Laboratorium 1 - Analiza błędów

Mateusz Podmokły - II rok Informatyka WI

29 luty 2024

1 Treść zadania

Zadanie 1. Oblicz przybliżoną wartość pochodnej funkcji, używajac wzoru

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

Sprawdź działanie programu dla funkcji $tan(x)$ oraz $x = 1$. Wyznacz błąd, porównujac otrzymaną wartość numerycznej pochodnej z prawdziwą wartością. Na wspólnym rysunku przedstaw wykresy wartości bezwględnej błędu metody, błędu numerycznego oraz błędu obliczeniowego w zaleznosci od h dla $h = 10^{-k}$, $k = 0, ..., 16$. Porównaj wyznaczoną wartość h_{min} z wartością otrzymaną ze wzoru

$$h_{min} \approx 2\sqrt{\frac{\epsilon_{mach}}{M}},$$

$$M \approx \left|f''(x)\right|.$$

Powtórz cwiczenie uzywajac wzoru róznic centralnych

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

Porównaj wyznaczoną wartość h_{min} z wartością otrzymaną ze wzoru

$$h_{min} \approx \sqrt[3]{\frac{3\epsilon_{mach}}{M}},$$

$$M \approx \left|f'''(x)\right|.$$

Zadanie 2. Napisz program generujący pierwsze n wyrazów ciągu zdefiniowanego równaniem róznicowym:

$$x_{k+1} = 2.25x_k - 0.5x_{k-1}$$

z wyrazami początkowymi:

$$x_0 = \frac{1}{3},$$

$$x_1 = \frac{1}{12}.$$

Wykonaj obliczenia:

- używajżc pojedynczej precyzji oraz przyjmując $n = 225$,
- uzywając podwójnej precyzji oraz przyjmując $n = 60$,
- uzywając reprezentacji z biblioteki `fractions` oraz przyjmujac $n = 225$.

Narysuj wykres wartości ciągu w zależności od k. Następnie narysuj wykres przedstawiający wartość bezwględną błędu względnego w zależnosci od k.
Dokładne rozwiązanie równania róznicowego:

$$x_k = \frac{4^{-k}}{3}$$

maleje wraz ze wzrostem k. Czy otrzymany wykres zachowuje się w ten sposób?

2 Specyfikacja użytego środowiska

Specyfikacja:

- Środowisko: Visual Studio Code,
- Język programowania: Python,
- System operacyjny: Microsoft Windows 11,
- Architektura systemu: x64.

3 Rozwiązanie problemu

W realizacji rozwiązania wykorzystane zostały następujące biblioteki:

```
import numpy as np
import matplotlib.pyplot as plt
from fractions import Fraction
```

3.1 Zadanie 1.

Prawdziwa wartość $(tan(x))'$ obliczona została ze wzoru $(tan(x))' = tan^2(x) + 1$. Błąd obliczeniowy obliczany jest jako różnica numerycznej i prawdziwej wartości pochodnej w stosunku do wartości prawdziwej.
Dla wzoru różnic w przód mamy

$$\frac{\left|tan^2(x) + 1 - \frac{f(x+h)-f(x)}{h}\right|}{tan^2(x) + 1},$$

a dla wzrou różnic centralnych

$$\frac{\left|tan^2(x) + 1 - \frac{f(x+h)-f(x-h)}{2h}\right|}{tan^2(x) + 1}.$$

Błąd metody otrzymujemy ze wzoru

$$\frac{Mh}{2},$$

$$M = |f''(x)| = tan^3(x) + tan(x)$$

dla różnic w przód oraz

$$\frac{Mh^2}{6},$$

$$M = |f'''(x)| = 6tan^4(x) + 8tan^2(x) + 2.$$

dla wzoru różnic centralnych. Natomiast błąd numeryczny ze wzorów

$$\frac{2\epsilon}{h}$$

oraz

$$\frac{\epsilon}{h}.$$

3.2 Zadanie 2.

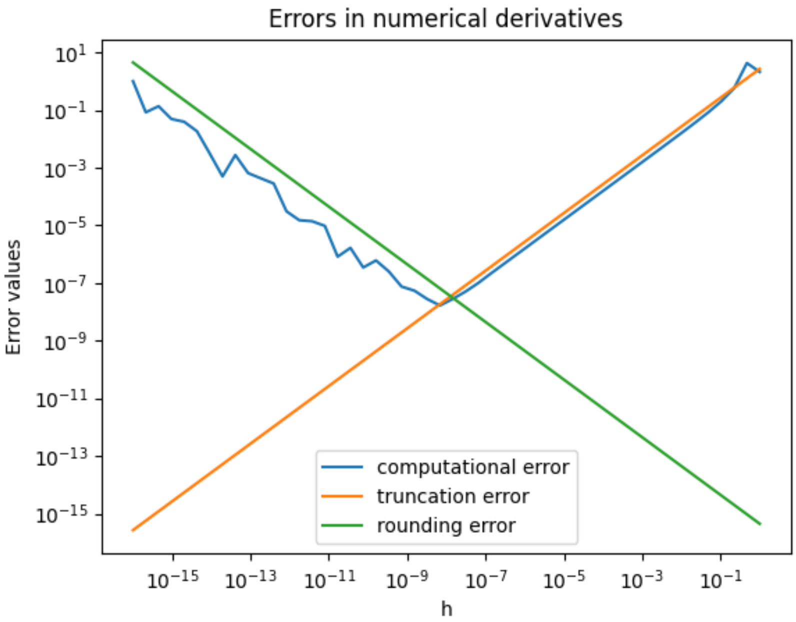
Kolejne elementy ciągu były generowane iteracyjnie na podstawie poprzednich wartości. Pojedyncza precyzja została ustawiona z wykorzystaniem funkcji biblioteki `NumPy` `np.float32`, a podwójna precyzja `np.float64`.

```
sequence3 = np.array([Fraction(1,3), Fraction(1,12)])
for i in range(n - 2):
    sequence3 = np.append(sequence3, Fraction(9,4) * sequence3[-1] - Fraction(1,2) * sequence3[-2])
```

Rysunek 1: Generowanie kolejnych elementów ciągu z użyciem klasy `Fraction` z biblioteki `fractions`.

4 Przedstawienie wyników

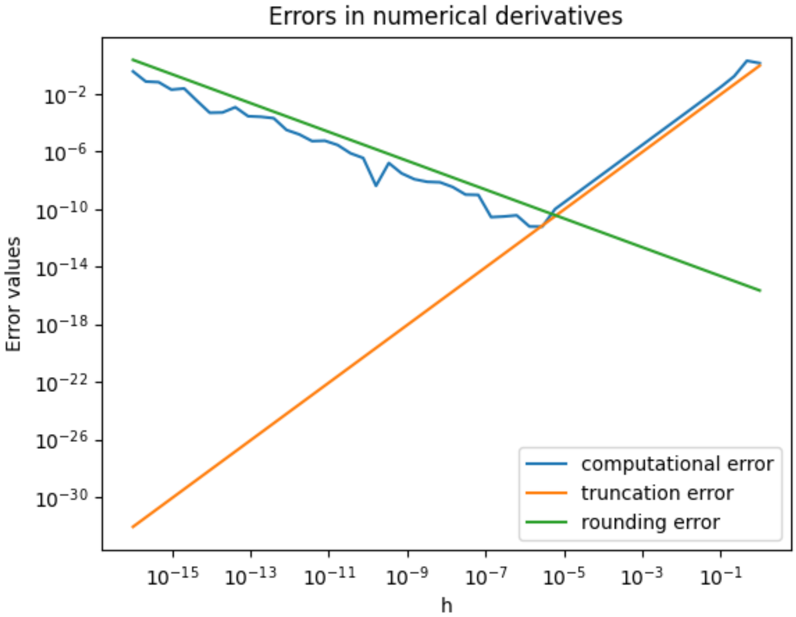
4.1 Zadanie 1.



Rysunek 2: Błędy dla wzoru różnic w przód.

$$h_{min} = 6.866488450042998 \cdot 10^{-9}$$

$$h_{min} \approx 2\sqrt{\frac{\epsilon_{mach}}{M}} \approx 1.2902853526408846 \cdot 10^{-8}$$

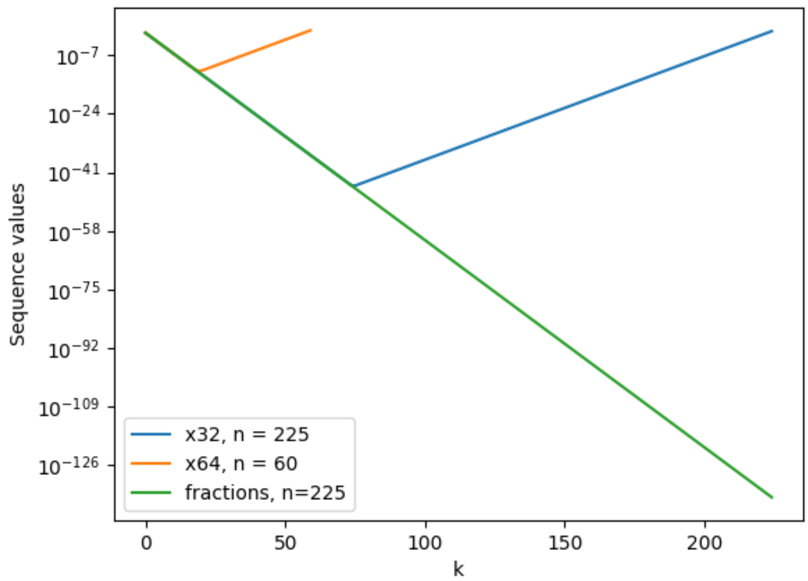


Rysunek 3: Błędy dla wzoru różnic centralnych.

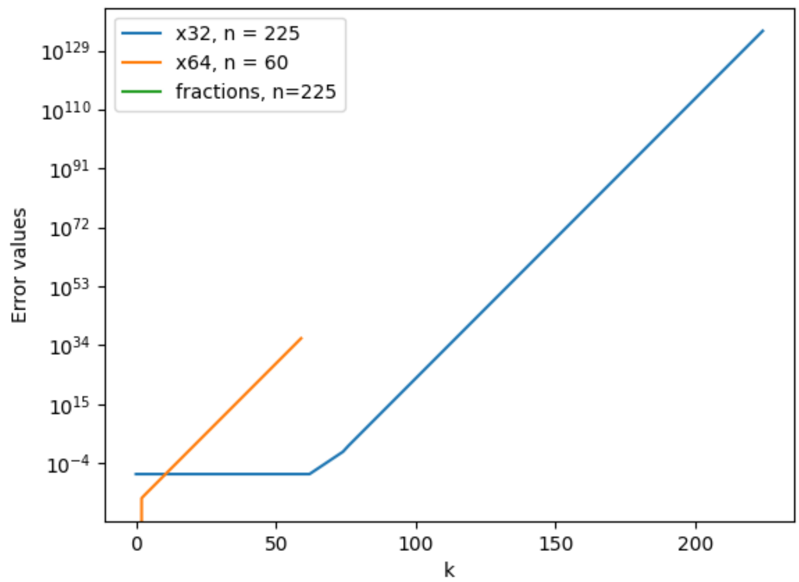
$$h_{min} = 2.811768697974231 \cdot 10^{-6}$$

$$h_{min} \approx \sqrt[3]{\frac{3\epsilon_{mach}}{M}} \approx 2.2732741568390613 \cdot 10^{-6}$$

4.2 Zadanie 2.



Rysunek 4: Wartości ciągu generowane różnymi metodami.



Rysunek 5: Wartości błędu względnego generowane różnymi metodami.

W przypadku użycia biblioteki `fractions` wartość błędu dla $n = 225$ jest stale równa 0.

5 Wnioski

5.1 Zadanie 1.

W obliczeniach z użyciem wzoru różnic centralnych wartości błędu obliczeniowego mają znacznie większy zakres górny i dolny. Najmniejsza wartość błędu obliczeniowego przyjmowana jest dla $h_{min} \approx 6.87 \cdot 10^{-9}$ w przypadku wzoru różnic w przód, a dla wzoru różnic centralnych $h_{min} \approx 2.81 \cdot 10^{-6}$. Jest ono zbliżone do prawdziwego. Dla mniejszych h niż h_{min} wartość błędu obliczeniowego zaczyna rosnąć. Może to być spowodowane zbyt małymi wartościami h wykraczającymi poza precyzję obliczeń.

5.2 Zadanie 2.

Kolejne wartości ciągu, wyznaczane z użyciem pojedynczej precyzji, znacznie szybciej zaczynają rosnąć, niż w przypadku podwójnej precyzji. Ze wzoru jawnego ciągu wiemy, że jest on malejący. Zmiana monotoniczności wynika prawdopodobnie z przekroczenia dokładności liczby zmiennoprzecinkowej w języku Python. Użycie klasy `Fraction` z biblioteki `fractions` pozwala na dokładne wyznaczanie wartości ciągu dla $n = 225$, ponieważ przechowujemy osobno licznik i mianownik, dzięki czemu możemy uniknąć utraty

precyzji.
Z powyższego wynika błąd względny, który dla obliczeń pojedynczej precyzji znacznie szybciej zaczyna rosnąć niż w przypadku obliczeń podwójnej precyzji. Obliczenia z użyciem biblioteki `fractions` pozwalają na utrzymanie błędu względnego na poziomie 0.

5.3 Podsumowanie

Dzięki wykonanym obliczeniom można zauważyć, że arytmetyka zmiennoprzecinkowa w komputerze nie jest idealna. Wykonując obliczenia numeryczne należy pamiętać o ograniczeniach sprzętowych komputerów, ponieważ nierozważne wykonywanie obliczeń może prowadzić do poważnych błędów.

6 Bibliografia

`http://heath.cs.illinois.edu/scicomp/notes/cs450_chapt01.pdf`
`https://pl.wikipedia.org/wiki/Liczba_zmiennoprzecinkowa`
`https://pl.wikipedia.org/wiki/B%C5%82%C4%85d_przybli%C5%BCenia`