МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВОЛИНСЬКИЙ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ ІМЕНІ ЛЕСІ УКРАЇНКИ
Кафедра комп'ютерних наук та кібербезпеки

На правах рукопису

МАТВІЙЧУК ОЛЕГ МИКОЛАЙОВИЧ

АНАЛІЗ МЕТОДІВ ЧІТКОЇ ТА НЕЧІТКОЇ КЛАСТЕРИЗАЦІЇ НА ОСНОВІ ОСВІТНІХ ДАНИХ НАВЧАННЯ СТУДЕНТІВ

Спеціальність: 122 Комп'ютерні науки
Освітньо-професійна програма: Комп'ютерні науки та інформаційні технології
Робота на здобуття освітнього ступеня «магістр»

Науковий керівник:
МАМЧИЧ ТЕТЯНА ІВАНІВНА,
кандидат фізико-математичних наук,
доцент кафедри комп'ютерних наук
та кібербезпеки

РЕКОМЕНДОВАНО ДО ЗАХИСТУ
Протокол №_____
засідання кафедри комп'ютерних
наук та кібербезпеки
від _____ 2025 р.
Завідувач кафедри

(_____)___Гришанович Т. О._____

ЛУЦЬК – 2025

ЗМІСТ

ВСТУП

Актуальність теми. У сучасному світі система освіти зазнає значних трансформацій, пов’язаних із цифровізацією, масовим збором та використанням освітніх даних. Заклади освіти все частіше звертаються до інструментів аналізу даних для оцінювання результатів навчання, виявлення закономірностей у поведінці студентів та формування персоналізованих освітніх траєкторій. Одним із перспективних підходів є кластеризація освітніх даних, що дозволяє групувати студентів за схожими характеристиками або навчальними досягненнями. Це дає можливість не лише покращити якість освітнього процесу, а й сприяти більш обґрунтованому прийняттю управлінських рішень у сфері освіти.

Особливо цінним є порівняння методів чіткої та нечіткої кластеризації. Традиційні «чіткі» алгоритми, такі як K-Means, дозволяють однозначно віднести кожного студента до певної групи, проте вони не враховують ситуації, коли студент може проявляти риси декількох кластерів одночасно. У свою чергу, нечітка або м’яка кластеризація (fuzzy clustering) передбачає ймовірнісний підхід, що краще відображає реальні процеси у навчанні, де межі між групами студентів не завжди є строго визначеними. Порівняльний аналіз цих методів допоможе визначити найбільш ефективні підходи до кластеризації даних про студентів та підвищить точність моделей освітньої аналітики у перспективі.

Отже, дослідження методів чіткої та нечіткої кластеризації освітніх даних є актуальним як з наукової, так і з практичної точок зору. Воно сприятиме розвитку нових інструментів для адаптивного навчання, дозволить підвищити мотивацію студентів через персоналізовані рекомендації, а також надасть можливість адміністрації навчальних закладів більш ефективно планувати навчальний процес. В умовах зростання обсягів освітніх даних та необхідності їх якісного аналізу, результати такої роботи матимуть прикладне значення та потенціал для впровадження у системи електронного навчання та освітньої аналітики.

Метою роботи є дослідження методів чіткої та нечіткої кластеризації в

задачах класифікації на прикладі освітніх даних студентів. У дослідженні порівнюється використання методів K-Means та Fuzzy C-Means.

Для досягнення мети необхідно виконати такі завдання:

- розглянути теоретичну базу методів чіткої та нечіткої кластеризації;
- проаналізувати особливості використання кластерного аналізу в аналітиці даних студентів;
- підготувати та опрацювати вибірку освітніх даних для проведення експериментів;
- розглянути використання методів K-Means та Fuzzy C-Means за допомогою мови програмування R;
- провести експериментальний аналіз результатів роботи алгоритмів K-Means та Fuzzy C-Means на базі освітніх даних;
- порівняти швидкодію, інтерпретованість та гнучкість застосування цих методів кластеризації;
- визначити найбільш доцільний метод кластеризації для аналізу освітніх даних у контексті визначення основних типів студентів за для інтеграції цих результатів в роботу шкіл та університетів.

Об’єкт дослідження: методи кластеризації даних.

Предмет дослідження: застосування та порівняння методів K-Means та Fuzzy C-Means кластеризації на даних опитування ENAPE 2021 (National Survey on Educational System Access and Retention in Mexico).

Публікації:

1. Матвійчук О., Мамчич Т. І. Використання нечіткої кластеризації для ідентифікації груп студентів у освітніх даних. Інформаційні технології і автоматизація – 2025: Матеріали XVIII міжнародної науково-практичної конференції. Одеса, 30-31 жовтня 2025 р. - Одеса, Вид-тво ОНТУ, 2025. С. 134-136.

РОЗДІЛ 1.
МЕТОДИ ЧІТКОЇ ТА НЕЧІТКОЇ КЛАСТЕРИЗАЦІЇ ДАНИХ

1.1. Теоретичні основи кластеризації як методу аналізу даних

Кластеризація (кластерний аналіз) [1-4] – це один із базових методів інтелектуального аналізу даних (Data Mining) та машинного навчання (Machine Learning), сутність якого полягає у розподілі множини об'єктів на групи (кластери) таким чином, щоб об'єкти в межах одного кластера були максимально схожими між собою за певними характеристиками, а відмінності між об'єктами різних кластерів – якомога більшими. Іншими словами, кластеризація дозволяє виявити приховану структуру даних, яка не є очевидною при звичайному аналізі.

