main

September 18, 2021

[35]:
```
import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt
```

[71]:
```
class Function:
    def __init__(self, function, derivative, name):
        self.__function = function
        self.__derivative = derivative
        self.__name = name
    def getFunction(self): return self.__function
    def getDerivative(self): return self.__derivative
    def getName(self): return self.__name
```

[72]:
```
def func1(x): return np.sin(x**2)
def realDerivative1(x): return 2. * x * np.cos(x**2)
f1 = Function(func1, realDerivative1, "$sin(x^2)$")

def func2(x): return np.cos(np.sin(x))
def realDerivative2(x): return -np.sin(np.sin(x)) * np.cos(x)
f2 = Function(func2, realDerivative2, "$cos(sin(x))$")

def func3(x): return np.exp(np.sin(np.cos(x)))
def realDerivative3(x): return -np.exp(np.sin(np.cos(x))) * np.cos(np.cos(x)) *␣
 ↪np.sin(x)
f3 = Function(func3, realDerivative3, "$\exp(sin(cos(x)))$")

def func4(x): return np.log(x + 3)
def realDerivative4(x): return 1. / (x + 3.)
f4 = Function(func4, realDerivative4, "$ln(x + 3)$")

def func5(x): return (x+3)**0.5
def realDerivative5(x): return 1. / (2. * ((x + 3.)**0.5))
f5 = Function(func5, realDerivative5, "$(x + 3)^{0.5}$")

funcs = [f1, f2, f3, f4, f5]
```

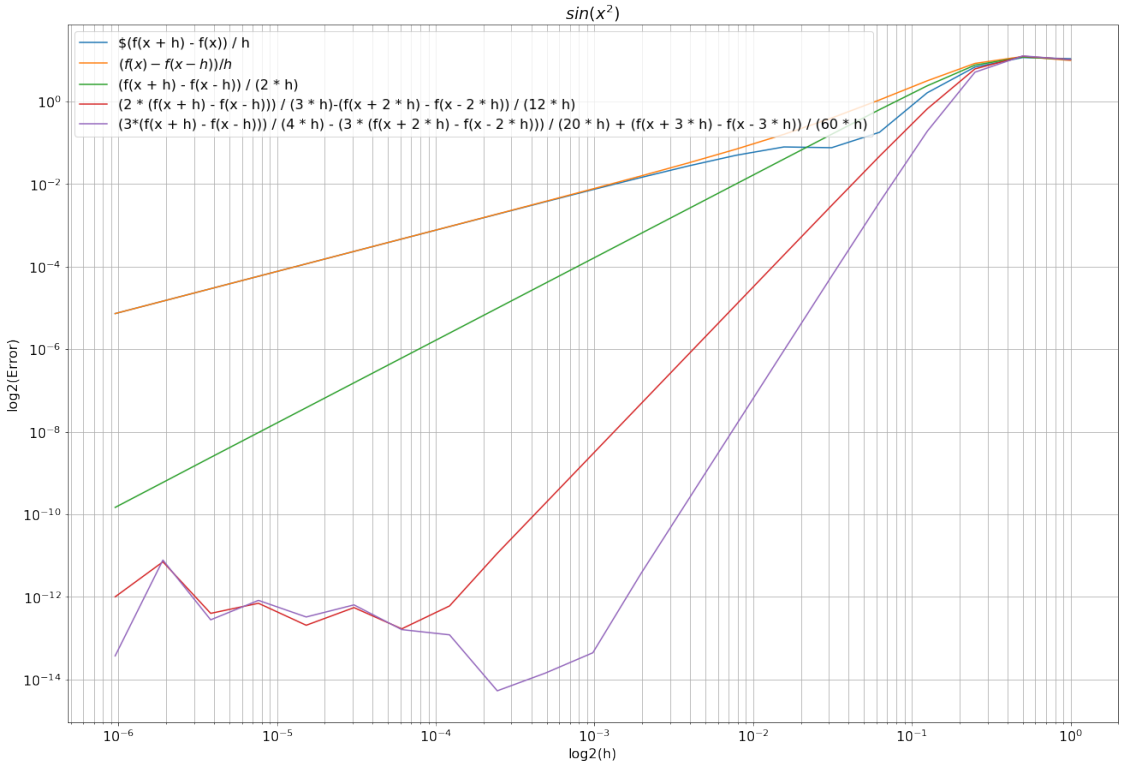
[81]:
```
class Method:
    def __init__(self, method, name):
        self.__method = method
        self.__name = name
    def getMethod(self): return self.__method
    def getName(self):   return self.__name
```

