right)$$

Logo, podemos escrever:

$$f'(x) = \frac{f(x+h) - f(x)}{h} - h\left(\frac{1}{2!}f''(x) + \frac{h}{3!}f'''(x) + \dots\right)$$

Método derivada a direita (forward difference -fd):

Aproximação para derivada:

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

Além disso,

$$f'(x) = \underbrace{\frac{f(x+h) - f(x)}{h}}_{f'(x)} \underbrace{- h\left(\frac{1}{2!}f''(x) + \frac{h}{3!}f'''(x) + \dots\right)}_{\text{Erro proporcional a h}}$$

Método derivada a esquerda (backward difference -bd):

De maneira análoga ao feito anteriormente em relação a série de Taylor:

$$f(x-h) = f(x) - \frac{h}{1!}f'(x) + \frac{h^2}{2!}f''(x) - \frac{h^3}{3!}f'''(x) + \dots$$

$$\Downarrow$$

$$f'(x) = \frac{f(x) - f(x-h)}{h} + h\left(\frac{1}{2!}f''(x) - \frac{h}{3!}f'''(x) + \dots\right)$$

Método derivada a esquerda (backward difference -bd):

Aproximação para derivada:

$$f'(x) \approx \frac{f(x) - f(x-h)}{h}$$

Além disso,

$$f'(x) = \underbrace{\frac{f(x) - f(x-h)}{h}}_{\text{f'(x)}} + \underbrace{h\left(\frac{1}{2!}f''(x) - \frac{h}{3!}f'''(x) + \dots\right)}_{\text{Erro proporcional a h}}$$

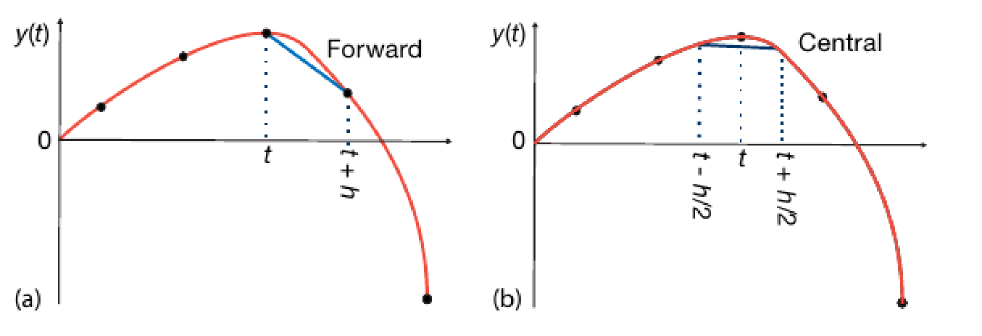
Ambos métodos fd e bd possuem erros da ordem de h

Método da diferença central:

Uma melhoria dos métodos anteriores é obtida usando um ponto adicional (método de três pontos):

$$f(x+h) = f(x) + \frac{h}{1!}f'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(x) + \dots \quad \text{(I)}$$

$$f(x-h) = f(x) - \frac{h}{1!}f'(x) + \frac{h^2}{2!}f''(x) - \frac{h^3}{3!}f'''(x) + \dots \quad \text{(II)}$$



Fazendo (I)-(II):

$$f(x+h) - f(x-h) = 2\frac{h}{1!}f'(x) + 2\frac{h^3}{3!}f'''(x) \underbrace{+ \dots}_{\text{Termos ímpares em h}}$$

Método da diferença central:

Aproximação para derivada:

$$f(x+h) - f(x-h) = 2\frac{h}{1!}f'(x) + 2\frac{h^3}{3!}f'''(x) + \dots$$

$$f'(x) = \underbrace{\frac{f(x+h) - f(x-h)}{2h}}_{f'(x)} - h^2\left(\frac{f'''(x)}{3!} + \dots\right)$$

Erro proporcional a h^2

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

Método de ordem $O(h^2)$

Método da diferença central (5 points, 5-point stencil):

Podemos utilizar mais pontos para a obtenção de aproximações com maior precisão. Para isso, temos que considerar:

$$f(x+h) = f(x) + \frac{h}{1!}f'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(x) + \dots \quad \text{(I)}$$

$$f(x-h) = f(x) - \frac{h}{1!}f'(x) + \frac{h^2}{2!}f''(x) - \frac{h^3}{3!}f'''(x) + \dots \quad \text{(II)}$$

$$f(x-2h) = f(x) - \frac{2h}{1!}f'(x) + \frac{4h^2}{2!}f''(x) - \frac{8h^3}{3!}f'''(x) + \dots \quad \text{(III)}$$

$$f(x+2h) = f(x) - \frac{2h}{1!}f'(x) + \frac{4h^2}{2!}f''(x) + \frac{8h^3}{3!}f'''(x) + \dots \quad \text{(IV)}$$

Método da diferença central (5 points, 5-point stencil):

Sendo (I)-(II):

$$f(x+h) - f(x-h) = 2\frac{h}{1!}f'(x) + 2\frac{h^3}{3!}f'''(x) + \ldots \quad \text{(V)}$$

Sendo (III)-(IV):

$$f(x+2h) - f(x-2h) = \frac{4h}{1!}f'(x) + \frac{16h^3}{3!}f'''(x) + \ldots \quad \text{(VI)}$$

Por fim, 8x(V) - (VI) (eliminação de termo de ordem h^3):

$$8f(x+h) - 8f(x-h) - f(x+2h) + f(x-2h) = \frac{12h}{1!}f'(x) + O(h^4)$$

Método da diferença central (5 points, 5-point stencil):

Aproximação para derivada:

$$8f(x+h) - 8f(x-h) - f(x+2h) + f(x-2h) = \frac{12h}{1!} f'(x) + O(h^4)$$

$$\Downarrow$$

$$f'(x) \approx \frac{1}{12h}(f(x-2h) - 8f(x-h) + 8f(x+h) - f(x+2h))$$

Erro da ordem de h^4

Necessidade de se conhecer o valor da função em mais pontos (4 pontos) para se obter a derivada em x.

Exemplo:

Seja f(x) = bx^2 + a, tal que f'(x) = 2bx:

fd: $$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$ → f'(x)=2bx + bh

Erro da ordem de h (boa aproximação se h << 1/b.

bd: $$f'(x) \approx \frac{f(x) - f(x-h)}{h}$$ → f'(x)=2bx - bh

Diferença central: $$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$ → f'(x)=2bx

Derivada de segunda ordem:

Podemos utilizar as expansões em série de Taylor anteriores para obter uma aproximação para a segunda derivada f''(x):

$$f(x+h) = f(x) + \frac{h}{1!}f'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(x) + \dots \qquad \text{(I)}$$

$$f(x-h) = f(x) - \frac{h}{1!}f'(x) + \frac{h^2}{2!}f''(x) - \frac{h^3}{3!}f'''(x) + \dots \qquad \text{(II)}$$

Obtendo (I)+(II), temos que:

$$f''(x) \approx \frac{f(x-h) - 2f(x) + f(x+h)}{h^2}$$

Erro de ordem h^2

Implementação (exemplo 1):

fd:

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

cd:

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

** Implementacao primeira derivada (fd e cd) **

$f(x) = e^x$,

Derivada em x = 1: $f'(x) = e^x$, $f'(1) = e$

Vale lembrar que e = 2.7182818284590 ...

```
import numpy as np

x = 1.0    #### ponto no qual queremos calcular a derivada

def fd(hs):
    fl = (1/hs)*(np.exp(x+hs)-np.exp(x))        ### forward difference
    return fl

def cd(hs):
    flcd = (1/(2*hs))*(np.exp(x+hs)-np.exp(x-hs))   #### central difference
    return flcd

h = 0.1   ### Passo

print '(FD) Derivada de f(x) = exp(x) em x = 0 igual a ', fd(h)

print '(CD) Derivada de f(x) = exp(x) em x = 0 igual a ', cd(h)
```

```
(FD) Derivada de f(x) = exp(x) em x = 0 igual a  2.858841954873883
(CD) Derivada de f(x) = exp(x) em x = 0 igual a  2.7228145639474177
```

Implementação (exemplo 2):

** Implementacao primeira derivada (fd e cd) **

$f(x)=ex$

,

Derivada em x = 1: $f'(x)=ex$, $f'(1)=e$

Vale lembrar que e = 2.7182818284590 ... Variacao do passo h

```
import numpy as np
import math

x = 1.0    #### ponto no qual queremos calcular a derivada

def fd(hs):
    fl = (1/hs)*(np.exp(x+hs)-np.exp(x))         ### forward difference
    return fl

def cd(hs):
    flcd = (1/(2*hs))*(np.exp(x+hs)-np.exp(x-hs))   #### central difference
    return flcd

h = [0.5, 0.1, 0.01, 0.001, 0.0001]   ### diferentes passos

for i in range(0,len(h)):

    print 'Passo h = ', h[i]

    print '(FD) Derivada de f(x) = exp(x) em x = 1 igual a ', fd(h[i]) , 'erro = ', fd(h[i])-math.e

    print '(CD) Derivada de f(x) = exp(x) em x = 1 igual a ', cd(h[i]) , 'erro = ', cd(h[i])-math.e
```

Implementação (exemplo 2):

```
Passo h =  0.5
(FD) Derivada de f(x) = exp(x) em x = 1 igual a  3.526814483758039 erro =  0.8085326552989938
(CD) Derivada de f(x) = exp(x) em x = 1 igual a  2.8329677996379363 erro =  0.11468597117889123
Passo h =  0.1
(FD) Derivada de f(x) = exp(x) em x = 1 igual a  2.858841954873883 erro =  0.14056012641483795
(CD) Derivada de f(x) = exp(x) em x = 1 igual a  2.7228145639474177 erro =  0.004532735488837263
Passo h =  0.01
(FD) Derivada de f(x) = exp(x) em x = 1 igual a  2.7319186557871245 erro =  0.013636827328079936
(CD) Derivada de f(x) = exp(x) em x = 1 igual a  2.718327133382714 erro =  4.530492366905392e-05
Passo h =  0.001
(FD) Derivada de f(x) = exp(x) em x = 1 igual a  2.7196414225332255 erro =  0.0013595940741804036
(CD) Derivada de f(x) = exp(x) em x = 1 igual a  2.718282281505724 erro =  4.530466788388831e-07
Passo h =  0.0001
(FD) Derivada de f(x) = exp(x) em x = 1 igual a  2.718417747082924 erro =  0.00013591862387896114
(CD) Derivada de f(x) = exp(x) em x = 1 igual a  2.718281832989611 erro =  4.53056570037802e-09
```

Implementação (exemplo 3):

** Exemplo 3 **

Derivada a partir de um conjunto discreto de pontos:

x = 0.398, 0.399, 0.400, 0.401, 0.402

f(x) = 0.408591, 0.409671, 0.410752, 0.411834, 0.412915

Desejamos calcular f'(x = 0.400).

```
import numpy as np

x = [0.398, 0.399, 0.400, 0.401, 0.402]

f = [0.408591, 0.409671, 0.410752, 0.411834, 0.412915]

valorex = np.cosh(0.4)       ##### valor de referencia

### Derivada pelo metodo central difference ####

h = x[2]-x[1]

fl = (1/(2*h))*(f[3]-f[1])

print 'Valor da prim. derivada em x = 0.4 , eh igual a  ', fl , ' erro = ' , abs(fl-valorex)
```

```
Valor da prim. derivada em x = 0.4 , eh igual a   1.0815  erro =  0.00042762816152941 42
```

Implementação (exemplo 3):

```
#### Derivada pelo metodo CD com 5 points ######

fl5p1 = f[0] + (8*f[3])
fl5p2 = 8*f[1] + f[4]
fl5p = (1/(12*h))*(fl5p1-fl5p2)

print 'Valor da prim. derivada em x = 0.4 (5points), eh igual a  ', fl5p, 'erro = ', abs(fl5p-valorex)
```

```
Valor da prim. derivada em x = 0.4 (5points), eh igual a   1.08166666667 erro =  0.0005942948281916216
```

```
#### Segunda derivada

fll = (1/h**2)*(f[3] - 2*f[2] + f[1])

print 'Segunda derivada em x = 0.4 , eh igual a ', fll
```

```
Segunda derivada em x = 0.4 , eh igual a  0.999999999973
```

```
print np.sinh(0.4)
```

```
0.4107523258028155
```

Implementação (exemplo 4):

```
import numpy as np

def f(a):
    return np.sinh(a)


h = 0.001     #### passo
x = 0.4

#### central difference
f1cd = (1/(2*h))*(f(x+h)-f(x-h))

print 'Primeira derivada (CD)  = ', f1cd, 'erro = ' , abs(np.cosh(0.4)-f1cd)
```

```
Primeira derivada (CD)  =  1.0810725520171982 erro =  1.8017874325870764e-07
```

```
f15p = (1/(12*h))*(f(x-(2*h)) - 8*f(x-h) + 8*f(x+h) - f(x+(2*h)))

print 'Primeira derivada (CD -5points) =  ', f15p , 'erro = ', abs(np.cosh(0.4)-f15p)
```

```
Primeira derivada (CD -5points) =   1.0810723718384234 erro =  3.1530333899354446e-14
```

Implementação (exemplo 4):

```
#### Segunda derivada

fll = (1/h**2)*(f(x+h) - 2*f(x) + f(x-h))

print 'Segunda derivada =  ', fll , 'erro = ', abs(np.sinh(0.4)-fll)
```

```
Segunda derivada =   0.41075236006937743 erro =  3.4266561921292293e-08
```

Atividades práticas:

1. Escreva um programa para calcular a primeira e segunda derivada da função cos(x) em x = pi/6.
Utilize as diferentes aproximações dadas em aula (fd, bd, cd, cd-5points) para a primeira derivada. Escreva os erros absolutos e relativos. Utilize h = 0.1, 0.01 e 0.001 .

2. Seja uma partícula de massa = 10 Kg, que possui suas coordenadas em função do tempo dadas por:

$$x(t) = 30tcos(60°)$$

$$y(t) = 30tsin(60°) - \frac{gt^2}{2}$$

g=9.8 m/s2

Escreva um programa que calcula as forças sobre a partícula nas direções x e y.