M 348 HOMEWORK 9

ZUN CAO

1. Chapter 5.1 Ex # 4c

2. Chapter 5.1 Ex # 10

3. Chapter 5.1 CP#2

4. Chapter 5.2 Ex # 12

5. Additional Problem:

Date: 4/16/2024.

M348HW9

5.1 #4c. $f'(\frac{\pi}{3}) \approx \frac{f(\frac{\pi}{3}+0.001)-f(\frac{\pi}{3}-0.001)}{2\cdot 0.001}$

$= \frac{\sin(\frac{\pi}{3}+0.001)-\sin(\frac{\pi}{3}-0.001)}{0.002}$

≈ 0.49999991667

$f(x)=\sin x$, $f'(x)=\cos x$, $f''(x)=-\sin x$, $f'''(x)=-\cos x$

error $\approx -\frac{h^2}{6}f'''(c)$ since $c\in(\frac{\pi}{3}-0.001,\frac{\pi}{3}+0.001)$

$-\frac{(0.001)^2}{6}\cdot(-\cos(\frac{\pi}{3}+0.001)) \le \text{error} \le -\frac{(0.001)^2}{6}\cdot(-\cos(\frac{\pi}{3}-0.001))$

$8.3189\times10^{-8} \le \text{error} \le 8.3478\times10^{-8}$

Real number =

$f'(\frac{\pi}{3}) = \cos(\frac{\pi}{3}) = \frac{1}{2} = 0.5$

error $\approx |0.49999991667-0.5| \approx 8.3333\times10^{-8}$

$8.3189\times10^{-8} < 8.3333\times10^{-8} < 8.3478\times10^{-8}$

We find the error lies between the bounds.

10. Write the Taylor theorem for $f(x+h)$ and $f(x-2h)$:

$f(x+h) = f(x)+hf'(x)+\frac{1}{2}h^2f''(x)+\frac{1}{6}h^3f'''(x)+\cdots$ ①

$f(x-2h) = f(x)-2hf'(x)+2h^2f''(x)-\frac{4}{3}h^3f'''(x)+\cdots$ ②

Multiply ① by 4, multiply ② by -1, add them together, we get:

$4f(x+h)-f(x-2h) = 3f(x)+6hf'(x)+0+2h^3f'''(x)+\cdots$ ③

Solving ③ for $f'(x)$, we obtain

$f'(x) = \frac{4f(x+h)-3f(x)-f(x-2h)-2h^3f'''(x)+\cdots}{6h}$

$f'(x) = \frac{4f(x+h)-3f(x)-f(x-2h)}{6h} - \frac{1}{3}h^2f'''(x)+\cdots$

Hence, the order of the FD approximation is 2.

```
C:\Users\13464\AppData\Local\Programs\Python\Python310\python.exe C:\Users\13464\Desktop\M348\HW9\cp2.py
               h  Approximate f'(x)         Error
0   1.000000e-01          -0.250627  6.265664e-04
1   1.000000e-02          -0.250006  6.250156e-06
2   1.000000e-03          -0.250000  6.249998e-08
3   1.000000e-04          -0.250000  6.250280e-10
4   1.000000e-05          -0.250000  9.964474e-12
5   1.000000e-06          -0.250000  3.494449e-11
6   1.000000e-07          -0.250000  1.315890e-10
7   1.000000e-08          -0.250000  1.519368e-09
8   1.000000e-09          -0.250000  2.068509e-08
9   1.000000e-10          -0.250000  2.068509e-08
10  1.000000e-11          -0.250000  2.068509e-08
11  1.000000e-12          -0.250022  2.222515e-05

Process finished with exit code 0
```

FIGURE 1. CP2 table

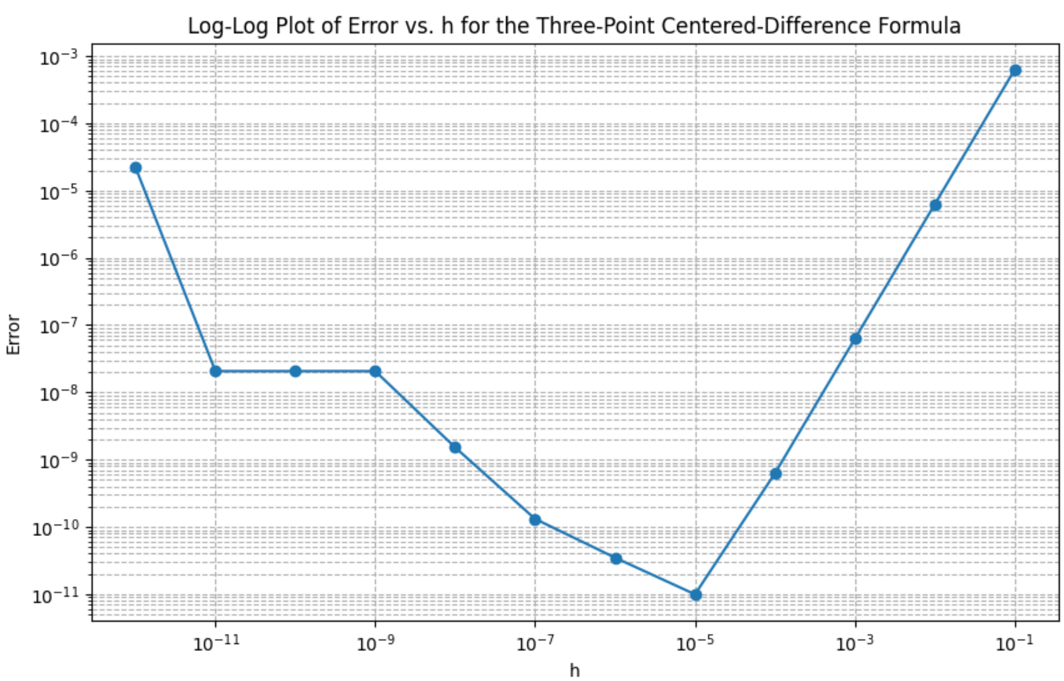

FIGURE 2. CP2 Plot

According to the table and the plot, the minimum error indeed correspond to the theoretical expectation, which is $\left(\frac{3F\epsilon}{M}\right)^{\frac{1}{3}}$, around 10^{-5}. As we can see from the table, when $h = 10^{-5}$, the error is at minimum, which is 9.964474×10^{-12}.

5.2 #12 Need to find C_1, C_2, C_3 s.t. $\int_0^1 f(x)dx \approx C_1 f(0) + C_2 f(0.5) + C_3 f(1)$ d.o.p ≥ 1.

To have degree of precision of 0, we need:

$$\int_0^1 1\,dx = 1 = C_1 f(0) + C_2 f(0.5) + C_3 f(1) = C_1 + C_2 + C_3.$$

To have degree of precision of 1, we need:

$$\int_0^1 x\,dx = \frac{1}{2} = C_1 f(0) + C_2 f(0.5) + C_3 f(1) = 0.5C_2 + 1C_3.$$

To have degree of precision of 2, we need:

$$\int_0^1 x^2 dx = \frac{1}{3} = C_1 f(0) + C_2 f(0.5) + C_3 f(1) = 0.25C_2 + 1C_3.$$

Summarize them:

$$\begin{cases} C_1 + C_2 + C_3 = 1 \\ 0.5C_2 + C_3 = \frac{1}{2} \\ 0.25C_2 + C_3 = \frac{1}{3} \end{cases} \quad \left[\begin{array}{ccc|c} 1 & 1 & 1 & 1 \\ 0 & 0.5 & 1 & \frac{1}{2} \\ 0 & 0.25 & 1 & \frac{1}{3} \end{array}\right] \xrightarrow{R_3 - \frac{1}{2}R_2} \left[\begin{array}{ccc|c} 1 & 1 & 1 & 1 \\ 0 & 0.5 & 1 & \frac{1}{2} \\ 0 & 0 & 0.5 & \frac{1}{12} \end{array}\right]$$

$$\begin{cases} C_1 + C_2 + C_3 = 1 \\ \frac{1}{2}C_2 + C_3 = \frac{1}{2} \\ \frac{1}{2}C_3 = \frac{1}{12} \Rightarrow C_3 = \frac{1}{6}. \end{cases} \quad C_1 = 1 - \frac{2}{3} - \frac{1}{6} = \frac{1}{6} \quad C_2 = \left(\frac{1}{2} - \frac{1}{6}\right) \times 2 = \frac{2}{3} \quad \therefore C = \begin{bmatrix} \frac{1}{6} \\ \frac{2}{3} \\ \frac{1}{6} \end{bmatrix} \begin{matrix} = C_1 \\ = C_2 \\ = C_3. \end{matrix}$$

Plug in C, we find that

$$\int_0^1 f(x)dx = \frac{1}{6} f(0) + \frac{2}{3} f(0.5) + \frac{1}{6} f(1)$$

$h = \frac{1-0}{2} = \frac{1}{2}$ $\rightarrow$ This is same with the Simpon's Rule

$$\left(\int_{x_0}^{x_2} f(x)dx = \frac{h}{3}(y_0 + 4y_1 + y_2) - \frac{h^5}{90} f^{(4)}(c) \right)$$

$x_0 = 0$ $\quad h = \frac{1}{2}$ $\quad y_0 = f(0)$

$x_2 = 1$ $\quad y_1 = f(0.5)$

$y_2 = f(1)$

AP1. $\int_0^1 e^x dx$

(a) Using Composite Midpoint Rule, we get:

$x_i = a + ih$

$$\int_0^1 e^x dx \approx h\sum_{i=1}^{m} f\left(\frac{x_{i-1}+x_i}{2}\right) + \underbrace{\frac{b-a}{24} f''(c) h^2}_{\text{error}}$$

$h = \frac{b-a}{m}$ $\quad c \in [a,b]$

$f(x) = e^x$

$f''(x) = e^x \rightarrow |f''(c)| = e^c \quad , c \in [0,1]$

$\text{Error} = \left|\frac{b-a}{24} f''(c) h^2\right| = \left|\frac{h^2}{24} e^c\right| \leq \frac{h^2 e}{24} \leq \frac{1}{2}\cdot 10^{-2}$ for 2 decimal places.

$h^2 \leq \frac{12}{e}\cdot 10^{-2} \qquad h \leq \frac{\sqrt{12}}{10\sqrt{e}} \approx 0.210108$

$m = \frac{1}{h} \Rightarrow m \geq \frac{10\sqrt{e}}{\sqrt{12}} \approx 4.75945 \quad \boxed{\therefore m = 5 \ , \ h = \frac{1}{m} = \frac{1}{5} = 0.2}$

$\therefore$ we need at least $m=5$ panels for 2 correct decimal places.

(b) $\int_0^1 e^x dx \approx \frac{1}{5}\sum_{i=1}^{5} f\left(\frac{x_{i-1}+x_i}{2}\right)$

$= \frac{1}{5}\left[f\left(\frac{x_0+x_1}{2}\right) + f\left(\frac{x_1+x_2}{2}\right) + f\left(\frac{x_2+x_3}{2}\right) + f\left(\frac{x_3+x_4}{2}\right) + f\left(\frac{x_4+x_5}{2}\right)\right]$

$= \frac{1}{5}\left[f(0.1) + f(0.3) + f(0.5) + f(0.7) + f(0.9)\right]$

$= \frac{1}{5}\left[e^{0.1} + e^{0.3} + e^{0.5} + e^{0.7} + e^{0.9}\right]$

$\approx \underline{1.715421}$

(c) $\int_0^1 e^x = e^x\Big|_0^1 = e^1 - 1 \approx \underline{1.718282}$

(d) $\text{error} = |1.715421 - 1.718282| \approx 0.00286047$

$= \underbrace{0.286047}_{<0.5} \times 10^{-2}$ 2 correct decimal places

```python
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd


def f(x):
    return (1 + x) ** (-1)

def f_prime(x):
    return -(1 + x) ** (-2)


x = 1
exact_derivative = f_prime(x)

h = np.logspace(-1, -12, num=12, base=10)

# Calculate derivatives using the three-point centered-difference formula
approx_derivatives = (f(x + h) - f(x - h)) / (2 * h)

# Calculate errors
errors = np.abs(approx_derivatives - exact_derivative)

data = {'h': h, 'Approximate f\'(x)': approx_derivatives, 'Error': errors}
data_frame = pd.DataFrame(data)
print(data_frame)

# Plotting the results
plt.figure(figsize=(10, 6))
plt.loglog(h, errors, marker='o')
plt.xlabel('h')
plt.ylabel('Error')
plt.title('Log-Log Plot of Error vs. h for the Three-Point Centered-
                                        Difference Formula')
plt.grid(True, which="both", ls="--")
plt.show()
```