УДК 004.94

ПОРІВНЯННЯ АЛГОРИТМІВ КЛАСИФІКАЦІЇ BIG DATA МЕТОДАМИ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ

DOI 10.36994 / 2788-5518-2023-01-05-15

***Одегов М.А.**, к.т.н.,доц. Державний університет інтелектуальних технологій і зв'язку, Одеса, Україна. onick_64@ukr.net*

***Гаджиєв М.М.**, д.т.н., проф. Державний університет інтелектуальних технологій і зв'язку, Одеса, Україна. gadjievmm@ukr.net*

***Буката Л.М.** Державний університет інтелектуальних технологій і зв'язку, Одеса, Україна. ygrikluda@gmail.com*

***Глазунова М.В.**, к.ф-м.н. Державний університет інтелектуальних технологій і зв'язку, Одеса, Україна. lvglazun@gmail.com*

***Кочеткова М.В.** Державний університет інтелектуальних технологій і зв'язку, Одеса, Україна., jubdvg@gmail.com*

***Анотація:** У статті вирішується задача порівняльного аналізу швидких алгоритмів класифікації, що можуть застосовувати для вирішення задач з надвеликими об'ємами даних (Big Data). Задача розв'язується методами імітаційного моделювання за допомогою програми Adaptive Metrics. Алгоритми найближчих сусідів, центрів класів та адаптивних правил порівнюються за критеріями надійності та продуктивності. Отримані результати дозволяють зробити висновок, що алгоритми. засновані на принципах M-means можуть ефективно використовуватись в задачах класифікації за певних умов, оскільки мають значну перевагу за критерієм продуктивності.*

***Ключові слова:** алгоритми найближчих сусідів, алгоритм центрів класів, алгоритм адаптивних правил, критерій надійності, критерій продуктивності, методи імітаційного моделювання, програма імітаційного моделювання.*

COMPARISON OF BIG DATA CLASSIFICATION ALGORITHMS BY SIMULATION METHODS

***Nick Odegov**, Ph.D., Ass. Prof. State University of Intellectual Technologies and Communication, Odesa, Ukraine. onick_64@ukr.net*

***Matin Hadzhyiev**, Dr. habil., Prof. State University of Intellectual Technologies and Communication, Odesa, Ukraine. gadjievmm@ukr.net*

***Liudmyla Bukata.** State University of Intellectual Technologies and Communication, Odesa, Ukraine. ygrikluda@gmail.com*

***Liudmyla Glazunova**, Ph.D., Ass. Prof. State University of Intellectual Technologies and Communication, Odesa, Ukraine. lvglazun@gmail.com*

***Marina Kochetkova.** State University of Intellectual Technologies and Communication, Odesa, Ukraine., jubdvg@gmail.com*

***Abstract:** With the development of information transmission and storage technologies, the volumes of data that require processing and analysis are growing rapidly. Therefore, the task of developing algorithms for solving various artificial intelligence problems for Big Data volumes is urgent. In our works, this informal term "Big Data" refers to situations when known processing algorithms do not allow solving a problem in a practically acceptable time. With regard to classification tasks, such conditions are possible when the first place is not even high reliability (that is, the minimum number of errors), but productivity (classification speed).*

The well-known method of nearest neighbors is one of the most productive. However, the indicator of the order of growth (the number of typical operations) for it is K x M x N, where K is the number of nearest neighbors, M is the number of classes, N is the typical number of class elements.

Along with this, we propose to consider algorithms based on the principles of M-means, where classes are replaced by only a small number of their characteristics. Among such algorithms, the article considers: the algorithm of class centers and the algorithm of adaptive rules. The order of growth for these algorithms is only M according to the number of classes.

The comparative analysis of these algorithms is performed by the method of simulation modeling. Simulation models are implemented by the Adaptive Metrics program, developed at the Department of Software

Engineering at DUITZ. In this program, the classification problem is solved using the example of the dichotomy problem for classes A and B. The program has the possibility of very flexible setting of models. Problems can be solved in 1-dimensional, 2-dimensional,..., 6-dimensional spaces. The distribution of factor values for classes A and B can have quite different statistical characteristics - from uniform and triangular distribution functions to functions approaching a normal distribution. The graphical interface of the program allows you to dynamically observe the solution of the classification problem in one-dimensional, two-dimensional and 6-dimensional projections.

As a result of multiple runs of the program, it was established that the algorithms of the nearest neighbors slightly outperform the algorithms of class centers and adaptive rules according to the criterion of reliability, and also comply with the principle of compactness (concentration of the largest number of erroneous solutions in the hypercube of errors). Algorithms based on M-means principles significantly outperform this algorithm in terms of performance. Also, the algorithm of adaptive rules best corresponds to the principle of equality of classes and is the most productive of the considered ones

Key words: *nearest neighbor algorithms, class centers algorithm, adaptive rule algorithm, reliability criterion, performance criterion, simulation modeling methods, simulation simulation program.*

Вступ

З поняттям Big Data автори цієї статті зіткнулися вже у простій задачі дихотомії, коли базові сукупності даних у класі А і класі В складали лише 10000 об'єктів у 6-мірному просторі. Треба було вирішити задачу віднесення 1000 тестових об'єктів кожного класу за допомогою алгоритму К найближчих сусідів [1], який вважається одним з найбільш ефективних з точки зору продуктивності (швидкості прийняття рішень) [2]. Число К для обох класів визначалось рівним 100. У авторів було враження, що досить сучасний комп'ютер «зависає». Втім, він напружено працював над задачею. Дійсно, у даному випадку значення порядку росту алгоритму [3] складає 2 x 6 x 10000 x 1000 x 100 = 12 x 10^9 (дванадцять мільярдів!) типових операцій, кожна з яких – визначення метрики у евклідовому просторі.

Порядок росту алгоритму К найближчих сусідів складає:

$$П=M\cdot K\cdot N, \tag{1}$$

де M – кількість класів; K – кількість найближчих сусідів; N – кількість відомих (вже раніше класифікованих) об'єктів класів.

В статті [4] обґрунтовано ряд швидких алгоритмів класифікації, які засновані на принципах M-means, що зазвичай використовуються для вирішення задач кластеризації [1, 2]. Порядок росту для цих алгоритмів складає лише $П = M$.

Виконання досліджень

Метою даної статті є порівняльний аналіз алгоритмів К найближчих сусідів з алгоритмами центрів класів та адаптивних метрик [4] за критеріями надійності та продуктивності.

Оскільки реальні задачі класифікації можуть вирішуватись в умовах досить різних варіантів розподілення значень факторів, то аналітичної оцінки у вигляді замкнених формул отримати в загальному випадку неможливо. Тому у даній статті задача вирішується методами імітаційного моделювання [5].

З часу виходу статті [6] (Ніколас, Метрополіс, 1949 р.) сфера застосування методів імітаційного моделювання поширилась на майже безмежну кількість наукових та практичних задач. Наведемо лише декілька прикладів. Застосування у сільському господарстві [7], у сфері будівництва [8], в задачах управління водними ресурсами [9], в задачах оптимізації протоколів передачі даних [10], в задачах оптимізації проектування авіаційної техніки [11], та, навіть, в медицині [12]. Тобто, головна проблема застосування цих методів – не визначення сфери, а синтез адекватної моделі. У даній статті ця проблема вирішується за допомогою гнучкого налаштування моделі з можливістю обрання дуже великої кількості варіантів розподілень значень факторів.