Математичне формулювання задачі кластеризації виглядає так: вибірку $X^n = \{x_1, \dots, x_n\} \subset X$ необхідно розбити на підмножини, які не перетинаються, так, щоб елементи всередині кожної підмножини були максимально близькими за метрикою відстані $d(x, x')$, а елементи різних підмножин – максимально далекими. Таким чином, алгоритм кластеризації описує функцію $X \to Y$, яка кожному об'єкту $x \in X$ ставить у відповідність мітку кластеру $y \in Y$.

На відміну від методів класифікації, де належність об'єкта до певного класу відома заздалегідь і задача зводиться до побудови моделі прогнозування, кластеризація відноситься до задач *навчання без учителя (unsupervised learning)*. Це означає, що алгоритм не має попередньо визначених "правильних" відповідей, а самостійно шукає закономірності у даних. Завдяки цьому кластеризація є одним із найпоширеніших інструментів для роботи з великими масивами інформації, де немає маркованих даних або неможливо задати чіткі правила сегментації.

У системі методів інтелектуального аналізу даних (Data Mining) кластеризація посідає важливе місце поруч із класифікацією, регресією, пошуком асоціативних правил і методами зниження розмірності. Вона використовується як самостійний метод для описового аналізу даних, а також як

проміжний етап у складніших моделях. Наприклад, кластери можуть застосовуватися для попереднього групування об’єктів перед побудовою прогнозної моделі або для виявлення нетипових (аномальних) спостережень.

У машинному навчанні (Machine Learning) кластеризація є ключовим підходом для задач сегментації та структурування даних. Її часто використовують у випадках, коли важко визначити чіткі еталонні класи або коли потрібно вивчити внутрішню організацію даних. Наприклад, у сфері освіти кластеризація може допомогти сегментувати студентів за стилями навчання, рівнем підготовки чи соціально-економічними характеристиками, у маркетингу – для створення портретів клієнтів, а у біоінформатиці – для класифікації генів чи білків за функціональними ознаками. Таким чином, кластеризація виступає універсальним інструментом, який забезпечує початкове структурування даних, необхідне для прийняття рішень та побудови подальших моделей.

Загалом, можна стверджувати, що кластеризація є невід’ємним елементом сучасних інформаційно-аналітичних систем, оскільки вона дозволяє отримати нові знання з даних, що раніше не мали чіткого опису або класифікації. Завдяки цьому вона стала важливою складовою як академічних досліджень, так і прикладних завдань у бізнесі, науці та соціальній сфері. Результат кластеризації може мати як чіткий характер (коли кожен об’єкт належить лише до одного кластера), так і нечіткий (коли об’єкт може одночасно належати до кількох кластерів із різними ступенями приналежності), тому варто розглянути теоретичні засади та відмінності цих підходів окремо.

1.2. Теоретичні основи чіткої кластеризації

Зазвичай говорячи про кластеризацію даних, мають на увазі чіткий варіант, оскільки він є очевиднішим та простішим, однак не варто плутати ці поняття. *Чітка кластеризація* [5] передбачає такий поділ даних, за якого кожен об’єкт належить тільки до одного кластера. На відміну від нечітких методів, де об’єкт може мати різний ступінь приналежності до кількох груп, у чітких алгоритмах

належність є однозначною та взаємовиключною. Такий підхід дозволяє отримати прозору структуру даних, що особливо важливо у випадках, коли потрібно чітко інтерпретувати результати аналізу.

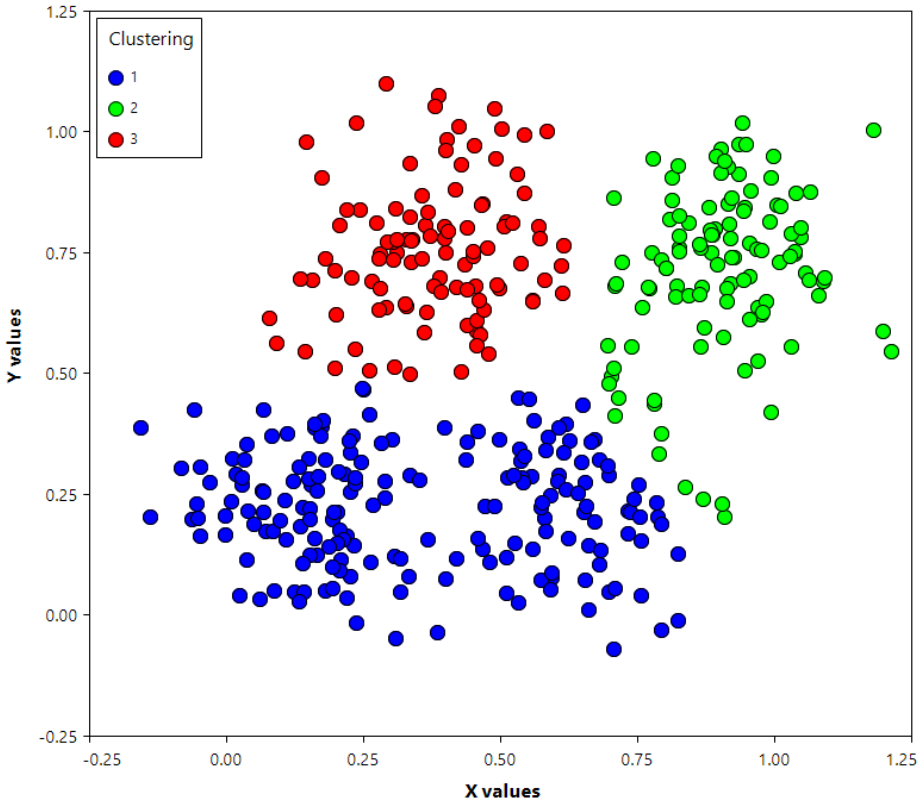

Рис. 1.1. Приклад чіткої кластеризації даних на 3 групи – кожне спостереження зафарбоване таким кольором, до якого кластеру воно відноситься

Основною метою чіткої кластеризації є мінімізація внутрішньокластерних відстаней та максимізація відстаней між кластерами. Метрика відстані для таких досліджень може варіюватися – найчастіше використовують евклідову відстань (що вимірює "пряму" відстань між точками у багатовимірному просторі), мангеттенську відстань (що обчислює суму модулів різниць координат, подібно до переміщення вулицями міста за сітковим планом) та узагальнену відстань Мінковського, яка дозволяє налаштовувати ступінь впливу різниці між координатами і охоплює як евклідову, так і мангеттенську метрики як окремі випадки.

Для більшості алгоритмів кластеризації вхідним параметром є кількість

кластерів. Важко передбачити, яка кількість кластерів буде оптимальна для кожного випадку вибірки. Тому використовують різні оцінки результатів кластеризації, які включать сума квадратів відстаней об’єктів від цетру свого кластеру (SSE), силуетний коефіцієнт, відстані між кластерами та інші підходи.

Ієрархічна кластеризація [6] є методом, який не потребує кількості кластерів як вхідного параметру алгоритму та ґрунтується на поступовому об’єднанні або розділенні об’єктів у кластери, утворюючи ієрархічну структуру у вигляді дендрограми. Вона має два основні підходи: агломеративний (знизу вгору) та дивізивний (згори вниз). В агломеративному варіанті кожен об’єкт на початку розглядається як окремий кластер, після чого найближчі пари поступово об’єднуються, доки не утвориться єдина структура. Дивізивний підхід працює навпаки – починаючи з одного загального кластера, він поетапно розділяє його на менші. Кількість кластерів може бути визначена шляхом «обрізання» дендрограми (рис. 1. 2) на певному рівні, тому, незважаючи на те, що цей вхідний параметр відсутній, метод потребує людського втручання для результативності роботи. Недоліком є висока обчислювальна складність для великих наборів даних, а також чутливість до вибору метрики відстані та методу злиття (linkage).

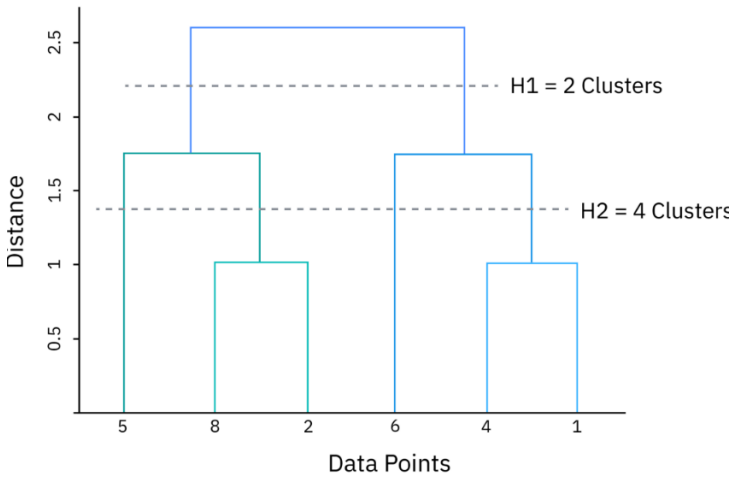

Рис. 1.2. Приклад «обрізання» сформованої дендрограми для отримання кластерів даних

DBSCAN (density-based spatial clustering of applications with noise) [7-8] належить до методів кластеризації на основі густини та особливо ефективний для

виявлення кластерів довільної форми у даних. Алгоритм визначає кластери як області з високою щільністю точок, відокремлені зонами низької щільності. Основними параметрами є радіус околу (eps) та мінімальна кількість точок у кластері ($minPts$). Об’єкти, що не належать до жодного кластеру через низьку щільність, позначаються як «шум» (рис. 1.3).

