

УКРАЇНА

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І
ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ**

Кафедра комп'ютерних наук

**МЕТОДИЧНІ ВКАЗІВКИ ДО ВИКОНАННЯ
ЛАБОРАТОРНИХ РОБІТ**

З ДИСЦИПЛІНИ

ЧИСЕЛЬНІ МЕТОДИ

для студентів спеціальності

“Комп'ютерні науки”

(денна форма навчання)

Київ – 2026

УДК 378.022.51

Викладено методичні вказівки до виконання лабораторних робіт з дисципліни “Чисельні методи”.

Постановка і проведення лабораторних робіт передбачає використання мови програмування Python для реалізації чисельних алгоритмів (або іншої мови програмування).

Рекомендовано вченою радою факультету інформаційних технологій Національного університету біоресурсів і природокористування України. Протокол №6 від 19.03.2026

Укладачі:

доц. **Кириченко В.В.**, ст.викладач **Панкратьєв В.О.**

Рецензенти: д.е.н., проф. Руденський Р.А.

д.т.н., проф. Смолій В.М.

Навчальне видання

ЧИСЕЛЬНІ МЕТОДИ

Методичні вказівки до виконання лабораторних робіт

для студентів спеціальності “Комп’ютерні науки”

(денна форма навчання)

Укладачі: КИРИЧЕНКО Віктор Вікторович

ПАНКРАТЬЄВ Віктор Олександрович

ЗМІСТ

Вступ	4
Лабораторна робота № 1 Чисельні методи розв’язування нелінійних рівнянь. Графічні та аналітичні методи відокремлення коренів. Уточнення коренів. Табличний метод та метод діхотомії	5
Лабораторна робота № 2 Методи уточнення коренів. Метод хорд, метод дотичних	11
Лабораторна робота № 3 Методи уточнення коренів. Метод ітерацій	21
Лабораторна робота № 4 Ітераційні методи розв’язування системи лінійних алгебраїчних рівнянь (СЛАР)	25
Лабораторна робота № 5 Апроксимація. Методи інтерполяції.	32
Лабораторна робота № 6 Регресія. Метод найменших квадратів	51
Лабораторна робота № 7 Чисельне інтегрування	54
Лабораторна робота № 8 Розв’язання звичайних диференціальних рівнянь першого порядку із застосуванням методу Ейлера	72
Лабораторна робота № 9 Розв’язання звичайних диференціальних рівнянь першого порядку із застосуванням методу Рунге–Кутта	77
Навчально–методична література	83

Вступ

Методичні вказівки містять завдання до лабораторних робіт, які виконуються студентами під час вивчення курсу. Метою проведення робіт є ознайомлення студентів з чисельними методами розв'язування задач.

Теоретичний матеріал подається стисло і розрахований на студентів, які вперше вивчають цю дисципліну. Наведено приклади розв'язування типових задач.

Перша, друга і третя роботи знайомлять студентів з чисельними методами розв'язування нелінійних рівнянь.

Четверта робота присвячена ітераційним методам розв'язування системи лінійних рівнянь.

У п'ятій та шостій роботах студенти знайомляться з методами апроксимації функції та регресії.

У сьомій роботі розглянуто чисельні методи обчислення визначених інтегралів.

Восьма та дев'ята робота присвячені чисельним методам розв'язування диференціальних рівнянь.

У ході виконання робіт студенти повинні засвоїти особливості використання алгоритмічних мов програмування для чисельного розв'язання математичних задач.

Після виконання лабораторної роботи студенти здають звіт, який містить:

- 1) титульний лист;
- 2) мету роботи;
- 3) умови завдання;
- 4) розрахунки;
- 5) висновки.

При захисті роботи студенти відповідають на контрольні запитання, наведені в кінці кожної роботи.

Лабораторна робота № 1

Чисельні методи розв'язування рівнянь. Графічні та аналітичні методи відокремлення коренів. Уточнення коренів. Табличний метод та метод діхотомії

Мета роботи: ознайомлення з прямими та ітераційними методами розв'язування рівнянь. Вивчити способи відокремлення коренів та методи уточнення коренів.

Теоретичні відомості

Задача знаходження коренів нелінійних рівнянь виду $f(x) = 0$ досить часто зустрічається в наукових дослідженнях.

Нелінійні рівняння можна поділити на два класи – алгебраїчні та трансцендентні.

Алгебраїчними називаються рівняння, які містять тільки алгебраїчні функції, тобто рівняння виду

$$a_0x^n + a_1x^{n-1} + a_2x^{n-2} + \dots + a_{n-1}x + a_n = 0$$

Для алгебраїчних рівнянь до четвертого ступеня включно відомі прямі методи розв'язування.

До **трансцендентних** рівнянь належать ті, які містять у собі показникові (a^x), логарифмічні ($\log_a x$), тригонометричні ($\sin x$, $\cos x$, $\tan x$...) функції, тобто не є алгебраїчними.

Методи розв'язування нелінійних рівнянь поділяють на прямі та ітераційні. Прямі методи дають змогу записати корені у вигляді деякого кінцевого співвідношення.

Але більшу кількість рівнянь неможливо розв'язати прямими (аналітичними) методами. Для їх розв'язування використовують так звані ітераційні методи, тобто методи послідовних наближень. Алгоритм цього методу включає два етапи: відокремлення коренів (локалізацію) та їх уточнення.

Відокремлення кореня

Відокремлення (локалізація) кореня - це визначення відрізка $[a, b]$ на осі Ox , на якому крива функції $y = f(x)$ перетинає вісь, змінюючи знак. Точка перетину є коренем рівняння.

Існують три основних методи відокремлення коренів функції $y = f(x)$:

1) графічний метод, при якому будується крива функції $y = f(x)$ та знаходиться точка перетину з віссю Ox ;

2) метод заміни функції $f(x) = 0$ функціями $\varphi(x) = \psi(x)$. Точка перетину кривих на графіку дає приблизне знаходження кореня рівняння. Наприклад, $x^3 + x^2 - 3x + 1 = 0$ можна записати як $x^3 + 1 = 3x - x^2$. Таким чином, $\varphi(x) = x^3 + 1$, $\psi(x) = 3x - x^2$.

Побудувавши графіки, знайдемо точку перетину кривих, абсциса якої буде значенням кореня;

3) табличний метод, або метод прямої підстановки, при якому відрізок $[a, b]$ розбивається на кілька менших відрізків і визначається відрізок, на якому функція змінює знак. Далі цей відрізок знову розбивається на кілька менших відрізків і так далі до необхідної точності. Додамо, що цей метод дозволяє визначати корені з необхідною точністю вже на першому етапі розв'язування та обійтись без другого етапу – уточнення кореня.

Слід зазначити, якщо рівняння просте (наприклад, квадратне або лінійне) або має стандартне рішення, то використовувати чисельні методи немає потреби. Але якщо рівняння, наприклад, таке:

$$f(x) = x^7 - 16x^3 + 4x + e^x = 0,$$

то знайти аналітичний метод рішення навряд чи вдасться. В такому випадку використаємо чисельний метод, наприклад метод половинного поділення, яки розглянемо нижче.

Уточнення коренів

Метод половинного поділення або діхотомії (бісекції) при знаходженні кореня рівняння $f(x)=0$ полягає в поділенні навіпіл відрізка $[a, b]$, на якому знаходиться корінь. Потім аналізується зміна знака функції на половинних відрізках і одна з границь відрізка $[a, b]$ переноситься в його середину. Переноситься та границя, зі сторони якої функція на половині відрізка знака не змінює. Якщо виконується умова $f(a) \cdot f(c) < 0$, то праву границю інтервалу переносять у середню точку c ($b = c$). Якщо ця умова не виконується, тобто $f(a) \cdot f(c) > 0$, то в середню точку переносять ліву границю ($a = c$). Далі дії повторюються. Коли виконується умова $|a - b| \leq 2\varepsilon$, ітерації припиняються. За значення кореня з точністю ε приймають середину відрізка $[a, b]$, на якому виконується умова.

Метод простий для машинної реалізації і використовується в багатьох стандартних програмних засобах, хоча існують інші, більш ефективні за затратами часу методи.

Приклад. Методом діхотомії знайти корінь рівняння $x^3 - 3x + 1 = 0$ з точністю до $0,001$ на інтервалі $[0, 1]$.

Розв'язання. За рівнянням складаємо функцію:

$$f(x) = x^3 - 3x + 1 = 0.$$

Знайдемо значення функції на кінцях відрізка:

$$f(a) = f(0) = 1 > 0,$$

$$f(b) = f(1) = (1)^3 - 3 \cdot 1 + 1 = -1 < 0.$$

Перевіримо виконання нерівності:

$$f(a) \cdot f(b) = f(0) \cdot f(1) = 1 \cdot -1 = -1 < 0.$$

Умова виконується, тобто можна застосувати метод половинного поділення.

Складаємо таблицю.

Результати обчислень

N	a_n	b_n	$x_n = (a_n + b_n)/2$	$f(x_n)$
0	0	1	0.5	-0,375
1	0	0,5	0,25	0,2656
2	0,25	0,5	0,375	-0,0723
3	0,25	0,375	0,3125	0,0930
4	0,3125	0,375	0,3438	0,0092
5	0,3438	0,375	0,3594	-0,0318
6	0,3438	0,3594	0,3516	-0,0113
7	0,3438	0,3516	0,3477	-0,0011
8	0,3438	0,3477	0,3458	0,0040
9	0,3458	0,3477	0,3468	0,0013
10	0,3468	0,3477	0,3473	

Як видно з таблиці корінь дорівнює приблизно $0,3473$. Кількість ітерацій 10 . Точність обчислення $a - b = 0,0009 < 0,001$.

Індивідуальні завдання за темою
“ Прямі та ітераційні методи розв’язування рівнянь.
Графічні та аналітичні методи відокремлення коренів. Уточнення
коренів. Табличний метод та метод діхотомії. ”

- | | |
|--|-------------------------------------|
| 1. a) $x - \sin x = 0,25;$ | b) $x^3 - 3x^2 + 9x - 8 = 0;$ |
| 2. a) $\operatorname{tg}(0,85x + 0,1) = x^2;$ | b) $x^3 - 6x - 8 = 0;$ |
| 3. a) $\sqrt{x} - \cos(0,387x) = 0;$ | b) $x^3 - 3x^2 + 6x + 3 = 0;$ |
| 4. a) $\operatorname{tg}(0,4x + 0,4) = x^2;$ | b) $x^3 - 0,1x^2 + 0,4x - 1,5 = 0;$ |
| 5. a) $\lg x - \frac{7}{2x+6} = 0;$ | b) $x^3 - 3x^2 + 9x + 2 = 0;$ |
| 6. a) $\operatorname{tg}(0,5x + 0,2) = x^2;$ | b) $x^2 + x - 5 = 0;$ |
| 7. a) $3x - \cos x - 1 = 0;$ | b) $x^3 + 0,2x^2 + 0,5x - 1,2 = 0;$ |
| 8. a) $x + \lg x = 0,5;$ | b) $x^3 + 3x + 1 = 0;$ |
| 9. a) $\operatorname{tg}(0,5x + 0,2) = x^2;$ | b) $x^3 + 0,2x^2 + 0,5x - 2 = 0;$ |
| 10. a) $x^2 + 4\sin x = 0;$ | b) $x^3 - 3x^2 + 12x - 9 = 0;$ |
| 11. a) $\operatorname{ctg} 1,05x - x^2 = 0;$ | b) $x^3 - 0,2x^2 + 0,3x - 12 = 0;$ |
| 12. a) $\operatorname{tg}(0,45x + 0,3) = x^2;$ | b) $x^3 - 3x^2 + 6x - 2 = 0;$ |
| 13. a) $x \lg x - 1,2 = 0;$ | b) $x^3 - 0,1x^2 + 0,4x - 1,5 = 0;$ |
| 14. a) $1,8x^2 - \sin 10x = 0;$ | b) $x^3 + 3x^2 + 6x - 1 = 0;$ |
| 15. a) $\operatorname{ctg} x - x/4 = 0;$ | b) $x^3 + 0,1x^2 + 0,4x - 1,2 = 0;$ |
| 16. a) $\operatorname{tg}(0,3x + 0,4) = x^2;$ | b) $x^3 + 4x - 6 = 0;$ |

- | | |
|---|-------------------------------------|
| 17. a) $x^2 - 24\sin x = 0;$ | b) $x^3 + 0,2x^2 + 0,5x + 1 = 0;$ |
| 18. a) $\operatorname{ctg} x - x/3 = 0;$ | b) $x^3 - 3x^2 + 12x - 12 = 0;$ |
| 19. a) $\operatorname{tg} (0,47x + 0,2) = x^2;$ | b) $x^3 - 0,2x^2 + 0,3x + 1,2 = 0;$ |
| 20. a) $x^2 + 4\sin x = 0;$ | b) $x^3 - 2x + 4 = 0;$ |
| 21. a) $\operatorname{ctg} x - x/2 = 0;$ | b) $x^3 - 0,2x^2 + 0,5x - 1,4 = 0;$ |
| 22. a) $2x - \lg x - 7 = 0;$ | b) $x^3 - 3x^2 + 6x - 5 = 0;$ |
| 23. a) $\operatorname{tg} (0,5x + 0,2) = x^2;$ | b) $x^3 - 0,1x^2 + 0,4x + 1,2 = 0;$ |
| 24. a) $3x - \cos x - 1 = 0;$ | b) $x^3 - 0,2x^2 + 0,5x - 1 = 0;$ |
| 25. a) $\operatorname{ctg} x - x/10 = 0;$ | b) $x^3 + 3x^2 + 12x + 3,2 = 0;$ |
| 26. a) $x^2 + 4\sin x = 0;$ | b) $x^3 - 0,1x^2 + 0,4x + 2 = 0;$ |
| 27. a) $\operatorname{tg} (0,36x + 0,4) = x^2;$ | b) $x^3 - 0,2x^2 + 0,4x - 2,4 = 0;$ |
| 28. a) $x + \lg x - 0,5 = 0;$ | b) $x^3 + 0,4x^2 + 0,6x - 1,6 = 0;$ |
| 29. a) $\operatorname{ctg} x - x/5 = 0;$ | b) $x^3 + x - 3 = 0;$ |
| 30. a) $2 \lg x - x/2 + 1 = 0;$ | b) $x^3 - 0,2x^2 + 0,5x + 1,4 = 0;$ |

Необхідно:

1) відокремити корені трансцендентного рівняння (a) графічно (двома способами) та уточнити один з них табличним методом з точністю до 0,001. Для виконання завдання можна використовувати мову Python (або іншу мову програмування).

2) запрограмувати метод діхотомії на алгоритмічній мові та розв'язати рівняння (b), використавши власну програму.

Контрольні запитання

1. Що називається відокремленням (локалізацією) коренів?
2. Які існують методи відокремлення коренів?
3. Які класи рівнянь існують?
4. Коли слід використовувати чисельні методи?
5. У чому полягає табличний метод?
6. У чому полягає метод діхотомії?

Лабораторна робота № 2

Методи уточнення коренів. Метод хорд, метод дотичних

Мета роботи: ознайомлення з прямими та ітераційними методами розв'язування рівнянь. Вивчити методи уточнення коренів.

Метод хорд

Нехай задане рівняння $f(x) = 0$, де $f(x)$ – неперервна функція, яка має в інтервалі $[a, b]$ похідні першого та другого порядків.

Корінь відокремлений, знаходиться на відрізку $[a, b]$, тобто $f(a) \cdot f(b) < 0$. Ідея методу полягає в тому, що на достатньо малому відрізку $[a, b]$ дуга кривої $y = f(x)$ замінюється стягуючою її хордою. Як наближене значення кореня приймається точка перетину хорди з віссю Ox .

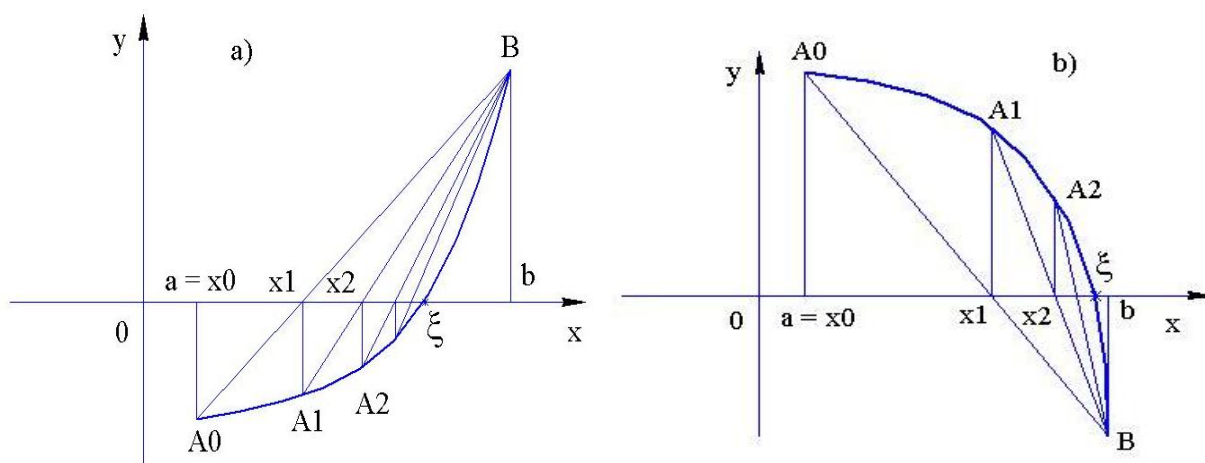


Рис. 2.1. Геометрична інтерпретація методу хорд

Розглянемо випадки, коли перша та друга похідні мають однакові знаки, тобто $f'(x) \cdot f''(x) > 0$.

Нехай $f(a) < 0$, $f(b) > 0$, $f'(x) > 0$ та $f''(x) > 0$. Графік функції $f(x) = 0$ проходить через точки $A_0(a, f(a))$ та $B(b, f(b))$. Корінь рівняння $f(x) = 0$ є абсцисою точки перетину графіка функції $y = f(x)$ з віссю Ox . Ця точка поки невідома. Якщо ми замінимо дугу A_0B хордою A_0B , то точка перетину хорди з віссю Ox x_1 буде приблизно дорівнювати значенню кореня.

Рівняння хорди, яка проходить через точки A_0 та B , має вигляд:

$$\frac{y - f(a)}{f(b) - f(a)} = \frac{x - a}{b - a}.$$

Знайдемо значення $x = x_1$, для якого $y = 0$. Воно дорівнює:

$$x_1 = a - \frac{f(a)(b - a)}{f(b) - f(a)}.$$

Ця формула називається формулою хорд.

Тепер корінь рівняння знаходиться в середині відрізка $[x_1, b]$. Якщо значення кореня нас не задовольняє, його можна уточнити, застосовуючи метод хорд для відрізка $[x_1, b]$.

З'єднаємо точки $A_1(x_1, f(x_1))$ з точкою $B(b, f(b))$ і знайдемо точку перетину хорди A_1B з віссю Ox .

$$x_2 = x_1 - \frac{f(x_1)(b - x_1)}{f(b) - f(x_1)}.$$

Продовжуючи процес, знаходимо

$$x_3 = x_2 - \frac{f(x_2)(b - x_2)}{f(b) - f(x_2)},$$

взагалі

$$x_{n+1} = x_n - \frac{f(x_n)(b - x_n)}{f(b) - f(x_n)}.$$

Процес продовжуємо поки не визначимо корінь із заданою точністю, тобто $|x_{n-1} - x_n| \leq \varepsilon$, де ε – задана точність.

За вище наведеними формулами обчислюються корені для випадку, коли $f(a) > 0$, $f(b) < 0$, $f'(x) < 0$ та $f''(x) < 0$.

Розглянемо випадки, коли перша та друга похідні мають різні знаки, тобто $f'(x) \cdot f''(x) < 0$.

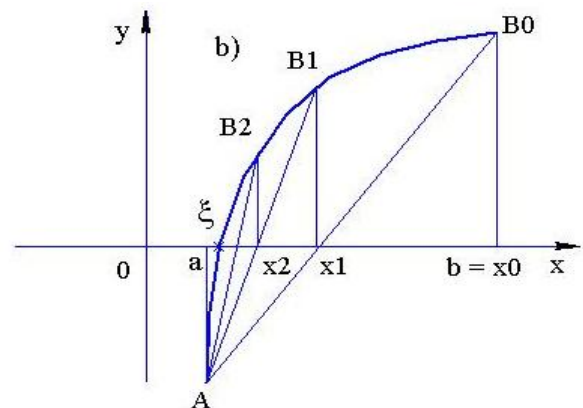
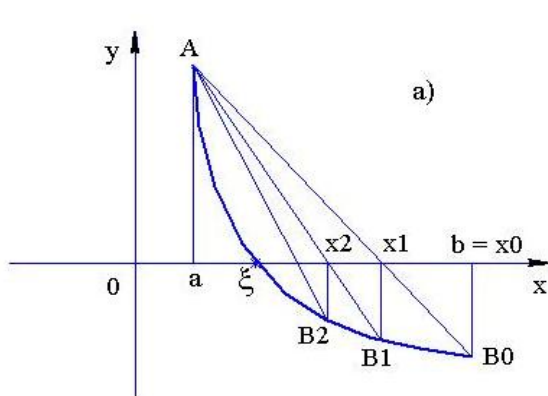


Рис. 2.2. Геометрична інтерпретація методу хорд

Нехай, наприклад, $f(a) > 0$, $f(b) < 0$, $f'(x) < 0$ та $f''(x) > 0$. З'єднаємо точки $A(a, f(a))$ та $B_0(b, f(b))$ і запишемо рівняння хорди, що проходить через A та B_0 :

$$\frac{y - f(b)}{f(b) - f(a)} = \frac{x - b}{b - a}.$$

Далі, діючи як і в попередньому випадку, отримаємо формулу:

$$x_{n+1} = x_n - \frac{f(x_n)(x_n - a)}{f(x_n) - f(a)}.$$

За такими саме формулами знаходимо наближене значення кореня і для випадку, коли $f(a) < 0$, $f(b) > 0$, $f'(x) > 0$ та $f''(x) < 0$.

Вибір формул для уточнення кореня за методом хорд виконують за таким правилом.

Якщо $f'(x) \cdot f''(x) > 0$ на відрізку $[a, b]$, то нерухомим кінцем буде “b”, і всі наближення до кореня ξ знаходяться з боку кінця “a”.