У даній статті задача порівняння методів класифікації вирішується для задачі дихотомії як базової задачі [4]. Класи умовно позначаються як А («зелений») та В («синій»).

1. Загальний опис імітаційної моделі

Імітаційне моделювання здійснюється за допомогою програми Adaptive Metrics (мова програмування C# , IDE Visual Studio 2022), яка розроблена співробітниками кафедри ІПЗ ДУІТЗ. Загальний вигляд вікна програми у певному варіанті показаний на рис. 1.

Моделювання виконується у фактор-просторі розмірності $D=6$. Окремо для кожного фактору для класів A і B визначаються функції розподілення значень. Базовий метод імітації значень у даному випадку – отримання псевдовипадкових (надалі – просто випадкових) чисел. Ці числа формуються відомим конгруентним алгоритмом [5] спочатку у діапазоні $r=[0,1]$. Такі числа приблизно рівномірно розподілені у даному діапазоні. Для отримання значень факторів класу A для рівномірних розподілень використовуються формула:

$$1R_A = SA \cdot r \cdot \qquad (2)$$

У формулі (2) і далі символ 1R розуміється як рівномірне розподілення. Конкретно у даній моделі параметр S_A дорівнює 300 (що не суттєво). Різні інші унімодальні розподілення для класу A позначаються як 2R, 3R,…, 10R і отримуються в результаті підсумовування відповідної кількості рівномірно розподілених чисел:

$$HR=\frac{1}{H}\sum_{h=1}^{H}1R, \quad H=1,2,...,10. \qquad (3)$$

Очевидно, що всі ці розподілення є симетричними відносно центру, який у графічному відображенні позначається як ім'я класу. У даному для «зеленого» класу завжди A = 150.

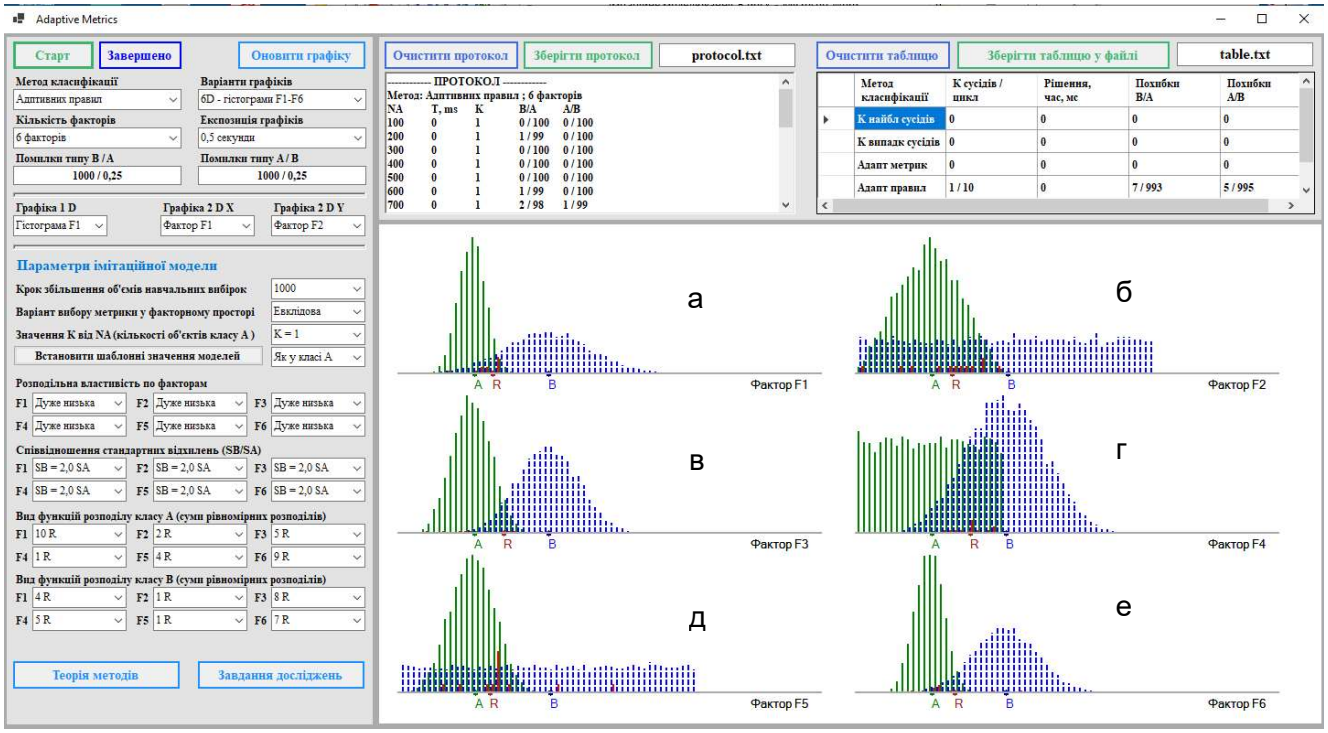


Рис. 1. Загальний вигляд вікна програми моделювання Adaptive Metrics з різними варіантами функцій розподілення окремих факторів для класів A і B:
а) A : 10R, B : 4R; **б)** A : 2R, B : 1R; **в)** A: 5R, B : 9R; **г)** A : 1R, B : 5R; **д)** A : 4R, B : 1R; **е)** A : 9R, B : 7R

Алгоритми генерування випадкових чисел намагаються зробити такими, щоб окремі послідовності були слабо корельованими, а краще – статистично незалежними. Відомо, що

щільність розподілення суми двох незалежних випадкових величин X та Y є згорткою їх щільностей розподілення [13]: $p(X+Y)=p(X)\otimes p(Y)$. Тому правило (3) формування послідовностей значень факторів для варіанту 2R можна представити у такій формі:

$$p(2R)=p(1R)\otimes p(1R), \tag{4}$$

а у загальному вигляді дана згортка виконується $H-1$ разів:

$$p(HR)=p(1R)\otimes p(1R)\otimes\ldots\otimes p(1R). \tag{5}$$

Із теорії спектрів [14] відомо, що операції згортки у часовому просторі відповідає операція множення у частотному просторі. Не важко провести аналогію між щільністю рівномірного розподілення випадкової величини та прямокутним відео імпульсом, спектр якого має форму функції $\mathrm{sinc}(x)=\sin(x)/x$. Відповідно, спектр згортки двох прямокутних імпульсів буде мати форму функції $\mathrm{sinc}^2(x)$. А такий спектр має трикутний імпульс [14]. Таким чином, розподілення типу 2R (4) повинно бути приблизно трикутним, якщо одномірні розподілення статистично незалежні. Приблизно такий варіант і спостерігається на рис. 1,б.

Функції виду $\mathrm{sinc}^H(x)$ досить швидко сходяться до функцій виду $\exp(-x^2)$ (гаусової функції). Втім, зворотне перетворення Фур'є гаусової функції знову ж таки приводить до гаусової функції, а у термінах теорії ймовірностей – до нормальної щільності розподілення. Це добре видно на прикладі рис. 1, в.

Висновок: послідовності випадкових чисел, що використовуються у даній роботі, можна вважати приблизно статистично незалежними.

