Київський національний університет імені Тараса Шевченка
Факультет комп’ютерних наук та кібернетики

Чисельні методи в інформатиці

Лабораторна робота №4

“Інтерполяція”

Варіант №7

Виконала студентка групи ІПС-31

Сенечко Дана Володимирівна

Київ - 2025

Постановка задачі

Виконати наближення функції, побудувати графіки функції та наближення.

1. Побудувати інтерполяційний поліном Лагранжа за п’ятьма вузлами для функції $2 \cdot x^8 + 3 \cdot x^7 + 5 \cdot x^5 - 2$ на проміжку $[1..5]$. Вузли обрати як нулі полінома Чебишева того ж порядку для відповідного проміжку. Оцінити похибку інтерполяції.
2. Знайти деякий розв’язок рівняння з попереднього пункту, використовуючи пряму та обернену інтерполяцію. Порядок полінома взяти той же, що і у попередньому пункті. Якщо у дослідженому інтервалі є розв’язок – для прямої інтерполяції можна використати вже побудований поліном.

Теоретичні відомості

Інтерполяція

Нехай функція $f(x) \in C[a, b]$. Задача інтерполяції полягає у відшуканні невідомих значень функції $f(x)$ за її відомими значеннями $f(x_k)$ в точках $x_k \in [a; b]$, $k = \overline{0, n}$, які називають *вузлами інтерполяції*. Розв’язок шукаємо у вигляді полінома $P_n(x)$, що відповідає *інтерполяційним умовам*: $f(x_k) = P_n(x_k)$, $k = \overline{0, n}$.

Інтерполяційний поліном Лагранжа

Формула для побудови поліному у формі Лагранжа:

$$L_n(x) = \sum_{k=0}^{n} \frac{\prod\limits_{\substack{j=0 \\ j \neq k}}^{n} (x - x_j)}{\prod\limits_{\substack{j=0 \\ j \neq k}}^{n} (x_k - x_j)} f(x_k),$$

для зручності її можна переписати в іншому вигляді:

$$L_n(x) = \sum_{k=0}^{n} \frac{\omega(x)}{(x - x_k)\omega'(x_k)} f(x_k),$$

де $\omega(x) = (x - x_0)(x - x_1)...(x - x_n)$.

Похибка інтерполяції

Для оцінки похибки інтерполяції можна використати оцінку залишкового члену у формі Лагранжа:

$$|f(x) - P_n(x)| \leqslant \frac{M_{n+1}}{(n+1)!}|\omega(x)|,$$

де $M_{n+1} = \max\limits_{x\in[a;b]} |f^{(n+1)}(x)|$, $\omega(x) = (x-x_0)(x-x_1)\cdot\ldots\cdot(x-x_n)$,

а також у формі Ньютона:
$|f(x) - L_n(x)| = \omega(x)f(x, x_0, x_1, ..., x_n)$.

Для рівновіддалених вузлів оцінку для залишкового члену для інтерполяційної формули Ньютона вперед подано у вигляді:

$$R_n(t) = \frac{f^{(n+1)}(\xi)}{(n+1)!}h^{n+1}t(t-1)(t-2)\cdots(t-n), \quad t = \frac{x-x_0}{h}$$

та для інтерполяційної формули назад:

$$R_n(t) = \frac{f^{(n+1)}(\xi)}{(n+1)!}h^{n+1}t(t+1)(t+2)\cdots(t+n), \quad t = \frac{x-x_n}{h}$$

Оптимальний вибір вузлів (поліном Чебишева)

Для зменшення похибки інтерполяції необхідно в якості вузлів взяти нулі полінома Чебишева 1 роду. Для їх визначення використовують рекурентні співвідношення:

$$T_n(x) = \frac{(x+\sqrt{x^2-1})^n + (x-\sqrt{x^2-1})^n}{2}, \ \ |x| \geqslant 1,$$

$$T_n(x) = \cos(n \arccos x), \ \ |x| \leqslant 1.$$

Нулі полінома Чебишева: $x \in [-1; 1]$: $x_k = cos\frac{(2k+1)\pi}{2n}$, $k = \overline{0, n-1}$.