Якщо $f'(x) \cdot f''(x) < 0$, то нерухомий кінець “a”, і всі наближення до кореня ξ знаходяться з боку кінця “b”.

Можна записати:

$$x_{n+1} = x_n - \frac{f(x_n)(x_n - c)}{f(x_n) - f(c)},$$

де c – нерухома точка.

Приклад 1. Методом хорд знайти корінь рівняння $x^4 - 2x - 4 = 0$ з точністю до 0.01 на інтервалі $[1; 1.7]$.

Розв’язання. Корінь рівняння знаходиться на інтервалі $[1; 1.7]$, оскільки

$$f(1) = -5 < 0, \text{ а } f(1.7) = 0.952 > 0.$$

Розрахунок виконують за формулою:

$$x_{n+1} = x_n - \frac{f(x_n)(b - x_n)}{f(b) - f(x_n)},$$

тому що перша і друга похідні на інтервалі $[1; 1.7]$ додатні.

Знаходимо перше наближення при $x_0 = a$.

$$x_1 = a - \frac{f(a)(b-a)}{f(b) - f(a)} = 1 - \frac{f(1)(1,7-1)}{f(1,7) - f(1)} = 1,588;$$

$$f(1,588) = -0,817 < 0.$$

Так само знаходимо друге наближення.

$$x_2 = 1,588 - \frac{f(1,588)(1,7-1,588)}{f(1,7) - f(1,588)} = 1,639;$$

$$f(1,639) = -0,051 < 0.$$

Знаходимо третє наближення.

$$x_2 = 1,639 - \frac{f(1,639)(1,7-1,639)}{f(1,7) - f(1,639)} = 1,642;$$

$$f(1,642) = -0,016 < 0.$$

Знаходимо четверте наближення.

$$x_3 = 1,642 - \frac{f(1,642)(1,7-1,642)}{f(1,7) - f(1,642)} = 1,643;$$

$$f(1,643) = 0,004 > 0.$$

Тобто обчислення можна припинити і прийняти приблизне значення кореня $x = 1,64$ з точністю до $0,01$.

Метод дотичних (метод Ньютона)

Нехай задане рівняння $f(x) = 0$, де $f(x)$ – безперервна функція, яка має в інтервалі $[a, b]$ похідні першого та другого порядків і вони зберігають знаки на цьому відрізку.

Корінь відділений, знаходиться на відрізку $[a, b]$, тобто $f(a) \cdot f(b) < 0$. Геометричний сенс **методу Ньютона** полягає в тому, що на достатньо малому відрізку $[a, b]$ дуга кривої $y = f(x)$ замінюється

дотичною до цієї кривої. Як наближене значення кореня приймається точка перетину дотичної з віссю Ox .

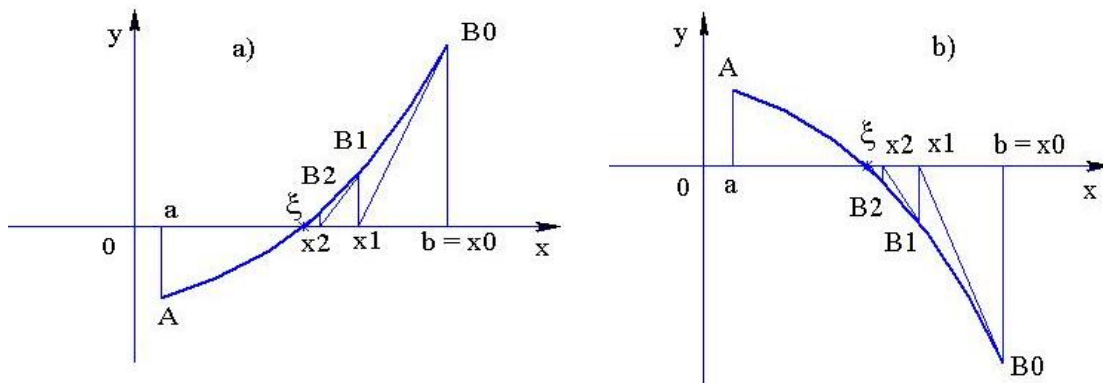


Рис.2.3. Геометрична інтерпретація методу дотичних

Розглянемо випадки, коли перша та друга похідні мають однакові знаки, тобто $f'(x) \cdot f''(x) > 0$.

Нехай $f(a) < 0$, $f(b) > 0$, $f'(x) > 0$ та $f''(x) > 0$. Графік функції $f(x) = 0$ проходить через точки $A(a, f(a))$ та $B_0(b, f(b))$. Корінь рівняння $f(x) = 0$ є абсцисою точки перетину графіка функції $y = f(x)$ з віссю Ox . Ця точка поки невідома. Якщо провести дотичну до кривої в точці A , то вона перетне вісь Ox у точці, що не належить відрізку $[a, b]$. Тоді проведемо дотичну у точці B_0 .

Рівняння дотичної в точці $B_0(b; f(b))$ має вигляд:

$$y - f(b) = f'(b) \cdot (x - b).$$

Знайдемо значення $x = x_1$, для якого $y = 0$:

$$x_1 = b - \frac{f(b)}{f'(b)}.$$

Ця формула називається формулою дотичних.

Тепер корінь рівняння знаходиться в середині відрізка $[a, x_1]$. Якщо значення кореня нас не задовольняє, його можна уточнити, застосовуючи метод дотичних для відрізка $[a, x_1]$.

Проведемо дотичну в точці $B_1(x_1, f(x_1))$ і знайдемо точку перетину дотичної з віссю Ox .

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}.$$

Продовжимо процес і знаходимо

$$x_3 = x_2 - \frac{f(x_2)}{f'(x_2)},$$

взагалі

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Процес продовжуємо поки не визначимо корінь із заданою точністю, тобто $|x_{n-1} - x_n| \leq \varepsilon$, де ε – задана точність.

За вище наведеними формулами обчислюються корені для випадку, коли $f(a) > 0$, $f(b) < 0$, $f'(x) < 0$ та $f''(x) < 0$.

Розглянемо випадки, коли перша та друга похідні мають різні знаки, тобто $f'(x) \cdot f''(x) < 0$.

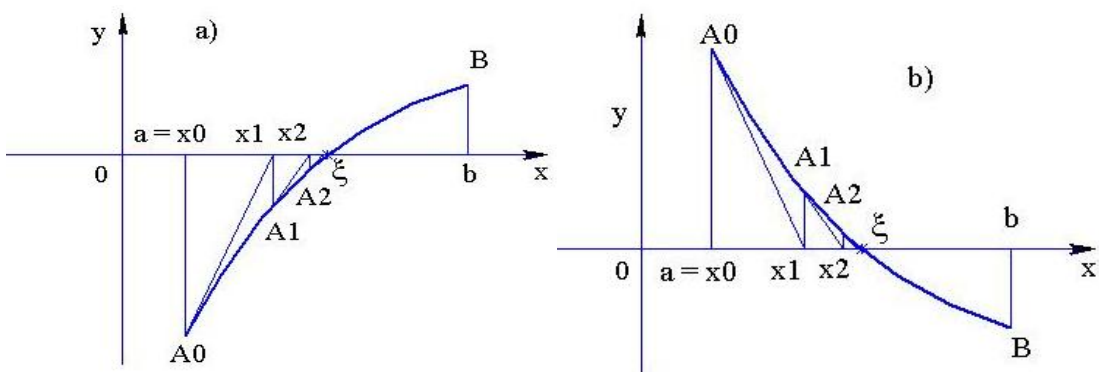


Рис.2.4. Геометрична інтерпретація методу дотичних

Нехай, наприклад, $f(a) < 0$, $f(b) > 0$, $f'(x) > 0$ та $f''(x) < 0$ або $f(a) > 0$, $f(b) < 0$, $f'(x) < 0$ та $f''(x) > 0$. Якщо провести дотичну до кривої в точці В, то вона перетинатиме вісь Ox у точці, що не належить відрізку $[a, b]$. Тоді проведемо дотичну в точці A_0 . Рівняння дотичної в точці $A_0(a; f(a))$ має вигляд:

$$y - f(a) = f'(a) \cdot (x - a).$$

Знайдемо значення $x = x_1$, для якого $y = 0$:

$$x_1 = a - \frac{f(a)}{f'(a)}.$$

Продовжуючи цей процес, одержимо формулу:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Таким чином, ми отримуємо послідовність наближених значень $x_1, x_2, x_3, \dots, x_n$. Розрахунки закінчуємо тоді, коли $|x_{n+1} - x_n| \leq \varepsilon$, де ε – задана точність.

У першому і другому випадках формули відрізняються тільки початковим наближенням: в першому випадку за x_0 приймаємо кінець відрізка b , у другому – початок a .

Вибір формул для уточнення кореня за методом дотичних виконують за таким правилом.

Якщо $f'(x) \cdot f''(x) > 0$ на відрізку $[a, b]$, то нерухомим кінцем буде “а”, і всі наближення до кореня ξ знаходяться з боку кінця “b”.

Якщо $f'(x) \cdot f''(x) < 0$, то нерухомий кінець “b” і всі наближення до кореня ξ знаходяться з боку кінця “а”.

Для оцінки похибки можна скористатися загальною формулою

$$|\xi - x_n| \leq |f(x_n)| / m$$

$$\text{де } m = \min_{[a, b]} |f'(x_n)|$$

(цю формулу можна застосовувати і для методу хорд).

Приклад 2. Методом дотичних знайти корінь рівняння $x^4 - 2x - 4 = 0$ з точністю до 0.01 на інтервалі $[1; 1.7]$.

Розв’язання. В цьому рівнянні

$$f(x) = x^4 - 2x - 4;$$

$$f'(x) = 4x^3 - 2;$$

$$f''(x) = 12x^2.$$

Перша і друга похідні мають один знак на інтервалі. Тому нерухомим кінцем буде “а”. Тоді $x_0 = b = 1,7$. Розрахунок будемо вести за формулою:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Знаходимо перше наближення.

$$f(1,7) = 1,7^4 - 2 \cdot 1,7 - 4 = 0,952;$$

$$f'(1,7) = 4 \cdot 1,7^3 - 2 = 17,652.$$

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = 1,7 - \frac{f(1,7)}{f'(1,7)} = 1,7 - \frac{0,952}{17,652} = 1,646.$$

Так само знаходимо друге наближення.

$$f(1,646) = 1,646^4 - 2 \cdot 1,646 - 4 = 0,048;$$

$$f'(1,646) = 4 \cdot 1,646^3 - 2 = 15,838.$$

$$x_2 = 1,646 - \frac{0,048}{15,838} = 1,643.$$

Знаходимо третє наближення.

$$f(1,643) = 1,643^4 - 2 \cdot 1,643 - 4 = 0,004;$$

$$f'(1,643) = 4 \cdot 1,643^3 - 2 = 15,740.$$

$$x_3 = 1,643 - \frac{0,004}{15,740} = 1,6427.$$

Тобто, можна прийняти приблизне значення кореня $x = 1,64$ з точністю до $0,01$.

Метод однієї дотичної

Існує спрощений метод розв'язання рівнянь, який базується на методі Ньютона. Це так званий метод однієї дотичної. Суть методу полягає в тому, що похідна $f'(x)$ обраховується тільки один раз (у точці “ a ” або “ b ” інтервалу $[a, b]$) і це значення використовують при подальших обчисленнях. Тобто, всі прямі лінії, які проводяться в наступних ітераціях, паралельні першій дотичній, і самі вже не є дотичними до кривої $f(x)$.

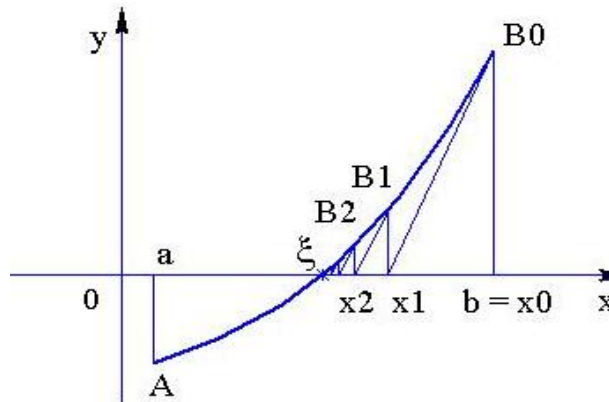


Рис.2.5. Геометрична інтерпретація методу однієї дотичної
Обчислення ведеться за формулою:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Метод простіший, але “повільніший” за метод Ньютона, оскільки для досягнення необхідної точності потрібно зробити більшу кількість ітерацій.

Комбінований метод (хорд – дотичних)

Один з найчастіше використовуваних методів уточнення коренів – метод одночасного використання методу хорд і методу дотичних. Це так званий комбінований метод. Його зручно використовувати, коли на відрізку $[a, b]$, де існує тільки один корінь, друга похідна $f''(x)$ не змінює знака. Постійність знаку $f''(x)$ означає: крива або опукла ($f''(x) < 0$), або увігнута ($f''(x) > 0$).

У цьому випадку наближення до кореня здійснюється з двох сторін – дотична перетинає вісь Ox зі сторони опуклості, а хорда – зі сторони увігнутості графіка функції $y = f(x)$. Інтервал як би “стискається з обох сторін”.

Як і в методі хорд та методі дотичних, розглянемо два випадки.

Нехай задане рівняння $f(x) = 0$, де $f(x)$ – неперервна функція, яка має в інтервалі $[a, b]$ похідні першого та другого порядків.

Розглянемо випадок, коли перша та друга похідні мають однакові знаки, тобто $f'(x) \cdot f''(x) > 0$.

Нехай $f(a) < 0$, $f(b) > 0$, $f'(x) > 0$ та $f''(x) > 0$. Графік функції $f(x) = 0$ проходить через точки $A_0(a, f(a))$ та $B(b, f(b))$. Тоді, користуючись методом хорд і методом дотичних, можна записати:

$$a_{n+1} = a_n - \frac{f(a_n)(b - a_n)}{f(b) - f(a_n)};$$

$$b_{n+1} = x_n - \frac{f(b_n)}{f'(b_n)}.$$

Якщо перша та друга похідні мають різні знаки, тобто $f'(x) \cdot f''(x) < 0$, тоді можна записати:

$$b_{n+1} = b_n - \frac{f(b_n)(b_n - a)}{f(b_n) - f(a)};$$

$$a_{n+1} = a_n - \frac{f(a_n)}{f'(a_n)}.$$

Додамо, що при розрахунках можна використовувати інші комбіновані методи. Наприклад, замість методу Ньютона використати спрощений метод. Можна також прискорити обчислення, якщо, наприклад, у випадку $f'(x) \cdot f''(x) > 0$ в перше рівняння при кожній наступній ітерації підставляти не "b", яке в методі хорд є нерухомим, а "b_n", яке знайдене в попередній ітерації із другого рівняння, тобто:

$$a_{n+1} = a_n - \frac{f(a_n)(b_n - a_n)}{f(b_n) - f(a_n)};$$

$$b_{n+1} = x_n - \frac{f(b_n)}{f'(b_n)}.$$

Якщо задана допустима похибка ε , то обчислення припиняється, як тільки $|a_n - b_n| \leq \varepsilon$. За значення кореня слід прийняти $x = (a_n + b_n)/2$.

Необхідно:

1) Розв'язати рівняння (a) з попереднього завдання методом хорд і Ньютона або комбінованим методом з точністю до 0,01.

2) Порівняти результати з результатами, отриманими в попередній роботі.

3) Перевірити розрахунки реалізувавши їх на мові Python або іншій мові програмування.

Контрольні запитання

1. Які існують методи уточнення коренів?
2. В чому полягає метод хорд?
3. В чому полягає метод дотичних?
4. Яка умова сходимості ітераційного процесу?
5. Які ще методи уточнення коренів ви знаєте?

Лабораторна робота № 3

Методи уточнення коренів. Метод ітерацій

Метод ітерацій (метод послідовних наближень)

Якщо яким-небудь способом одержано наближене значення x_0 кореня рівняння $f(x) = 0$, то уточнення кореня можна виконати методом послідовних наближень або методом ітерацій. Для цього рівняння $f(x) = 0$ записують у вигляді $x = \varphi(x)$. Функція $x = \varphi(x)$ називається ітераційною функцією.

Виберемо будь-яке значення x_0 із відрізка $[a, b]$ і підставимо його в праву частину рівняння $x = \varphi(x)$. Отримаємо значення $x_1 = \varphi(x_0)$. Потім значення x_1 підставимо знову у рівняння та одержимо x_2 . Повторюючи цей процес, отримаємо послідовність чисел $x_1, x_2, \dots, x_n$. Обчислювальний процес закінчимо тоді, коли $|x_{i+1} - x_i| \leq \mu$ (μ – задана точність обчислень).

При обчисленнях можливі два випадки:

а) послідовність чисел $x_1, x_2, \dots, x_n$... **збігається**, тобто є межа і ця межа буде коренем рівняння $f(x) = 0$;

б) послідовність чисел $x_1, x_2, \dots, x_n$... **розбігається**, тобто немає межі.

Метод ітерацій використовується тоді, коли виконується умова:

$|\varphi'(x)| < 1$. При цьому чим менше $\varphi'(x)$, тим краще збіжність ітераційного процесу.

Записати рівняння можна різними способами. Наприклад, рівняння $x^4 - 2x - 4 = 0$ можна замінити такими:

$$x = x^4 - x - 4; \quad (\varphi(x) = x^4 - x - 4);$$

$$x = (x^4 - 4)/2; \quad (\varphi(x) = (x^4 - 4)/2);$$

$$x = \sqrt[4]{2x + 4}; \quad (\varphi(x) = \sqrt[4]{2x + 4}).$$

Ітераційний процес збігається, якщо на відрізку $[a, b]$ виконується умова:

$$\max_{a \leq x \leq b} |\varphi'(x)| \leq q < 1.$$

Як перше наближення можна довільно прийняти значення з інтервалу, в якому знаходиться корінь рівняння.

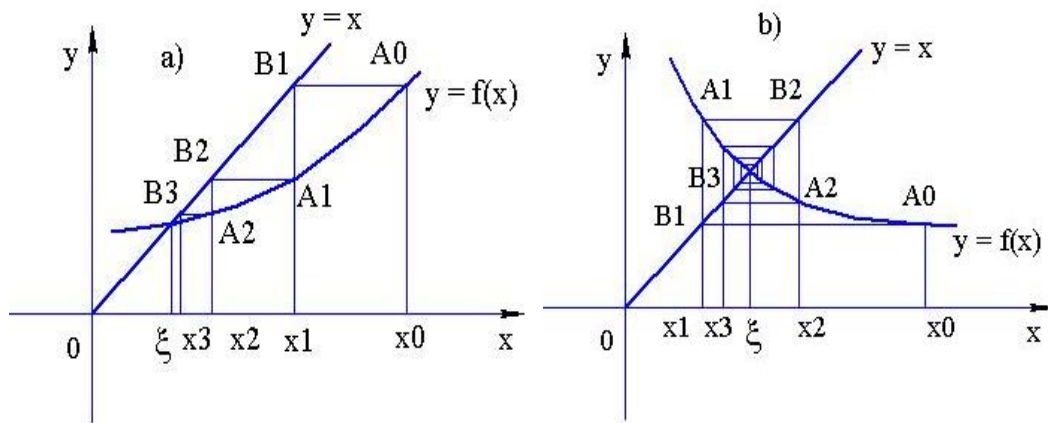


Рис. 3.1. Геометрична інтерпретація методу ітерацій

Приклад 3. Методом ітерацій розв'язати рівняння $5x^3 - 20x + 3 = 0$ на відрізку $[0, 1]$.

Розв'язання. Це рівняння можна привести до вигляду $x = \varphi(x)$ двома способами:

$$x = x + 5x^3 - 20x + 3; \quad \text{тоді } \varphi_1(x) = 5x^3 - 19x + 3;$$

$$x = (5x^3 + 3)/2; \quad \text{тоді } \varphi_2(x) = (5x^3 + 3)/20.$$

Визначимо, яку з функцій слід використовувати в методі ітерацій.

Обчислимо та оцінимо значення похідних обох функцій на заданому інтервалі.

$$\varphi'(x) = 15x^2 - 19 > 1 \quad \text{і} \quad [0, 1];$$

$$\varphi'(x) = 15x^2 / 20 < 1 \quad \text{і} \quad [0, 1].$$

Очевидно, що слід скористатися другою формулою для уточнення кореня рівняння $5x^3 - 20x + 3 = 0$ на відрізок $[0, 1]$.

Для оцінки похибки методу ітерацій використовується формула:

$$|\xi - x_n| \leq \frac{q^n}{1-q} |x_1 - x_0|$$

А якщо $q \leq 0.5$ похибка визначається за формулою:

$$|\xi - x_n| \leq |x_n - x_{n-1}|$$

У цій формулі ξ – корінь рівняння з інтервалу.

Приклад 4. Методом ітерацій знайти корінь рівняння $x^3 - 5x + 1 = 0$ з точністю до $0,0001$ на інтервалі $[0; 0,5]$.

Розв'язання. Запишемо рівняння у вигляді:

$$x = \frac{x^3 + 1}{5}, \quad \text{де} \quad \frac{x^3 + 1}{5} = \varphi(x).$$

Похідна від $\varphi(x)$

$$\varphi'(x) = \frac{3x^2}{5}.$$

Оскільки

$$\max_{0 \leq x \leq 0.5} |\varphi'(x)| = \frac{3}{20} < \frac{1}{2},$$

процес ітерацій збігається.

Прийmemo $x_0 = 0,25$ (середина інтервалу) і почнемо обчислювання за формулою:

$$x_{n+1} = \frac{x_n^3 + 1}{5}.$$

Результати обчислень запишемо в таблицю.

Результати обчислень

N	x_n	$x_{n+1} = \frac{x_n^3 + 1}{5}$	$ x_n - x_{n+1} $
0	0,25	0,020313	0,04687
1	0,20313	0,020168	0,00145
2	0,20168	0,020164	0,00004

Як бачимо, корінь рівняння $x = 0,020164$, обчислений з точністю $|\xi - x_n| = |\xi - 0,020164| \leq |x_2 - x_1| = |0,020164 - 0,20168| = 0,00004 < \varepsilon = 0,0001$.

Необхідно:

- 1) Розв'язати рівняння (b) з попереднього завдання методом ітерацій з точністю до 0,01.
- 2) Порівняти результати з результатами, отриманими в попередній роботі.
- 3) Перевірити розрахунки за допомогою мови Python (або іншої мови програмування).

