

РОЛЬ ПРОГРАМУВАННЯ У ВИЩІЙ МАТЕМАТИЦІ

Вінницький національний технічний університет

Анотація

У роботі розглянуто можливості застосування мов програмування, зокрема Python, MATLAB та C++, для розв'язання задач математичного аналізу, лінійної алгебри, диференціальних рівнянь і теорії ймовірностей. Проаналізовано освітні переваги використання алгоритмічних підходів та чисельних методів, а також потенційні проблеми, пов'язані з надмірною автоматизацією та недостатнім розумінням математичної суті алгоритмів. Особливу увагу приділено ролі програмування у формуванні аналітичного мислення студента та розвитку навичок математичного моделювання.

Ключові слова: програмування, чисельні методи, математичне моделювання, вища математика, алгоритми.

Abstract

This paper examines the use of programming languages, including Python, MATLAB, and C++, for solving problems in mathematical analysis, linear algebra, differential equations, and probability theory. The educational benefits of algorithmic approaches and numerical methods, as well as potential issues related to excessive automation and insufficient understanding of mathematical concepts, are analyzed. Particular attention is paid to the role of programming in fostering students' analytical thinking and developing skills in mathematical modeling.

Keywords: programming, numerical methods, mathematical modeling, higher mathematics, algorithms.

Вступ

У ХХІ столітті вища математика остаточно перетворилася на обчислювальну дисципліну. Якщо раніше теоретичні результати отримували переважно аналітично, то сьогодні більшість практичних задач розв'язується за допомогою комп'ютерних алгоритмів. Програмування стало не просто допоміжним інструментом, а повноцінною мовою сучасної математики, що дозволяє перевіряти гіпотези, проводити чисельні експерименти, візуалізувати абстрактні об'єкти та розв'язувати задачі, які не мають аналітичного розв'язку. У технічних університетах програмування вже інтегровано у курси лінійної алгебри, математичного аналізу, теорії ймовірностей, досліджень операцій та чисельних методів.

Результати дослідження

Математичне програмування є ключовим розділом прикладної математики, що вивчає методи знаходження екстремумів функцій на множинах, заданих системами рівностей та нерівностей [1]. Воно охоплює лінійне, нелінійне, цілочисельне, стохастичне та динамічне програмування. Сучасні задачі цього типу розв'язуються програмно за допомогою спеціалізованих бібліотек, таких як SciPy, PuLP, CVXPY у Python, а також Gurobi, CPLEX чи вбудованих оптимізаційних інструментів MATLAB та Mathematica [2]. Такий підхід дозволяє ефективно обробляти великі обсяги даних і складні обмеження, що виникають у реальних економічних, інженерних та наукових моделях.

Чисельні методи становлять основу обчислювальної математики і застосовуються для розв'язання задач інтегрування, диференціальних рівнянь, наближення функцій та нелінійних рівнянь, де аналітичні розв'язки неможливі або надто складні. Програмна реалізація цих методів здійснюється через бібліотеки на кшталт NumPy та SciPy у Python, Symbolic Math Toolbox у MATLAB, DifferentialEquations.jl у Julia чи NDSolve у Mathematica [3]. Завдяки цим інструментам складні обчислення, які вручну займали б значний час, виконуються миттєво, що робить їх невід'ємною частиною навчального та дослідницького процесів.

Символьні обчислення дозволяють виконувати аналітичні перетворення, такі як диференціювання, інтегрування, спрощення виразів, розклад у ряди та точне розв'язання рівнянь. Системи комп'ютерної алгебри, зокрема Mathematica, SymPy (безкоштовна бібліотека для Python) та Maxima, інтегруються з програмним кодом і полегшують роботу з абстрактними математичними об'єктами [4]. Це особливо

корисно для студентів, які вивчають вищу алгебру чи математичний аналіз, оскільки поєднує теоретичні знання з практичною верифікацією.

Візуалізація математичних об'єктів є важливим аспектом програмування у вищій математиці, адже дозволяє представляти графіки функцій, параметричні криві, поверхні, векторні поля та фрактали. Інструменти на кшталт Matplotlib, Plotly, Mayavi у Python, вбудованих функцій плоту в MATLAB чи 3D-графіки в Mathematica роблять абстрактні поняття доступними для сприйняття [5]. Така візуалізація сприяє глибшому розумінню геометричних і аналітичних властивостей об'єктів.

У сфері машинного навчання та обробки даних програмування безпосередньо застосовує знання з теорії ймовірностей, математичної статистики та оптимізації. Бібліотеки pandas, scikit-learn, TensorFlow, PyTorch у Python чи спеціалізовані пакети в R дозволяють реалізувати алгоритми градієнтного спуску, лінійної регресії та нейронних мереж [6]. Студенти, опановуючи вищу математику, одночасно розвивають навички програмної імплементації цих методів, що готує їх до реальних завдань у даних науках та штучному інтелекті.

Загалом, програмування пропонує значні переваги, такі як швидке прототипування гіпотез, розв'язання практичних задач та розвиток алгоритмічного мислення. Водночас існують виклики, зокрема ризик використання інструментів як «чорного ящика» без розуміння алгоритмів, необхідність паралельного вивчення математики та кодування, а також потреба в критичному аналізі комп'ютерних результатів.

Висновок

Програмування стало невід'ємною частиною вищої математики. Воно не заміняє теоретичні знання, а значно їх розширює, дозволяючи студентам і дослідникам переходити від абстрактних формул до реальних обчислень та моделей. Найефективнішим є інтегрований підхід, коли математична теорія одразу супроводжується програмною реалізацією (наприклад, курси «Чисельні методи з практикумом на Python», «Лінійна алгебра та аналітична геометрія з MATLAB», «Дослідження операцій та оптимізація»). У майбутньому роль програмування у вищій математиці лише зростатиме, особливо з появою нових мов (Julia) та хмарних обчислювальних платформ.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Вікіпедія. Математичне програмування [Електронний ресурс] Доступ: https://uk.wikipedia.org/wiki/Математичне_програмування
2. Математичне програмування. Wiki ТНТУ [Електронний ресурс] Доступ: https://wiki.tntu.edu.ua/Математичне_програмування
3. Енциклопедія Сучасної України. Математичне програмування [Електронний ресурс] Доступ: <https://esu.com.ua/article-66938>
4. Кафедра дослідження операцій КНУ. Математичне програмування. [Електронний ресурс] Доступ: https://do.csc.knu.ua/?page_id=66
5. Математична база для програміста. DOU.ua [Електронний ресурс] Доступ: <https://dou.ua/lenta/columns/mathematical-basis-for-programmer/>
6. Наскільки програмісту потрібно знати математику. Cases Media [Електронний ресурс] Доступ: <https://cases.media/article/naskilki-programistu-potribno-znati-matematiku-yaka-potribna-matematika-programistu-de-vchiti-matematiku-dlya-programistiv>

Петренчук Артем Миколайович - студент групи 2ПКТ-24Б, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний, [e-mail:Artempetrenchukcompany@gmail.com](mailto:Artempetrenchukcompany@gmail.com)

Petrenchuk Artem Mykolayovych - student of group 2PKT-24B, Faculty of Intellectual Information Technologies and Automation, Vinnytsia National Technical University, Vinnytsia, email:Artempetrenchukcompany@gmail.com