Поліном Чебишева 1 роду на проміжку [a;b]

За допомогою заміни $x = \frac{1}{2}((b-a)z + (b+a))$ переведемо проміжок $[-1, 1]$ в $[a, b]$. Тоді поліноми Чебишова запишуться таким чином:

$$T_n^{[a;b]}(x) = T_n^{[-1;1]}\left(\frac{2x-(b+a)}{b-a}\right)$$

Його нулі: $x_k = \dfrac{a+b}{2} + \dfrac{b-a}{2}\cos\dfrac{(2k+1)\pi}{2n}, \qquad k = \overline{0, n-1}$

При використанні цих вузлів похибка має вигляд:

$$|f(x) - L_n(x)| \leqslant \frac{M_{n+1}}{(n+1)!}\frac{(b-a)^{n+1}}{2^{2n+1}}.$$

Пряма інтерполяція

Нехай функція $y = f(x) \in C[a, b]$, що задана таблично (x_i, y_i), $i = \overline{0, n}$, *немонотонна*. Для знаходження x^* застосовуємо такий алгоритм:

За заданою таблицею (x_i, y_i), $i = \overline{0, n}$ будуємо інтерполяційний поліном

$$L_n(x) = \sum_{i=0}^{n} f(x_i)\frac{\omega_{n+1}(x)}{(x - x_i)\omega'_{n+1}(x_i)}$$

та розв'язують нелінійне рівняння $L_n(x) = y^*$.

На підставі теореми про середнє:

$$f(x) - f(x^*) = f'(\xi)(x - x^*)$$

одержимо

$$x - x^* = \frac{f(x) - f(x^*)}{f'(\xi)}.$$

Отже, похибка інтерполяції має вигляд:

$$|x - x^*| \leqslant \frac{M_{n+1}|\omega(x^*)|}{m_1(n+1)!},$$

де $m_1 = \min\limits_{x} |f'(x)|; \quad M_{n+1} = \max\limits_{x} |f^{n+1}(x)|.$

Обернена інтерполяція

Нехай функція $y = f(x) \in C[a, b]$, що задана таблично (x_i, y_i), $i = \overline{0, n}$, *монотонна*. Для знаходження x^* застосовуємо такий алгоритм:

Будуємо за таблицею (x_i, y_i), $i = \overline{0, n}$ таку таблицю: (y_i, x_i), $i = \overline{0, n}$. На підставі останньої таблиці інтерполянт набуває вигляду:

$$L_n(y) = \sum_{i=0}^{n} x_i \frac{\omega_{n+1}(y)}{(y - y_i)\omega'_{n+1}(y_i)},$$

де $\omega_{n+1}(y) = (y - y_0)(y - y_1)\cdots(y - y_n)$ та $L(y^*) \approx x^*$.

Залишковий член в цьому випадку утворюється із залишкового члена формули Ньютона, якщо в останньому поміняти місцями x та y, а похідну $f'(x)$ замінити на похідну від оберненої функції. Похибка інтерполяції має вигляд:

$$|x - x^*| \leqslant \frac{\widetilde{M}_{n+1}}{(n+1)!}|\omega(y^*)|; \quad \widetilde{M}_{n+1} = \max_y \left| \frac{d^{n+1}}{dy^{n+1}} x(y) \right|.$$

За допомогою інтерполяції можна знаходити корені нелінійних рівнянь, для цього знаходять x^* при $y^* = 0$.

Хід роботи

Мова програмування – Python. Використані бібліотеки: numpy, matplotlib.

Функція: $2x^8 + 3x^7 + 5x^5 - 2$. Досліджувати її за умовою вказано на проміжку [1; 5], але оскільки функція монотонно зростає на даному проміжку ми не отримаємо дійсних коренів, а лише величезну похибку:

```
 Choosing 5 nodes by Chebyshov polynomial on [1, 5]
k   xk         f(xk)
0   4.9021     885189.1948
1   4.1756     257562.4350
2   3.0000     20896.0000
3   1.8244     546.4050
4   1.0979     15.9652

Lagrange Polynomial constructed on [1, 5] with 5 nodes.

 Interpolation Error Estimation
Test point x             = 0.5
Function value f(x)      = -1.812500
Lagrange polynomial L(x) = 86354.437500
Absolute difference      = |f(x) - L(x)| = 86356.250000
Theoretical error bound  = 31160.000000

Note: Error might exceed bound if test point is outside [a,b] (extrapolation).
```

Тому змінимо проміжок на $[0,5;\ 1]$ для більшої точності.

Завдання №1.

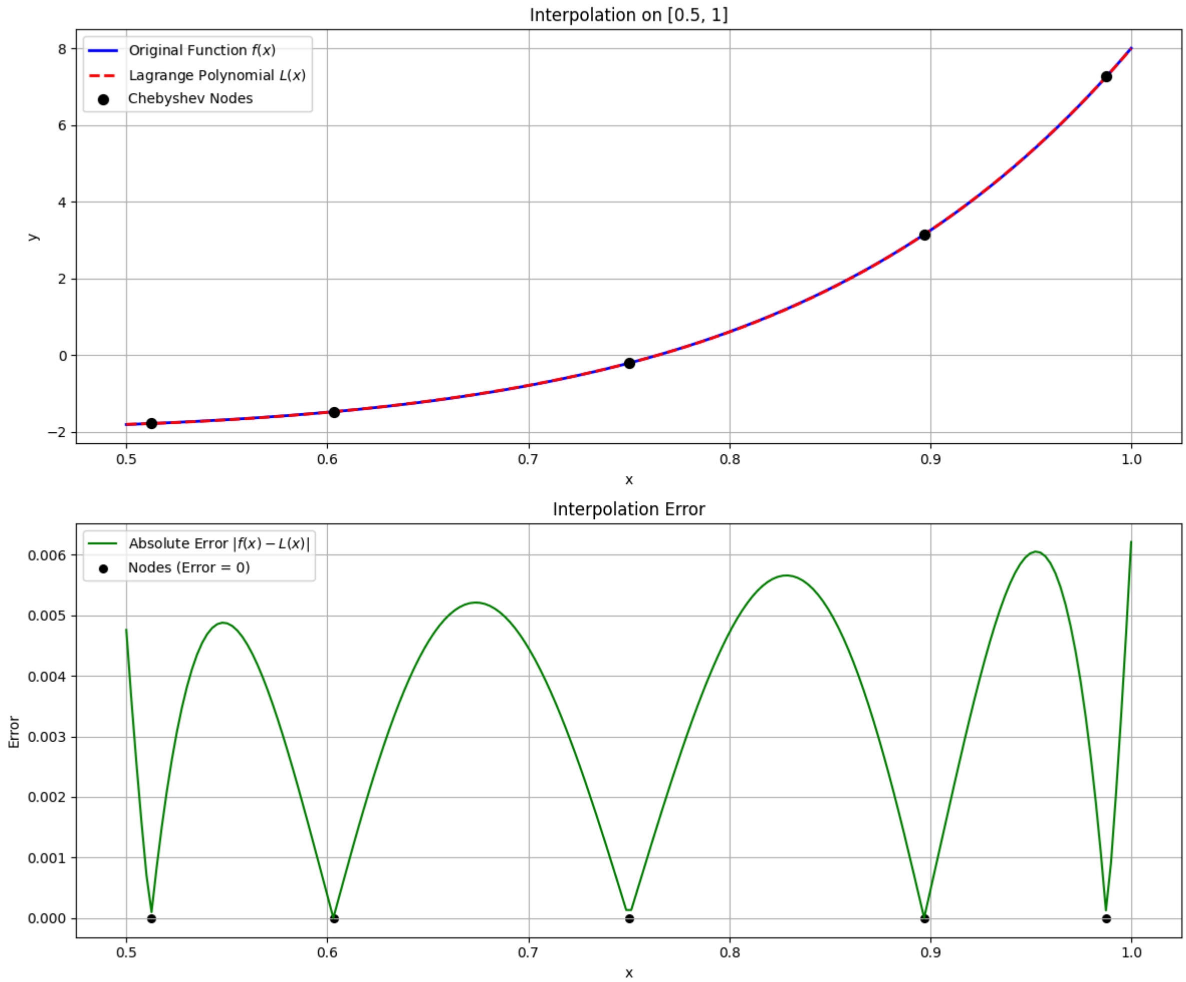
Шукаємо вузли як нулі полінома Чебишева.
Оскільки за умовою потрібно взяти 5 вузлів (N = 5), то максимальний степінь полінома – 4 (n = 4). За формулою знаходимо 5 вузлів і відповідне їм значення функції в цих точках:
$x_k = \frac{a+b}{2} + \frac{b-a}{2} cos\frac{(2k+1)\pi}{2N},\ k = \overline{0,\ N - 1},\ N$ – кількість вузлів.

```
 Choosing 5 nodes by Chebyshov polynomial on [0.5, 1]
k   xk         f(xk)
0   0.9878     7.2662
1   0.8969     3.1417
2   0.7500     -0.2128
3   0.6031     -1.4792
4   0.5122     -1.7864

Lagrange Polynomial constructed on [0.5, 1] with 5 nodes.
```

Для перевірки точності побудованого полінома Лагранжа було обрано тестову точку $x = 0.5$ на новому проміжку:

```
 Interpolation Error Estimation
Test point x             = 0.5
Function value f(x)      = -1.812500
Lagrange polynomial L(x) = -1.807745
Absolute difference      = |f(x) - L(x)| = 0.004755
Theoretical error bound  = 0.010986

Interpolation error is within the theoretical bound.
```

Оскільки 0.004755<0.010986, фактична похибка не перевищує теоретичну. Інтерполяція виконана коректно.



Завдання №2.

Теоретично точний корінь на досліджуваному проміжку:

```
 Searching for theoretical root on [-1, 2]
Theoretical Root x ≈ 0.7646989822
```

Метод прямої інтерполяції

Розв’язуємо рівняння $L(x) = 0$ на проміжку $[0,5;\ 1]$:

```
 Direct Interpolation Method
 Choosing 5 nodes by Chebyshov polynomial on [0.5, 1]
k   xk        f(xk)
0   0.9878    7.2662
1   0.8969    3.1417
2   0.7500    -0.2128
3   0.6031    -1.4792
4   0.5122    -1.7864

Searching for root of L(x) = 0 on [0.5, 1]
Root found by Direct Interpolation: x ≈ 0.7648048401
Check f(root) ≈ 0.0016379828
```

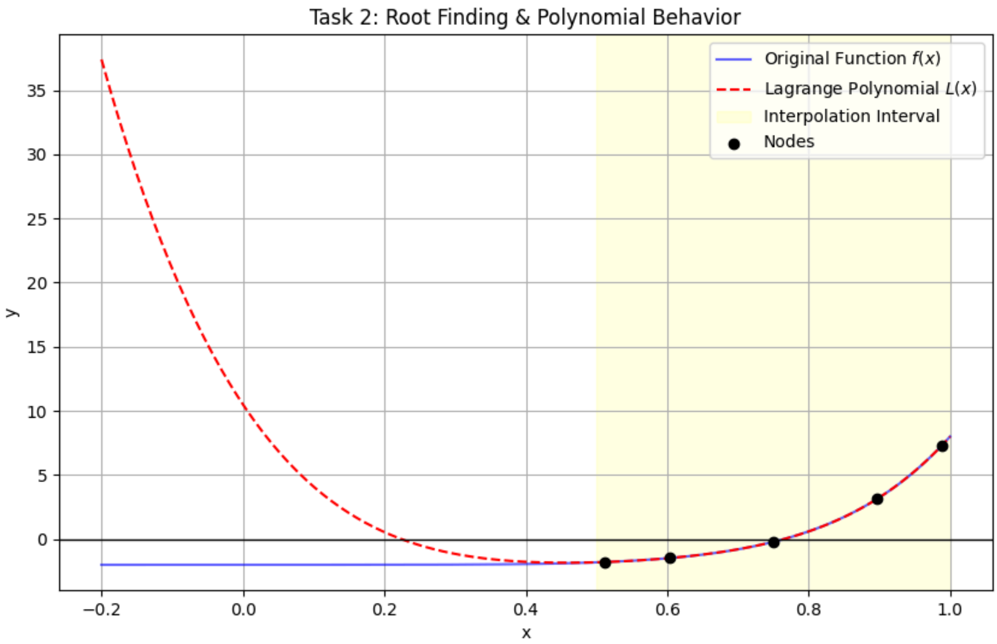
Перевірка підстановкою у функцію показує досить високу точність – отримане значення розбігається з теоретичним після 4 знаку.

Метод оберненої інтерполяції

Будуємо інтерполяційний поліном для оберненої функції $x(y)$, де вузлами є значення $f(x_k)$, а значеннями – x_k. Шукаємо значення цього полінома у точці $y = 0$:

```
 Inverse Interpolation Method
 Choosing 5 nodes by Chebyshov polynomial on [0.5, 1]
k   xk        f(xk)
0   0.9878    7.2662
1   0.8969    3.1417
2   0.7500    -0.2128
3   0.6031    -1.4792
4   0.5122    -1.7864
Building polynomial x(y) (swapping nodes)...
Root found by Inverse Interpolation: x ≈ 0.7536263334
Check f(root) ≈ -0.1622064313
```

Точність методу оберненої інтерполяції виявилася нижчою за пряму, що пояснюється властивостями похідної функції на лівій межі проміжку (похідна близька до нуля, що спричиняє різке зростання оберненої функції).


Task 2: Root Finding & Polynomial Behavior
Original Function $f(x)$
Lagrange Polynomial $L(x)$
Interpolation Interval
Nodes
y
x
35
30
25
20
15
10
5
0
−0.2
0.0
0.2
0.4
0.6
0.8
1.0