Контрольні запитання

1. Які існують методи уточнення коренів?
2. В чому полягає метод ітерацій?
3. Яка умова сходимості ітераційного процесу?
4. Які ще методи уточнення коренів ви знаєте?

Лабораторна робота № 4

Ітераційні методи розв'язування системи лінійних алгебраїчних рівнянь

Мета роботи: ознайомитися з ітераційними методами розв'язування систем лінійних рівнянь.

Теоретичні відомості

Як уже зазначалося, ітераційні методи належать до наближених методів. Вони характеризуються великою кількістю обчислювань, що однак не заважає їм бути за своїй структурою достатньо простими. За рахунок наближень ми отримуємо нові наближення і, якщо система задовольняє умові збіжності, ці наближення все менше відрізнятимуться від аналітичного рішення.

Метод ітерацій (від латинського iteration – повторення) можна назвати методом послідовних наближень. Він цікавий тим, що дозволяє знайти рішення з деякою заздалегідь заданою точністю, крім того, невеликі помилки, зроблені в процесі обчислень, потім виправляються, тобто метод є самовиправним. Метод ітерацій легко програмується, його зручно використовувати в комп'ютерних обчисленнях.

Метод прямої ітерації

Розглянемо систему трьох лінійних рівнянь з трьома невідомими.

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2 \\ a_{31}x_1 + a_{32}x_3 + a_{33}x_3 = b_3 \end{cases} \quad (4.1)$$

Запишемо цю систему в такому вигляді:

$$\begin{cases} x_1 = \frac{1}{a_{11}}(b_1 - a_{12}x_2 - a_{13}x_3) \\ x_2 = \frac{1}{a_{22}}(b_2 - a_{21}x_1 - a_{23}x_3) \\ x_3 = \frac{1}{a_{33}}(b_3 - a_{31}x_1 - a_{32}x_2) \end{cases} \quad (4.2)$$

Задамо нульові наближення:

$$x_1 = x_1^{(0)}, \quad x_2 = x_2^{(0)}, \quad x_3 = x_3^{(0)}.$$

Тоді,

$$\begin{cases} x_1^{(1)} = \frac{1}{a_{11}}(b_1 - a_{12}x_2^{(0)} - a_{13}x_3^{(0)}) \\ x_2^{(1)} = \frac{1}{a_{22}}(b_2 - a_{21}x_1^{(0)} - a_{23}x_3^{(0)}) \\ x_3^{(1)} = \frac{1}{a_{33}}(b_3 - a_{31}x_1^{(0)} - a_{32}x_2^{(0)}) \end{cases}$$

Закінчена перша ітерація і визначені $x_1^{(1)}, x_2^{(1)}, x_3^{(1)}$.

Далі підставляємо одержані значення в (4.2) і виконуємо наступну ітерацію. Процес продовжується до отримання результату з необхідною точністю.

Необхідною умовою є наступне: модулі діагональних коефіцієнтів повинні бути не менше за суму модулів інших коефіцієнтів відповідних рівнянь, тобто

$$a_{ij} \geq \sum_{i \neq j} |a_{ij}|.$$

Приклад 1. Розв'язати систему трьох рівнянь за методом ітерацій.

$$\begin{cases} 4x_1 - x_2 + x_3 = 4 \\ 2x_1 + 6x_2 - x_3 = 7 \\ x_1 + 2x_2 - 3x_3 = 0 \end{cases}$$

Розв'язання. Запишемо цю систему в такому вигляді:

$$\begin{cases} x_1 = \frac{1}{4}(4 + x_2 - x_3) \\ x_2 = \frac{1}{6}(7 - 2x_1 + x_3) \\ x_3 = \frac{1}{3}(x_1 + 2x_2) \end{cases}$$

Задамо нульові наближення:

$$x_1^{(0)} = 0, \quad x_2^{(0)} = 0, \quad x_3^{(0)} = 0.$$

Тоді,

$$\begin{cases} x_1^{(1)} = \frac{1}{4}(4 + 0 - 0) = 1 \\ x_2^{(1)} = \frac{1}{6}(7 - 0 + 0) = \frac{7}{6} \\ x_3^{(1)} = \frac{1}{3}(1 + 0) = \frac{1}{3} \end{cases}$$

Закінчена перша ітерація і визначені:

$$x_1^{(1)} = 1, \quad x_2^{(1)} = \frac{7}{6}, \quad x_3^{(1)} = \frac{1}{3}.$$

Підставимо одержані значення та отримаємо:

$$\begin{cases} x_1 = \frac{1}{4}(4 + \frac{7}{6} - \frac{1}{3}) = \frac{29}{24} \\ x_2 = \frac{1}{6}(7 - 2 + \frac{1}{3}) = \frac{8}{9} \\ x_3 = \frac{1}{3}(1 + 2 \cdot \frac{7}{6}) = \frac{10}{9} \end{cases}$$

Закінчена друга ітерація і визначені:

$$x_1^{(1)} = \frac{29}{24}, \quad x_2^{(1)} = \frac{8}{9}, \quad x_3^{(1)} = \frac{10}{9}.$$

Знову підставляємо одержані значення в (4.1) і продовжуючи ітерації, нарешті отримаємо $x_1 = 1$, $x_2 = 1$, $x_3 = 1$.

Метод Зейделя

Метод Зейделя нагадує метод прямої ітерації, але є деякі відмінності.

Обчислення виконують в такому порядку:

а) задаються нульові наближення, потім із першого рівняння обчислюється перше невідоме і це значення першого невідомого підставляється в друге рівняння;

б) обчислюється друге невідоме, його значення разом з першим підставляється в третє рівняння;

с) обчислюється третє невідоме.

Далі виконується наступна ітерація.

Розглянемо систему трьох лінійних рівнянь з трьома невідомими.

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2 \\ a_{31}x_1 + a_{32}x_3 + a_{33}x_3 = b_3 \end{cases} \quad (4.3)$$

Як і в прикладі 1 запишемо цю систему в наступному вигляді:

$$\begin{cases} x_1 = \frac{1}{a_{11}}(b_1 - a_{12}x_2 - a_{13}x_3) \\ x_2 = \frac{1}{a_{22}}(b_2 - a_{21}x_1 - a_{23}x_3) \\ x_3 = \frac{1}{a_{33}}(b_3 - a_{31}x_1 - a_{32}x_2) \end{cases} \quad (4.4)$$

Задамо нульові наближення $x_2 = x_2^{(0)}$, $x_3 = x_3^{(0)}$.

Тоді,

$$\begin{cases} x_1^{(1)} = \frac{1}{a_{11}}(b_1 - a_{12}x_2^{(0)} - a_{13}x_3^{(0)}) \\ x_2^{(1)} = \frac{1}{a_{22}}(b_2 - a_{21}x_1^{(1)} - a_{23}x_3^{(0)}) \\ x_3^{(1)} = \frac{1}{a_{33}}(b_3 - a_{31}x_1^{(1)} - a_{32}x_2^{(1)}) \end{cases}$$

Закінчена перша ітерація і визначені $x_1^{(1)}$, $x_2^{(1)}$, $x_3^{(1)}$.

Далі підставляємо одержані значення в (4.4) і виконуємо наступну ітерацію. Процес продовжується до отримання результату з необхідною точністю.

Приклад 2. Розв'язати систему трьох рівнянь за методом Зейделя.

$$\begin{cases} 4x_1 - x_2 + x_3 = 4 \\ 2x_1 + 6x_2 - x_3 = 7 \\ x_1 + 2x_3 - 3x_3 = 0 \end{cases}$$

Розв'язання. Запишемо цю систему в такому вигляді:

$$\begin{cases} x_1 = \frac{1}{4}(4 + x_2 - x_3) \\ x_2 = \frac{1}{6}(7 - 2x_1 + x_3) \\ x_3 = \frac{1}{3}(x_1 + 2x_2) \end{cases}$$

Задамо нульові наближення $x_1^{(0)} = 0$, $x_2^{(0)} = 0$, $x_3^{(0)} = 0$.

Тоді,

$$\begin{cases} x_1^{(1)} = \frac{1}{4}(4 + 0 - 0) = 1 \\ x_2^{(1)} = \frac{1}{6}(7 - 2 \cdot 1 + 0) = \frac{5}{6} \\ x_3^{(1)} = \frac{1}{3}(1 + 2 \cdot \frac{5}{6}) = \frac{8}{9} \end{cases}$$

Закінчена перша ітерація і визначені

$$x_1^{(1)} = 1, \quad x_2^{(1)} = \frac{5}{6}, \quad x_3^{(1)} = \frac{8}{9}.$$

Підставимо одержані значення і отримаємо:

$$\begin{cases} x_1 = \frac{1}{4}(4 + \frac{5}{6} - \frac{8}{9}) = \frac{71}{72} \\ x_2 = \frac{1}{6}(7 - 2 \cdot \frac{71}{72} + \frac{8}{9}) = \frac{71}{72} \\ x_3 = \frac{1}{3}(\frac{71}{72} + 2 \cdot \frac{71}{72}) = \frac{71}{72} \end{cases}$$

Закінчена друга ітерація і визначені

$$x_1^{(1)} = \frac{71}{72}, \quad x_2^{(1)} = \frac{71}{72}, \quad x_3^{(1)} = \frac{71}{72}.$$

Знову підставляємо одержані значення і, продовжуючи ітерації, нарешті отримаємо $x_1 = 1$, $x_2 = 1$, $x_3 = 1$.

Індивідуальні завдання за темою
“ Ітераційні методи виРозв’язання системи лінійних
алгебраїчних рівнянь ”

ВАРІАНТ 1

$$\begin{aligned} 10x + y + z &= 12 \\ 2x + 10y + z &= 13 \\ 2x + 2y + 10z &= 14 \end{aligned}$$

ВАРІАНТ 2

$$\begin{aligned} 4x + 0,24y - 0,08z &= 8 \\ 0,09x + 3y + 0,015z &= 9 \\ 0,04x - 0,08y + 4z &= 20 \end{aligned}$$

ВАРІАНТ 3

$$\begin{aligned} 4x - y + z &= 4 \\ x + 6y + 2z &= 9 \\ -x - 2y + 5z &= 2 \end{aligned}$$

ВАРІАНТ 4

$$\begin{aligned} 2x - y - z &= -3 \\ 3x + 5y - z &= 1 \\ x - 4y + 10z &= 0 \end{aligned}$$

ВАРІАНТ 5

$$\begin{aligned} 10x + 2y + 6z &= 28 \\ x + 10y + 8z &= 7 \\ 2x - 7y - 10z &= -17 \end{aligned}$$

ВАРІАНТ 6

$$\begin{aligned} 7,6x + 0,5y - 2,4z &= 1,9 \\ 2,2x + 9,1y + 4,4z &= 9,7 \\ -1,3x + 0,2y + 5,8z &= -1,4 \end{aligned}$$

ВАРІАНТ 7

$$\begin{aligned} 20x + 2y + 6z &= 38 \\ x + 20y + 9z &= -23 \\ 2x - 7y - 20z &= -57 \end{aligned}$$

ВАРІАНТ 8

$$\begin{aligned} x - 0,2y - 0,2z &= 0,6 \\ -0,1x + y - 0,2z &= 0,7 \\ -0,1x - 0,1y + z &= 0,8 \end{aligned}$$

ВАРІАНТ 9

$$\begin{aligned} 5x + 0,5y + 0,5z &= 6 \\ x + 5y + 0,5z &= 6,5 \\ x + y + 5z &= 7 \end{aligned}$$

ВАРІАНТ 10

$$\begin{aligned} 7,6x + 0,5y + 2,4z &= 1,9 \\ 2,2x + 9,1y + 4,4z &= 9,7 \\ -1,3x + 0,2z + 5,8z &= -1,4 \end{aligned}$$

ВАРІАНТ 11

$$\begin{aligned} x + 0,1y + 0,1z &= 1,2 \\ 0,2x + y + 0,1z &= 1,3 \\ 0,2x + 0,2y + z &= 1,4 \end{aligned}$$

ВАРІАНТ 12

$$\begin{aligned} 2x + 0,12y - 0,04z &= 4 \\ 0,03x + y - 0,05z &= 3 \\ 0,01x + 0,02y + z &= 5 \end{aligned}$$

ВАРІАНТ 13

$$\begin{aligned} 2x - 0,5y + 0,5z &= 2 \\ 0,5x + 3y + z &= 4,5 \\ -2x - 4y + 10z &= 4 \end{aligned}$$

ВАРІАНТ 14

$$\begin{aligned} 4x - 2y - z &= 6 \\ 1,5x + 2,5y - 0,5z &= 0,5 \\ 0,25x - y + 2,5z &= 0 \end{aligned}$$

ВАРІАНТ 15

$$\begin{aligned} 10x + y + z &= 12 \\ 2x + 10y + z &= 13 \\ 2x + 2y + 10z &= 14 \end{aligned}$$

ВАРІАНТ 16

$$\begin{aligned} 4x + 0,24y - 0,08z &= 8 \\ 0,09x + 3y + 0,015z &= 9 \\ 0,04x - 0,08y + 4z &= 20 \end{aligned}$$

ВАРІАНТ 17

$$\begin{aligned} 4x - y + z &= 4 \\ x + 6y + 2z &= 9 \\ -x - 2y + 5z &= 2 \end{aligned}$$

ВАРІАНТ 18

$$\begin{aligned} 2x - y - z &= -3 \\ 3x + 5y - 3z &= 1 \\ x - 4y + 10z &= 0 \end{aligned}$$

ВАРІАНТ 19

$$\begin{aligned} x - 0,2y - 0,2z &= 0,6 \\ -0,1x + y - 0,2z &= 0,7 \\ -0,1x - 0,1y + z &= 0,8 \end{aligned}$$

ВАРІАНТ 20

$$\begin{aligned} 5x + 0,5y + 0,5z &= 6 \\ x + 5y + 0,5z &= 6,5 \\ x + y + 5z &= 7 \end{aligned}$$

ВАРІАНТ 21

$$\begin{aligned} 7,6x + 0,5y + 2,4z &= 1,9 \\ 2,2x + 9,1y + 4,4z &= 9,7 \\ -1,3x + 0,2y + 5,8z &= -1,4 \end{aligned}$$

ВАРІАНТ 22

$$x + 0,1y + 0,1z = 1,2$$

$$0,2x + y + 0,1z = 1,3$$

$$0,2x + 0,2y + z = 1,4$$

ВАРІАНТ 23

$$x - 0,2y - 0,2z = 0,4$$

$$-0,1x + y - 0,2z = 0,8$$

$$-0,1x - 0,1y + z = 0,7$$

ВАРІАНТ 24

$$4x + 0,24y - 0,08z = 7$$

$$0,1x + 3y + 0,02z = 9$$

$$0,05x - 0,08y + 4z = 21$$

Необхідно:

1) Розв'язати систему трьох лінійних рівнянь за методами ітерацій та Зейделя, зробивши три ітерації.

2) Розв'язати систему, використавши відповідні функції пакету NumPy мови Python (або іншу мову програмування) і порівняти результат з результатами, отриманими методами ітерацій та Зейделя.

Контрольні запитання

1. Яка система називається сумісною?
2. Як називається система, яка не має жодного рішення?
3. Які існують методи розв'язування систем лінійних рівнянь?
4. Коли слід використовувати ітераційні методи?
5. У чому полягає відмінність методів прямих ітерацій та Зейделя?
6. Яка умова можливості використання ітераційних методів?
7. Який метод, на вашу думку, забезпечує більшу швидкість досягнення потрібної точності обчислень?

Лабораторна робота № 5

Апроксимація

Мета роботи: засвоїти методи апроксимації функцій.

Теоретичні відомості

Поняття про наближення функції

Апроксимацією називають знаходження такої функції, яка найточніше описує функцію, задану у вигляді таблиці, тобто *заміну функції $f(x)$ функцією $\varphi(x)$, близькою до першої*.

Якщо наближення виконується на заданому дискретному інтервалі множини точок $\{x_i\}$, то таку апроксимацію називають точковою. При побудові наближення на неперервній множині точок, наприклад на відрізку $[a, b]$, апроксимація називається неперервною, або інтегральною.

Інтерполяція

Одним з основних типів точкової апроксимації є інтерполяція.

Найпростіша задача інтерполяції полягає в такому: на відрізку задані точки, які називаються вузлами інтерполяції, і значення деякої функції у цих точках.

Необхідно побудувати таку функцію (так звану інтерполяційну функцію), яка приймає у вузлах інтерполяції ті самі значення, що й функція. Тобто треба знайти криву певного типу, що проходить через задану систему точок. Знайшовши таку функцію, ми зможемо знайти її значення в кожній точці відрізка. Можна дати таке визначення інтерполяції.

Інтерполяція – це спосіб знаходження проміжних значень функції за дискретним набором відомих значень.

Приклад 1. Дана таблиця, в якій вказана температура тіла в різні моменти часу.

t	0'	10'	20'	30'
T	60°	90°	80°	20°

Необхідно знайти функцію, яка описує залежність температури від часу, тобто дозволяє знайти температуру тіла в заданий момент часу.

Розв’язання. Для рішення задачі спробуємо знайти найпростішу функцію, яка при заданих значеннях аргумента (часу) приймала би ті самі значення функції (температури), що вказані в таблиці. Найпростішою функцією, зручною для обчислень, є поліном (ціла раціональна функція). Оскільки в таблиці вказані чотири значення функції, будемо шукати поліном третього ступеня.

$$\begin{aligned} \varphi(t) &= at^3 + bt^2 + ct + d, \\ \text{або} \quad T &= at^3 + bt^2 + ct + d. \end{aligned} \quad (5.1)$$

У такому виразі $t = 0, 10, 20, 30$.

Коефіцієнти a, b, c, d нам невідомі. Підберемо їх так, щоб при $t = 0, 10, 20, 30$ поліном (5.1) прийняв відповідно значення $T = 60, 90, 80, 20$.

Підставимо послідовно значення t у поліном (5.1) і прирівняємо відповідним значенням T . Отримаємо систему рівнянь:

$$\begin{aligned} d &= 60 && (\text{при } t = 0, T = 60) \\ 1000a + 100b + 10c + d &= 90 && (\text{при } t = 10, T = 90) \\ 8000a + 400b + 20c + d &= 80 && (\text{при } t = 20, T = 80) \\ 27000a + 900b + 30c + d &= 20 && (\text{при } t = 30, T = 20) \end{aligned}$$

Розв’язавши систему, одержимо:

$$a = -1/600, \quad b = -3/20, \quad c = 14/3.$$

Поліном буде мати вигляд:

$$T = -(1/600)t^3 - (3/20)t^2 + (14/3)t + 60.$$

Користуючись виразом, ми можемо знайти температуру тіла в кожен момент часу від $0'$ до $30'$. Наприклад, в момент $t = 17'$:

$$T = -(1/600)17^3 - (3/20)17^2 + (14/3)17 + 60 = 87,8, \text{ тобто } T = 87,8^\circ.$$

Весь цей процес називається інтерполяцією, а знайдена функція називається інтерполяційною функцією. Саму функцію записують у вигляді **полінома**:

$$\varphi(x) = a_0x^n + a_1x^{n-1} + a_2x^{n-2} + \dots + a_n$$

Інтерполяція за допомогою полінома називається **параболічною**.

Точки $x_0, x_1, x_2, \dots, x_n$, в яких функція має значення $y_0, y_1, y_2, \dots, y_n$, називають вузлами інтерполяції.

Якщо між різними вузлами поліноми різні, то така інтерполяція називається локальною.

Якщо поліном один для всього інтервалу інтерполяції, то така інтерполяція називається глобальною.

За допомогою полінома можна знаходити значення функції у вузлах, які знаходяться за межами інтервалу. В такому випадку процес знаходження значень функції називають **екстраполяцією**.

Інтерполяційний поліном Лагранжа

В загальному випадку для $n + 1$ вузлів інтерполяційний поліном має вигляд

$$L_n(x) = \sum_{i=0}^n \frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)} y_i. \quad (5.2)$$

Ця формула називається інтерполяційний поліном Лагранжа.

Лінійна інтерполяція

Якщо задані значення функції в двох вузлах x_0 та x_1 і ці значення дорівнюють y_0 та y_1 , то лінійну функцію на відрізку можна записати формулою:

$$y = Ax + B,$$

в якій при $x = x_0$ $y = y_0$, а при $x = x_1$ $y = y_1$.

Тоді можна записати:

$$y = \frac{(x-x_1)}{(x_0-x_1)} y_0 + \frac{(x-x_0)}{(x_1-x_0)} y_1$$

Приклад 2. Потрібно побудувати інтерполяційний поліном для функції, яка задана у вигляді таблиці:

x	x_0	x_1
y	y_0	y_1

з такими чисельними значеннями:

x	1	3
y	1	9

Розв’язання. В даному випадку маємо два вузли інтерполяції і рішення будемо шукати за формулою

$$y = \frac{(x - x_1)}{(x_0 - x_1)} y_0 + \frac{(x - x_0)}{(x_1 - x_0)} y_1.$$

Підставивши чисельні значення, одержимо:

$$y = \frac{(x - 3)}{(1 - 3)} 1 + \frac{(x - 1)}{(3 - 1)} 9 = 4x - 3,$$

або $y = 4x - 3$.

Графік функції має вигляд прямої лінії.

Квадратична інтерполяція

Якщо задані значення функції в трьох вузлах x_0 , x_1 та x_2 і ці значення дорівнюють y_0 , y_1 та y_2 , то інтерполяційний поліном буде мати вигляд параболи, яку можна описати рівнянням виду:

$$y = Ax^2 + Bx + C.$$

Він може бути знайдений за формулою:

$$y = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} y_0 + \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} y_1 + \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} y_2.$$

Приклад 3. Потрібно побудувати інтерполяційний поліном для функції, яка задана у вигляді таблиці:

x	x_0	x_1	x_2
y	y_0	y_1	y_2

з такими чисельними значеннями:

x	1	3	4
y	12	4	6

Розв’язання. В даному випадку маємо три вузли інтерполяції і рішення будемо шукати за формулою:

$$y = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}y_0 + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}y_1 + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}y_2.$$

Підставивши чисельні значення, отримаємо:

$$y = \frac{(x-3)(x-4)}{(1-3)(1-4)}12 + \frac{(x-1)(x-4)}{(3-1)(3-4)}4 + \frac{(x-1)(x-3)}{(4-1)(4-3)}6 = 2x^2 - 12x + 22,$$

$$\text{або } y = 2x^2 - 12x + 22.$$

Побудуємо графік функції. Він має вигляд параболи (рис. 5.1.).

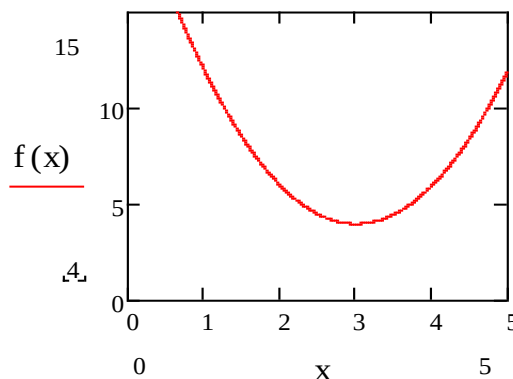


Рис. 5.1. Графік функції $y = 2x^2 - 12x + 22$

Кубічна інтерполяція

Якщо задані значення функції в чотирьох вузлах x_0 , x_1 , x_2 та x_3 і ці значення дорівнюють y_0 , y_1 , y_2 та y_3 , то інтерполяційний поліном буде мати вигляд кубічної параболи, яку можна описати рівнянням виду:

$$y = Ax^3 + Bx^2 + Cx + D.$$

Він може бути знайдений за формулою:

$$y = \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} y_0 + \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_2-x_3)} y_1 +$$

$$+ \frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} y_2 + \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_2-x_2)} y_3.$$

Приклад 4. Побудувати інтерполяційний поліном Лагранжа за наступними даними:

x	0	2	3	5
y	1	3	2	5

Розв'язання. В цьому випадку маємо чотири вузли інтерполяції і рішення будемо шукати за формулою:

$$y = \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} y_0 + \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_2-x_3)} y_1 +$$

$$+ \frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} y_2 + \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_2-x_2)} y_3.$$

Підставивши чисельні значення, отримаємо:

$$y = \frac{(x-2)(x-3)(x-5)}{(0-2)(0-3)(0-5)} \cdot 1 + \frac{x(x-3)(x-5)}{2(2-3)(2-5)} \cdot 3 + \frac{x(x-2)(x-5)}{2(3-2)(3-5)} \cdot 2 + \frac{x(x-2)(x-3)}{5(5-2)(5-3)} \cdot 5 =$$

$$= \frac{2}{15} x^3 - x^2 + \frac{37}{15} x + 1.$$

Побудуємо графік функції. Він має вигляд кубічної параболи (рис. 5.2).

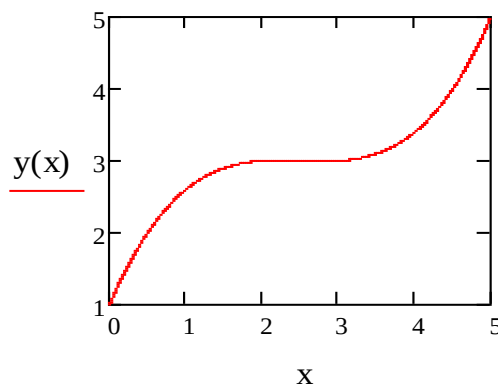


Рис. 5.2. Графік функції $y = \frac{2}{15} x^3 - x^2 + \frac{37}{15} x + 1$

Приклад 5. Знайти інтерполяційний поліном Лагранжа для функції $y = \sin(\pi x)$ за такими значеннями функції в вузлах: $x_0 = 0$, $x_1 = 1/6$, $x_2 = 1/4$, $x_3 = 1/2$.

Розв'язання. В заданих вузлах функція дорівнює: $y_0 = 0$, $y_1 = 1/2$, $y_2 = \sqrt{\frac{1}{2}}$, $y_3 = 1$.

Користуючись формулою Лагранжа, отримуємо інтерполяційний поліном:

$$y = -3,764 x^3 - 0,4903 x^2 + 3,186 x.$$

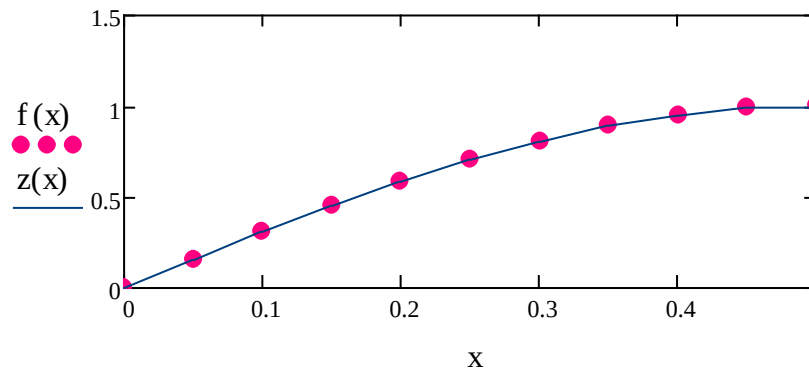


Рис. 5.3. Графік функції $y = \sin(\pi x)$

На відрізку $0 \leq x \leq 0,5$

$$\sin(\pi x) \approx -3,764 x^3 - 0,4903 x^2 + 3,186 x.$$

Ми бачимо практично повний збіг графіків функцій:

$$y = f(x) = \sin(\pi x) \quad (\text{точки});$$

$$z(x) = -3,764x^3 - 0,4903x^2 + 3,186x \quad (\text{тонка лінія}).$$

Інтерполяція функцій в Python

Для розрахунків та побудови графіків зручно користуватися пакетами Python для чисельних розрахунків, наприклад, підмодуль `scipy.interpolate`. Документація та опис всіх функцій модуля тут: <https://docs.scipy.org/doc/scipy/reference/interpolate.html#module-scipy.interpolate>. Нижче наведено приклад використання функцій із `scipy.interpolate` для проведення інтерполяції функції, яка задана масивами точок `x_data` та `y_data`.

Підмодуль [scipy.interpolate](#) містить методи для інтерполяції.

Задача інтерполювання полягає у тому, щоб за значенням функції $f(x)$ у кількох точках відрізка відновити її значення інших точках цього відрізка. Зрозуміло, така задача допускає скільки завгодно багато розв'язків. Задача інтерполювання виникає, наприклад, у тому випадку, коли відомі результати вимірювання $y_h = f(x, k)$ деякої фізичної величини $f(x)$ в точках $x_k, k = 0, 1, \dots, n$ і потрібно визначити її значення в інших точках. Інтерполювання використовується також при згущенні таблиць, коли обчислення значень $f(x)$ трудомістке. Іноді виникає необхідність наближеної заміни або апроксимації цієї функції іншими функціями, які легко обчислити. Зокрема, розглядається задача про найкраще наближення в нормованому просторі H , коли задану функцію $f \in H$ потрібно замінити лінійною комбінацією φ заданих елементів із H так, щоб відхилення $\|f - \varphi\|$ було мінімальним. Результати та методи теорії інтерполювання та наближення функцій знайшли широке застосування у чисельному аналізі, наприклад при виведенні формул чисельного диференціювання та інтегрування, при побудові сіткових аналогів задач математичної фізики.

Щоб розглянути інтерполяцію, згенеруємо дані. Тобто згенеруємо масив вузлів інтерполяційної сітки `x_data` та значень `y_data` деякої функції f у вузлах цієї сітки при різній кількості відомих точок.

```
import numpy as np
import matplotlib.pyplot as plt

L = 0
R = np.pi
n_points = (4, 10, 64, 65)
f = np.sin
xh = np.linspace(L, R, 100)
yh = f(xh)
```

```

def generate_data(f, L, R, n):
    x_data = np.linspace(L, R, n-1)
    y_data = f(x_data)
    return x_data, y_data

def plot_problem(ax, x_data, y_data):
    ax.plot(xh, yh, linewidth=0.5, label="Вихідна залежність")
    ax.scatter(x_data, y_data, marker="x", color="red", label="Відомі дані")
    ax.set_xlabel("$x$")
    ax.set_ylabel("$y$")

fig, axs = plt.subplots(ncols=2, nrows=2, figsize=(16, 16), layout="tight")
for ax, N in zip(axs.flatten(), n_points):
    x_data, y_data = generate_data(f, L, R, N)
    plot_problem(ax, x_data, y_data)
    ax.set_title(f"Вихідні дані  $N={N}$ ")
    ax.legend()

```

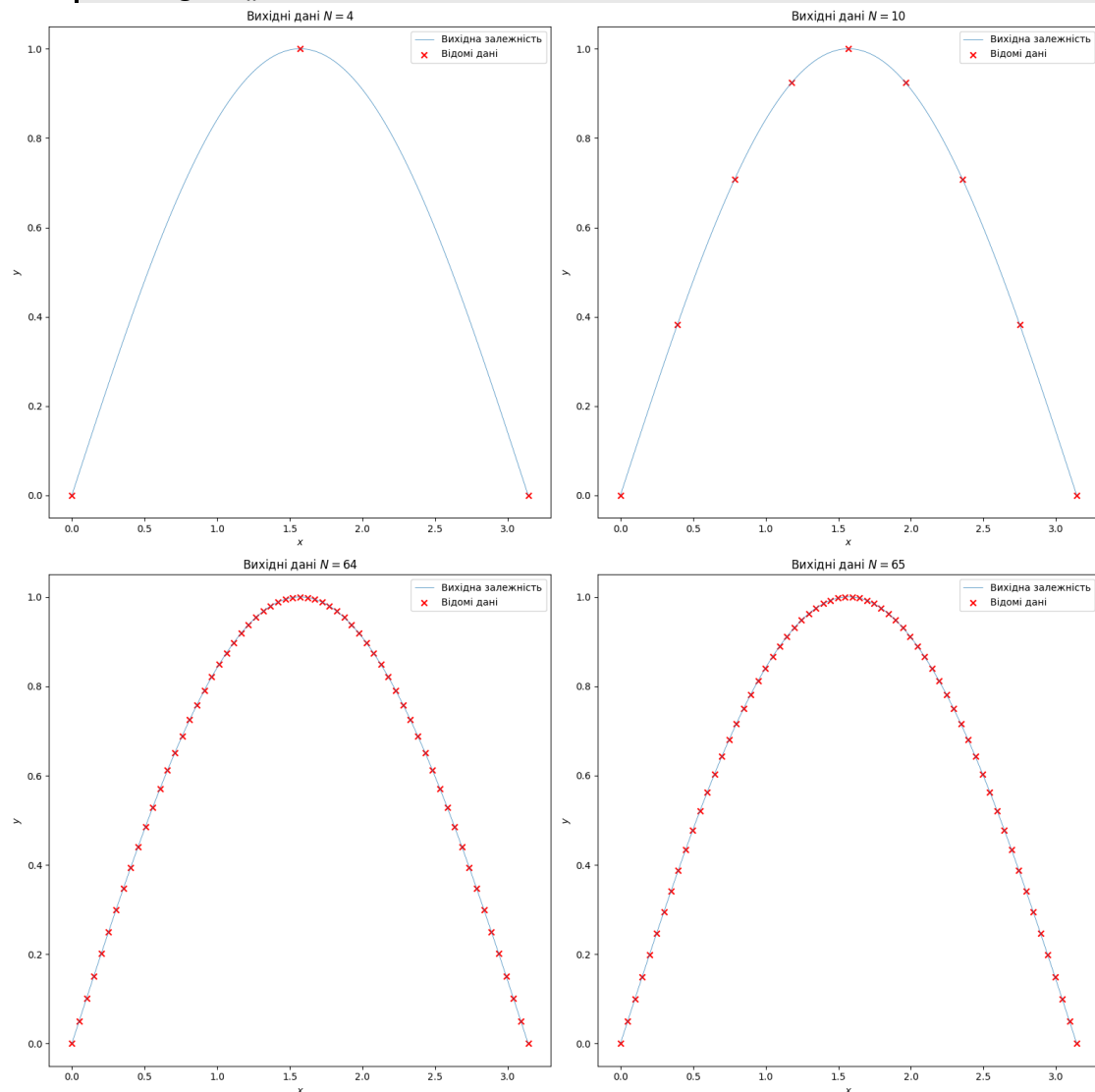


Рис. 5.4. Генерація вхідних даних

Тепер, коли дані готові, залишилося зробити інтерполяцію. Методів інтерполяції є дуже багато. Розглянемо поліноміальну інтерполяцію та сплайн інтерполяцію.

Поліноміальна інтерполяція ґрунтується на тому факті, що існує єдиний поліном $P_N(x) = a_0 + a_1x + \dots + a_Nx^N$ ступеня N , що проходить через точки $\{(x_1, y_1), \dots, (x_{N+1}, y_{N+1})\}$, якщо $x_i \neq x_j, \forall i, j = 1, \dots, N + 1, i \neq j$.

Недоліком поліноміальної інтерполяції є її нестабільність. Поліноміальна інтерполяція поводитьися дуже погано при великих N .

```
from scipy import interpolate

fig, axs = plt.subplots(ncols=2, nrows=2, figsize=(16, 16), layout="tight")
for ax, N in zip(axs.flatten(), n_points):
    x_data, y_data = generate_data(f, L, R, N)
    plot_problem(ax, x_data, y_data)

    poly = interpolate.KroghInterpolator(x_data, y_data)
    y_poly = poly(xh)
    ax.plot(xh, y_poly, label="Інтерполуючий поліном")

    ax.set_title(f"Результат інтерполяції  $\{N=\}$ ")
    ax.legend()
```

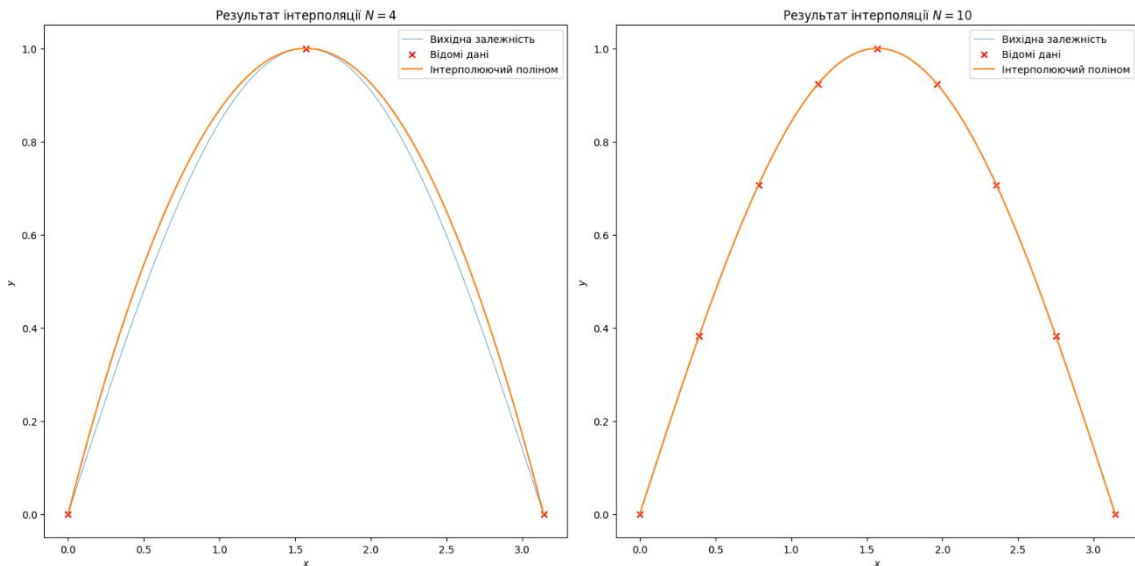


Рис. 5.5. Результат інтерполяції при $N = 4, N = 5$.

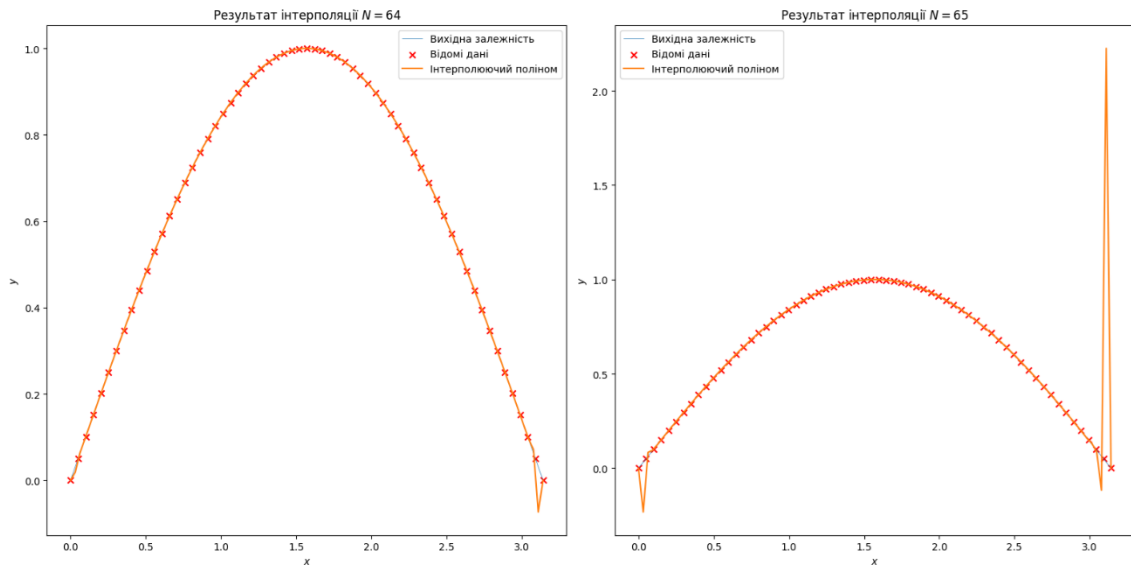


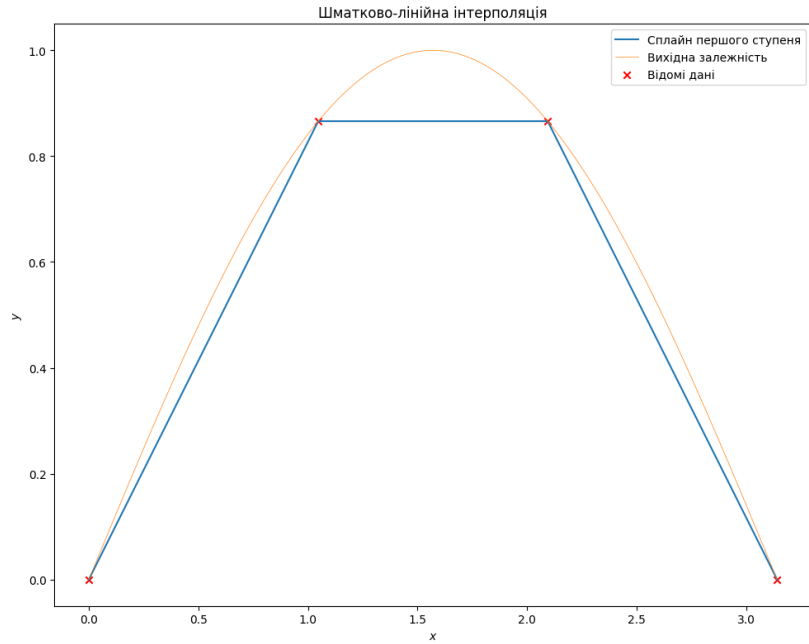
Рис. 5.6. Результат інтерполяції при $N = 64$, $N = 65$.

Сплайн інтерполяція. Інтерполювання многочленом $P_N(x)$ на всьому відрізку $[a, b]$ з використанням великої кількості вузлів інтерполяції часто призводить до поганого наближення, що пояснюється сильним накопиченням похибок у процесі обчислень. З іншого боку, через розбіжності процесу інтерполяції збільшення кількості вузлів має призводити до підвищення точності. Щоб уникнути великих похибок, весь відрізок $[a, b]$ розбивають на часткові відрізки і кожному з часткових відрізків приблизно замінюють функцію $f(x)$ многочленом невисокого ступеня (так звана кусочно-поліноміальна інтерполяція). Одним із способів інтерполювання по всьому відрізку є інтерполювання за допомогою сплайн-функцій. Сплайн-функцією або сплайном називають шматково-поліноміальну функцію, визначену на відрізку $[a, b]$ яка має на цьому відрізку кілька безперервних похідних.

Наприклад, якщо взяти як сплайн-функцію поліном першого ступеня, то на виході вийде шматково-лінійна інтерполяція.

```
x_data, y_data = generate_data(f, L, R, 5)
linear = interpolate.interp1d(x_data, y_data, kind="linear")
y_linear = linear(xh)

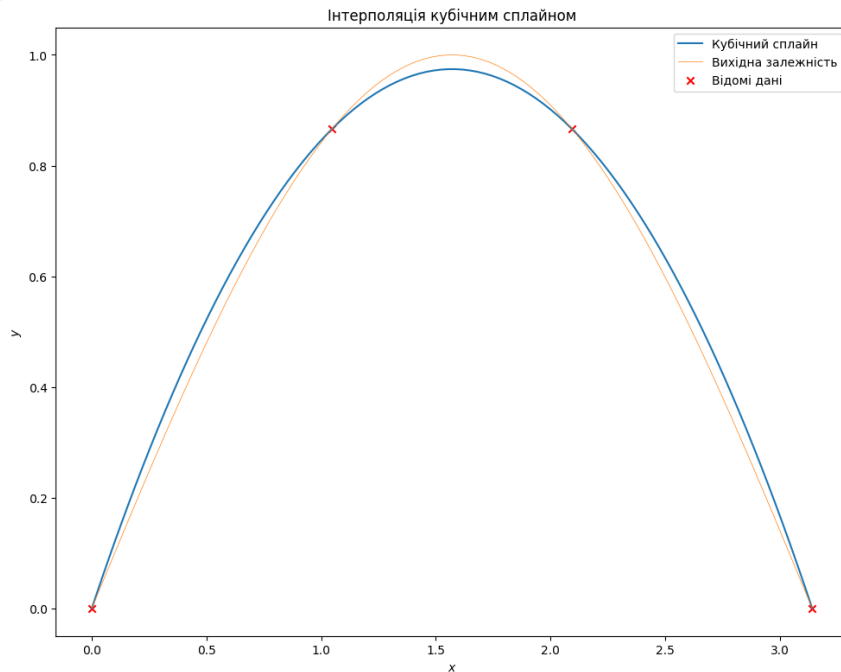
fig, ax = plt.subplots(figsize=(10, 8), layout="tight")
ax.plot(xh, y_linear, label="Сплайн першого ступеня")
plot_problem(ax, x_data, y_data)
ax.legend()
ax.set_title("Шматково-лінійна інтерполяція")
```



Дуже часто вибирають поліном третього ступеня, які дозволяють будувати гладкі інтерполіруючі поліному і розумно поведуться між точками у багатьох випадках.

```
cubic = interpolate.interp1d(x_data, y_data, kind="cubic")
y_cubic = cubic(xh)

fig, ax = plt.subplots(figsize=(10, 8), layout="tight")
ax.plot(xh, y_cubic, label="Кубічний сплайн")
plot_problem(ax, x_data, y_data)
ax.legend()
ax.set_title("Інтерполяція кубічним сплайном")
```



Апроксимація функцій

При інтерполюванні ставиться за мету відновити вихідну залежність $y = f(x)$ в такий спосіб по набору точок, якими функція відбувається у точності. Але якщо значення вхідної функції f в точках $x_i, i = 0, \dots, N$ відомо не точно, а в межах якоїсь похибки або з точністю до якоїсь випадкової величини, тобто, наприклад, $y_i = f(x_i) + \varepsilon_i, i = 0, \dots, N$. У такому разі така сувора вимога на те, щоб розв'язок $\varphi(x)$ точно проходив через усі точки $(x_i, y_i), i = 0, \dots, N$ не має великого сенсу і може бути ослабленим, наприклад, до пошуку такої функції φ з певного класу функцій, яка найменше відхиляється від значень $(x_i, y_i), i = 0, \dots, N$.

Дуже поширеним підходом у разі є метод найменших квадратів, у якому відхилення формалізується як суми квадратів відхилень апроксимуючої функції φ в точках $x_i, i = 0, \dots, N$ від вимірів $y_i, i = 0, \dots, N$. Нехай розв'язок шукається у класі функцій $\{\varphi_p: \mathbb{R} \rightarrow \mathbb{R} \mid p \in P\}$, тоді завдання апроксимації зводиться до задачі мінімізації

$$\sum_{i=0}^N \|\varphi_p(x_i) - y_i\|^2 \sim \min_{p \in P}.$$

Багато методів апроксимації легко узагальнюються на багатовимірні випадки.

При такій постановці часто виділяють лінійну та нелінійну апроксимацію. Апроксимація є **лінійною**, якщо $\varphi_p(x)$ лінійно залежить від параметрів p .

Апроксимація функцій реалізована в багатьох інших бібліотек, таких як statsmodels (<https://www.statsmodels.org/stable/index.html>), scikit-learn (<https://scikit-learn.org/stable/>) та деяких інших.

Лінійна апроксимація. Дуже поширеним випадком лінійної апроксимації є поліноміальна апроксимація, при якій розв'язок шукається у вигляді полінома $P_N(x) = a_0 + a_1x + \dots + a_nx^n$. Тобто розв'язується задача

$$\sum_{i=0}^N \|a_0 + a_1x_i + \dots + a_nx_i^n - y_i\|^2 \sim \min_{a_0, a_1, \dots, a_n \in \mathbb{R}^{n+1}}.$$

Якщо всі $x_i, i = 0, \dots, N$ всі різні, то при $n = N$ існує глобальний мінімум, який відповідає інтерполюючому поліному, який проходить через усі точки, тому зазвичай розв'язок шукається при $n < N$.

Згенеруємо дані для тестування поліноміальної інтерполяції при різних n . В якості вхідної залежності виберемо функцію

$$f(x) = 2e^{-3x} + 4,$$

згенеруємо $N = 11$ точок $x_i, i = 0, \dots, N$ рівномірно розподілених на відрізок $[0, 1]$ і штучно додамо розподілений по Гауссу шум до значень функції f в цих точках: $\{y_i = f(x_i) + \varepsilon_i, \varepsilon_i \sim \mathcal{N}(0, \sigma^2), i = 0, \dots, 10\}$

```
L = 0
R = 1

def f(x):
    return 2*np.exp(-3*x) + 4

N = 11
sigma = 0.2
x_data = np.linspace(L, R, N)
y_data = f(x_data) + np.random.normal(loc=0, scale=sigma, size=N)

xh = np.linspace(0, 1, 100)
yh = f(xh)

fig, ax = plt.subplots(figsize=(8, 8), layout="tight")
plot_problem(ax, x_data, y_data)
ax.legend()
```

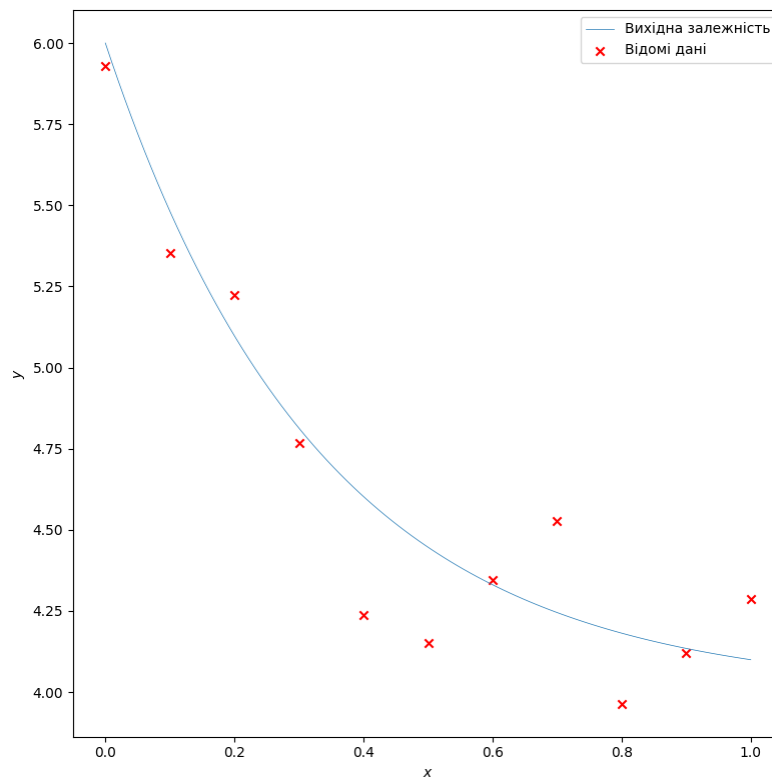


Рис.5.9. Генерація вхідних даних з додаванням шуму

Розв'язати задачу поліноміальної апроксимації можна засобами модуля `numpy.polynomial`. Зокрема, підігнати поліном заданого ступеня методом найменших квадратів можна за допомогою методу `numpy.polynomial.Polynomial.fit`.

(<https://numpy.org/doc/stable/reference/generated/numpy.polynomial.polynomial.Polynomial.fit.html>).

```
from numpy.polynomial import Polynomial
```

```
degrees = (1, 2, 3, 4, 5, 10)
```

```
fig, axs = plt.subplots(figsize=(16, 24), nrows=3, ncols=2, layout="tight")
```

```
for ax, degree in zip(axs.flatten(), degrees):
```

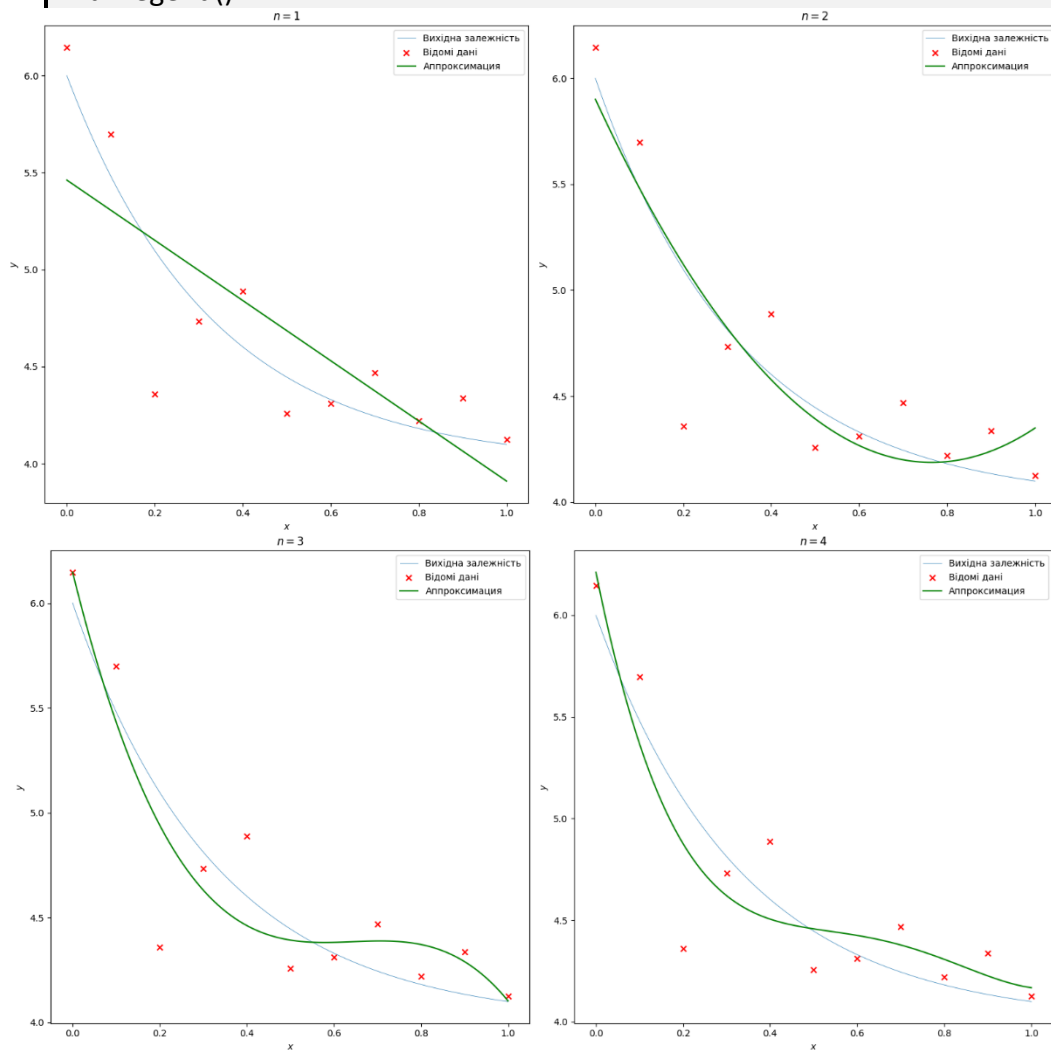
```
    approximation = Polynomial.fit(x_data, y_data, degree)
```

```
    plot_problem(ax, x_data, y_data)
```

```
    ax.plot(xh, approximation(xh), color="green", label="Аппроксимация")
```

```
    ax.set_title(f"$n = {degree}$")
```

```
    ax.legend()
```



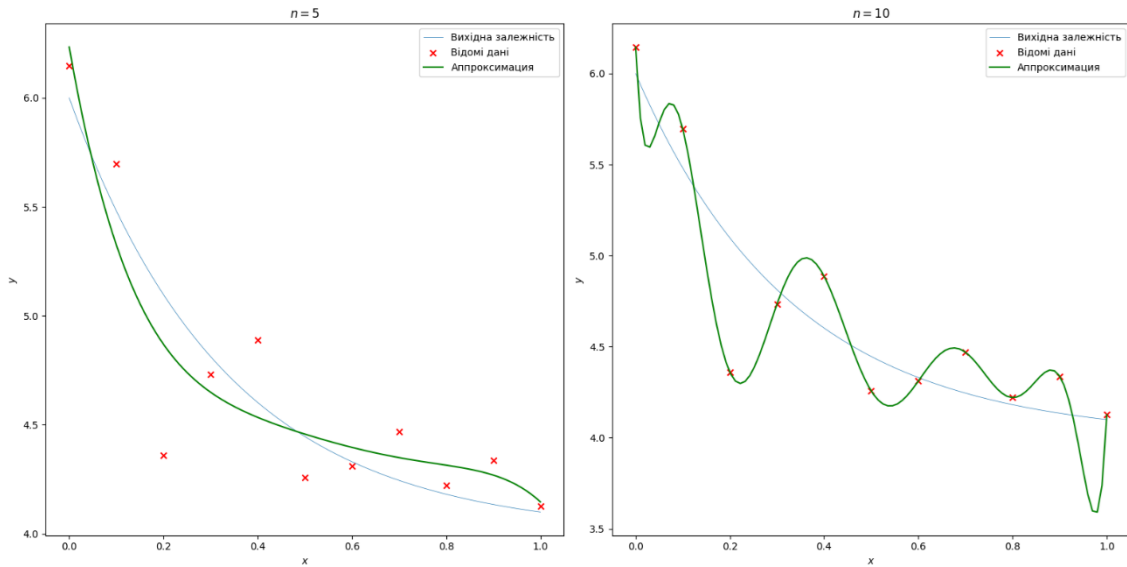


Рис.5.10. Апроксимація поліномами різних ступеней

У більш загальному випадку при лінійній апроксимації розв'язок шукається у вигляді лінійної комбінації будь-яких функцій $\varphi(x) = c_1\varphi_1(x) + \dots + c_n\varphi_n(x)$ та задача апроксимації зводиться до підбору коефіцієнтів за допомогою мінімізації

$$\sum_{i=0}^N \|c_1\varphi_1(x) + \dots + c_n\varphi_n(x_i) - y_i\|^2 \sim \min_{c_1, \dots, c_n \in \mathbb{R}^n}.$$

Поліноміальна інтерполяція виходить при $\varphi_i(x) = x^i$. Можна показати, що розв'язання оптимізаційної задачі вище еквівалентної мінімізації нев'язки системи лінійних алгебраїчних рівнянь

$$\begin{pmatrix} \varphi_1(x_0) & \cdots & \varphi_k(n_0) \\ \vdots & \ddots & \vdots \\ \varphi_1(x_N) & \cdots & \varphi_k(n_N) \end{pmatrix} \begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} y_0 \\ \vdots \\ y_N \end{pmatrix}$$

За розв'язання таких задач відповідає метод `linalg.lstsq` (<https://numpy.org/doc/stable/reference/generated/numpy.linalg.lstsq.html>). Як приклад апроксимуємо той же набір точок, але розв'язок будемо шукати у вигляді

$$\varphi_{\alpha, \beta, \gamma}(x) = \alpha + \beta e^{-x} + \gamma x^2.$$

```
from scipy import linalg

def approximation(x, alpha, beta, gamma):
    return alpha + beta*np.exp(-x) + gamma*np.square(x)
```

```

A = np.vstack([
    np.ones_like(x_data), np.exp(-x_data), np.square(x_data)
]).T

b = y_data

(alpha, beta, gamma), *_ = linalg.lstsq(A, b)

fig, ax = plt.subplots(figsize=(8, 8), layout="tight")
plot_problem(ax, x_data, y_data)
ax.plot(xh, approximation(xh, alpha, beta, gamma), color="green",
        label="Аппроксимация")
ax.legend()

```

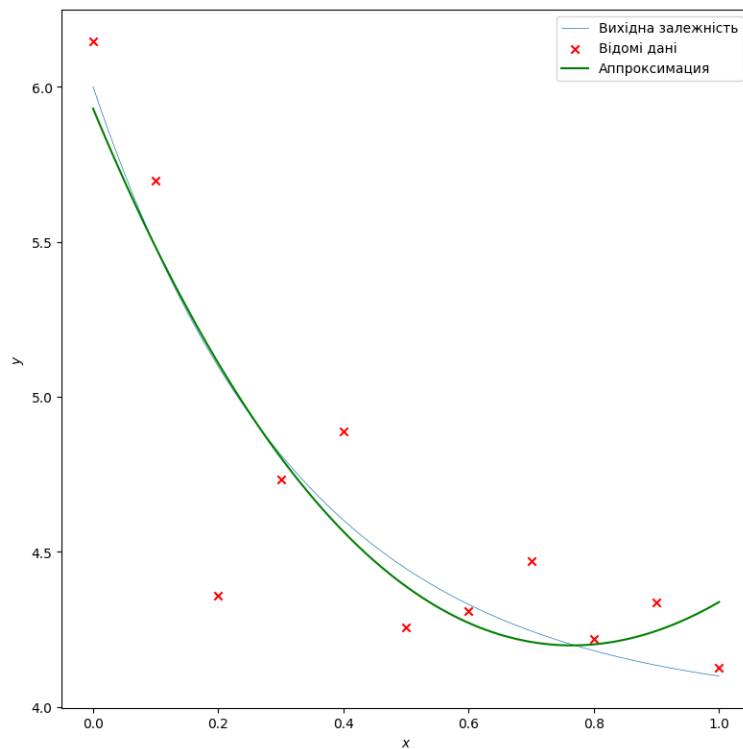


Рис. 5.11. Апроксимація функцією $\varphi_{\alpha,\beta,\gamma}(x)$

Індивідуальні завдання за темою “Апроксимація”

Функція $y = f(x)$, задана таблицею значень.

№	Таблиця					X_0	№	Таблиця					x_0
1	x	-2	-1	0	1	-1,25	16	X	-3	-2	-1	0	-2,25
	y	4	1	-2	-3			y	-2	-3	-1	0	
2	x	-1	0	1	2	-0,25	17	x	-2	-1	0	1	-1,25
	y	1	-2	-3	-1			y	-3	-1	0	7	
3	x	0	1	2	3	0,75	18	x	-1	0	1	2	-0,25
	y	-2	-3	-1	0			y	-1	0	7	4	
4	x	1	2	3	4	1,75	19	x	0	1	2	3	0,75
	y	-3	-1	0	7			y	0	7	4	1	
5	x	2	3	4	5	2,75	20	x	-4	-3	-2	-1	-3,25
	y	-1	0	7	4			y	-3	-1	0	7	
6	x	3	4	5	6	3,75	21	x	-5	-4	-3	-2	-4,25
	y	0	7	4	1			y	4	1	-2	-3	
7	x	1	2	3	4	1,75	22	x	4	5	6	7	4,75
	y	4	1	-2	-3			y	1	-2	-3	-1	
8	x	-4	-3	-2	-1	-3,25	23	x	5	6	7	8	5,75
	y	1	-2	-3	-1			y	-2	-3	-1	0	
9	x	3	4	5	6	3,75	24	x	6	7	8	9	6,75
	y	-2	-3	-1	0			y	-3	-1	0	7	
10	x	4	5	6	7	4,75	25	x	-3	-2	-1	0	-2,25
	y	-3	-1	0	7			y	-1	0	7	4	
11	x	5	6	7	8	5,75	26	x	-2	-1	0	1	-1,25
	y	-1	0	7	4			y	0	7	4	1	
12	x	6	7	8	9	6,75	27	x	0	1	2	3	0,75
	y	0	7	4	1			y	0	1	2	3	
13	x	-5	-4	-3	-2	-4,25	28	x	1	2	3	4	1,75
	y	-2	-3	-1	0			y	1	-2	-3	-1	
14	x	3	4	5	6	3,75	29	x	4	5	6	7	4,75
	y	4	1	-2	-3			y	-1	0	7	4	
15	x	2	3	4	5	2,75	30	x	5	6	7	8	5,75
	y	1	-2	-3	-1			y	0	7	4	1	

Необхідно:

1) побудувати інтерполяційний поліном за методом складання системи рівнянь; розрахувати наближене значення функції в точці x_0 ; побудувати графік. Зробити це саме в середовищі за допомогою Python (або іншої мови програмування).

2) побудувати інтерполяційний поліном у формі Лагранжа; розрахувати наближене значення функції в точці x_0 ; побудувати графік. Зробити це саме за допомогою мови Python (або іншої мови програмування).

Контрольні запитання

1. Що таке інтерполяція?
2. Що таке локальна інтерполяція?
3. Що таке глобальна інтерполяція?
4. Що таке апроксимація?
5. Що таке лінійна інтерполяція?
6. Що таке квадратична інтерполяція?
7. Що таке екстраполяція?

Лабораторна робота № 6

Регресія. Види регресії. Метод найменших квадратів

Мета роботи: засвоїти методи регресії.

Регресія

Результати експериментів завжди мають якусь похибку і ця похибка часто буває досить великою. В таких випадках інтерполяційна крива не буде відповідати справжній залежності. Подібні ситуації виникають доволі часто, тому інтерполяція рідко використовується при аналізі результатів експериментів. Частіше використовується так званий регресійний аналіз. *Регресією називається підгонка параметрів простої функції для найкращої апроксимації експериментальних даних (від латинського *approximo* – наближаюсь).*

Для пошуку оптимальних значень параметрів функцій часто використовують *метод найменших квадратів*. Суть цього методу полягає в тому, щоб знайти такі коефіцієнти поліному, при яких сума квадратів відхилень (різниць між обчисленими та експериментальними значеннями функції) була мінімальною. Розглянемо застосування цього методу на прикладі лінійної регресії.

Лінійна регресія

Лінійна регресія є найпростішою, але використовується частіше за інші види регресії. Суть її в знаходженні таких значень параметрів a і b , щоб пряма $y = a + bx$ найкращим чином апроксимувала заданий набір точок.

Нехай відомо, що величини x та y зв'язані деякою формулою $y = f(x)$, але в формулі є невідомі параметри a та b . Запишемо залежність між ними, як

$$y = f(x, a, b).$$

Припустимо, що значення функції y обчислені за формулою, яка поки ще нам невідома, і ці значення дорівнюють $\hat{y}_1, \hat{y}_2, \dots, \hat{y}_n$.

На практиці маємо такі результати у вигляді таблиці.

x_1	x_2	x_3	...	x_n
y_1	y_2	y_3	...	y_n

Необхідно знайти такі a і b , щоб сума квадратів відхилень була мінімальною.

Відхиленнями називають $y_1 - \hat{y}_1, y_2 - \hat{y}_2, \dots, y_n - \hat{y}_n$. Тоді сума квадратів відхилень

$$\sum_{k=1}^n (y_k - \hat{y}_k)^2 = (y_1 - \hat{y}_1)^2 + (y_2 - \hat{y}_2)^2 + \dots + (y_n - \hat{y}_n)^2.$$

Можна записати:

$$\hat{y}_1 = x_1 + ab; \hat{y}_2 = x_2 + ab; \dots; \hat{y}_n = x_n + ab.$$

Тоді

$$\begin{aligned} \sum_{k=1}^n (ax_k + b - y_k)^2 &= (x_1 + ab - y_1)^2 + (ax_2 + b - y_2)^2 + \dots + \\ &+ (ax_n + b - y_n)^2. \end{aligned}$$

Значення a і b , задовільняючи вимогам, знаходять із системи рівнянь:

$$\begin{cases} M_{xx} + M_x b = M_{xy} \\ M_x a + b = M_y \end{cases},$$

де

$$M_x = \frac{1}{n} \sum_{k=1}^n x_k; \quad M_y = \frac{1}{n} \sum_{k=1}^n y_k; \quad M_{xx} = \frac{1}{n} \sum_{k=1}^n x_k^2; \quad M_{xy} = \frac{1}{n} \sum_{k=1}^n x_k y_k$$

Розв'язавши систему рівнянь, знайдемо значення a і b та запишемо рішення у вигляді рівняння: $y = a + bx$, з відповідними значеннями a та b .

Приклад 6. Нехай дана таблиця, отримана шляхом експерименту. Потрібно методом найменших квадратів знайти емпіричну формулу залежності у вигляді лінійної функції.

x	0	3	5	8	10	14	17	20	22	24
y	1,0	2,	3,9	5,1	6,8	8,3	10,7	11,8	13,6	12,9
	2	5	2	6	2	6	4	2	4	6

Розв'язання. Визначимо значення $M_x, M_{xy}, M_{xx}, M_{xy}$.

$$M_x = 1/10 (0 + 3 + 5 + 8 + 10 + 14 + 17 + 20 + 22 + 24) = 12,3.$$

$$M_y = 1/10 (1,02 + 2,5 + 3,92 + 5,16 + 6,82 + 8,36 + 10,74 + 11,82 + 13,64 + 12,96) = 7,694.$$

$$M_{xx} = 1/10 ((0^2 + 3^2 + 5^2 + 8^2 + 10^2 + 14^2 + 17^2 + 20^2 + 22^2 + 24^2) = 214,3.$$

$$M_{xy} = 1/10 (0 \cdot 1,02 + 3 \cdot 2,5 + 5 \cdot 3,92 + 8 \cdot 5,16 + 10 \cdot 6,82 + 14 \cdot 8,36 + 17 \cdot 10,74 + 20 \cdot 11,82 + 22 \cdot 13,64 + 24 \cdot 12,96) = 128,372.$$

Параметри a та b знайдемо з системи

$$\begin{cases} 214,3 a + 12,3 b = 128,4 \\ 12,3 a + b = 7,694 \end{cases}$$

Розв'язавши систему, отримаємо: $a = 0,54$; $b = 1,05$.

Лінійна функція має вигляд: $y = 0,54x + 1,05$.

На відміну від лінійної регресії поліноміальна регресія апроксимує дані поліномом заданого ступеня. Це означає, що замість прямої лінії (як у лінійній регресії) модель може бути представлена у вигляді кривої, наприклад параболи або кубічної кривої.

Завдання за темою “ Регресія. Види регресії. Метод найменших квадратів”

Необхідно:

1) методом найменших квадратів знайти емпіричну формулу залежності у вигляді лінійної функції; побудувати графік.

2) провести лінійну та поліноміальну регресію за допомогою мови Python (або іншої мови програмування).

Дані для розрахунків взяти з попереднього завдання.

Контрольні запитання

1. Що таке регресія?
2. Види регресії.
3. Суть методу найменших квадратів.

Лабораторна робота № 7

Чисельне інтегрування

Мета роботи: засвоїти методи чисельного інтегрування.

Теоретичні відомості

Чисельне інтегрування

При розв'язуванні задач фізики або геометрії, часто буває необхідно обчислити визначений інтеграл від функції в межах від a до b (формула Ньютона – Лейбніца):

$$\int_a^b f(x)dx,$$

де $f(x)$ – функція неперервна на відрізку $[a, b]$.

Інтеграл обчислюється за формулою Ньютона – Лейбніца

$$\int_a^b f(x)dx = F(b) - F(a),$$

де $F(x)$ – первісна від функції для заданої функції $f(x)$.

Геометрична інтерпретація: визначений інтеграл при $f(x) > 0$ дорівнює площі криволінійної трапеції, яка обмежена функцією $f(x)$, віссю абсцис та прямими $x = a$ і $x = b$ (границями інтегрування).

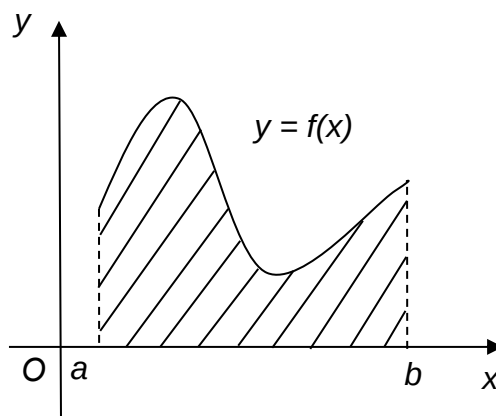


Рис.7.1. Геометрична інтерпретація визначеного інтеграла

Однак у багатьох випадках не вдається знайти аналітичної формули для визначення $F(x)$ або функція задана таблицею.

Таким інтегралом може бути, наприклад інтеграл

$$\int_2^3 \frac{\sin(x)}{x} dx$$

Для цієї функції неможливо виразити первісну $F(x)$ через елементарні функції. З іншого боку, площа криволінійної трапеції, що задається таким інтегралом, існує. Тоді, повинно існувати значення інтегралу, яке, однак, ми не можемо знайти точно.

У таких випадках використовують методи чисельного інтегрування. Суть методу полягає у такому. При чисельному інтегруванні інтервал інтегрування розбивається на відрізки, на яких крива $y = f(x)$ замінюється простішими функціями, інтеграли від яких можна обчислити. Таким чином, площа криволінійної трапеції приблизно замінюється сумою площин елементарних геометричних фігур.

Формули наближеного інтегрування називаються **квадратурними формулами**.

Метод прямокутників

Поділимо інтервал інтегрування $[a, b]$ на n рівних частин, кожна довжиною $h = \frac{b-a}{n}$. Точками поділу будуть $x_0 = a$, $x_1 = a + h$, $x_2 = a + 2h$, ... $x_{n-1} = a + (n-1)h$, $x_n = b$. Ці точки називають вузлами. Будемо обчислювати значення функції в вузлах, позначивши їх $y_0, y_1, \dots, y_n$.

Розглянемо метод “лівих”, “правих” та “середніх” прямокутників.

Метод “лівих” прямокутників

Приблизне значення інтегралу отримаємо у вигляді суми площин n прямокутників, висота яких дорівнює значенню $f(x)$ на лівому краю кожного підінтервалу.

$$S = \int_a^b f(x) dx \approx hy_0 + hy_1 + \dots + hy_{n-1} = h(y_0 + y_1 + \dots + y_{n-1}).$$

Тобто, формула чисельного інтегрування має вигляд:

$$S = \int_a^b f(x) dx \approx h \sum_{i=0}^{n-1} y_i$$

и називається формулою “лівих” прямокутників.

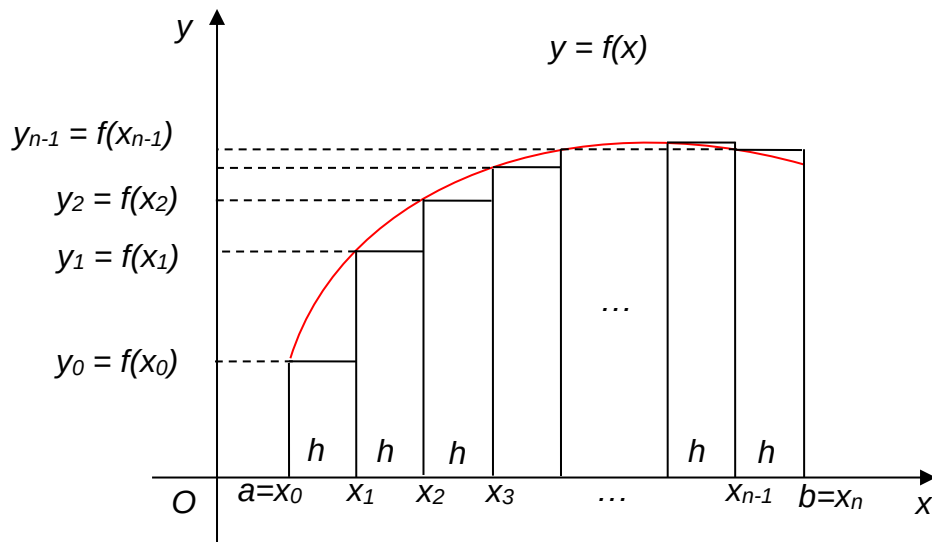


Рис.7.2. Геометрична інтерпретація метода “лівих” прямокутників

Метод “правих” прямокутників

Якщо як наближене значення площі для кожного підінтервалу прийняти площу прямокутника, висота якого дорівнює значенню $f(x)$ на правому краю підінтервалу, то формула чисельного інтегрування має вигляд:

$$S = \int_a^b f(x)dx \approx h \sum_{i=1}^n y_i$$

і називається формулою “правих” прямокутників.

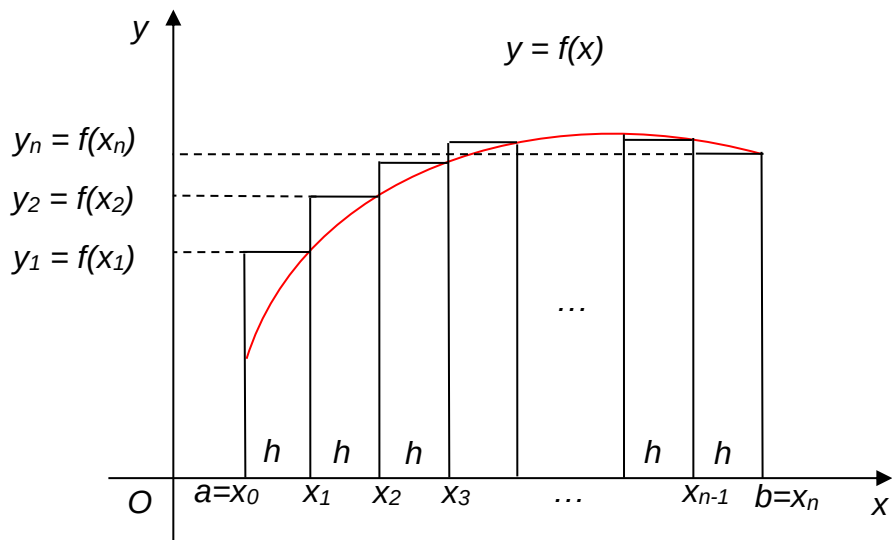


Рис.7.3. Геометрична інтерпретація метода “правих” прямокутників

Метод “середніх” прямокутників

Існує третя модифікація методу прямокутників – метод “середніх” прямокутників. У такому випадку в якості наближеного значення площі для кожного підінтервалу можна прийняти площу

прямокутника, висота якого дорівнює значенню $f(x)$ у середній точці підінтервалу.

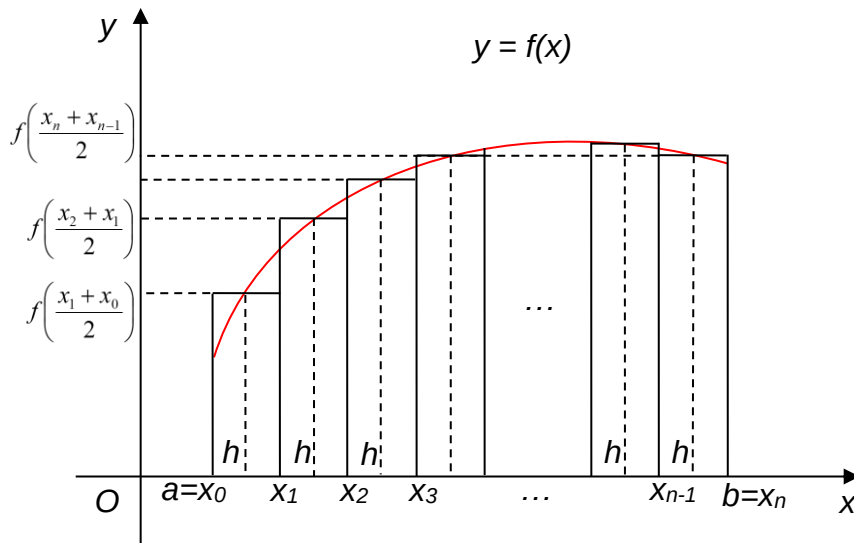


Рис.7.4. Геометрична інтерпретація метода “середніх” прямокутників

Тоді формула чисельного інтегрування має вигляд:

$$S = \int_a^b f(x)dx \approx h \sum_{i=0}^{n-1} y(x_i + \frac{h}{2})$$

Метод прямокутників – це найпростіший, але разом с тим найбільш приблизний метод чисельного інтегрування. Очевидно, що чим більшем буде кількість n відрізків поділу інтервалу, тим більш точним буде результат. Однак збільшення кількості відрізків проміжку інтегрування не завжди можливе. Тому часто використовують інші формули, які дають точніші результати при тому самому числі точок поділу.

Приклад 1. Обчислимо за методом “лівих” та “правих” прямокутників інтеграл:

$$\int_1^9 \frac{1}{x+2} dx.$$

Розв’язання. Прийемо, що інтервал поділено на $n = 4$ рівних частин. Тоді крок інтегрування $h = \frac{b-a}{n} = \frac{9-1}{4} = 2$.

Точки поділу $x_0 = 1$; $x_1 = 3$; $x_2 = 5$; $x_3 = 7$; $x_4 = 9$.

Визначимо значення функції $y_0 = f(x) = \frac{1}{x+2}$ у цих точках:

$$y_0 = f(x_0) = \frac{1}{1+2} = \frac{1}{3}; y_1 = f(x_1) = \frac{1}{3+2} = \frac{1}{5}; y_2 = f(x_2) = \frac{1}{1+2} = \frac{1}{7};$$

$$y_3 = f(x_3) = \frac{1}{7+2} = \frac{1}{9}; y_4 = f(x_4) = \frac{1}{9+2} = \frac{1}{11}.$$

Чисельне значення інтегралу за формулою “лівих” прямокутників:

$$I = h (y_0 + y_1 + y_2 + y_3) = \frac{1}{3} + \frac{1}{5} + \frac{1}{7} + \frac{1}{9} = 1,6024.$$

Чисельне значення інтегралу по формулі “правих” прямокутників:

$$I = h (y_1 + y_2 + y_3 + y_4) = \frac{1}{5} + \frac{1}{7} + \frac{1}{9} + \frac{1}{11} = 1,1053.$$

Ми бачимо велику розбіжність у результатах обчислень, тобто метод прямокутників дає велику похибку. Для підвищення точності розрахунків слід поділити інтервал на більшу кількість відрізків, тобто збільшити кількість кроків.

Меншу похибку дає інший метод – метод трапецій.

Метод трапецій

Якщо функцію на кожному відрізку апроксимувати прямою, яка проходить через кінцеві значення, то одержимо метод трапецій.

У цьому методі відрізок $[a, b]$ також розбивається на n рівних частин. На кожному відрізку $[x_i, x_{i+1}]$ крива $y = f(x)$ замінюється прямою, що проходить через дві відомі точки і створюється прямокутна трапеція з висотою $h = (b - a)/n$.

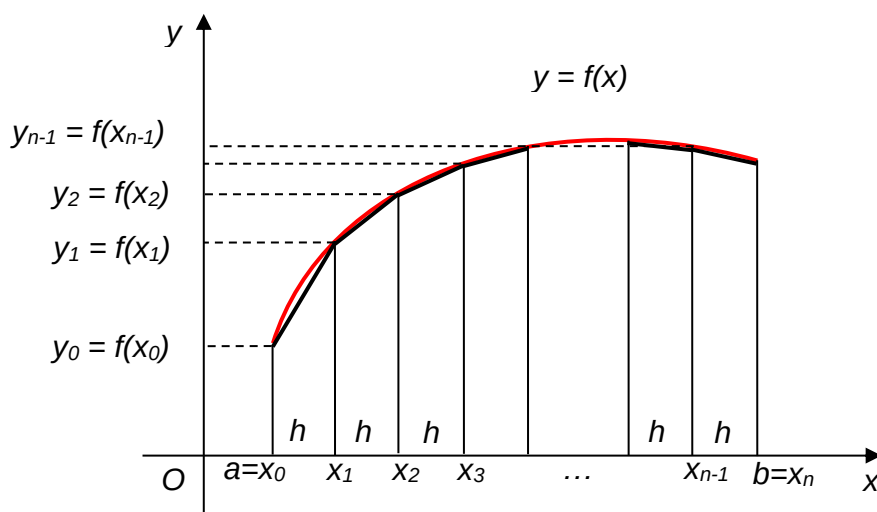


Рис.7.5. Геометрична інтерпретація метода трапецій

Таким чином, площа криволінійної трапеції наближено замінюється сумою площин елементарних геометричних трапецій

(площа трапеції висотою h і основами a, b обчислюється за формулою: $S = h \cdot (a + b)/2$. Очевидно, що площа такої фігури буде точніше відповідати площі криволінійної трапеції, ніж площа ступінчастої фігури, обчисленої за методом прямокутників.

Тоді

$$S = \int_a^b f(x)dx \approx$$

$$\approx h \frac{f(x_0) + f(x_1)}{2} + h \frac{f(x_1) + f(x_2)}{2} + \dots + h \frac{f(x_{n-2}) + f(x_{n-1})}{2} + h \frac{f(x_{n-1}) + f(x_n)}{2} =$$

$$= h \left(\frac{f(x_0)}{2} + f(x_1) + f(x_2) + \dots + f(x_{n-1}) + \frac{f(x_n)}{2} \right).$$

Таким чином, $S = \int_a^b f(x)dx \approx$

$$h \left(\frac{f(x_0)}{2} + f(x_1) + f(x_2) + \dots + f(x_{n-1}) + \frac{f(x_n)}{2} \right).$$

Отриману формулу можна записати у вигляді

$$S = \int_a^b f(x)dx \approx h \left(\frac{f(x_0)}{2} + \frac{f(x_n)}{2} + \sum_{i=1}^{n-1} f(x_i) \right)$$

Приклад 2. Обчислимо за методом трапецій інтеграл:

$$\int_0^1 \frac{1}{1+x^2} dx$$

Розв'язання. Прийнемо, що інтервал поділен на $n = 4$ рівних частин. Тоді крок інтегрування $h = \frac{b-a}{n} = \frac{1-0}{4} = 0.25$,

Точки поділу $x_0 = 0$; $x_1 = 0,25$; $x_2 = 0,5$; $x_3 = 0,75$; $x_4 = 1$.

Функція в точках:

$$y_0 = f(x_0) = 1; \quad y_1 = f(x_1) = \frac{1}{1+0.25^2}; \quad y_2 = f(x_2) = \frac{1}{1+0.5^2};$$

$$y_3 = f(x_3) = \frac{1}{1+0.75^2}; \quad y_4 = f(x_4) = \frac{1}{2}.$$

Тоді за формулою трапецій маємо:

$$\int_0^1 \frac{1}{1+x^2} dx = 0,25(1 + \frac{1}{1+0,25^2} + \frac{1}{1+0,5^2} + \frac{1}{1+0,75^2} + \frac{1}{2}) \approx 0,764$$

Метод парабол (Сімпсона)

Значне підвищення точності наближених формул чисельного інтегрування дає метод парабол (метод Сімпсона). Ідея методу виходить з того, що на частковому проміжку дуга параболі в загальному випадку “тісніше” прилягає до кривої $y = f(x)$, ніж хорда, яка з’єднує кінці дуги цієї кривої (метод трапецій). Тому значення площин відповідних елементарних трапецій, які обмежені зверху дугами парабол, є ближчими до значень площин відповідних часткових криволінійних трапецій, обмежених зверху дугою кривої $y = f(x)$, ніж значення площин відповідних прямолінійних трапецій.

Розглянемо функцію $y = f(x)$. Будемо вважати, що на відрізку $[a, b]$ вона додатна і неперервна. Знайдемо площу криволінійної трапеції $aABb$.

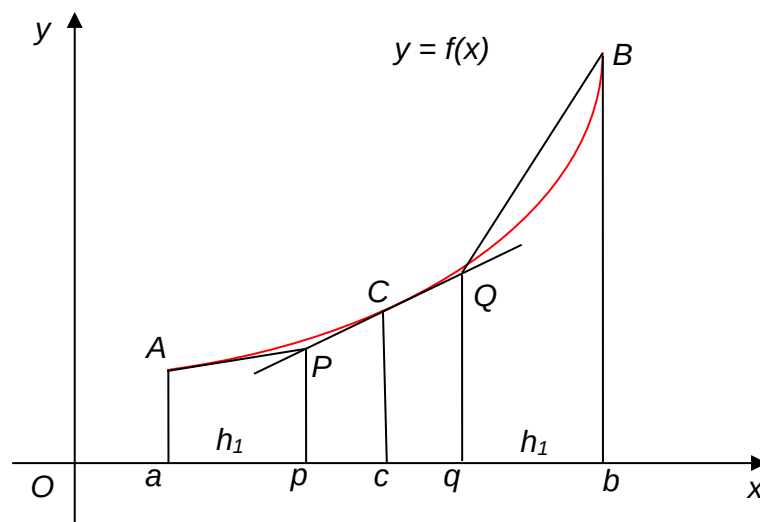


Рис.7.6. Геометрична інтерпретація метода Сімпсона

Поділимо відрізок $[a, b]$ точкою $c = \frac{a+b}{2}$ навпіл і в точці $C(c, f(c))$ проведемо дотичну до лінії $y = f(x)$. Після цього розділимо відрізок $[a, b]$ точками p і q на три рівні частини і проведемо через них прямі $x = p$ і $x = q$. Нехай P і Q – точки перетину цих прямих з дотичною. З’єднавши A з P і B з Q , отримаємо три прямолінійні трапеції $aAPp$, $pPQq$, $qQBb$. Тоді площу трапеції $aABb$ можна приблизно обчислити за формулою:

$$I \approx \frac{aA + pP}{2} \cdot h_1 + \frac{pP + qQ}{2} \cdot h_1 + \frac{qQ + bB}{2} \cdot h_1, \quad \text{де } h_1 = \frac{b-a}{3}.$$

Таким чином, отримуємо так звану “малу” формулу Сімпсона.

$$I \approx \frac{b-a}{6} \cdot (f(a) + 4f(c) + f(b))$$

В цьому випадку дуга ACB замінюється параболою, яка проходить через точки A, P, Q, B .

“Мала” формула Сімпсона дає інтеграл з достатньою точністю, коли графік функції мало “зігнутий”, в іншому випадку, коли функція більш складна, необхідно поділити відрізок $[a, b]$ на n частин и до кожної з них використати “малу” формулу Сімпсона. Обов’язковою вимогою є та, що n повинно бути парним.

Тоді отримуємо так звану “велику” формулу Сімпсона, яка має вигляд:

$$\int_a^b f(x) dx \approx \frac{h}{3} [f(x_0) + 4(f(x_1) + f(x_3) + \dots + f(x_{n-1})) + 2(f(x_2) + f(x_4) + \dots + f(x_{n-2})) + f(x_n)]$$

Загальний вигляд формули:

$$\int_a^b f(x) dx \approx \frac{h}{3} (Y_{\text{êð}} + 4Y_{\text{îäîäð}} + 2Y_{\text{îäð}}),$$

де $Y_{\text{êð}} = y_0 + y_n$, $Y_{\text{îäîäð}} = y_1 + y_3 + \dots + y_{n-1}$, $Y_{\text{îäð}} = y_2 + y_4 + \dots + y_{n-2}$, $h = (b-a)/n$.

Особливістю використання формули Сімпсона є те, що кількість відрізків поділу інтервалу інтегрування – парна. Якщо кількість відрізків поділу непарна, то слід використати формулу, яку називають **формулою «трьох восьмих» або формулою Ньютона**.

$$\int_a^b f(x) dx \approx \frac{3h}{8} (y_0 + 3y_1 + 3y_2 + 2y_3 + 3y_4 + 3y_5 + 2y_6 + \dots + y_n), \quad \text{або}$$

$$\int_a^b f(x) dx \approx \frac{3h}{8} (\Sigma_1 + 3\Sigma_2 + 2\Sigma_3),$$

де $h = \frac{b-a}{n}$;

$$\Sigma_1 = y_0 + y_n; \quad \Sigma_2 = y_1 + y_2 + y_4 + y_5 + \dots; \quad \Sigma_3 = y_3 + y_6 + y_9 + \dots$$

Зверніть увагу, число поділу n повинно бути кратним трьом.

Приклад 3. Обчислити інтеграл

$$I = \int_0^1 \frac{dx}{1+x^2}.$$

Розв'язання. Цей інтеграл легко обчислюється:

$$I = \arctg x \Big|_0^1 = \frac{\pi}{4} \approx 0,785398.$$

Для порівняння точності наближених формул обчислимо ще раз інтеграл

$$\int_0^1 \frac{dx}{1+x^2},$$

але тепер за формулою Сімпсона при $n = 4$. Розіб'ємо відрізок $[0, 1]$ на чотири рівні частини точками $x_0 = 0, x_1 = 1/4, x_2 = 1/2, x_3 = 3/4, x_4 = 1$ та обчислимо приблизно значення функції $f(x) = 1/(1+x)$ в точках: $y_0 = 1,0000, y_1 = 0,8000, y_2 = 0,9411765, y_3 = 0,64, y_4 = 0,5000$.

За формулою Сімпсона отримаємо:

$$\begin{aligned} \int_0^1 \frac{dx}{1+x^2} &\approx \frac{b-a}{3n} [y_0 + y_4 + 2y_2 + 4(y_1 + y_3)] = \\ &= \frac{1-0}{12} [1,0000 + 0,5000 + 2 \cdot 0,9411765 + 4(0,8000 + 0,64)] \approx 0,785. \end{aligned}$$

Очевидно, що обчислення за методом “лівих” або “правих” прямокутників дають найнижчу точність. Більш високу точність дає метод “середніх” прямокутників, трапецій та, особливо, Сімпсона.

Правило Рунге оцінки похибки

У кожній конкретній задачі необхідно визначити число точок поділу n для обчислювання інтегралу з необхідною точністю ε .

Для визначення n зручно користуватися правилом Рунге. Нехай ε – необхідна точність обчислювання інтегралу, тоді крок h повинен задовольняти умові:

$$h \leq \sqrt[4]{\varepsilon}.$$

За значенням h із співвідношення $h = \frac{b-a}{n}$ визначається n . При цьому для методу Сімпсона як n береться найближче парне ціле число, більше за $\frac{b-a}{n}$, а для методів прямокутників і трапецій – найближче ціле, більше за $\frac{b-a}{n}$.

Оцінку похибки можна провести також методом Рунге.

Нехай I_h – приблизне значення інтегралу, обчислене з кроком h , а I_{2h} – значення цього інтегралу, обчислене з кроком $2h$. Зверніть увагу, що чим менше крок h (тобто, більше n), тим точнішим буде наближене значення інтегралу.

Якщо

$$\frac{|I_h - I_{2h}|}{15} < \varepsilon, \quad (7.1)$$

де I_h і I_{2h} обчислені за методом Сімпсона, або

$$\frac{|I_h - I_{2h}|}{3} < \varepsilon, \quad (7.2)$$

де I_h і I_{2h} обчислені за по методом прямокутників або трапецій, то як приблизне значення інтегралу беруть значення I_h .

Якщо нерівність для відповідного методу не виконується, тоді знайдене значення інтегралу не відповідає заданій точності.

Тоді проводять нові обчислення з кроком $\frac{h}{2}$ і знову перевіряють виконання нерівності (7.1) або (7.2). Цю процедуру зменшення кроку виконують доти, поки відповідна нерівність не стане дійсною.

Приклад 4. Обчислити інтеграл $\int_{0,215}^{1,523} e^{\cos x} dx$ з точністю $\varepsilon = 0,001$ за методом Сімпсона.

Розв'язання. Оцінимо величину кроку h :

$$h \leq \sqrt[4]{0,001} \approx 0,1778.$$

Выберемо $n = \frac{b-a}{h} = \frac{1,523-0,215}{0,1718} = 7,3566$. Оскільки в методі

Сімпсона n обов'язково повинно бути парним, його следует округлити в більший бік, тобто беремо $n = 8$. Для цього n визначаємо крок

$$h = \frac{1,523-0,215}{8} = 0,1635.$$

(оскільки точність $\varepsilon = 0,001$, то в обчисленнях залишаємо чотири цифри після коми).

Використовуючи формулу Сімпсона, отримуємо

$$\int_{0,215}^{1,523} e^{\cos x} dx \approx 2,4755.$$

Перевіримо точність обчислювань за методом Рунге. Позначимо знайдене значення інтегралу з кроком $h = 0,1635$ через I_h . Збільшимо крок інтегрування в два рази і з новим кроком $2h = 0,327$ обчислимо інтеграл (при $n = 4$). Отримаємо значення

$$I_{2h} = \int_{0,215}^{1,523} e^{\cos x} dx \approx 2,4754.$$

За формулою (7.2) маємо:

$$\frac{I_h - I_{2h}}{15} = \frac{2,4755 - 2,4754}{15} = \frac{0,0001}{15} = 0,000007 < \varepsilon.$$

Умова (7.2) виконана. Таким чином, наближене значення інтегралу з точністю 0,001 дорівнює $I_h = 2,476$.

Нагадаємо, що для оцінки точності знайденого значення інтегралу за правилом Рунге необхідно, щоб значення n було кратним чотирьом.

Інтегрування функцій в Python

Підмодуль `scipy.integrate`

<https://docs.scipy.org/doc/scipy/tutorial/integrate.html>

дозволяє приблизно обчислювати значення певних інтегралів.

Функція `scipy.integrate.quad` дозволяє інтегрувати функцію $f: \mathbb{R} \rightarrow \mathbb{R}$. Виклик функції `quad(f, a, b)` приблизно знаходить значення інтеграла

$$\int_a^b f(x) dx.$$

Як приклад знайдемо чисельне значення інтегралу

$$I = \int_0^{\pi} \sin x \, dx = -\cos x \Big|_0^{\pi} = 2.$$

```
import numpy as np
from scipy import integrate
from matplotlib import pyplot as plt

f = np.sin
a, b = 0, np.pi
I, _ = integrate.quad(f, 0, np.pi)
x = np.linspace(0, np.pi, 100)
y = f(x)
print(I)

fig, ax = plt.subplots(figsize=(10, 6), layout="tight")
ax.plot(x, y, label=r"$f(x)=\sin x$")
ax.set_xlabel("$x$")
ax.set_ylabel("$y$")
ax.fill_between(x, np.zeros_like(x), y, where=(y>0), facecolor='green',
alpha=0.30)
ax.set_xticks(np.linspace(0, np.pi, 7))
ax.text(np.pi/2, 0.5, fr"$S = \{I\}$", fontdict={"size": 24, "ha": "center"})
ax.legend()
ax.grid()
```

2.0

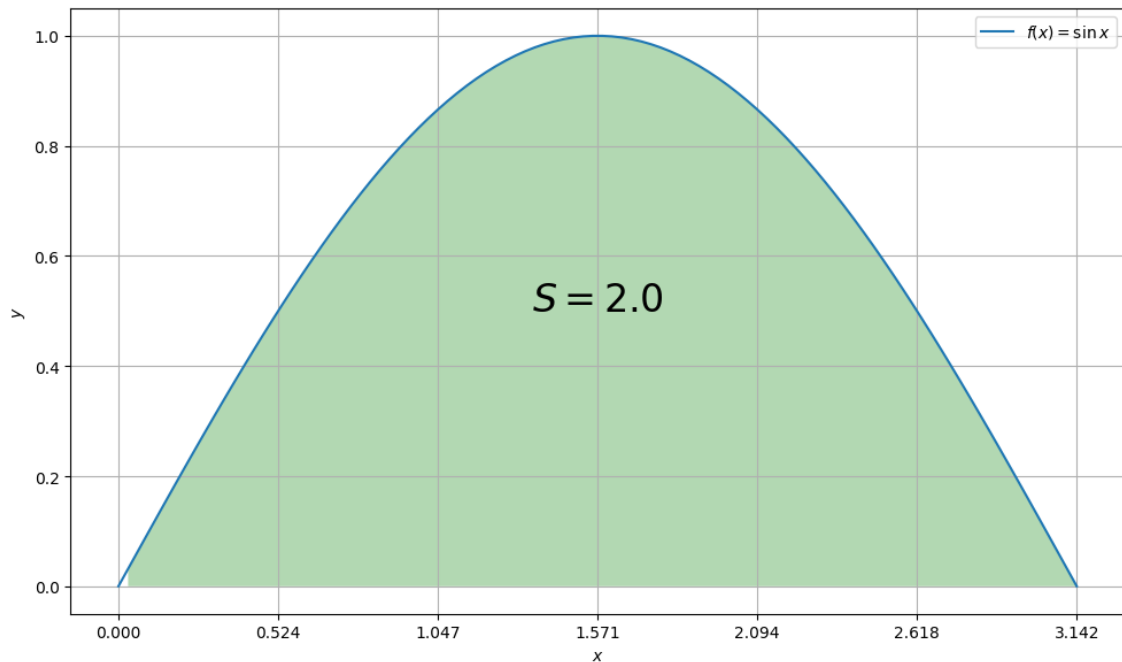


Рис.7.7. Інтегрування функції $f(x) = \sin x$ на інтервалі $[0, \pi]$.

Параметри a і b можуть набувати $-\infty$ і $+\infty$, щоб брати невласні інтеграли. Продемонструємо це на прикладі інтегралу

$$I = \int_1^{\infty} \frac{1}{x^2} dx = 1.$$

```
def f(x):
    return 1. / np.square(x)

a, b = 1, np.inf
l, _ = integrate.quad(f, a, b)
print(l)

x = np.linspace(a, 7, 100)
y = f(x)

fig, ax = plt.subplots(figsize=(10, 6), layout="tight")
ax.plot(x, y, label=r"$f(x)=\dfrac{1}{x^2}$")
ax.set_xlabel("$x$")
ax.set_ylabel("$y$")
ax.fill_between(x, np.zeros_like(x), y, where=(y>0), facecolor='green',
alpha=0.30)
ax.set_xticks(np.linspace(1, 7, 7))
ax.text(1.5, 0.1, fr"$S = {l}$", fontdict={"size": 24, "ha": "center"})
ax.legend()
ax.grid()
```

1.0

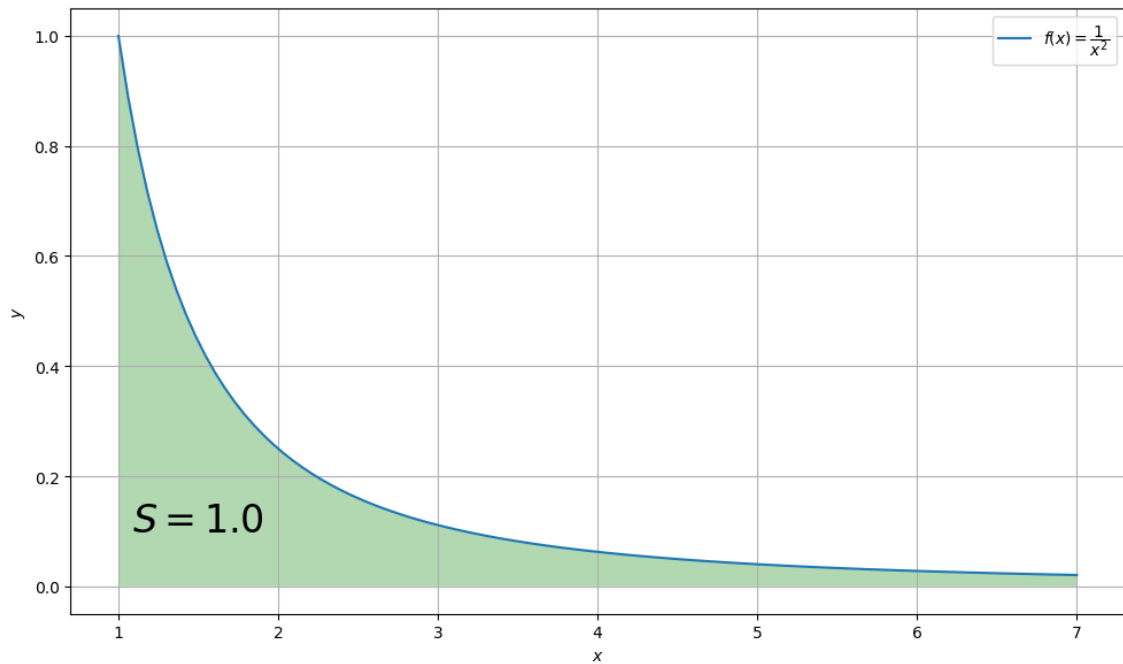


Рис.7.8. Інтегрування функції $f(x) = \frac{1}{x^2}$ на інтервалі $[1, \infty]$.

Проте чисельне взяття невластних інтегралів зовсім не тривіальне завдання. Наприклад, інтеграл Френеля

$$\int_0^{\infty} \sin x^2 dx = \sqrt{\frac{\pi}{8}}$$

просто так порахувати не вдасться.

```
def f(t):
    return np.sin(t**2)

l, _ = integrate.quad(f, 0, np.inf)
print(l)
```

-2395.5825842271347

IntegrationWarning: The integral is probably divergent, or slowly convergent.
l, _ = integrate.quad(f, 0, np.inf)

Індивідуальні завдання за темою “Чисельне інтегрування”

Функція $f(x)$ задана формулою.

№ п/п	Функція $f(x)$	Відрізок інтегрування $[a, b]$
1	$f(x) = \sin \frac{1}{3} \ln(0,1 + x^2)$	$[1; 3]$
2	$f(x) = \sqrt{1 + 0,1x + \ln^2 x}$	$[1; 3]$
3	$f(x) = e^{\operatorname{tg} \frac{x}{3} + 0,1}$	$[0; 2]$
4	$f(x) = \frac{0,1 + \sqrt{x}}{1 + \ln^2 x}$	$[2; 4]$
5	$f(x) = \arcsin \frac{1}{\sqrt{1,1 + x}}$	$[1; 3]$
6	$f(x) = \cos \left(e^{\frac{x}{3}} + 0,1x \right)$	$[0; 2]$
7	$f(x) = e^{\sqrt{1+x+0,1x^2}}$	$[0; 2]$
8	$f(x) = \operatorname{tg} 0,1(x^2 + \sqrt{1+0,1x})$	$[1; 3]$
9	$f(x) = \ln \left(1,1 + \operatorname{tg} \frac{x}{2} \right)$	$[0; 2]$
10	$f(x) = \ln(2,1 + \sin 2x)$	$[0; 2]$
11	$f(x) = \sin \frac{1}{3} \cdot \ln(0,2 + x^2)$	$[1; 3]$
12	$f(x) = \sqrt{1 + 0,2x + \ln^2 x}$	$[1; 3]$
13	$f(x) = e^{\operatorname{tg} \frac{x}{3} + 0,2}$	$[0; 2]$
14	$f(x) = \frac{0,2 + \sqrt{x}}{1 + \ln^2 x}$	$[2; 4]$

Продовження

15	$f(x) = \arcsin \frac{1}{\sqrt{1,2+x}}$	$[1; 3]$
16	$f(x) = \cos \left(e^{\frac{x}{3}} + 0,2x \right)$	$[0; 2]$
17	$f(x) = e^{\sqrt{1+x+0,2x^2}}$	$[0; 2]$
18	$f(x) = \operatorname{tg} 0,1(x^2 + \sqrt{1+0,2x})$	$[1; 3]$
19	$f(x) = \ln \left(1,2 + \operatorname{tg} \frac{x}{2} \right)$	$[0; 2]$
20	$f(x) = \ln(2,2 + \sin 2x)$	$[0; 2]$
21	$f(x) = \sin \frac{1}{3} \cdot \ln(0,5 + x^2)$	$[1; 3]$
22	$f(x) = \sqrt{1 + 0,5x + \ln^2 x}$	$[1; 3]$
23	$f(x) = e^{\operatorname{tg} \frac{x}{3} + 0,5}$	$[0; 2]$
24	$f(x) = \frac{0,5 + \sqrt{x}}{1 + \ln^2 x}$	$[2; 4]$
25	$f(x) = \arcsin \frac{1}{\sqrt{1,5+x}}$	$[1; 3]$
26	$f(x) = \cos \left(e^{\frac{x}{3}} + 0,5x \right)$	$[0; 2]$
27	$f(x) = e^{\sqrt{1+x+0,5x^2}}$	$[0; 2]$
28	$f(x) = \operatorname{tg} 0,1(x^2 + \sqrt{1+0,5x})$	$[1; 3]$
29	$f(x) = \ln \left(1,5 + \operatorname{tg} \frac{x}{2} \right)$	$[0; 2]$
30	$f(x) = \ln(2,5 + \sin 2x)$	$[0; 2]$

Необхідно:

1) обчислити інтеграл

$$I = \int_0^1 \frac{dx}{1+x^2}$$

за методами: “лівих”, “правих”, “середніх” прямокутників.

2) обчислити визначений інтеграл від заданої функції за методами:

а) прямокутників;

б) трапецій;

в) сімпсона (або за формулою “трьох восьмих”).

3) оцінити похибку за правилом Рунге. Зробити це саме, використавши Python (або іншу мову програмування).

4) обчислити функцію різними методами в Python (або в іншій мові програмування).

Контрольні запитання

1. Обчислення інтегралу $\int_a^b f(x)dx$ відповідає визначенню:

а) об'єму будь якої фігури;

б) площі будь якої фігури;

в) об'єму тіла, отриманого обертанням криволінійної трапеції, в якої $x = a$, $x = b$, $y = 0$, $y = f(x)$;

г) площі криволінійної трапеції, обмеженої лініями $x = a$, $x = b$, $y = 0$, $y = f(x)$.

2. Формула чисельного інтегрування методу “лівих” прямокутників має вигляд:

3. Суть методу Сімпсона полягає в тому, що через три послідовні ординати поділу інтервалу проводиться:

а) квадратична парабола;

б) будь яка крива;

в) синусоида;

г) гіпербола.

4. Методи чисельного інтегрування використовуються тоді, коли:

а) неможливо визначити первісну $F(x)$;

б) неможливо визначити похідну $f'(x)$;

в) невідомий інтервал інтегрування $[a, b]$;

г) функція $y = f(x)$ задана графічно.

5. Найбільш приблизним методом чисельного інтегрування є метод:

- a) прямокутників;
- b) трапецій;
- c) парабол;
- d) Сімпсона.

6. Необхідна умова використання формул Сімпсона є: число точок поділу повино бути:

- a) парним числом;
- b) цілим числом;
- c) непарним числом;
- d) кратним "4".

7. Відомо, що функція описується поліномом другого ступеня (квадратним рівнянням). Вкажіть метод, який дозволить обчислити визначений інтеграл з мінімальною похибкою:

- a) метод Сімпсона;
- b) метод трапецій;
- c) метод "лівих" прямокутників;
- d) метод "середніх" прямокутників.

Лабораторна робота № 8

Розв'язання звичайних диференціальних рівнянь першого порядку з застосуванням методу Ейлера

Мета роботи: Ознайомлення з розв'язанням диференціальних рівнянь методом Ейлера.

Теоретичні відомості

Методи наближеного розв'язання диференціальних рівнянь, які в залежності від форми отримання рішення розподіляються на дві групи:

- 1) аналітичні методи, які дають наближене рішення диференціального рівняння у вигляді аналітичного виразу;
- 2) чисельні методи, які дають наближене рішення у вигляді таблиці.

Розв'язання звичайних диференціальних рівнянь першого порядку з застосуванням чисельних методів виконуються за:

- а) методом Ейлера;
- б) методом Рунге–Кутта.

Взагалі рішенням (або інтегралом) диференціального рівняння називається будь-яка диференційована функція $y = \varphi(x)$, що задовольняє цьому рівнянню, тобто така, після підстановки якої в рівняння воно обертається в тотожність

Але розв'язати диференціальне рівняння $y' = f(x, y)$ чисельним методом – це означає, що необхідно, не з'ясовуючи функцію $y = f(x)$, отримати таблицю значень цієї функції для заданої послідовності значень аргументу. Величина $h = x_{k+1} - x_k$ називається кроком інтегрування.

Розглянемо один з методів розв'язання диференціальних рівнянь – метод Ейлера.

Метод Ейлера

Цей метод є порівнянно неточним і використовується переважно для наближених розрахунків, а також для визначення точок входу для інших методів.

Нехай дано диференціальне рівняння першого порядку $y' = f(x, y)$ з початковою умовою $x = x_0$, $y(x_0) = y_0$. Необхідно знайти рішення цього рівняння на інтервалі $[a, b]$.

Розіб'ємо інтервал на k рівних частин і отримаємо послідовність $x_0, x_1, x_2, \dots, x_k$, де $x_i = x_0 + k \cdot i$ ($i = 1, 2, 3, \dots, k$). $h = (b - a)/n$ – крок інтегрування.

Значення функції у відповідних точках позначимо $y_0, y_1, y_2, \dots, y_k$.

Для розв'язання рівняння слід скористатися формулою:

$$y_{k+1} = y_k + h f(x_k, y_k)$$

Тоді,

$$y_1 = y_0 + h f(x_0, y_0)$$

$$y_2 = y_1 + h f(x_1, y_1)$$

.....

$$y_{k+1} = y_k + h f(x_k, y_k)$$

Продовжуючи цей процес та кожного разу приймаючи підінтегральну функцію на відповідній ділянці постійною та рівною її значенню на початку кожної ділянки, отримаємо таблицю рішень диференціального рівняння на заданному інтервалі $[a, b]$. Отримана формула означає, що на інтервалі $[x_k, x_{k+1}]$ інтегральна крива наближено замінюється прямолінійним відрізком, що виходить із точки з координатами (x_k, y_k) з кутовим коефіцієнтом $f(x_k, y_k)$. Як наближену інтегральну криву отримуємо ломану лінію (верхня лінія), яка показує збільшення відхилення від теоретичної кривої (нижня лінія), тобто збільшення погрішності впродовж обчислень.

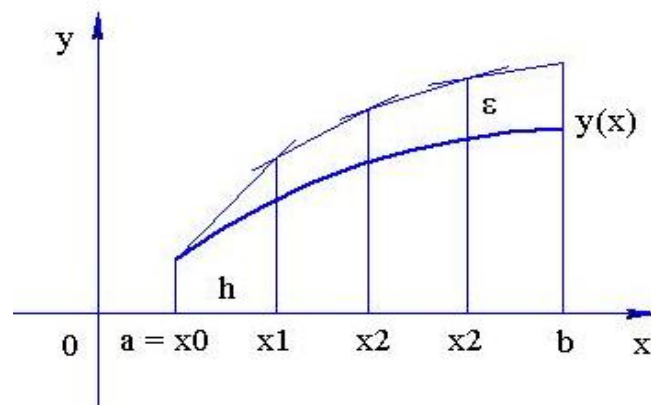


Рис. 8.1. Геометрична інтерпретація метода Ейлера

Похибка обчислювань $R_k = 0,5h^2 \cdot y''(\epsilon)$, $x_k \leq \epsilon \leq x_{k+1}$.

Приклад. Розв'язати рівняння $y' = y + x^2$ при початкових умовах: $y_0 = y(0) = 1$; $0 \leq x \leq 0,5$. Прийняти $h = 0,1$.

Розв'язання. Для розв'язання рівняння скористаємося формулою

$$y_{k+1} = y_k + h \cdot f(x_k, y_k), \text{ де } f(x_k, y_k) = y_k + x_k^2.$$

$$\text{Тоді отримаємо вираз } y_{k+1} = y_k + h \cdot (y_k + x_k^2).$$

Підставляючи відповідні значення $x_0 = 0$, $x_1 = 0,1$, $x_2 = 0,2$, $x_3 = 0,3$, $x_4 = 0,4$, $x_5 = 0,5$, а також початкові умови y_0 та обчислені значення y_k в рівняння, отримаємо:

$$y_1 = y_0 + h \cdot (y_0 + x_0^2) = 1 + 0,1 \cdot (1 + 0) = 1,1;$$

$$y_2 = y_1 + h \cdot (y_1 + x_1^2) = 1,1 + 0,1 \cdot (1 + 0,1^2) = 1,211;$$

$$y_3 = y_2 + h \cdot (y_2 + x_2^2) = 1,211 + 0,1 \cdot (1 + 0,2^2) = 1,3361;$$

$$y_4 = y_3 + h \cdot (y_3 + x_3^2) = 1,3361 + 0,1 \cdot (1 + 0,3^2) = 1,4787;$$

$$y_5 = y_4 + h \cdot (y_4 + x_4^2) = 1,4787 + 0,1 \cdot (1 + 0,4^2) = 1,6426.$$

Результати обчислень запишемо у вигляді таблиці і побудуємо графік (рис. 8.2).

K	x_k	y_k
0	0	1
1	0,1	1,1
2	0,2	1,211
3	0,3	1,3361
4	0,4	1,4787
5	0,5	1,6426

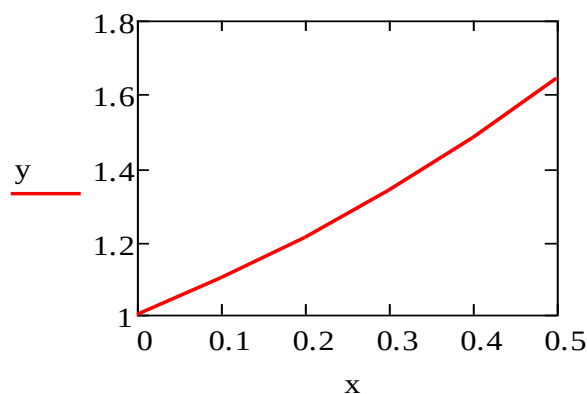


Рис. 8.2. Графік функції $y = f(x)$

Індивідуальні завдання за темою “Розв’язання звичайних диференціальних рівнянь першого порядку з застосуванням методу Ейлера”

Задано рівняння.

№ п/п	Рівняння	Початкові умови
1	$y' = 2y + 3x + 1$	$y(0)=0, \ 0 \leq x \leq 1$
2	$y' = x - 2y$	$y(0)=0, \ 0 \leq x \leq 1$
3	$y' = y^2 + x$	$y(0)=1, \ 0 \leq x \leq 1$
4	$y' = y^2 + x^2$	$y(0)=1, \ 0 \leq x \leq 1$
5	$y' = yx^2 + x^3$	$y(0)=1, \ 0 \leq x \leq 1$
6	$y' = x - y$	$y(0)=-1, \ 0 \leq x \leq 1$
7	$y' = x^2 - y$	$y(0)=2, \ 0 \leq x \leq 1$
8	$y' = y + x + 1$	$y(0)=1, \ 0 \leq x \leq 1$
9	$y' = y - 2x$	$y(0)=0, \ 0 \leq x \leq 1$
10	$y' = x - y + 2$	$y(0)=0, \ 1 \leq x \leq 2$
11	$y' = y - 3x$	$y(0)=1, \ 0 \leq x \leq 1$
12	$y' = y - 4x$	$y(0)=1, \ 0 \leq x \leq 1$
13	$y' = y + x + 2$	$y(0)=0, \ 0 \leq x \leq 1$
14	$y' = 2y + 2x + 1$	$y(0)=0, \ 0 \leq x \leq 1$
15	$y' = 3y + 2x + 1$	$y(0)=0, \ 0 \leq x \leq 1$
16	$y' = 2y + 3x + 1$	$y(0)=0, \ 1 \leq x \leq 2$
17	$y' = x - 2y$	$y(0)=0, \ 1 \leq x \leq 2$
18	$y' = x - 2y$	$y(0)=0, \ 1 \leq x \leq 2$

Продовження

19	$y' = yx^2 + x^3$	$y(0)=1, 1 \leq x \leq 2$
20	$y' = x - y$	$y(0)=-2, 0 \leq x \leq 1$
21	$y' = x^2 - y$	$y(0)=2, 1 \leq x \leq 2$
22	$y' = y + x + 1$	$y(0)=1, 1 \leq x \leq 2$
23	$y' = y - 2x$	$y(0)=0, 1 \leq x \leq 2$
24	$y' = x - y + 2$	$y(0)=0, 0 \leq x \leq 12$

Необхідно:

- 1) розв'язати рівняння методом Ейлера, прийнявши $h = 0,1$, побудувати графік.
- 2) розв'язати рівняння відповідним методом в Python (або в іншій мові програмування).

Контрольні запитання

- 1 В чому суть метода Ейлера?
- 2 Яка геометрична інтерпретація метода Ейлера?
- 3 Яка точність метода Ейлера?
4. Яка послідовність розв'язання рівняння за методом Ейлера?
5. Коли слід використовувати метод Ейлера?

Лабораторна робота № 9

Розв'язання звичайних диференціальних рівнянь першого порядку з застосуванням методу Рунге–Кутта

Мета роботи: Ознайомлення з розв'язанням рівнянь методом Рунге–Кутта.

Теоретичні відомості

Метод Рунге–Кутта четвертого порядку

В обчислювальній практиці найбільш часто використовується метод Рунге–Кутта четвертого порядку. Цей метод може бути застосован до звичайних диференціальних рівнянь вищого порядку, а також до систем диференціальних рівнянь першого порядку. Він реалізований в різних математических пакетах (Maple, MathCAD, Maxima).

Суть методу полягає в послідовному обчисленні приблизних значень рішення задачі Коши за формулами:

$$y_{i+1} = y_i + h (k_1 + 2 \cdot k_2 + 2 \cdot k_3 + k_4)/6, \text{ де } i = 1, 2, 3, \dots, k$$

В цій формулі

$$\begin{aligned}k_1 &= f(x_i, y_i) \\k_2 &= f(x_i + h/2, y_i + h \cdot k_1/2) \\k_3 &= f(x_i + h/2, y_i + h \cdot k_2/2) \\k_4 &= f(x_i + h, y_i + h \cdot k_3)\end{aligned}$$

А також

y_0 – початкова умова, h – крок, $x_i = x_0 + i \cdot h$

Приклад. Розв'язати рівняння $y' = y + x^2$ при початкових умовах: $y_0 = y(0) = 1$; $0 \leq x \leq 0,5$. Прийняти $h = 0,1$.

Розв'язання. В нашому випадку $x_0 = 0$, $x_1 = 0,1$, $x_2 = 0,2$, $x_3 = 0,3$, $x_4 = 0,4$, $x_5 = 0,5$.

На першому кроці, підставляючи $x_0 = 0$ та $y_0 = 1$ в відповідні рівняння, отримуємо:

$$k_1 = f(x_0, y_0) = y_0 + x_0^2 = 1 + 0 = 1.$$

$$k_2 = f(x_0 + h/2, y_0 + h \cdot k_1/2) = y_0 + k_1/2 + (x_0 + h/2)^2 = 1 + 0,05 + (0 + 0,05)^2 = 1,0525.$$

$$k_3 = f(x_0 + h/2, y_0 + h \cdot k_2/2) = y_0 + k_2/2 + (x_0 + h/2)^2 = 1 + 1,0525/2 + (0 + 0,05)^2 = 1,0551.$$

$$k_4 = f(x_0 + h/2, y_0 + h \cdot k_3) = y_0 + k_3/2 + (x_0 + h/2)^2 = 1 + 1,0551 + (0 + 0,05)^2 = 1,1155.$$

Тоді,

$$y_1 = y_0 + h (k_1 + 2 \cdot k_2 + 2 \cdot k_3 + k_4)/6 = 1 + 0,1 (1 + 2 \cdot 1,0525 + 2 \cdot 1,0551 + 1,1155)/6 = 1,1055117.$$

На другому кроці, підставляючи $x_1 = 0,1$ та $y_1 = 1,1055117$ в відповідні рівняння, отримуємо значення y_2 , і так далі по всіх кроках до $x_5 = 0,5$ (всі розрахунки не наводимо).

Результати обчислень запишемо у вигляді таблиці.

k	x_i	y_i
0	0	1
1	0,1	1,10551
2	0,2	1,22421
3	0,3	1,35958
4	0,4	1,51548
5	0,5	1,69617

Графік буде мати майже такий самий вигляд, як і при розв'язанні рівняння із застосуванням методу Ейлера.

Приклад розв'язання диференціального рівняння в Python

Підмодуль `scipy.integrate`

(<https://docs.scipy.org/doc/scipy/reference/integrate.html#module-scipy.integrate>)

містить у собі безліч методів розв'язання систем звичайних диференціальних рівнянь (ОДП, ordinary differential equation, ODE).

Так як скрізь нижче в цьому розділі будуть будуватися графіки отриманого та точного розв'язків рівняння, визначимо функцію для побудови графіків цих розв'язків.

```
from matplotlib import pyplot as plt

def plot_solution(ax, x, exact_sol, sol_x, sol_y, xlabel="$t$"):
    ax.plot(x, exact_sol(x), color="green", label="Точний розв'язок")
    ax.scatter(sol_x, sol_y, color="red", marker="o", label="Наближений розв'язок")
    ax.set_xlabel(xlabel)
    ax.set_ylabel("$y$")
    ax.legend()
```

Функція `scipy.integrate.solve_ivp` (solve initial value problem) дозволяє чисельно розв'язувати задачу Коші виду

$$\begin{cases} \frac{dy}{dt} = f(t, y), \\ y(t_0) = y_0. \end{cases}$$

В якості найпростішого прикладу розв'яжемо рівняння

$$\begin{cases} y' = y, \\ y(0) = 1, \end{cases}$$

точний розв'язок якого $y = e^t$.

Щоб чисельно розв'язати це рівняння засобами SciPy, необхідно визначити функцію правої частини ($f(t, y)$ в прикладі коду нижче), задати початкове значення $y(0)$ у вигляді списку з одного елемента (y_0 в прикладі коду нижче), а також задати інтервал незалежної змінної, де необхідно розв'язати диференціальне рівняння.

```
import numpy as np
from scipy import integrate
from matplotlib import pyplot as plt

def f(t, y):
    return y
```

```
def exact_solution(t):
    return np.exp(t)

y_0 = [1]
t_0 = 0
t_final = 3
solution = integrate.solve_ivp(f, (t_0, t_final), y_0)
print(solution)
```

```
message: The solver successfully reached the end of the integration interval.
success: True
status: 0
      t: [ 0.000e+00  1.000e-01  1.066e+00  2.304e+00  3.000e+00]
      y: [[ 1.000e+00  1.105e+00  2.904e+00  1.002e+01  2.009e+01]]
      sol: None
t_events: None
y_events: None
      nfev: 26
      njev: 0
      nlu: 0
```

Метод повертає структуру з порівняно великою кількістю полів, з яких можна отримати налагоджувальну інформацію про збіжність алгоритму, а також отримати наближену оцінку $\tilde{y}(t)$ в наборі точок $\{t_i, i = 1, \dots, n\}$.

```
fig, ax = plt.subplots(figsize=(8, 8), layout="tight")
x = np.linspace(t_0, t_final, 100)
plot_solution(ax, x, exact_solution, solution.t, solution.y[0])
```

Хоча права частина рівняння i не залежить явно від t , функція $f(t, x)$ все одно оголошується з першим параметром t .

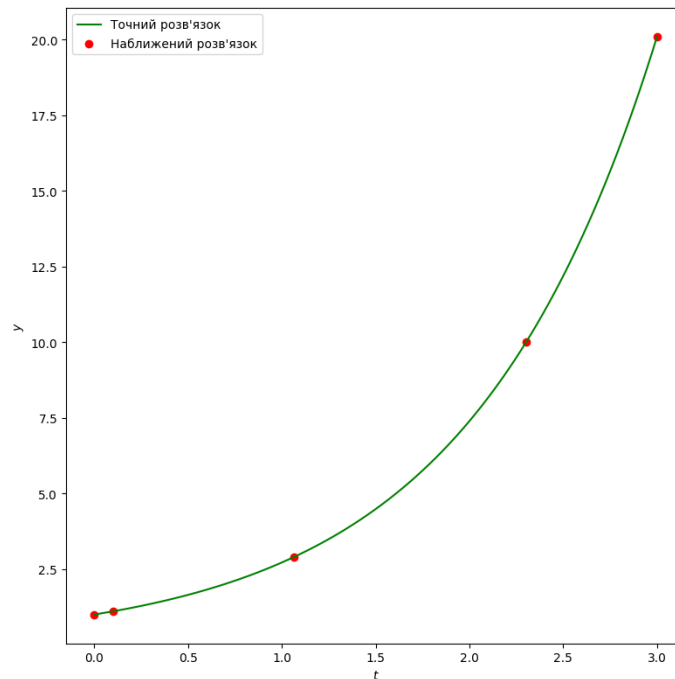


Рис.9.1. Точний та наближений розв'язки

Для цього рівняння функція `scipy.integrate.solve_ivp` видала розв'язок, що містить зовсім невелику кількість точок, оскільки метод швидко зійшовся. Параметром `t_eval` можна явно задати сітку, у вузлах якої потрібно отримати оцінку точного розв'язку.

```
solution = integrate.solve_ivp(f, (t_0, t_final), y_0, t_eval=np.linspace(t_0, t_final, 50))
fig, ax = plt.subplots(figsize=(8, 8), layout="tight")
plot_solution(ax, x, exact_solution, solution.t, solution.y[0])
```

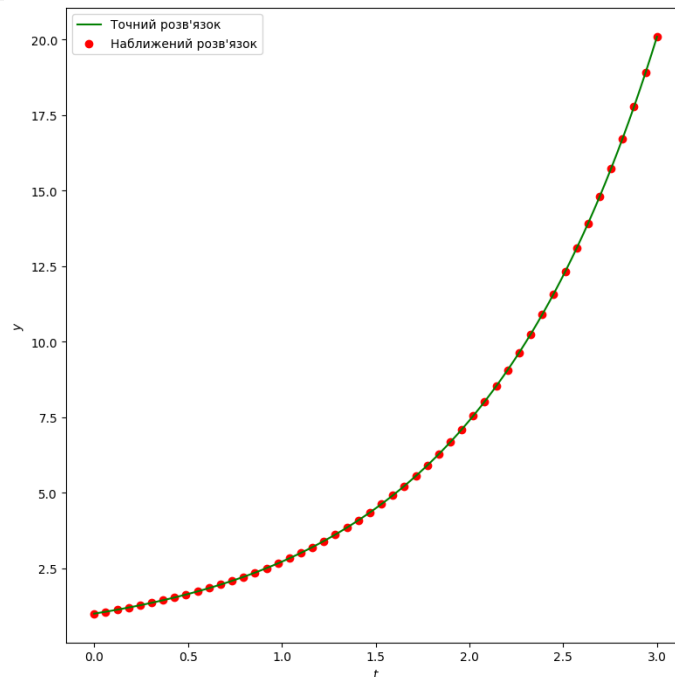


Рис.9.2. Збільшення кількості вузлів

**Індивідуальні завдання за темою “Розв’язання звичайних
диференціальних рівнянь першого порядку з застосуванням
методу Рунге–Кутта”**

Необхідно:

- 1) розв’язати рівняння із попередньої роботи методом Рунге–Кутта, прийнявши $h = 0,1$, побудувати графік.
- 2) розв’язати рівняння відповідним методом за допомогою Python (або іншої мови програмування).

Контрольні запитання

- 1 У чому суть методу Рунге–Кутта?
- 2 Яка геометрична інтерпретація методу Рунге–Кутта?
- 3 Яка точність методу Рунге–Кутта?
4. Яка послідовність розв’язання рівняння за методом Рунге–Кутта?
5. Коли слід використовувати метод Рунге–Кутта?

НАВЧАЛЬНО–МЕТОДИЧНА ЛІТЕРАТУРА

1. Чисельні методи: теорія і практика. Навч. посіб. / А. Л. Литвинов; Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. – Харків: ХНУМГ ім. О. М. Бекетова, 2022. – 166 с.
URL: https://eprints.kname.edu.ua/62005/1/2022_печ.13Н_Чисельн_Методи_Литвинов.pdf
2. Чисельні методи в комп'ютерних науках: навч. посіб. / Андруник В.А., Висоцька В.А., Пасічник В.В., Чирун Л.Б., Чирун Л.В. – Львів: Видавництво «Новий світ – 2000», 2020. – 470 с.
URL: <https://ns2000.com.ua/wp-content/uploads/2019/07/CHysel-ni-metody-1-t..pdf>
3. Чисельні методи: навчальний посібник / Голубєва К.М., Кашпур О.Ф., Ключин Д.А. – Київ: КНУ ім. Тараса Шевченка, 2022. – 145 с.
URL: http://csc.knu.ua/media/filer_public/43/bc/43bc9641-ef96-441f-8e5f-a5f07e2d6805/posibnik_chm_dlia_sa.pdf
4. Чисельні методи: Навчальний посібник. / Волонтир Л.О, Зелінська О.В., Потапова Н.А., Чіков І.А., Вінницький національний аграрний університет. – Вінниця: ВНАУ, 2020 – 322 с.
URL: <http://repository.vsau.org/getfile.php/27703.pdf>
5. Чисельні методи розв'язання прикладних задач : навч. посіб. / О.А. Гончаров, Л.В. Васильєва, А.М. Юнда. – Суми : Сумський державний університет, 2020. – 142 с.
URL: <https://essuir.sumdu.edu.ua/server/api/core/bitstreams/0f8e860a-c101-4db8-9e54-c823abb78b76/content>
6. Чисельні методи комп'ютерного аналізу: навч. посіб. / В.Я. Воропаєва, І.К. Локтіонов, Л.П. Мироненко та ін.; за заг. ред. В.В. Турупалова. – Львів: «Магнолія 2006», 2025. – 224 с.
URL: https://magnolia.lviv.ua/wp-content/uploads/2025/01/CHyselni-metody-kompiuternoho-analizu_zmist.pdf
7. Johan Larsson. Introduction to Engineering Numerical Methods. – Dep. of Mechanical Engineering University of Maryland, 2024. – 86 p.
URL: <https://larsson.umd.edu/wp-content/uploads/sites/18/2023/01/engineeringNumericalMethods.pdf>
8. Python Programming And Numerical Methods: A Guide For Engineers And Scientists, 2020.
URL: <https://pythonnumericalmethods.studentorg.berkeley.edu/notebooks/Index.html>
9. Numerical Methods in Scientific Computing / J. van Kan, A. Segal, F. Vermolen. – Delft University of Technology, 2023. – 290 p.
URL: <https://books.open.tudelft.nl/home/catalog/view/169/297/544>