Розподілення (3) при різних значеннях H будуть відрізнятись стандартними відхиленнями, а також ексцесами. Відомо, що дисперсія суми незалежних випадкових величин дорівнює сумі дисперсій [15]. Дисперсія кожного доданку у сумі (4) складає $D=D(1R)/H^2$, тоді дисперсія суми дорівнює $D(HR)=D(1R)/H$ і для стандартних відхилень буде справедлива теоретична формула:

$$\sigma(HR)=\frac{\sigma(1R)}{\sqrt{H}}. \tag{6}$$

Розподілення факторів класу B («синього») відрізняються лише параметрами зсуву та масштабу, тобто залежність (3) модифікується таким чином:

$$1R_B = SB \cdot SA \cdot r - B(SB), \quad B(SB) > 0, \tag{7}$$

де параметр SB у даному варіанті програми може приймати значення 1, 1,5 або 2. Відповідно, у нашому випадку розмах значень для класу B може складати 300, 450 або 600.

2. Алгоритм навчання класів та тестування класифікаторів

Базові послідовності значень 6-ти факторів для кожного класу формуються за формулами (3) та (7) у такій послідовності: спочатку об'єми класів дорівнюють базовим значенням N_1. В програмі передбачено два варіанти значень N_1: 100 або 1000.

Навчання для алгоритму K найближчих сусідів полягає лише у тому, що значення факторів відносяться до відповідних класів. Тобто, моделюється ситуація, коли вже є передісторія досліджень, згідно якої визначено базові сукупності класів A і B.

Об'єми класів далі нарощуються з кроком N_1 9 разів. Тобто об'єм навчальних вибірок поступово збільшується: 100, 200,…, 1000 (або 1000, 2000,…, 10000).

Для алгоритмів центрів класів і адаптивних правил на кожному такому кроці за допомогою рекурентних формул визначаються параметри: середні значення A_d і B_d, стандартні відхилення $\sigma_{A,d}$ і $\sigma_{B,d}$, а також вирішальні границі R_d для кожного фактору $d=1,2...6$.

На кожному з десяти кроків отримання значень навчальних вибірок формуються за тими ж самими правилами тестові вибірки об'єму N_1 для кожного класу. Також на кожному з цих кроків обраним алгоритмом класифікації елементи тестових послідовностей відносяться до класу A або до класу B. При цьому визначається кількість вірних рішень (A=A, B=B) та кількість помилок 1-го та 2-го типів (A/B , B/A) [4].

На кожному кроці нарощування об'ємів вибірок проміжні результати заносяться у протокол, а останній варіант при максимальному об'ємі базових елементів класів – у таблицю (рис. 1).

3. Варіанти налаштувань імітаційної моделі

Програма Adaptive Metrics дозволяє обрати один з алгоритмів класифікації [4]:

- K найближчих сусідів;
- K випадкових сусідів;
- центрів класів;
- адаптивних метрик;
- адаптивних правил.

Для перших двох алгоритмів число сусідів K можна обрати: або K = 1, або як відсоток від об'ємів базових елементів класів A і B. Програма дозволяє обирати відсотки від 1% до 10%. Тобто, максимальне значення K може бути від 10 до 1000.

Класифікація може здійснюватись лише з урахуванням першого фактору (F1), лише з урахуванням факторів F1 та F2, …, з урахуванням всіх 6-ти факторів.

Для обчислення метрик передбачено два варіанти: або Евклідова метрика, або метрика Хемінга.

Для кожного фактора окремо може налаштовуватись параметр розподільної властивості, який визначає відстань між центрами значень факторів для класів A і B. Значення даного параметру: «Дуже висока», «Висока», «Середня», «Низька», «Дуже низька». Приклади розподілень відповідно значенням даного фактора наведено на рис. 2.

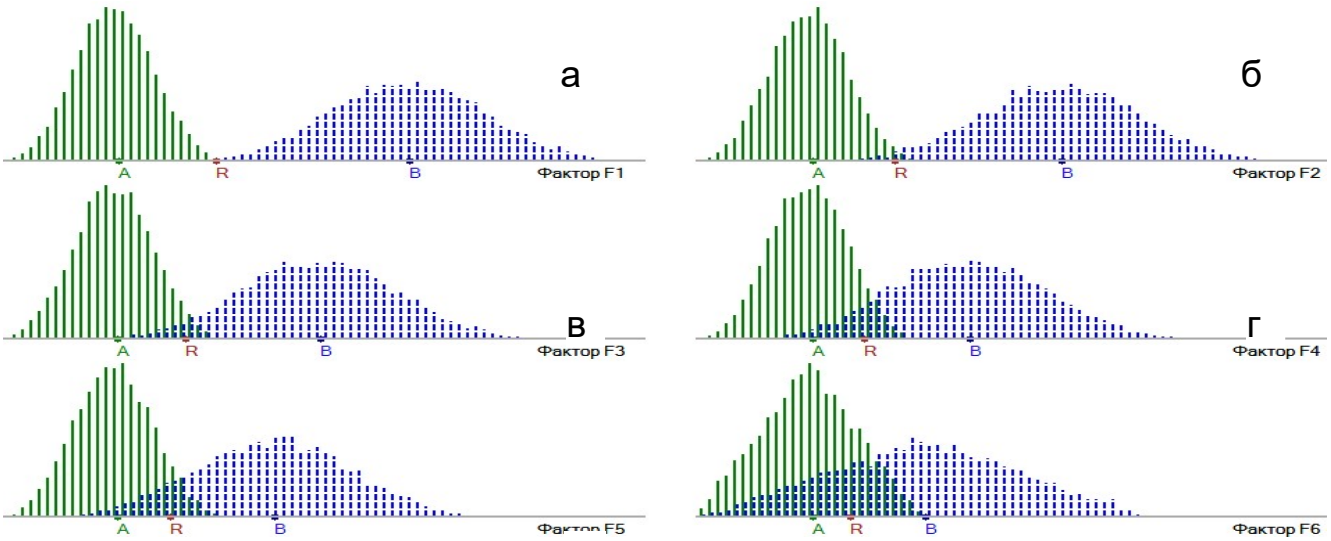


Рис. 2. Приклади розподілень значень факторів з різною розподільною властивістю:
а) «Дуже висока»; б) «Висока»; в) «Середня»; г) «Низька»; д) «Низька»; д) «Дуже низька»

Про налаштування параметрів розподілень сказано у п.1. Загалом програма дозволяє обрати:

- значення параметрів розподільної властивості – один з 30 варіантів;

- значення параметру SB (окремо для кожного фактора) – однин з 18 варіантів;
- варіантів вибору видів розподілень – по 60 окремо для кожного класу.

Тобто, всього можна обрати один з 64800 варіантів налаштування моделей. Тому імітаційна модель є достатньо гнучкою для вирішення задач порівняння характеристик алгоритмів класифікації.

Зауважимо, що серед моделей розподілень немає асиметричних функцій (для яких коефіцієнт асиметрії не дорівнює нулю). Це обмеження допущено навмисно: найбільше значення для алгоритмів класифікації має не стільки вид функцій розподілення, скільки доля елементів класів, які потрапляють в область перетину значень факторів.

4. Варіанти графічного відображення результатів

Графічному відображенню результатів навчання алгоритмів та класифікації закономірно приділяється значна увага [2]: звичайний (не штучний) інтелект людини дозволяє вирішувати різні задачі, якщо базова інформація представлена у логічному та наглядному вигляді. Тому програма Adaptive Metrics дозволяє представляти проміжні та кінцеві результати класифікації у дуже наочних графічних формах. А саме, передбачено три варіанти відображення:

- 1D – одномірна проекція розподілень значень факторів у вигляді гістограм для класів A і B (рис. 3);
- 6D – відображення розподілень по всім 6-ти факторам у вигляді гістограм (рис. 1, 2);
- 2D – проекція значень базових елементів та помилок класифікації на площину будь-яких різних двох факторів (рис. 4).

У варіантах 1D та 6D гістограми помилок відображаються в умовному масштабі без розрізнення кольору варіантів A/B і B/A. У варіанті 2D базові елементи відображаються окремими пікселями зеленого (для класу A) та синіми (для класу B) кольору; помилки відображаються крупними значками відповідно червоного (B/A) та коричневого кольорів (A/B) (рис. 4).

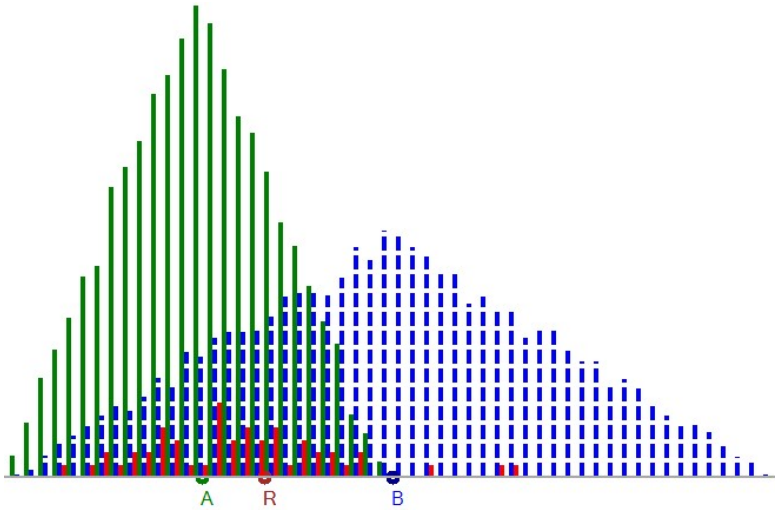


Рис. 3. Приклад проекції 1D на фактор F1

У кожному варіанті (рис. 1-4) на графіках букви A і B показують центри відповідних класів, а буквою R позначається вирішальна границя (у даному випадку від слова «Rule») для алгоритмів адаптивних метрик та адаптивних правил.

Всі результати розрахунків зберігаються в оперативній пам'яті, тому як під час виконання числових експериментів, так і після їх завершення можна вивести у вікно програми будь-який варіант графіки.

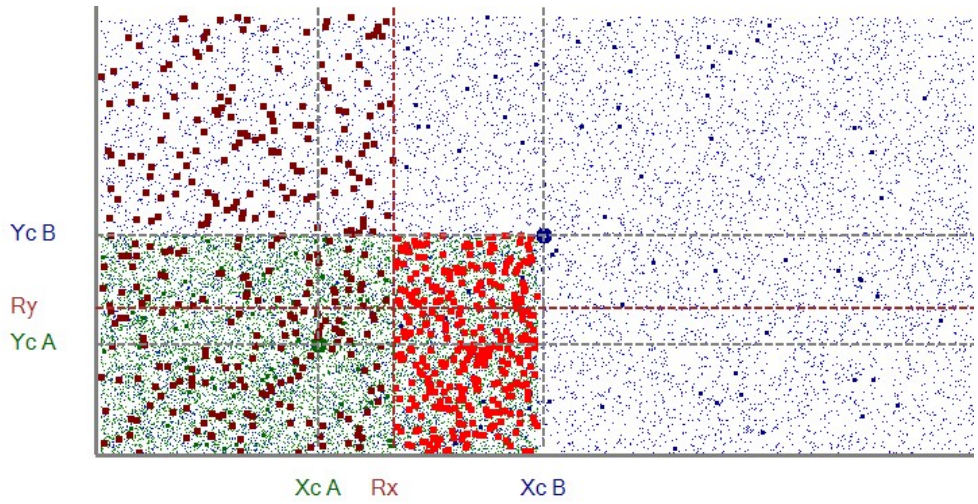


Рис. 4. Приклад 2D проекції: класифікація лише за фактором F1; метод адаптивних правил

5. Поняття гіперкубу помилок

У Евлкідовому просторі значення факторів елементів класів A і B можна представити у вигляді D – розмірного прямокутного паралелепіпеду з обмеженими ребрами. Після приведення значень факторів до єдиної шкали даний паралелепіпед перетворюється у D – розмірний гіперкуб. Саме такий термін ми й будемо використовувати надалі.

На наше переконання, у практичних задачах значення факторів завжди обмежені. Обмеження можна встановити як на підставі вивчення властивостей елементів класів у задачах Big Data, так, навіть, за допомогою здорового глузду. Наприклад, у «класичній задачі» – видавати чи не видавати кредит [2] – дихотомія вирішується у двомірному просторі: F1 – вік людини; F2 – дохід. Тут, мабуть, мова йде про споживчій кредит: на покупку смартфону, ноутбука та т. ін. Більш коштовні покупки потребують застави і алгоритм прийняття рішення там дуже сильно відрізняється від цього.

Отже, ми не дуже помилимось, якщо передбачимо можливі значення віку від 16 до 140 років, а місячний дохід – від мінімальної пенсії до, наприклад, 1 000 000 грн: навряд чи людина з великим доходом прийде за кредитом, який значно менше розміру його «кишенькових» грошей.

Таким чином, по кожному фактору для кожного класу можна визначити одномірну проекцію гіперкубу можливих значень як відрізок прямої між мінімальним та максимальним значенням, що ми позначимо як $HQ(A,d)$ та $HQ(B,d)$, де $d=1,2,...,D$. Далі, користуючись звичайними позначеннями теорії множин та теорії ймовірнісних просторів [15], визначимо одномірні проекції гіперкубів всіх значень для обох класів $HQ_d(A \cup B)$ та одномірні проекції областей можливих помилок $HQE_d(A \cap B)$. Тоді гіперкуби можливих значень окремих класів, значень для обох класів та гіперкубу помилок визначаються як відповідні Декартові добутки:

$$\begin{cases} HQ(A) = HQ(A,1) \times HQ(A,2) \times ... \times HQ(A,D) \\ HQ(B) = HQ(B,1) \times HQ(B,2) \times ... \times HQ(B,D) \\ HQ(A \cup B) = HQ_1(A \cup B) \times HQ_2(A \cup B) \times ... \times HQ_D(A \cup B) \\ HQE(A \cap B) = HQ_1(A \cap B) \times HQ_2(A \cap B) \times ... \times HQ_D(A \cap B) \end{cases} \quad (8)$$

Також визначимо об'єми цих гіперкубів як звичайні добутки довжин відрізків одномірних проекцій гіперкубів (8):

$$V(A),\ V(B),\ V(A \cup B),\ V(A \cap B) \quad (9)$$

та відносні об'єми цих гіперкубів:

$$\delta V(A)=\frac{V(A)}{V(A\bigcup B)},\ \delta V(B)=\frac{V(B)}{V(A\bigcup B)},\ \delta V(A\bigcap B)=\frac{V(A\bigcap B)}{V(A\bigcup B)}. \qquad (10)$$

Визначення (8-10) дозволяють виконувати експрес-аналіз наборів даних у практичних задачах класифікації, а також оцінювати надійність алгоритмів якісно, а в деяких випадках і кількісно, що ми покажемо на прикладах.

6. Приклади порівняльного аналізу алгоритмів класифікації методом імітаційного моделювання

По перше, розглянемо найважчий випадок, який тільки можливий у пропонованій імітаційній моделі: значення всіх факторів розподілені рівномірно, параметр SB=2, варіант розподільної властивості для всіх факторів «Дуже низька» (рис. 5). Алгоритми класифікації при цьому враховують всі 6 факторів.

З рис. 5 добре видно, що розмах значень для всіх факторів класу B вдвічі більша, ніж для класу A. Також видно, що мають місце досить складні для класифікації співвідношення між гіперкубами значень та помилок:

$$HQ\ (B)\equiv HQ\ (A\cup B),\quad HQ\ (A)\equiv HQE\ (A\cap B),$$

тобто гіперкуб класу B дорівнює гіперкубу всіх можливих значень обох класів, а гіперкуб класу A дорівнює гіперкубу помилок. Також видно, що по кожному фактору значення для класу A займають вдвічі менший відрізок, ніж для класу B. Тому для відносних об'ємів гіперкубів справедливі співвідношення:

$$\delta V\ (B)=1,\ \delta V\ (A)=\delta V\ (A\cap B)=1/64\ . \qquad (11)$$

Числові характеристики (11) вказують на те, що у даному випадку можна очікувати «перекіс» частоти помилок у сторону A/B (реальний клас елементу тесту B, але його віднесено до класу A). Саме така закономірність простежується у табл. 1.

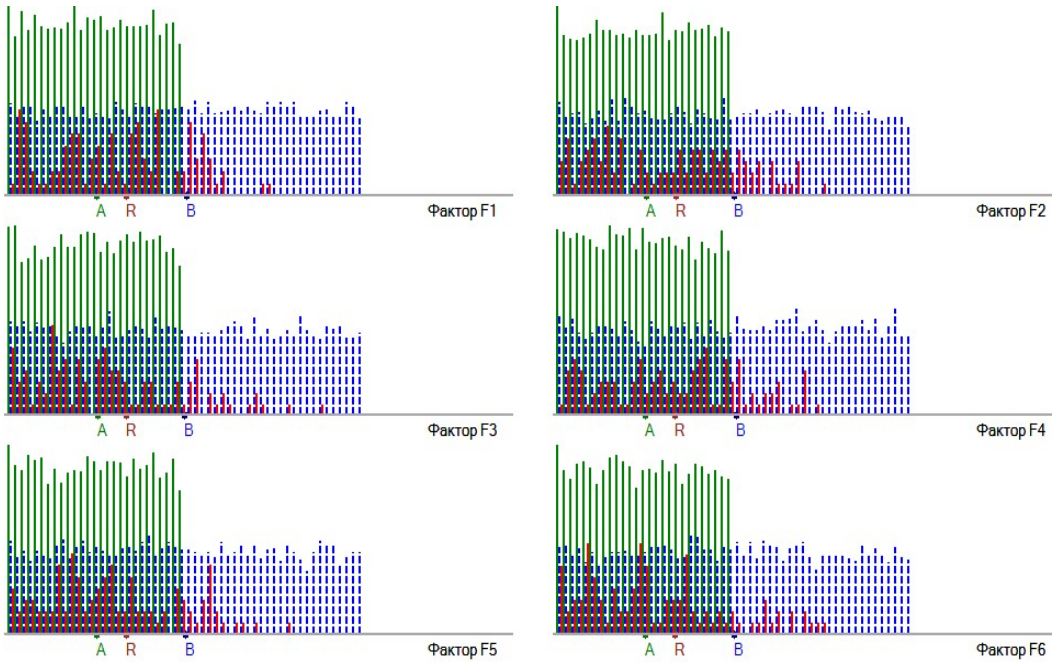


Рис. 5. Приклад класифікації алгоритмом K найближчих сусідів

Для досліджень якості алгоритмів за показником відповідності принципу «рівноправності класів» [4] доцільно ввести показник асиметрії помилок: співвідношення кількості помилок типу

B/A та A/B. При однаковій апріорній ймовірності появи у тестах представників класів A і B «хороший» алгоритм кластеризації повинен давати значення цього показника близьке до 1.

Таблиця 1.

Результати порівняльного аналізу алгоритмів класифікації

Метод	Найближчих сусідів, K=1			Центрів класів			Адаптивних метрик		
Об'єми класів	Час, мс	B/A	A/B	Час, мс	B/A	A/B	Час, мс	B/A	A/B
1000	18560	7	115	18	16	137	0	116	87
2000	36100	3	102	39	12	157	1	101	89
3000	53650	20	99	42	12	155	1	93	99
4000	73230	8	85	35	20	127	2	115	94
5000	96630	17	73	27	17	134	1	95	104
6000	120220	15	68	42	10	147	1	100	105
7000	129590	22	61	33	12	150	2	104	98
8000	154870	25	69	44	25	134	1	99	104
9000	165870	17	64	26	20	143	2	93	94
10000	190590	23	67	37	18	155	1	108	86
Загальний процент помилок, %			4,80			8,01			9,92
Середнє значення показника асиметрії помилок			5,11			8,88			0,94

Аналіз значень у табл. 1 дозволяє для даного прикладу налаштувань імітаційної моделі зробити наступні висновки:

- за критерієм надійності алгоритм найближчих сусідів (при K=1) приблизно вдвічі переважає алгоритми, що засновані на принципах M-means (алгоритми центрів класів та адаптивних правил);
- за критерієм продуктивності алгоритми, що засновані на принципах M-means суттєво переважають алгоритм K найближчих сусідів: алгоритм одного сусіда відпрацював більш як за 3 хвилини, тоді як алгоритм центрів класів приблизно за 20 мілісекунд, тоді як алгоритм адаптивних правил іноді вирішує задачу менше, ніж за 1 мілісекунду;
- за показником асиметрії помилок («рівноправності» класів) найкращі результати дає алгоритм адаптивних правил.

Як бачимо, у даному випадку алгоритми, засновані на принципах M-means поступаються алгоритму найближчих сусідів. Але не важко екстраполювати отримані результати, якщо у класах буде не 10000 об'єктів, а мільйон, або мільярд. Чи вистачить часу та терпіння дослідників для вирішення задачі класифікації, особливо у випадку, якщо кількість класів M>2?

У даному випадку метод K найближчих сусідів має ще одну перевагу: не потребує навчання: базові елементи класів A і B апріорно задані як статичні множини (хтось колись у попередній історії вже відніс елементи класів до базових сукупностей, як у прикладі з наданням кредиту). Але визначення оптимального числа K сусідів все ж таки потребує навчання алгоритму [1]. Тому виконаємо аналіз лише для алгоритму K найближчих сусідів при зміні значення K, що також дозволяє програма Adaptive Metrics. Приклади протоколів розрахунків для того ж самого варіанту розподілень значень факторів (рис. 5) наведено у табл. 2.

Таблиця 2.

Результати порівняльного аналізу алгоритму К найближчих сусідів при К=1% та К=2% від кількості елементів класів

Кількість елементів класів	К сусідів	Помилки А/В	Помилки В/А	Процент помилок	К сусідів	Помилки А/В	Помилки В/А	Процент помилок
1000	10	57	128	9,25	20	9	133	7,1
2000	20	1	103	5,2	40	0	123	6,15
3000	30	0	107	5,35	60	0	106	5,3
4000	40	0	96	4,8	80	0	131	6,55
5000	50	0	116	5,8	100	0	108	5,4
6000	60	0	118	5,9	120	0	112	5,6
7000	70	0	109	5,45	140	0	92	4,6
8000	80	0	92	4,6	160	0	117	5,85
9000	90	0	82	4,1	180	0	120	6,0
10000	100	0	77	3,85	200	0	98	4,9

Аналіз значень у табл. 2 дозволяє зробити наступні висновки:

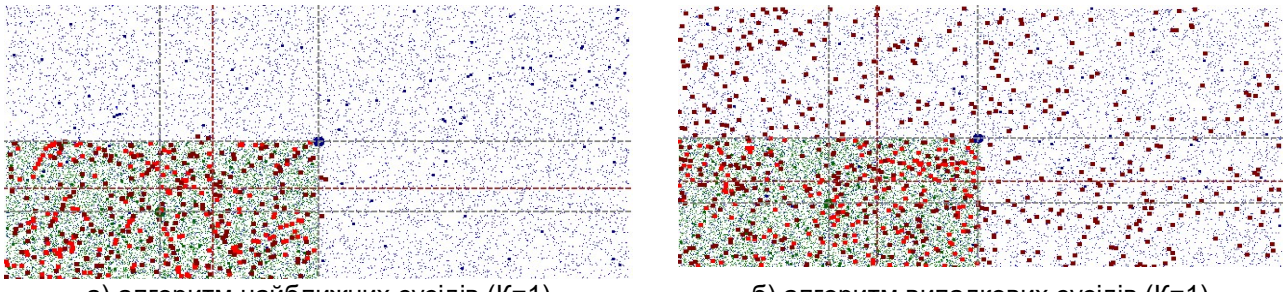
- процент помилок має незначну тенденцію зменшення при зростання кількості К найближчих сусідів;

- загальний процент помилок, навіть при великій кількості К найближчих сусідів, незначно менший, ніж для випадку К=1, тобто при даному налаштуванні моделі підвищення кількості сусідів не призводить до суттєвого покращення за критерієм надійності;

- варіація значень та відсутність чіткої функціональної залежності між кількістю К найближчих сусідів свідчить лише про те, що використані датчики випадкових чисел забезпечують досить впливову ймовірнісну складову, що можна побачити з рис. 5, де гістограми розподілень для дуже великих вибірок мають не зовсім рівномірний характер.

Наочно продемонструвати особливості алгоритмів, що розглядаються можна на прикладі того ж самого варіанту розподілень значень факторів, але у випадку класифікації лише з врахуванням двох факторів: F1 та F2. У даному випадку вичерпну інформацію у графічному вигляді можна отримати у варіанті відображення 2D (рис. 6, 7).

Можна прийняти без доказу, що «хороший» алгоритм класифікації забезпечує концентрацію основної кількості помилкових рішень всередині гіперкубу помилок (умова компактності помилок). Графічні відображення результатів класифікації показують, що алгоритм К найближчих сусідів (у даному випадку) відповідає цій умові. Алгоритм випадкових сусідів [4], хоча й дозволяє суттєво зменшити обсяг обчислень, але може давати значну кількість помилок класифікації (табл. 3). Також він зовсім не відповідає умові компактності помилок.

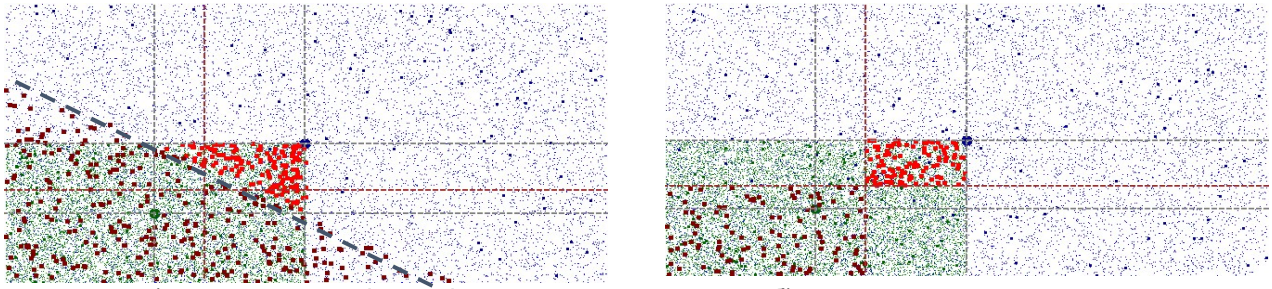


а) алгоритм найближчих сусідів (К=1) б) алгоритм випадкових сусідів (К=1)

Рис. 6. Результати класифікації алгоритмами К сусідів

Результати класифікації, що представлені на рис. 7, демонструють особливості досліджених алгоритмів, заснованих на принципах M-means. Алгоритм центрів класів фактично задає регресійну поверхню між класами (штрихова лінія на рис. 7. а). У найпростішому варіанті даного

алгоритму регресійна поверхня перпендикулярна направленню від центру класу А до центру класу В. Недолік: дана регресійна поверхня не чутлива до масштабування значень факторів для різних класів і завжди проходить по середині між центрами цих класів. Відповідно, знаходиться певна кількість помилкових рішень, які у даному варіанті моделювання виходять за межі гіперкубу помилок.



а) алгоритм центрів класів б) алгоритм адаптивних правил

Рис. 7. Результати класифікації алгоритмами, заснованими на принципах M-means

Алгоритм адаптивних правил у даному випадку (рис. 7, б) дає коректні результати класифікації: помилки розподіляються згідно обраного адаптивного правила [4] та цілком зосереджуються у гіперкубі помилок.

Таблиця 3.

Результати порівняльного аналізу алгоритмів найближчих сусідів та алгоритмів, заснованих на принципах M-means у двомірному фактор-просторі

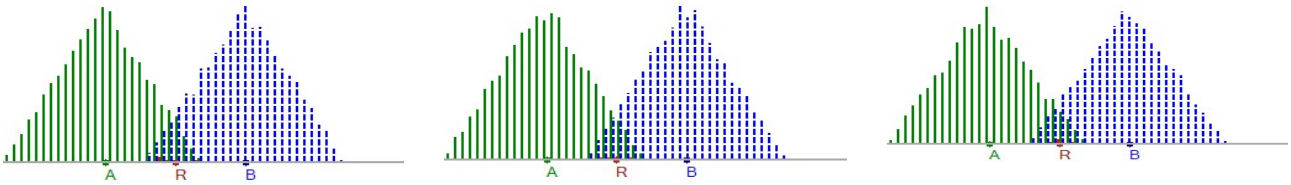
Алгоритм класифікації	Час прийняття рішень, мс	Помилки типу A/B	Помилки типу B/A	Загальний процент помилок, %	Значення показника асиметрії
К найближчих сусідів	53450	188 / 812	225 / 775	20,65	1,20
К випадкових сусідів	15	181 / 819	494 / 506	33,75	2,73
Центрів класів	14	147 / 853	277 / 723	21,2	1,88
Адаптивних метрик	0	109 / 891	104 / 896	10,65	0,95

Аналіз даних у табл. 3 дозволяє зробити певні висновки у даному варіанті моделювання:

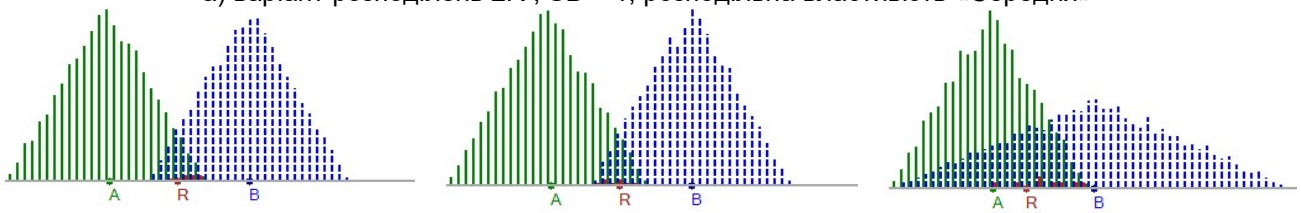
- найкращі результати за усіма показниками дає алгоритм адаптивних правил;
- алгоритм К найближчих сусідів (К=1) та алгоритм центрів класів дають приблизно одні й ти ж результати за кількістю помилок, але алгоритм центрі класів показує суттєво більшу ефективність по критерію продуктивності;
- алгоритм К випадкових сусідів практично дорівнює алгоритму центрів класів, оскільки використовує ті ж самі правила обчислення метрик, але за іншими показниками поступається всім іншим алгоритмам: помилки типу B/A навіть скоріше визначити підкиданням монетки. Також зауважимо. що цей алгоритм чутливий до кількості К сусідів: при збільшенні К пропорційно зростатиме час прийняття рішень.

Програма Adaptive Metrics дозволяє вирішувати задачі аналізу впливу розподільних властивостей окремих факторів на кількісні показники якості класифікації. Розглянемо лише три варіанти:

- класифікація здійснюється лише з врахуванням двох факторів F1 та F2, розподілення для обох класів і для обох факторів однакові, функції розподілення обрано за варіантом 2R (трикутні функції розподілення ймовірностей), варіант розподільної властивості «Середня»;
- варіант тих самих співвідношень по факторам F1 та F2, але додається ще третій фактор F3 з такими ж характеристиками (рис. 8, а);
- третій варіант відрізняється тим, що розподілення для класу В має вдвічі більший розмах значень (SB=2), а також розподільна властивість цього фактору встановлена як «Дуже низька» (рис. 8, б).



а) варіант розподілень 2R ; SB = 1; розподільна властивість «Середня»



б) для фактору F3: SB=2, розподільна властивість «Дуже низька»

Рис. 8. Варіанти розподілень значень трьох факторів

Результати прогонів імітаційної моделі дано у табл. 4, де кількість помилок дана як типові значення з протоколів за формою табл. 2.

Таблиця 4.

Помилки внаслідок впливу фактору з низькою розподільною властивістю

Алгоритм класифікації	Варіант налаштування імітаційної моделі					
	2 перші фактори		3 фактори, варіант а)		3 фактори, варіант б)	
	A/B	B/A	A/B	B/A	A/B	B/A
К найближчих сусідів	9	11	2	1	7	11
Центрів класів	6	4	1	0	3	5
Адаптивних правил	5	3	1	0	16	19

Аналіз даних у табл. 4 дозволяє зробити певні висновки з урахуванням того, що кількість помилок у даному випадку дуже мала, але:

- додавання інформативних факторів з хорошою розподільною властивістю призводить до підвищення надійності класифікації;
- додавання факторів з низькою розподільною властивістю може призводити до зниження надійності класифікації;
- найкращі результати за критерієм надійності у даному випадку дає алгоритм центрів класів;
- алгоритм адаптивних правил найбільш чутливий до включення в процес класифікації факторів з низькою розподільною властивістю.

З цих висновків витікає не дуже тривіальне твердження: алгоритм адаптивних правил завдяки його найвищої продуктивності може ефективно використовуватись у алгоритмах штучного інтелекту для попереднього відбракування «поганих факторів» (з низькою розподільною властивістю).

У даній статті розглянуто лише обмежену кількість варіантів налаштування імітаційної моделі, втім наведені приклади дозволяють зробити певні висновки стосовно застосування різних алгоритмів класифікації саме для випадків Big Data.

Висновки

1. Отримані результати дозволяють підтвердити основний висновок у статті [4:] різні методи (алгоритми) класифікації заздалегідь не є «поганими » чи «добрими», оскільки практика їх застосування потребує у кожному випадку вибору найбільш ефективного алгоритму для даної конкретної задачі.

2. При відносно невеликих об'ємах даних (кількість класів – одиниці або десятки, кількість елементів класів - тисячі) кращі результати за критерієм надійності дає алгоритм К найближчих

сусідів, але при зростанні об'ємів даних ефективність даного алгоритму нівелюється за критерієм продуктивності.

3. Ще одною перевагою алгоритмів К найближчих сусідів є відповідність принципу компактності помилок: помилкові рішення найбільше концентруються в межах гіперкубу помилок.

4. Алгоритми, засновані на принципах M-means дещо поступаються алгоритмам К найближчих сусідів за критерієм надійності, але суттєво переважають за критерієм продуктивності.

5. Алгоритм адаптивних правил не має конкурентів за критерієм продуктивності, оскільки виконує лише найбільш швидкі операції порівняння значень факторів тестових об'єктів з вирішальними границ ями.

6. Ще однією перевагою алгоритму адаптивних правил є відповідність принципу рівноправності класів: навіть у дуже складних для класифікації умовах частота помилок типу A/B приблизно дорівнює частоті помилок типу B/A.

7. Втім, алгоритм адаптивних правил найбільш чутливий до включення в обробку факторів з низькою розподільною властивістю. Даний факт можна використати для алгоритмів автоматизації відбракування малоінформативних факторів.

8. Окремо відмітимо можливість застосування розробленого програмного забезпечення і подібних розробок для вирішення дидактичних задач. Розроблене програмне забезпечення завдяки значній варіабельності налаштувань та наочності відображення результатів дозволяє вирішувати різні навчальні задачі:

- демонструвати особливості різних алгоритмів класифікації для вирішення задач класу Big Data на лекційних заняттях;
- сприяти розвитку абстрактного мислення: здобувачі освіти за 121 спеціальністю «Інженерія програмного забезпечення» та 122 спеціальністю «Комп'ютерні науки» повинні розвивати абстрактне мислення – навчитись мислити у багатомірних просторах;
- оскільки лише варіантів налаштування імітаційної моделі 64800, то якщо до цього додати варіанти вибору алгоритмів (5 різних), варіанти кроку зростання кількості елементів базових класів (100 або 1000), варіанти вибору метрик (Евклідова чи метрика Хемінга), то загальна кількість варіантів досліджень буде 1 296 000, тобто здобувачам освіти буде чим зайнятись на парах лабораторних занять.

Література

1. Марченко О.О., Россада Т. В. Актуальні проблеми Data Mining: навч. посібн. для студентів факультету комп'ютерних наук та кібернетики. Київ: КНУ, 2017. 150 с.
2. Чубукова І.О. Data Mining. Київ: КНЕУ ім. В. Гетьмана, 2016. 326 с. URL: http://kist.ntu.edu.ua/textPhD/Chubukova-Data_Mining.pdf
3. Кромен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ / пер. с англ. под ред. А. Шеня. М.: МЦНМО, 2002. – 960 с.
4. Одегов М.А., Гаджиєв М.М., Буката Л.М., Глазунова Л.В., Кочеткова М.В. Обґрунтування швидких алгоритмів класифікації на множинах Big Data за критеріями надійності і продуктивності. *Інфокомунікаційні та комп'ютерні технології*. Київ, 2023. No 1 (стаття у даному випуску журналу).
5. Буртняк І.В. Імітаційне моделювання. Івано-Франківськ: ПНУ ім. В. Стефаника, 2019. 97 с.
6. Nicholas; Metropolis. The Monte Carlo Method. *Journal of the American Statistical Association*, 1949. Vol. 44, no. 247. P. 335—341. ISSN 0162-1459. DOI: doi:10.2307/228023.
7. Andrea Parenti, Giovanni Cappelli, Walter Zegada-Lizarazu, Carlos Martín Sastre, Myrsini Christou, Andrea Monti, Fabrizio Ginaldi. SunnGro: A new crop model for the simulation of sunn hemp (Crotalaria juncea L.) grown under alternative management practices. *Biomass and Bioenergy*, Vol. 146, 2021, 105975, ISSN 0961-9534. DOI: https://doi.org/10.1016/j.biombioe.2021.105975.
8. Mayuri Rajput, Mostafa Reisi Gahrooei, Godfried Augenbroe. A statistical model of the spatial variability of weather for use in building simulation practice. *Building and Environment*, Vol. 206, 2021, 108331, ISSN 0360-1323. DOI: https://doi.org/10.1016/j.buildenv.2021.108331.
9. A. Araya, P.V.V. Prasad, I.A. Ciampitti, P.K. Jha. Using crop simulation model to evaluate influence of water management practices and multiple cropping systems on crop yields: A case study for Ethiopian highlands. *Field Crops Research*, Vol. 260, 2021, 108004, ISSN 0378-4290. DOI: https://doi.org/10.1016/ j.fcr.2020.108004.
10. El Arbi Abdellaoui Alaoui, Stephane Cedric Koumetio Tekouabou, Yassine Maleh, Anand Nayyar. Corrigendum to "Towards to intelligent routing for DTN protocols using machine learning techniques. *Simulation*

Modelling Practice and Theory. Vol. 123, 2023, 102689, ISSN 1569-190X. DOI: https://doi.org/10.1016 /j.simpat.2022.102689.
11. Jean-Charles Mare. Best practices for model-based and simulation-aided engineering of power transmission and motion control systems. *Chinese Journal of Aeronautics*. Vol. 32, Issue 1, 2019, P. 186-199, ISSN 1000-9361. DOI: https://doi.org/10.1016/j.cja.2018.07.015.
12. Tian Zhao, Xi Chen, Ke-Lun He, Qun Chen. A standardized modeling strategy for heat current method-based analysis and simulation of thermal systems. *Energy*, Vol. 217, 2021, 119403, ISSN 0360-5442. DOI: https://doi.org/10.1016/j.energy.2020.119403.
13. Закон распределения суммы двух случайных величин. URL: https://bstudy.net/ 717926/sotsiologiya/zakon_raspredeleniya_summy_dvuh_sluchaynyh_velichin_kompozitsiya_dvuh_zakonov_raspredeleniya.
14. Харкевич А.А. Спектры и анализ. М.: «ЛИБКОМ», 2009. 240 с.
15. Ширяев А.Н. Вероятность. М.: Наука, 1980. 576 с.

References

1. Marchenko O.O., Rossada T.V. Aktualni problemi Data Mining: navch. posibn. dlya studentiv fakultetu komp'yuternih nauk ta kibernetiki. Kiyiv: KNU, 2017. 150 s.
2. Chubukova I.O. Data Mining. Kiyiv: KNEU im. V. Getmana, 2016. 326 s. URL: http://kist.ntu.edu.ua/textPhD/Chubukova-Data_Mining.pdf
3. Kromen T., Lejzerson Ch., Rivest R. Algoritmy: postroenie i analiz / per. s angl. pod red. A. Shenya. M.: MCNMO, 2002. – 960 s.
4. Odegov M.A., Hadzhyiev M.M., Bukata L.M., Glazunova L.V., Kochetkova M.V. Obgruntuvannya shvidkih algoritmiv klasifikaciyi na mnozhinah Big Data za kriteriyami nadijnosti i produktivnosti. *Infokomunikacijni ta komp'yuterni tehnologiyi*. Kiyiv, 2023. No 1 (stattya u danomu vipusku zhurnalu).
5. Burtnyak I.V. Imitacijne modelyuvannya. Ivano-Frankivsk: PNU im. V. Stefanika, 2019. 97 s.
6. Nicholas; Metropolis. The Monte Carlo Method. *Journal of the American Statistical Association*, 1949. Vol. 44, no. 247. P. 335—341. ISSN 0162-1459. DOI: doi:10.2307/228023.
7. Andrea Parenti, Giovanni Cappelli, Walter Zegada-Lizarazu, Carlos Martin Sastre, Myrsini Christou, Andrea Monti, Fabrizio Ginaldi. SunnGro: A new crop model for the simulation of sunn hemp (Crotalaria juncea L.) grown under alternative management practices. *Biomass and Bioenergy*, Vol. 146, 2021, 105975, ISSN 0961-9534. DOI: https://doi.org/10.1016/j.biombioe.2021.105975.
8. Mayuri Rajput, Mostafa Reisi Gahrooei, Godfried Augenbroe. A statistical model of the spatial variability of weather for use in building simulation practice. *Building and Environment*, Vol. 206, 2021, 108331, ISSN 0360-1323. DOI: https://doi.org/10.1016/j.buildenv.2021.108331.
9. A. Araya, P.V.V. Prasad, I.A. Ciampitti, P.K. Jha. Using crop simulation model to evaluate influence of water management practices and multiple cropping systems on crop yields: A case study for Ethiopian highlands. *Field Crops Research*, Vol. 260, 2021, 108004, ISSN 0378-4290. DOI: https://doi.org/10.1016/ j.fcr.2020.108004.
10. El Arbi Abdellaoui Alaoui, Stephane Cedric Koumetio Tekouabou, Yassine Maleh, Anand Nayyar. Corrigendum to "Towards to intelligent routing for DTN protocols using machine learning techniques. *Simulation Modelling Practice and Theory*. Vol. 123, 2023, 102689, ISSN 1569-190X. DOI: https://doi.org/10.1016 /j.simpat.2022.102689.
11. Jean-Charles Mare. Best practices for model-based and simulation-aided engineering of power transmission and motion control systems. *Chinese Journal of Aeronautics*. Vol. 32, Issue 1, 2019, P. 186-199, ISSN 1000-9361. DOI: https://doi.org/10.1016/j.cja.2018.07.015.
12. Tian Zhao, Xi Chen, Ke-Lun He, Qun Chen. A standardized modeling strategy for heat current method-based analysis and simulation of thermal systems. *Energy*, Vol. 217, 2021, 119403, ISSN 0360-5442. DOI: https://doi.org/10.1016/j.energy.2020.119403.
13. Zakon raspredeleniya summy dvuh sluchajnyh velichin. URL: https://bstudy.net/ 717926/sotsiologiya/zakon_raspredeleniya_summy_dvuh_sluchaynyh_velichin_kompozitsiya_dvuh_zakonov_raspredeleniya.
14. Harkevich A.A. Spektry i analiz. M.: «LIBKOM», 2009. 240 s.
15. Shiryaev A.N. Veroyatnost. M.: Nauka, 1980. 576 s.