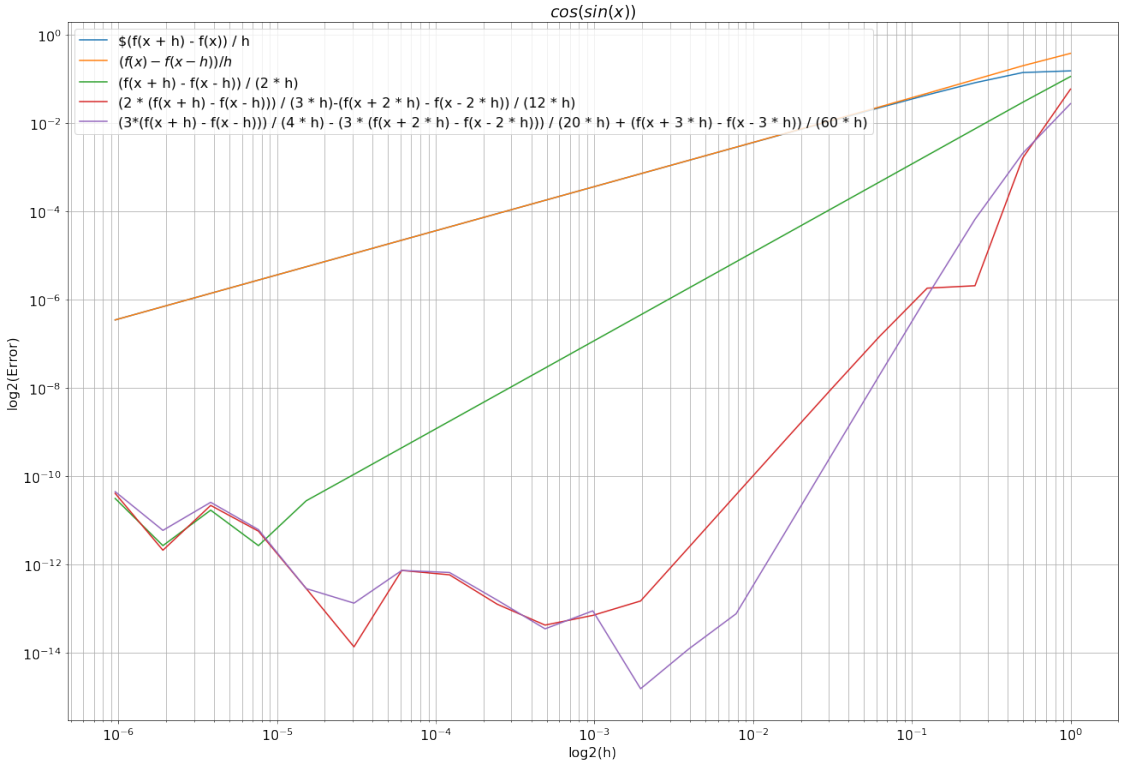
[114]:
```
def approximateDerivative1(f, x, h): return (f(x + h) - f(x)) / h
method1 = Method(approximateDerivative1, "$(f(x + h) - f(x)) / h")

def approximateDerivative2(f, x, h): return (f(x) - f(x - h)) / h
method2 = Method(approximateDerivative2, "$(f(x) - f(x - h)) / h$")

def approximateDerivative3(f, x, h): return (f(x + h) - f(x - h)) / (2 * h)
method3 = Method(approximateDerivative3, "(f(x + h) - f(x - h)) / (2 * h)")

def approximateDerivative4(f, x, h):
    return (2 * (f(x + h) - f(x - h))) / (3 * h)-(f(x + 2 * h) - f(x - 2 * h)) /␣
 ↪(12 * h)
method4 = Method(approximateDerivative4,
          "(2 * (f(x + h) - f(x - h))) / (3 * h)-(f(x + 2 * h) - f(x - 2 * h)) /␣
 ↪(12 * h)")

def approximateDerivative5(f, x, h):
    return (3*(f(x + h) - f(x - h))) / (4 * h) - (3 * (f(x + 2 * h) -
    f(x - 2 * h))) / (20 * h) + (f(x + 3 * h) - f(x - 3 * h)) / (60 * h)
method5 = Method(approximateDerivative5,
          "(3*(f(x + h) - f(x - h))) / (4 * h) - (3 * (f(x + 2 * h) - f(x - 2 *␣
 ↪h))) / (20 * h) + (f(x + 3 * h) - f(x - 3 * h)) / (60 * h)")

methods = [method1, method2, method3, method4, method5]
```

