Numerische-Differentiation

May 4, 2019

Wir zeigen zunächst, wie man mit Hilfe von Python Taylorentwicklungen ausrechnen kann.

In [1]:
```
import sympy
from IPython.display import display
sympy.init_printing(use_latex='mathjax')
import numpy as np
import scipy.special
import math
import matplotlib.pyplot as plt

h=sympy.Symbol("h")
x=sympy.Symbol("x")
f=sympy.Function("f")
```

In [2]:
```
# Entwickle die Funktion f um die Stelle 0 in der Variablen h bis zum 5. Glied
display(f(h).series(h,0,5))
```

$$f(0) + h \left. \frac{d}{dx} f(x) \right|_{x=0} + \frac{h^2 \left. \frac{d^2}{dx^2} f(x) \right|_{x=0}}{2} + \frac{h^3 \left. \frac{d^3}{dx^3} f(x) \right|_{x=0}}{6} + \frac{h^4 \left. \frac{d^4}{dx^4} f(x) \right|_{x=0}}{24} + O\left(h^5\right)$$

In [3]:
```
# Wir suchen eine Linearkombination aus f(0), f(h), f(-h), die f'(0) möglichst gut appro
# Also:
a=sympy.Symbol("a")
b=sympy.Symbol("b")
c=sympy.Symbol("c")
C=a*f(h).series(h,0,3)+b*f(-h).series(h,0,3)+c*f(0).series(h,0,3)
C=C.simplify()
C=C.factor()
C=C.collect(h)
C=C.simplify()
display(C)
```

$$h\,(a-b) \left. \frac{d}{dx} f(x) \right|_{x=0} + \frac{h^2\,(a+b) \left. \frac{d^2}{dx^2} f(x) \right|_{x=0}}{2} + c f(0) + b f(0) + a f(0) + O\left(h^3\right)$$

Wir erhalten:

$$a + b + c = 0 \tag{1}$$
$$a - b = 1/h \tag{2}$$
$$a + b = 0 \tag{3}$$

Es folgt $c = 0$, $a = \frac{1}{2h}$, $b = -\frac{1}{2h}$.
Wir testen die resultierende Formel.

In [4]:
```
C=1/h/2*f(h).series(h,0,4)-1/h/2*f(-h).series(h,0,4)
display(C.simplify())
```

$$\left.\frac{d}{dx}f(x)\right|_{x=0} + \frac{h^2 \left.\frac{d^3}{dx^3}f(x)\right|_{x=0}}{6} + O\left(h^3\right)$$

Die Formel ist exakt von der Ordnung 2, aber nicht von der Ordnung 3.
Wir testen die Genauigkeit einiger weiterer Formeln.

In [5]:
```
# Vorwärts-Differenzenquotient
C=(f(x+h).series(h,0,3)-f(x))/h
display(C.simplify())
# Exakt von der Ordnung 1.
```

$$\left.\frac{d}{d\xi_1}f(\xi_1)\right|_{\xi_1=x} + \frac{h \left.\frac{d^2}{d\xi_1^2}f(\xi_1)\right|_{\xi_1=x}}{2} + O\left(h^2\right)$$

In [6]:
```
# Rückwärts-Differenzenquotient
C=(-f(x-h).series(h,0,3)+f(x))/h
display(C.simplify())
# Exakt von der Ordnung 1.
```

$$\left.\frac{d}{d\xi_1}f(\xi_1)\right|_{\xi_1=x} - \frac{h \left.\frac{d^2}{d\xi_1^2}f(\xi_1)\right|_{\xi_1=x}}{2} + O\left(h^2\right)$$

In [7]:
```
# Zentraler Quotient (siehe oben)
C=((f(x+h/2).series(h,0,4)-f(x-h/2).series(h,0,4)))/h
display(C.simplify())
# Exakt von der Ordnung 2, aber nicht von der Ordnung 3.
```

$$\left.\frac{d}{d\xi_1}f(\xi_1)\right|_{\xi_1=x} + \frac{h^2 \left.\frac{d^3}{d\xi_1^3}f(\xi_1)\right|_{\xi_1=x}}{24} + O\left(h^3\right)$$

In [8]:
```
# Zentraler Quotient für die zweite Ableitung
C=(f(x+h).series(h,0,5)-2*f(x)+f(x-h).series(h,0,5))/(h*h)
display(C.simplify())
# Exakt von der Ordnung 2, aber nicht von der Ordnung 3.
```

$$\left.\frac{d^2}{d\xi_1^2}f(\xi_1)\right|_{\xi_1=x} + \frac{h^2 \left.\frac{d^4}{d\xi_1^4}f(\xi_1)\right|_{\xi_1=x}}{12} + O\left(h^3\right)$$

In [ ]: