Interpolacja

Joanna Klimek

Informatyka, WIET

05.04.2021r.

Niestety nie udało mi się wykonać 2 pierwszych zadań.

Materiały (kod w Pythonie i odręczne obliczenia) dołączone w archiwum.

Zadanie 3.

Wykonaj interpolacje funkcji $f(x) \; = |sin(x)|$ w przedziale $[-4, 4]$ przy użyciu wielomianów Lagrange'a 2-go, 5-go oraz 10-go stopnia, dla równoodległych węzłów interpolacji. Narysuj wykres i zinterpretuj go. Jakie wiele węzłów interpolacji byłoby wymaganych aby maksymalny błąd interpolacji nie przekraczał 10^{-10}?

Wskazówka. Wykorzystaj oszacowanie (6) z zadania 2.

a) wielomiany Lagrange'a 2-go stopnia

Przedział $[-4, 4]$ dzielimy na 2 równe części – stąd nasze węzły interpolacji i ich wartości:

$\boldsymbol{x}$	-4.0	0.0	4.0
$\boldsymbol{f(x) \; = |sin(x)|}$	0.7568	0.0	0.7568

Korzystając z wielomianu Lagrange'a:

$$w(x) = \; f(x_1)\frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} + f(x_2)\frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)} + f(x_3)\frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}$$

$$w(x) = \; 0.7568 * \frac{(x-0.0)(x-4.0)}{(-4.0-0.0)(-4.0-4.0)} + 0.0 * \frac{(x-(-4.0))(x-4.0)}{(0.0-(-4.0))(0.0-4.0)} + 0.7568\frac{(x-(-4.0))(x-0.0)}{(4.0-(-4.0))(4.0-0.0)}$$

$$w(x) = \; 0.0473x^2$$

b) wielomiany Lagrange'a 5-go stopnia

Przedział $[-4, 4]$ dzielimy na 5 równych części – stąd nasze węzły interpolacji i ich wartości:

$\boldsymbol{x}$	-4.0	- 2.4	- 0.8	0.8	2.4	4.0
$\boldsymbol{f(x)} = \boldsymbol{\lvert sin(x) \rvert}$	0.7568	0.6755	0.7174	0.7174	0.6755	0.7568

$$\begin{aligned} w(x) = \ & f(x_1)\frac{(x-x_2)(x-x_3)(x-x_4)(x-x_5)(x-x_6)}{(x_1-x_2)(x_1-x_3)(x_1-x_4)(x_1-x_5)(x_1-x_6)} \\ & + f(x_2)\frac{(x-x_1)(x-x_3)(x-x_4)(x-x_5)(x-x_6)}{(x_2-x_1)(x_2-x_3)(x_2-x_4)(x_2-x_5)(x_2-x_6)} \\ & + f(x_3)\frac{(x-x_1)(x-x_2)(x-x_{34})(x-x_5)(x-x_6)}{(x_3-x_1)(x_3-x_2)(x_3-x_4)(x_3-x_5)(x_3-x_6)} \\ & + f(x_4)\frac{(x-x_1)(x-x_2)(x-x_3)(x-x_5)(x-x_6)}{(x_4-x_1)(x_4-x_2)(x_4-x_3)(x_4-x_5)(x_4-x_6)} \\ & + f(x_5)\frac{(x-x_1)(x-x_2)(x-x_3)(x-x_4)(x-x_6)}{\big(x_5-x_1(x_5-x_2)\big)(x_5-x_3)(x_5-x_4)(x_5-x_6)} \\ & + f(x_6)\frac{(x-x_1)(x-x_2)(x-x_3)(x-x_4)(x-x_5)}{(x_6-x_1)(x_6-x_2)(x_6-x_3)(x_6-x_4)(x_6-x_5)} \end{aligned}$$

c) wielomiany Lagrange'a 10-go stopnia

Przedział $[-4, 4]$ dzielimy na 10 rówych części – stąd nasze węzły interpolacji i ich wartości:

$\boldsymbol{x}$	$\boldsymbol{f(x)} = \boldsymbol{\lvert sin(x) \rvert}$
-4.0	0.7568
- 3.2	0.0584
- 2.4	0.6755
- 1.6	0.9996
- 0.8	0.7174
0.0	0.000
0.8	0.7174
1.6	0.9996
2.4	0.6755
3.2	0.0584
4.0	0.7568

Wielomian budowany aanalogicznie jak w poprzednich podpunktach.

Kod pomocniczy do zadania (dodany również w materiałach):

```
from math import sin
import numpy as np
import matplotlib.pyplot as plt
from numpy.polynomial.polynomial import Polynomial

start = -4
end = 4

length = end - start

deg = [2, 5, 10]



for n in deg:
    dist = length / n
    i = 0

    result_x = []
    result_y = []

    print("\n\n=======================================")
    print("\nstopien = ", n)

    while i <= n:

        x = round(start + (dist * i), 1)
        y = abs(sin(x))

        print(i + 1, ". x =", x, "y =", y)

        result_x.append(x)
        result_y.append(y)

        i += 1

    wynikowy_wielomian = Polynomial([0.])

    for i in range(n + 1):
        pierwiastki = []

        for j in range(n + 1):
            if j != i:
                pierwiastki.append(result_x[j])

        wielomian = Polynomial.fromroots(pierwiastki)

        mianownik = 1

        for j in range(n + 1):
            if j != i:
                mianownik *= (result_x[i] - result_x[j])

        wspolczynnik = result_y[i] / mianownik

        #print("\nwspolczynnik:", wspolczynnik)
        #print("wielomian:", wielomian)

        wielomian = wielomian * wspolczynnik
```

```
        wynikowy_wielomian = wynikowy_wielomian + wielomian

    print("\nwynikowy wielomian:")
    print(wynikowy_wielomian)


    x = np.linspace(-4, 4, 100)
    y = wynikowy_wielomian(x)

    fig, ax = plt.subplots()
    ax.plot(x, y, label="w(x)")
    ax.set_xlabel("x")
    ax.set_ylabel("y")
    ax.legend()
    plt.show()
```

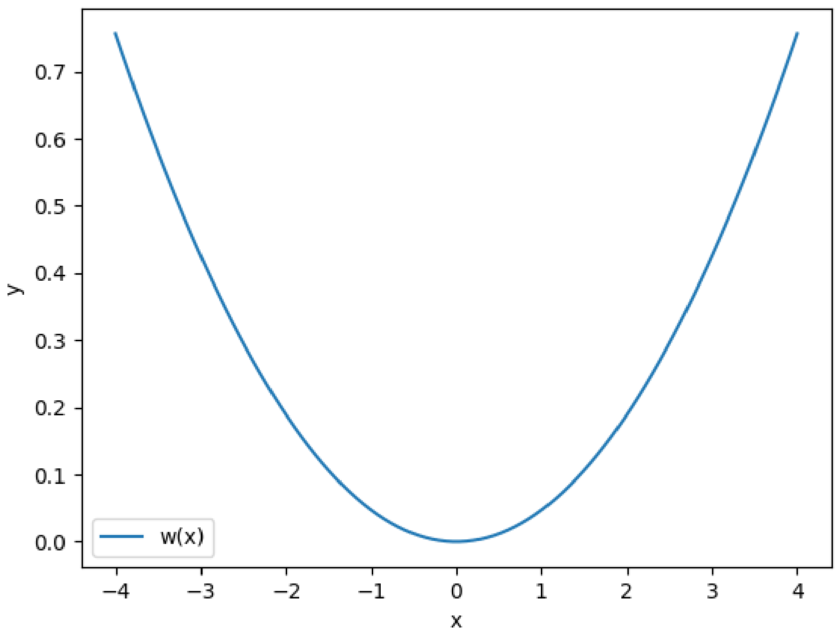
Wyniki działania:

a:

```
========================================


stopien =  2
1 . x = -4.0 y = 0.7568024953079282
2 . x = 0.0 y = 0.0
3 . x = 4.0 y = 0.7568024953079282

wynikowy wielomian:
0.0 + 0.0 x**1 + 0.04730015595674551 x**2
```

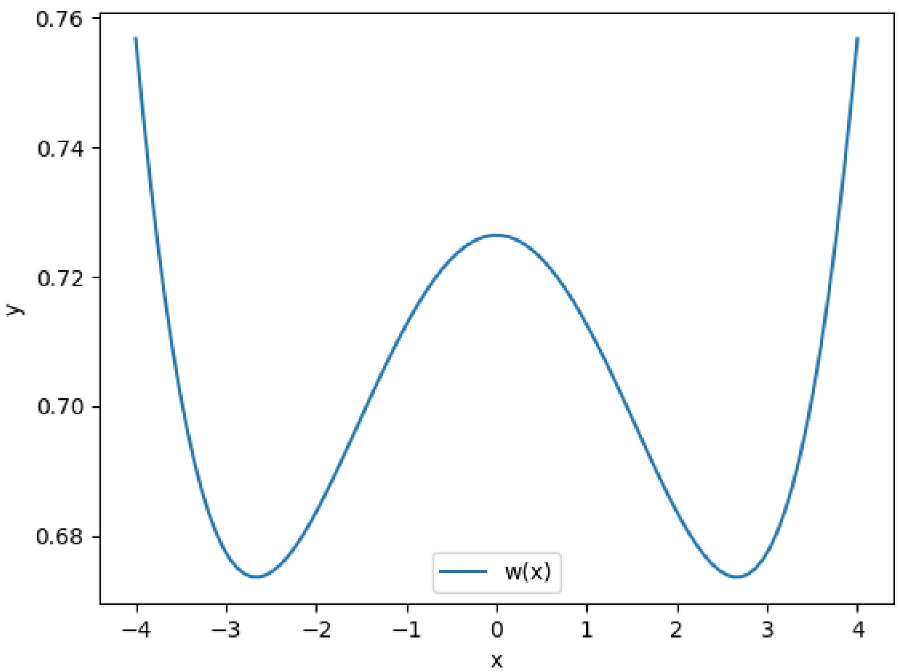


b:

```
======================================

stopien =  5
1 . x = -4.0 y = 0.7568024953079282
2 . x = -2.4 y = 0.675463180551151
3 . x = -0.8 y = 0.7173560908995228
4 . x = 0.8 y = 0.7173560908995228
5 . x = 2.4 y = 0.675463180551151
6 . x = 4.0 y = 0.7568024953079282

wynikowy wielomian:
0.7264628250552612 + 1.0885389811754465e-16 x**1 -
0.014901168014555087 x**2 - 2.168404344971009e-17 x**3 +
0.0010498373378341722 x**4 + 7.589415207398531e-19 x**5
```

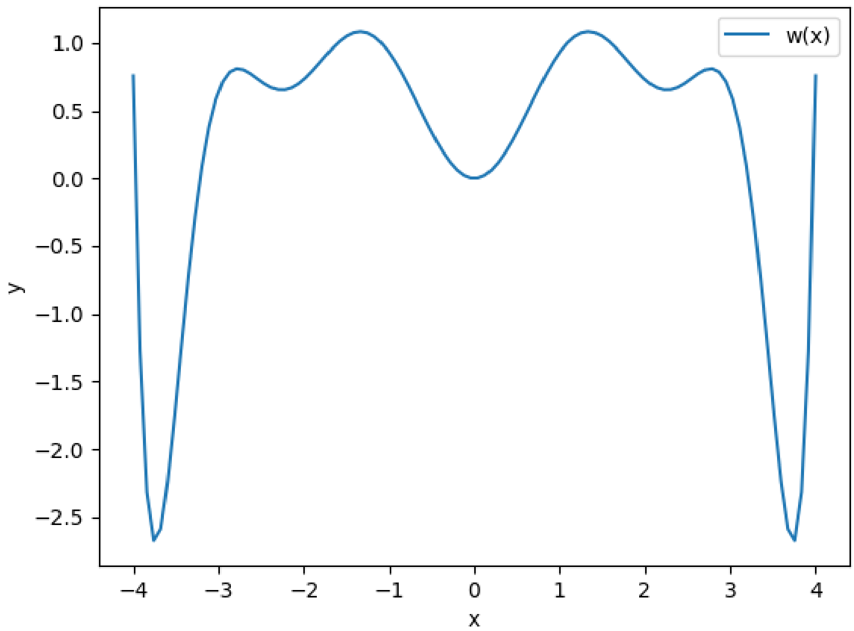


c:

```
=======================================

stopien =  10
1 . x = -4.0 y = 0.7568024953079282
2 . x = -3.2 y = 0.058374143427580086
3 . x = -2.4 y = 0.675463180551151
4 . x = -1.6 y = 0.9995736030415051
5 . x = -0.8 y = 0.7173560908995228
6 . x = 0.0 y = 0.0
7 . x = 0.8 y = 0.7173560908995228
8 . x = 1.6 y = 0.9995736030415051
9 . x = 2.4 y = 0.675463180551151
10 . x = 3.2 y = 0.058374143427580086
11 . x = 4.0 y = 0.7568024953079282

wynikowy wielomian:
0.0 + 3.6288246713089833e-16 x**1 + 1.5380540074476743 x**2 -
3.5366674866477155e-16 x**3 - 0.7364440342258896 x**4 +
1.4213890481284963e-16 x**5 + 0.13922774590338372 x**6 -
5.8817967857338616e-18 x**7 - 0.011220464534971936 x**8 +
3.3542504711270293e-19 x**9 + 0.00031446945923007427 x**10
```



Wszystkie reprezentacje odbiegają od rzeczywistego wyniku, jednak wraz ze wzrostem stopni wielomianów, poprawia się dokładność. W ostatnim wykresie warto zauważyć, że na końcach przedziałów wyniki znacznie odbiegają od poprawnych – jest to efekt Rungego, spowodowany nieodpowiednim rozłożeniem punktów interpolacji.

Zadanie 4.

a) Oblicz wielomian interpolacyjny dla danych (0.5, 5.5), (1, 14.5), (1.5, 32.5), (2, 62.5) przy pomocy jednomianów.
b) Oblicz wielomian interpolacyjny Lagrange'a dla tych samych danych i pokaz, ze wielomian będzie ten sam co w (a).
c) Oblicz wielomian interpolacyjny Newtona dla tych samych danych korzystając z metody trójkąta różnic dzielonych.

Pokaz, ze każda reprezentacja daje ten sam wielomian.

a) Postać wielomianu: $w(x) = ax^3 + bx^2 + cx + d$.

Po podstawieniu podanym punktów:

$$\begin{cases} a(0.5)^3 + b(0.5)^2 + c(0.5) + d = 5.5 \\ a(1)^3 + b(1)^2 + c(1) + d = 14.5 \\ a(1.5)^3 + b(1.5)^2 + c(1.5) + d = 32.5 \\ a(2)^3 + b(2)^2 + c(2) + d = 62.5 \end{cases}$$

$$\begin{cases} a = 4 \\ b = 6 \\ c = 2 \\ d = 2{,}5 \end{cases}$$

Otrzymany wielomian interpolacyjny: $\boldsymbol{w(x) = 4x^3 + 6x^2 + 2x + 2.5}$.

b) Korzystając ze wzorów:

$$L_k(x) = \prod_{i=0, i\neq k}^{n} \frac{x - x_i}{(x_k - x_i)}$$

$$P_n(x) = \sum_{k=0}^{n} f(x_k) * L_k(x)$$

$$w(x) = f(x_1)\frac{(x-x_2)(x-x_3)(x-x_4)}{(x_1-x_2)(x_1-x_3)(x_1-x_4)} + f(x_2)\frac{(x-x_1)(x-x_3)(x-x_4)}{(x_2-x_1)(x_2-x_3)(x_2-x_4)} +$$
$$f(x_3)\frac{(x-x_1)(x-x_2)(x-x_4)}{(x_3-x_1)(x_3-x_2)(x_3-x_4)} + f(x_4)\frac{(x-x_1)(x-x_2)(x-x_3)}{(x_4-x_1)(x_4-x_2)(x_4-x_3)}$$

$$w(x) = 5.5 * \frac{(x-1)(x-1.5)(x-2)}{(0.5-1)(0.5-1.5)(0.5-2)} + 14.5 * \frac{(x-0.5)(x-1.5)(x-2)}{(1-0.5)(1-1.5)(1-2)} +$$

$$32.5 * \frac{(x-0.5)(x-1)(x-2)}{(1.5-0.5)(1.5-1)(1.5-2)} + 62.5\frac{(x-0.5)(x-1)(x-1.5)}{(2-0.5)(2-1)(2-1.5)}$$

$$\boldsymbol{w(x) = \ 4x^3 + 6x^2 + 2x + 2.5}$$

c) Korzystając z trójkąta różnic dzielonych:

x_0	$f(x_0)$			
x_1	$f(x_1)$	$f[x_0, x_1]$		
x_2	$f(x_2)$	$f[x_1, x_2]$	$f[x_0, x_{1,} x_2]$	
x_3	$f(x_3)$	$f[x_2, x_3]$	$f[x_1, x_2, x_{3,}]$	$f[x_0, x_1, x_{2,} x_3]$

Po podstawieniu:

0.5	5.5			
1	14.5	$\frac{14.5-5.5}{1-0.5} = 18$		
1.5	32.5	$\frac{32.5-14.5}{1.5-1} = 36$	$\frac{36-18}{1.5-0.5} = 18$	
2	62.5	$\frac{62.5-32.5}{2-0.5} = 60$	$\frac{60-36}{2-1} = 24$	$\frac{24-18}{2-0.5} = 4$

Wielomian interpolacyjny:

$$W(x) = f(x_0) + (x-x_0) * f[x_0, x_1] + (x-x_0)(x-x_1) * f\left[x_0, x_{1,} x_2 \right] \\ + (x-x_0)(x-x_1)(x-x_2) * f[x_0, x_1, x_{2,} x_3]$$

$$W(x) = 5.5 + (x-0.5) * 18 + (x-0.5)(x-1) * 18 + (x-0.5)(x-1)(x-1.5) * 4$$

$$\boldsymbol{W(x) = 4x^3 + 6x^2 + 2x + 2.5}$$

Widać, że każda reprezentacja dała ten sam wielomian jako wynik.

Zadanie 5.

Wyraź następujący wielomian metodą Hornera: $p(t) = 3t^3 - 7t^2 + 5t - 4$.

$$p(t) = 3t^3 - 7t^2 + 5t - 4 = (3t^2 - 7t + 5)t - 4 = \boldsymbol{\big((3t - 7)t + 5\big)t - 4}$$

Zadanie 6.

Ile mnożeń trzeba wykonać do ewaluacji wielomianu p(t) stopnia n - 1 w danym punkcie t jeżeli wybieramy jako reprezentacje:

a) Jednomiany

$$w(x) = a_0 + a_1 x + \cdots + a_{n-1}x^{n-1}$$

Do obliczenia wyrażenia x^k potrzeba k-1 mnożeń. Wraz ze współczynnikiem a_k daje nam to k mnożeń ja k-tego jednomianu. Po zsumowaniu:

$$0 + 1 + 2 + .. + n - 1 = \boldsymbol{\frac{(n-1)n}{2}}$$

Natomiast stosując metodę Hornera (jak w zadaniu 5) otrzymamy **n** mnożeń.

b) wielomiany Lagrange'a

$$L_k(x) = \prod_{i=0, i \neq k}^{n-1} \frac{x - x_i}{(x_k - x_i)}$$

$$P_{n-1}(x) = \sum_{k=0}^{n-1} f(x_k) * L_k(x)$$

W przypadku $L_k(x)$ potrzebujemy wymnożyć n czynników (ułamków), co daje nam n-1 mnożeń. Każde $L_k(x)$ mnożymy jeszcze, przez $f(x_k)$ co daje dla każdego składnika sumy n mnożeń. Składników jest n, więc ostatecznie otrzymujemy $\boldsymbol{n^2}$ mnożeń.

Jeżeli rozważalibyśmy dzielenie jako mnożenie przez odwrotność mianownika to do obliczenia $L_k(x)$ potrzebowalibyśmy $n + n - 1 = 2n - 1$ mnożeń, a ostateczny wynik wyniósł by $(2n - 1 + 1) * n = \boldsymbol{2n^2}$.

c) wielomiany Newton'a

$$W(x) = \sum_{k=0}^{n-1} b_k * p_k(x)$$

$$p_k(x) = \prod_{i=0}^{k-1} (x - x_i)$$

Obliczenie wyrazu $p_k(x)$ wymaga k mnożeń, a co za tym idzie $b_k * p_k(x)$ potrzebuje k+1 mnożeń. Liczba k przyjmuje wartości od 0 do n-1, więc ilość mnożeń wynosi:

$$1 + 2 + .. + n = \frac{\boldsymbol{n(n+1)}}{\boldsymbol{2}}$$