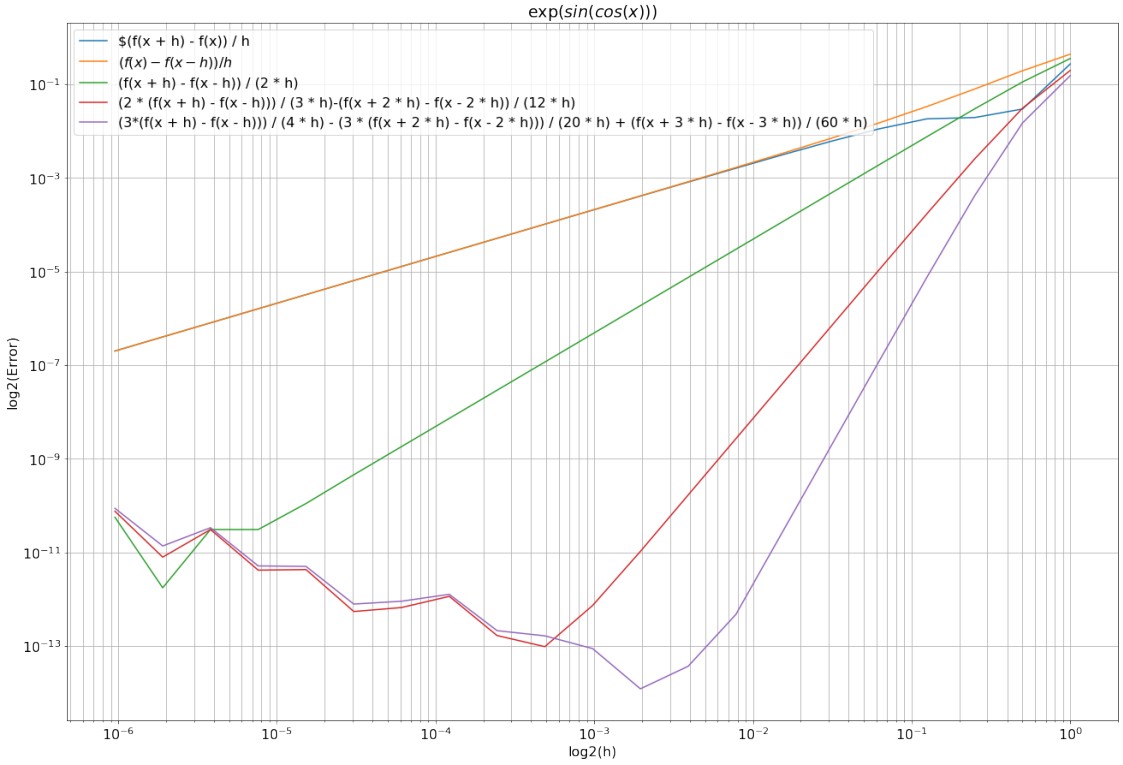
[115]:
```
n = 21
x = 1 / (2**np.arange(n))
x0 = 5
```

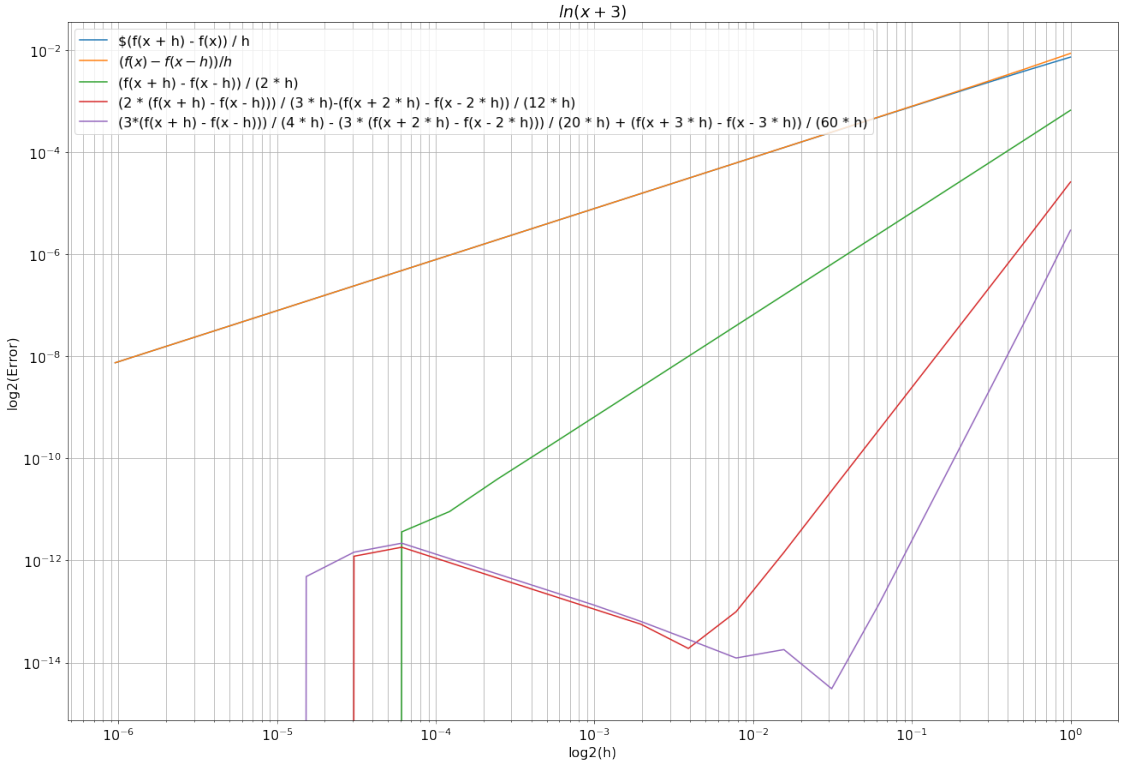
[116]:
```
mpl.rcParams['font.size']=16
for func in funcs:
    plt.figure(figsize=(22,15))
    plt.grid(b=True, which='major', axis='both')
    plt.grid(b=True, which='minor', axis='both')
    for method in methods:
        call = method.getMethod()
        sigma = np.abs(call(func.getFunction(), x0, x) - (func.
 ↪getDerivative())(x0))
        plt.loglog(x, sigma, label = method.getName())
    plt.title(func.getName())
```

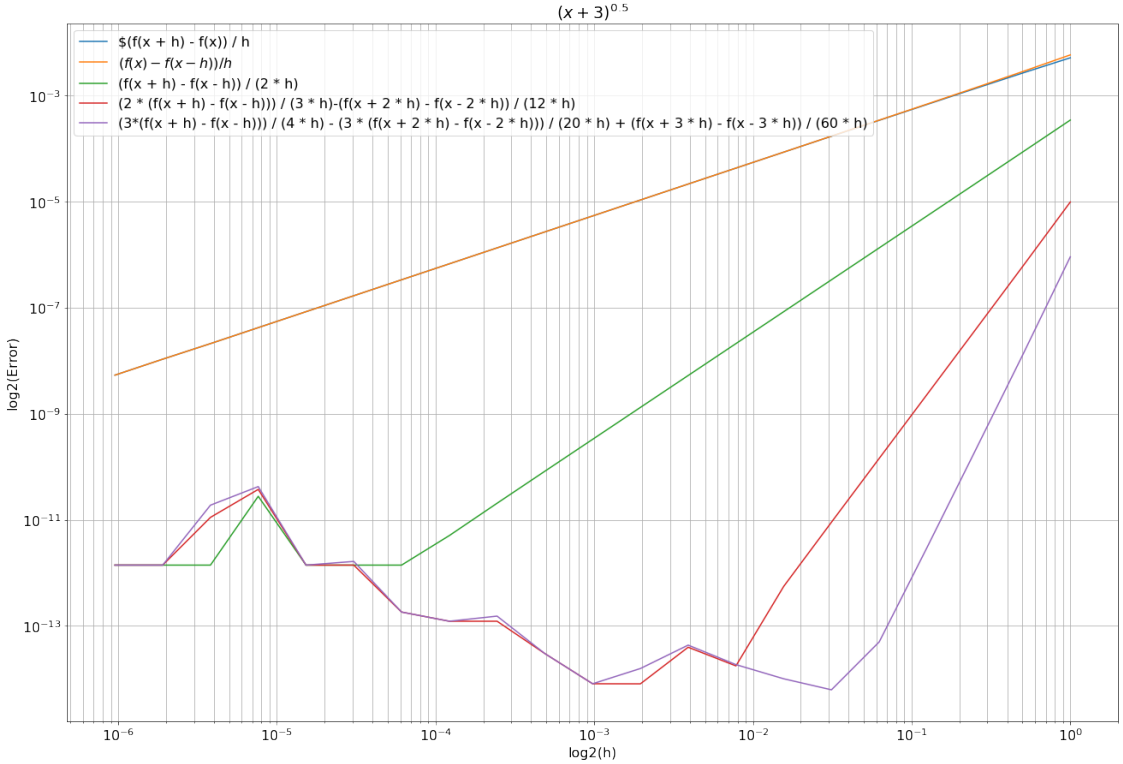
```
    plt.ylabel("log2(Error)")
    plt.xlabel("log2(h)")
    plt.legend()
    plt.show()
```




$cos(sin(x))$
$(f(x + h) - f(x)) / h$
$(f(x) - f(x - h))/h$
(f(x + h) - f(x - h)) / (2 * h)
(2 * (f(x + h) - f(x - h))) / (3 * h)-(f(x + 2 * h) - f(x - 2 * h)) / (12 * h)
(3*(f(x + h) - f(x - h))) / (4 * h) - (3 * (f(x + 2 * h) - f(x - 2 * h))) / (20 * h) + (f(x + 3 * h) - f(x - 3 * h)) / (60 * h)
log2(Error)
log2(h)


exp(sin(cos(x)))
$(f(x + h) - f(x)) / h
$(f(x) - f(x - h))/h
(f(x + h) - f(x - h)) / (2 * h)
(2 * (f(x + h) - f(x - h))) / (3 * h)-(f(x + 2 * h) - f(x - 2 * h)) / (12 * h)
(3*(f(x + h) - f(x - h))) / (4 * h) - (3 * (f(x + 2 * h) - f(x - 2 * h))) / (20 * h) + (f(x + 3 * h) - f(x - 3 * h)) / (60 * h)
log2(Error)
log2(h)


$ln(x + 3)$
$(f(x + h) - f(x)) / h$
$(f(x) - f(x - h))/h$
(f(x + h) - f(x - h)) / (2 * h)
(2 * (f(x + h) - f(x - h))) / (3 * h)-(f(x + 2 * h) - f(x - 2 * h)) / (12 * h)
(3*(f(x + h) - f(x - h))) / (4 * h) - (3 * (f(x + 2 * h) - f(x - 2 * h))) / (20 * h) + (f(x + 3 * h) - f(x - 3 * h)) / (60 * h)
log2(Error)
log2(h)


$(x + 3)^{0.5}$
$(f(x + h) - f(x)) / h$
$(f(x) - f(x - h))/h$
(f(x + h) - f(x - h)) / (2 * h)
(2 * (f(x + h) - f(x - h))) / (3 * h)-(f(x + 2 * h) - f(x - 2 * h)) / (12 * h)
(3*(f(x + h) - f(x - h))) / (4 * h) - (3 * (f(x + 2 * h) - f(x - 2 * h))) / (20 * h) + (f(x + 3 * h) - f(x - 3 * h)) / (60 * h)
log2(Error)
log2(h)

[ ]:

[ ]: