

Варто зазначити, що SciPy містить ще функцію *solve_ivp*, яка є більш новим інтерфейсом. Для кодів, які використовують останню версію бібліотеки SciPy, рекомендується використовувати *solve_ivp*, оскільки вона надає ширшу базу для розв'язання диференціальних рівнянь та їх систем. *Solve_ivp* підтримує декілька чисельних методів розв'язання диференціальних рівнянь (методи Рунге–Кутта, BDF-метод тощо) і надає можливість більш гнучкого налаштування цього процесу. До того ж є можливість вручну обрати метод інтегрування, наприклад, RK23, RK45, BDF, LSODA тощо, водночас *odeint* містить адаптивний метод LSODA. Для детальнішого ознайомлення з наведеними функціями та їх аргументами можна звернутися до [1] і [3].

Отже, розв'язання диференціальних рівнянь за допомогою Python є не тільки ефективним, але й швидким за допомогою можливостей, які надають бібліотеки SciPy та Matplotlib.

Список використаних джерел

1. Integration and ODEs (scipy.integrate) – SciPy v1.14.1 Manual. *Numpy and Scipy Documentation – Numpy and Scipy documentation*. URL: <https://docs.scipy.org/doc/scipy/reference/integrate.html> (date of access: 01.12.2024).
2. Solve Differential Equations with ODEINT. *APMonitor Optimization Suite*. URL: <https://apmonitor.com/pdc/index.php/Main/SolveDifferentialEquations> (date of access: 01.12.2024).
3. Müller-Komorowska D. Differential Equations with SciPy – odeint or solve_ivp. *Simulation-Based*. URL: https://simulationbased.com/2021/02/16/differential-equations-with-sciPy-odeint-or-solve_ivp/ (date of access: 01.12.2024).

УДК 519.22/.25:004.415]:332.624(043.2)

*Чемес В. С., здобувач вищої освіти,
Хмелівський Ю. С., асистент кафедри
інформаційних технологій*

ЗАСТОСУВАННЯ ЛІНІЙНОЇ РЕГРЕСІЇ ДЛЯ ОЦІНКИ ЦІНИ НЕРУХОМОСТІ ЗА РІЗНИМИ ПАРАМЕТРАМИ

Донецький національний університет імені Василя Стуса, м. Вінниця

Ринок нерухомості є однією з найбільш активних і важливих частин економіки. Визначення справедливої ціни об'єкта є однією з головних проблем для інвесторів, покупців і продавців нерухомості. Ціна нерухомості залежить від багатьох факторів, зокрема місцезнаходження, площі, стану будівлі, кількості кімнат, наявності додаткових зручностей тощо. Отже, передбачення ціни є складним завданням, і щоб отримати точну оцінку, необхідно використовувати математичні моделі.

Лінійна регресія є одним із найбільш поширених методів вирішення цієї проблеми. Лінійна регресія дає змогу оцінити вплив різних елементів на ціну нерухомості та створити модель, яка може прогнозувати ціну на основі значень цих елементів. Це особливо важливо на ринку нерухомості, оскільки зміна одного фактора може значно вплинути на кінцеву вартість нерухомості.

Однак оцінка ціни нерухомості залежить від кількох факторів, і застосування лінійної регресії з однією змінною не є доцільним, оскільки це є недостатнім для точного прогнозування. Тому для більш точних і реалістичних результатів доцільно використовувати множинну лінійну регресію, яка допомагає враховувати кілька незалежних змінних одночасно.

У такій моделі залежна змінна (ціна нерухомості) оцінюється на основі комбінації кількох факторів, що значно підвищує точність прогнозів та дає змогу врахувати складні взаємозв'язки між різними характеристиками нерухомості [1].

Множинна лінійна регресія є розширенням лінійної регресії, де для прогнозування залежної змінної використовується більше ніж одна незалежна змінна. Це один із основних методів статистичного навчання, який допомагає оцінити вплив кількох факторів на результат і побудувати математичну модель для прогнозування значення залежної змінної.

Множинна лінійна регресія дає змогу створювати рівняння, яке описує лінійний зв'язок між кількома незалежними змінними (предикторами) та залежною змінною. Математичне представлення моделі множинної регресії виглядає так:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n + e,$$

де y – залежна змінна (наприклад, ціна нерухомості);

$x_1, x_2, \dots, x_n$ – незалежні змінні (наприклад, площа, кількість кімнат, район та ін.);

β_0 – вільний член (інтерцепт);

$\beta_1, \beta_2, \dots, \beta_n$ – коефіцієнти регресії, які вказують на вплив кожної незалежної змінної на залежну;

e – випадкова похибка (залишкове значення) [1, 2].

Мета множинної лінійної регресії – знайти такі значення коефіцієнтів $\beta_1, \beta_2, \dots, \beta_n$, які мінімізують похибку моделі. Зазвичай для цього використовують метод найменших квадратів, який мінімізує суму квадратів різниць між прогнозованими і реальними значеннями залежної змінної. Процес виглядає так:

$$\min \sum_{i=1}^n (y_i - \hat{y}_i)^2,$$

де y_i – реальні значення залежної змінної;

$\hat{y}_i$ – прогнозовані значення залежної змінної на основі моделі.

Мета дослідження – побудувати модель множинної лінійної регресії для прогнозування ціни нерухомості (y) на основі різних характеристик квартири. У якості незалежних змінних обрано площу квартири (x_1), кількість кімнат (x_2), відстань до центру міста (x_3), наявність паркінгу (x_4) [4].

Необхідно знайти коефіцієнти моделі, які допомагають найкраще описати вплив зазначених факторів на ціну квартири.

Для побудови моделі використовується вибірка даних про квартири (табл. 1).

Таблиця 1 – Вибірка даних про квартири

Площа, м ²	Кількість кімнат	Відстань до центру, км	Паркінг	Ціна, \$
50	2	5	0	100 000
70	3	3	1	150 000
45	1	8	0	80 000
120	4	2	1	250 000
90	3	4	1	200 000

З огляду на умову завдання множинна лінійна регресія описується рівнянням:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + e.$$

Для оцінки параметрів регресії застосовуємо метод найменших квадратів. Модель у матричній формі:

$$y = X\beta + e,$$

де y – вектор залежної змінної;

X – матриця незалежних змінних;

β – вектор коефіцієнтів;

e – вектор похибок.

Для оцінки β використовується формула:

$$\beta = (X^T X)^{-1} X^T y [3].$$

Використовуючи дані таблиці, формуємо матриці:

$$X = \begin{bmatrix} 1 & 50 & 2 & 5 & 0 \\ 1 & 70 & 3 & 3 & 1 \\ 1 & 45 & 1 & 8 & 0 \\ 1 & 120 & 4 & 2 & 1 \\ 1 & 90 & 3 & 4 & 1 \end{bmatrix}, \quad y = \begin{bmatrix} 100\,000 \\ 150\,000 \\ 80\,000 \\ 250\,000 \\ 200\,000 \end{bmatrix}$$

Обчисливши матриці, отримуємо:

- $\beta_0 = -1\,210\,000$ – базова ціна;
- $\beta_1 = -4\,000$ – кожен додатковий м² зменшує ціну на \$4 000;
- $\beta_2 = 430\,000$ – кожна додаткова кімната додає \$430 000 до ціни;
- $\beta_3 = 130\,000$ – кожен кілометр віддаленості збільшує ціну на \$130 000;
- $\beta_4 = -40\,000$ – наявність паркінгу зменшує ціну на \$40 000.

Після визначення коефіцієнтів регресії отримано математичну модель, яка описує залежність ціни нерухомості від основних характеристик квартири:

$$y = -1\,210\,000 - 4\,000x_1 + 430\,000x_2 + 130\,000x_3 - 40\,000x_4.$$

Отримана модель дає змогу кількісно оцінити вплив основних факторів на формування ціни нерухомості, як-от площа, кількість кімнат, віддаленість від центру та наявність паркінгу. Вона також забезпечує можливість прогнозувати вартість житла на основі цих параметрів, що є корисним для покупців, продавців та ріелторів.

Висновки. У цій роботі було розглянуто застосування множинної лінійної регресії для прогнозування ціни нерухомості, що допомагає враховувати вплив кількох факторів на результат. Множинна лінійна регресія є потужним інструментом статистичного навчання, який дає змогу оцінити взаємозв'язок між кількома незалежними змінними та залежною змінною.

Прогнозування за допомогою цієї моделі є корисним інструментом для покупців, продавців і ріелторів, оскільки допомагає приймати обґрунтовані рішення. До того ж модель має потенціал для подальших удосконалень, зокрема додавання нових факторів, що можуть впливати на ціни на ринку нерухомості.

Список використаних джерел

1. Kanade V. What Is Linear Regression? Types, Equation, Examples, and Best Practices for 2022. Spiceworks. 2023. URL: <https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-linear-regression/> (дата звернення 30.11.2024).
2. Geeks for Geeks. Linear Regression in Machine learning. 2023. URL: <https://www.geeksforgeeks.org/ml-linear-regression/> (дата звернення 30.11.2024).
3. Wikipedia. General linear model. URL: https://en.wikipedia.org/wiki/General_linear_model (дата звернення 30.11.2024).
4. Yasser M. Housing Prices Dataset. Kaggle. 2022. URL: <https://www.kaggle.com/datasets/yasserh/housing-prices-dataset> (дата звернення 30.11.2024).

УДК 004.422.5

*Черніхевич О. В., здобувач вищої освіти,
Поремський Ю. В., канд. техн. наук,
старший викладач кафедри інформаційних
технологій*

АЛГОРИТМИ ДИНАМІЧНОГО ПРОГРАМУВАННЯ В ЗАДАЧАХ ОПТИМАЛЬНОГО ВИБОРУ

Донецький національний університет імені Василя Стуса, м. Вінниця

Динамічне програмування (Dynamic Programming, DP) – це метод оптимізації, що використовується для вирішення складних задач шляхом розбиття їх на простіші підзадачі. Динамічне програмування ефективно використовується для задач, які можна розділити на перекриваючі підзадачі з оптимальною структурою підзадач.

1. Мемоізація (зверху вниз): рекурсивний підхід, за якого результати підзадач зберігаються в пам'яті для уникнення повторних обчислень.
2. Табуляція (знизу вгору): ітеративний підхід, за якого результати підзадач обчислюються і зберігаються в таблиці (зазвичай масиви) і використовуються для обчислення кінцевого результату [1].

Метод динамічного програмування (ДП) є одним з основних інструментів оптимізації та розв'язання задач в інформатиці, економіці, біології та інших галузях. Інакше кажучи, динамічне програмування – це метод розв'язання задач, що ґрунтується на розбитті складної задачі на безліч дрібніших.