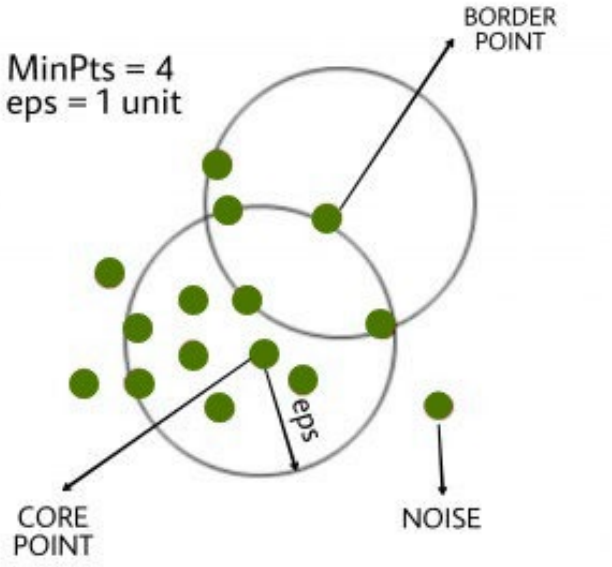

Рис. 1.3. Принцип роботи алгоритму DBSCAN - $eps = 1$, $minPts = 4$, тому точка Core Point є ядром, до сусідніх точок застосовується такий же підхід, що формує спільний кластер. Точка Noise є шумом, оскільки в околі радіуса 1 кількість точок менша ніж 4

Перевагою DBSCAN є здатність виявляти кластери довільної форми та коректно обробляти аномальні точки, що робить його корисним для аналізу складних освітніх чи соціальних даних. Проте вибір параметрів eps та minPts часто є нетривіальним завданням, і результати можуть суттєво змінюватися залежно від їх значень. Цей метод також автоматично визначає кількість кластерів розбиття, однак навантаженість іншими параметрами та вихідною інтерпретацією (наприклад, чи вважати шум як окремий кластер) також не робить цей метод повністю незалежним від людини.

1.3. Метод кластеризації K-means

Метод K-means (або K-середніх) [9] є найпоширенішим і найпростішим методом кластеризації даних. Його популярність зумовлена відносною простотою реалізації, зрозумілістю ітераційного процесу та достатньо високою ефективністю для великих обсягів даних. Він розбиває простір спостережень на k підмножин так, щоб кожне належало до свого найближчого центру (центроїда). При цьому метрика відстані, як уже було згадано, обирається залежно від особливостей спостережених даних.

Математична формалізація задачі виглядає так: нехай маємо вибірку з n спостережень, які описані m-вимірними векторами ознак:

$$X = \{x_1, x_2, \dots, x_n\}, x_i \in \mathbb{R}^m.$$

Необхідно розбити множину X на k кластерів $C_1, C_2, \dots, C_k$ так, щоб функція цільової якості (сума квадратів відстаней кожної точки до центроїда свого кластеру) була мінімізована:

$$J = \sum_{j=1}^{k} \sum_{x_i \in C_j} \| x_i - \mu_j \|^2,$$

де μ_j – центроїд j-го кластеру. Таким чином, алгоритм спрямований на мінімізацію внутрішньокластерної дисперсії. Окрім цього, можна помітити, що такий підхід явно описує розподіл простору даних на комірки діаграми Вороного.

Отже, послідовність кроків алгоритму K-means (рис. 1.4) включає:

1. Вибір кількості кластерів k.
2. Задання початкових центрів кластерів випадковим чином або за допомогою спеціальних стратегій (наприклад, k-means++).
3. Обчислення відстані для кожного об’єкта x_i до всіх центроїдів μ_j.
4. Призначення об’єктів відповідному найближчому кластеру:
$$C_j = \{x_i : \|x_i - \mu_j\| \leq \|x_i - \mu_l\|, \forall l = 1, \dots, k\}.$$
5. Обчислення нового центроїда для кожного кластеру як середнє

арифметичне об’єктів, що входять до цього кластеру:

$$\mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i.$$

6. Повторення кроків 3-5 допоки не досягнута одна з умов завершення ітераційного процесу: центроїди не змінюють своїх положень, центроїди змінили своє положення менше ніж на зазначений поріг, досягнута максимальна кількість ітерацій.

Задача K-Means пошуку оптимального розбиття спостережень вимірності d на k кластерів є NP-складною, оскільки кількість можливих розбиттів дуже велика – k^n, тому без евристичних методів тут не обійтися. Якщо ж k і d зафіксовані, то задачу можна розв’язати точно, однак час виконання складатиме $O(n^{dk+1})$, що робить точне вирішування непридатним для великих наборів даних. Така проблема вирішується за допомогою класичного алгоритму Ллойда, який забезпечує евристичний підхід та має обчислювальну складність $O(nkdi)$.

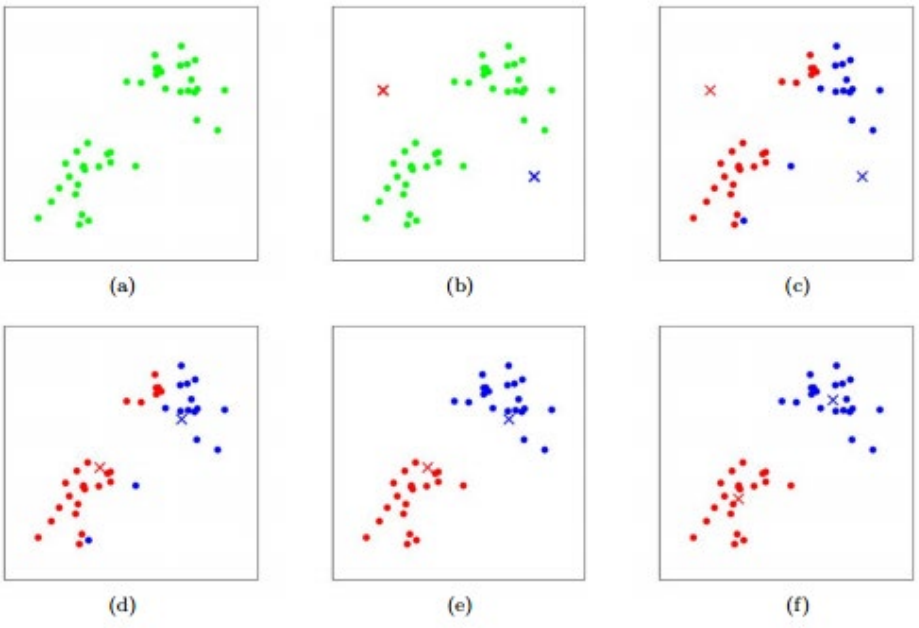

Рис. 1.4. Кроки алгоритму K-Means. a – оригінальний датасет, b – випадкові центроїди, c – розподіл спостережень за найближчим центроїдом, d-f – ітерації алгоритму із послідовним зсувом центроїдів на місце середнього арифметичного кожного кластеру

Метод K-середніх добре працює тоді, коли кластери мають приблизно однакові розміри, густину та «сферичну» форму. Якщо дані мають складнішу структуру (наприклад, витягнуті або з нерівномірною щільністю), алгоритм може давати некоректні результати. Окрім того, проблемою може бути чутливість до спостережень-аномалій – оскільки центроїди обчислюються як середнє арифметичне, на результат сильно впливають об’єкти, які є викидами із загальної маси об’єктів кластеру.

Проаналізувавши алгоритм, можна виділити його переваги – простота в реалізації, легка масштабованість на великі дані і інтуїтивно зрозуміла інтерпретація кластерів через центроїди. Серед недоліків можна виділити необхідність задання кількості кластерів k, чутливість до ініціалізації та викидів, а також обмеженість у випадках, коли кластери мають нерівну густину чи складну форму.

1.4. Теоретичні основи нечіткої кластеризації

Нечітка логіка (fuzzy logic) [10-12]— це розширення класичної двозначної логіки, запропоноване Лотфі Заде у 1965 році. На відміну від традиційної логіки, де кожне твердження може бути або істинним (1), або хибним (0), нечітка логіка дозволяє оперувати проміжними значеннями істинності. Основним інструментом виступають функції належності, які описують ступінь приналежності елемента до певної множини у межах інтервалу [0,1]. Такий підхід дозволяє формалізувати та математично обробляти неточність, невизначеність та розмитість, що природно зустрічаються в реальних даних і процесах. У математиці це створює основу для побудови нечітких множин, нечітких відношень та операцій над ними.

Завдяки своїй здатності враховувати неточність і неповноту інформації, нечітка логіка знайшла широке застосування в різних сферах науки і техніки. У керуванні технічними системами вона використовується для побудови нечітких регуляторів (наприклад, у побутовій техніці чи автомобілях), що забезпечують

гнучке прийняття рішень на основі нечітких правил типу «якщо – то». У сфері штучного інтелекту нечітка логіка застосовується для класифікації, прийняття рішень та моделювання поведінки, де класичні алгоритми неефективні через складність або невизначеність даних. Також вона активно використовується у біоінформатиці, економіці та аналізі даних.

Так, переймаючи концепцію нечіткості на задачу кластеризації, виникла нечітка кластеризація. На відміну від чіткої, де кожен об'єкт може належати лише до одного кластера, *нечітка кластеризація* дозволяє об'єктам мати певний ступінь належності до кількох кластерів одночасно (рис. 1.5). Вона є умовним логічним розширенням чіткого свого підходу. Такий метод краще відображає реальність, де об'єкти часто не можуть бути віднесені до лише однієї категорії. Наприклад, студент може одночасно демонструвати риси кількох груп поведінки, і жорстке віднесення до однієї з них спотворює картину.

Основна відмінність полягає у введенні функції належності. Якщо у чіткій кластеризації маємо:

$$u_{ij} \in \{0,1\}, \sum_{j=1}^{k} u_{ij} = 1,$$

де u_{ij} – це індикатор належності об'єкта x_i до кластеру j, то у нечіткій кластеризації:

$$u_{ij} \in [0,1], \sum_{j=1}^{k} u_{ij} = 1.$$

Це означає, що об'єкт x_i може мати часткову належність одразу до кількох кластерів. Наприклад, при чіткій кластеризації на 3 кластери спостереження x_1, яке належить до кластеру C_1, функція належності буде мати вигляд $u(x_1) = (1,0,0)$, а для спостереження x_2, яке частково належить до кластеру C_2 і C_3, і взагалі не належить до C_1 – $u(x_2) = (0, 0.5, 0.5)$.

Так само, як і для чіткої кластеризації, задача нечіткої полягає у мінімізації цільової функції, але зі змінами:

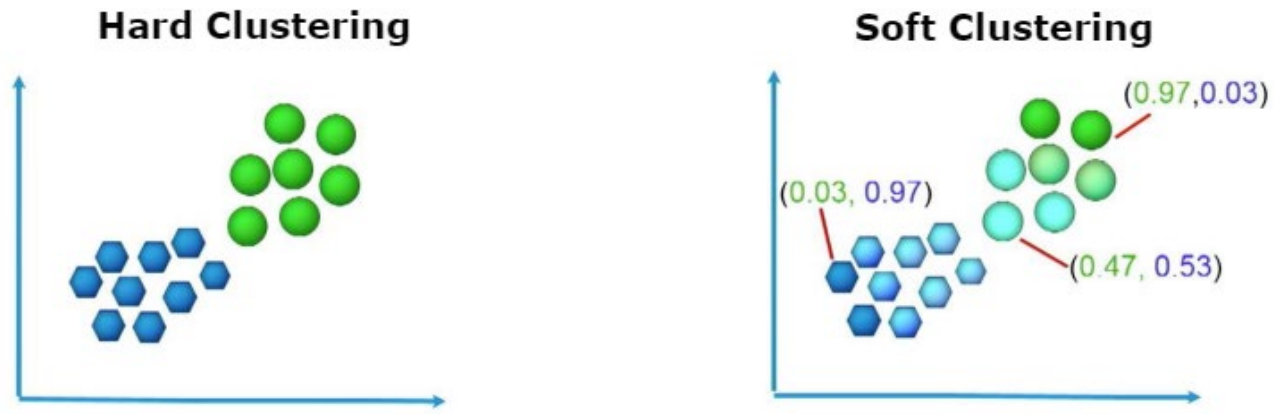

Рис. 1.5. Порівняльна візуалізація чіткої і нечіткої кластеризації. Ліворуч – чітка кластеризація, колір чітко розмежовує дані до одного чи іншого кластеру. Праворуч – колір показує ступінь належності до кластеру

$$J_m = \sum_{j=1}^{n}\sum_{j=1}^{k} u_{ij}^{m} * \| x_i - \mu_j \|^2,$$

де n – кількість об'єктів, k – кількість кластерів, $x_i \in \mathbb{R}^d$ – вектор ознак об'єкта, c_j – центр j-го кластеру, u_{ij} – ступінь належності x_i до кластеру j, $m > 1$ – параметр фазифікації, який керує розмитістю меж між кластерами.

Параметр фазифікації m (fuzzifier) є одним із ключових у нечіткій кластеризації, зокрема і в алгоритмі Fuzzy C-Means. Він визначає ступінь "розмитості" кластерів і безпосередньо впливає на розподіл значень функцій належності між кластерами. При значенні $m = 1$ алгоритм фактично зводиться до чіткої кластеризації, де кожен об'єкт належить лише одному кластеру. Зі збільшенням $m > 1$ значення функцій належності стають більш вирівняними, тобто об'єкти частіше належать одночасно кільком кластерам із близькими ступенями. Це дозволяє краще відображати невизначеність у даних, проте занадто велике m призводить до надмірного розмиття та втрати інтерпретованості кластерів. Тому вибір параметра фазифікації є критичним: зазвичай його встановлюють у діапазоні $m \in [1.5\ ;\ 3]$ залежно від природи даних та мети дослідження.

Одним із підходів до оцінки ступеня розмитості кластеризації є

використання *інформаційна ентропії Шеннона* [13], яка дозволяє виміряти рівень невизначеності у розподілі належностей об’єктів до кластерів. Для кожного об’єкта обчислюється ентропія його вектора належностей u_{ij}, що показує ймовірність приналежності об’єкта i до кластера j. Формула має вигляд:

$$h_i = -\sum_{j=1}^{c} u_{ij} * \log(u_{ij}),$$

де c – кількість кластерів, u_{ij} – ступінь належності i-го об’єкта до j-го кластеру. Загальна ентропія всієї системи визначається як середнє арифметичне значення по всіх об’єктах:

$$H = \frac{1}{n}\sum_{i=1}^{n} h_i,$$

де n – кількість об’єктів. Чим вищим є значення ентропії, тим більш розмитою є кластеризація, а низькі значення свідчать про близькість до чіткого розподілу. Такий підхід дозволяє кількісно оцінювати баланс між точністю та гнучкістю кластеризації.

Сьогодні існує безліч нових методів і модифікацій до старих, які використовують принципи нечіткої логіки для кластеризації даних. Алгоритм *FLAME (Fuzzy clustering by Local Approximation of Memberships)* [14] є підходом, що базується на локальних властивостях даних. Він спочатку визначає для кожного об’єкта його "щільність" відносно сусідів, після чого ідентифікує "об’єкти-представники" (cluster supports), що відіграють роль центрів кластерів. Потім для інших точок ступені належності розраховуються на основі локальної апроксимації через сусідні об’єкти. Такий підхід добре працює на наборах даних зі складною формою кластерів і дозволяє уникати проблеми з визначенням кількості кластерів.

Алгоритм *Fuzzy DBSCAN* [15] є модифікацією класичного DBSCAN, що поєднує кластеризацію за щільністю з нечітким підходом. На відміну від стандартного DBSCAN, який жорстко визначає, чи належить точка до кластера чи є шумом, у Fuzzy DBSCAN ступінь належності точки до кластера

визначається на основі її відстані до центрів щільних регіонів та кількості сусідів у радіусі ε. Це дозволяє уникнути різкого відсікання точок та більш гнучко працювати з даними, особливо якщо вони містять шум чи перетини між кластерами.

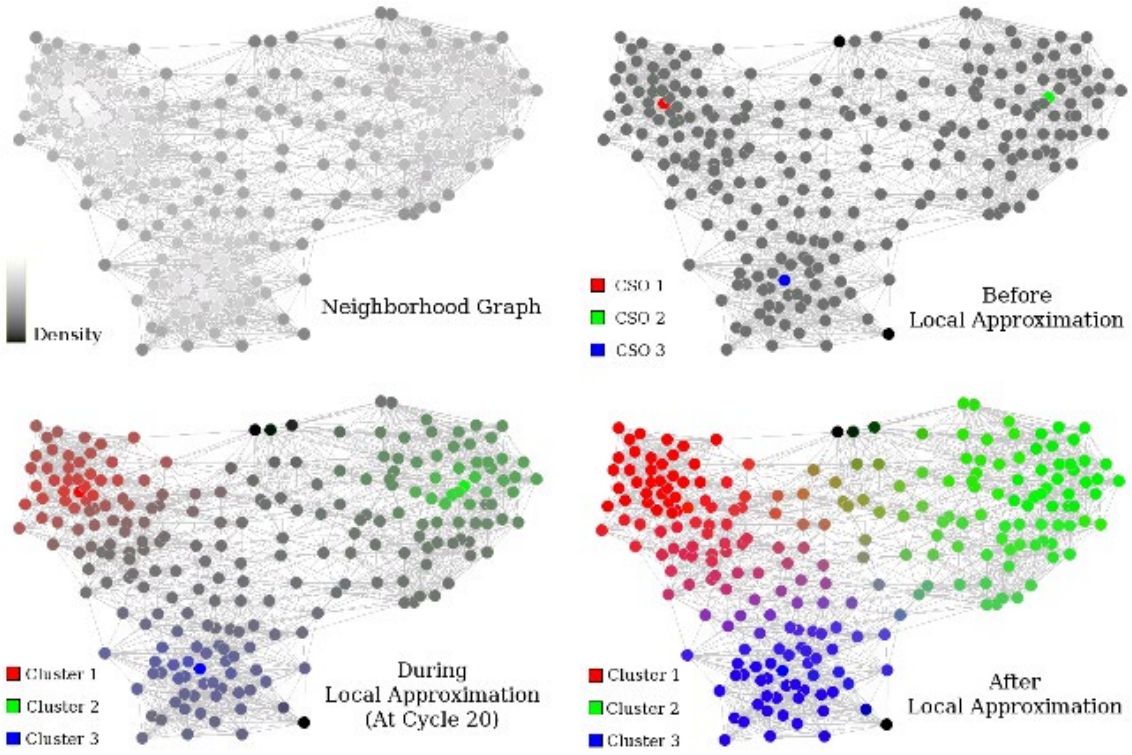

Рис. 1.6. Приклад роботи алгоритму FLAME

1.5. Метод кластеризації Fuzzy C-Means

Метод *нечітких c-середніх (Fuzzy C-Means, FCM)* [16], запропонований Дж. Данном та покращений Дж. Бездеком, є узагальненням алгоритму k-середніх, який дозволяє кожному об'єкту належати одночасно до кількох кластерів з різними ступенями належності.

Послідовність кроків алгоритму FCM:

1. Задання параметрів кількості кластерів c та ступеня фазифікації m.
2. Ініціалізація матриці належності $U = [u_{ij}] \in \mathbb{R}^{c \times n}$, де c – кількість

кластерів, n – кількість об’єктів. Зазвичай це відбувається випадковим чином, з дотриманням умови:

$$\sum_{j=1}^{c} u_{ij} = 1\,, \forall i.$$

3. Обчислення центрів кластерів v_j як зваженого середнього об’єктів, де вагами виступають ступені належності:

$$v_j = \frac{\sum_{i=1}^{n}(u_{ij})^m * x_i}{\sum_{i=1}^{n}(u_{ij})^m}, j = 1, \dots, c.$$

4. Оновлення матриці належності для кожного об’єкта x_i за формулою:

$$u_{ij} = \frac{1}{\sum_{k=1}^{c}(\frac{\left\|x_i - v_j\right\|}{\left\|x_i - v_k\right\|})^{\frac{2}{m-1}}}, j = 1, \dots, c.$$

Якщо об’єкт збігається з центром кластеру $\left\|x_i - v_j\right\| = 0$, то $u_{ij} = 1$ для цього кластеру і 0 – для інших.

5. Перевірка критерію зупинки алгоритму на основі зміни максимального значення в матриці належності:

$$\max_{i,j} \left|u_{ij}^{(t+1)} - u_{ij}^{(t)}\right| < \mathcal{E},$$

де $\mathcal{E}$ – поріг точності. Якщо умова виконується, алгоритм завершено. Інакше перехід до кроку 3.

FCM характеризується хорошою гнучкістю, оскільки дозволяє описувати дані, що перетинаються, ідентифікуючи нечіткі межі між групами. Параметр фазифікації дає змогу регулювати рівень невизначеності, що особливо важливо при роботі із соціальними, освітніми чи біомедичними даними, де межі між кластерами часто є неоднорідними. Так само, як і для k-means питання вибору метрики відстані залишається відкритим і залежить від особливостей задачі. FCM також може сходитися до локальних мінімумів, тому важлива коректна початкова ініціалізація.

Оскільки алгоритм вимагає на кожній ітерації оновлення $n \times c$ матриці ступенів належності та обчислення центрів кластерів, то його складність –

$O(ncdi)$, де d – розмірність простору ознак кожного спостереження, i – кількість ітерацій.

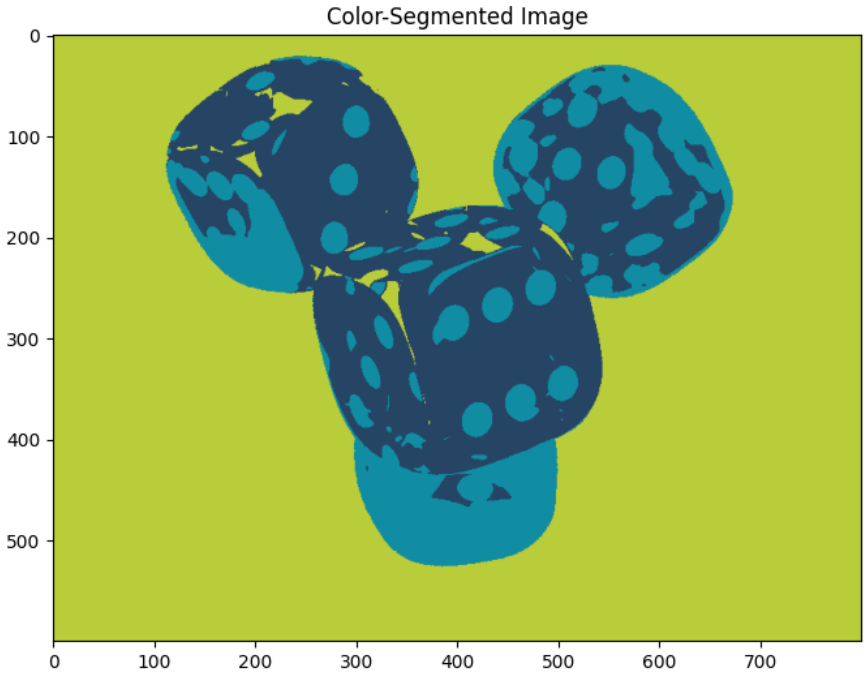

Рис. 1.7. Приклад сегментації зображення за допомогою FCM – зображення перетворене з реального фото у кластеризовану карту, де кожен піксель віднесено до певного сегмента за подібністю кольору

Алгоритм FCM є одним із класичних інструментів для Image Segmentation (сегментації зображень) [17] (рис. 1.7), оскільки він дозволяє більш гнучко працювати з даними, де границі між класами не є чіткими. Зображення розглядається як набір пікселів, де кожен піксель можна подати у вигляді вектора ознак. Найпростіший варіант — це лише інтенсивність (у градаціях сірого) або колір (наприклад, у просторі RGB чи Lab). У складніших випадках використовуються додаткові ознаки: текстурні характеристики, просторові координати пікселів, локальні градієнти тощо. Для візуалізації сегментації зазвичай кожному пікселю приписується той кластер, для якого його належність максимальна. Водночас, завдяки наявності «проміжних» значень, алгоритм краще працює у випадках з шумом чи невиразними межами. Це часто використовується у медичних зображення МРТ чи КТ, комп'ютерному зорі та

обробці супутникових зображень.

1.6. Оцінка якості кластеризації

Оцінка якості кластеризації є ключовим етапом аналізу даних, адже вона дозволяє визначити, наскільки добре обраний алгоритм і параметри відображають приховану структуру вибірки. На відміну від методів класифікації, де є відомі мітки класів, кластеризація є задачею без учителя, тому для її оцінювання використовуються спеціальні внутрішні метрики. Такі метрики, як правило, спираються на ідею максимізації подібності об'єктів усередині кластерів і мінімізації подібності між кластерами. У науковій практиці застосовують як метрики для «жорстких» методів, так і спеціальні показники для нечіткої кластеризації.

Силуетний коефіцієнт (Silhouette Coefficient) [18] вимірює, наскільки добре кожен об'єкт відповідає кластеру, до якого він віднесений, порівняно з найближчим альтернативним кластером. Для кожної точки i обчислюється середня відстань до всіх інших точок у її кластері $a(i)$ та найменша середня відстань до точок іншого кластера $b(i)$. Силует визначається як

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}.$$

Значення $s(i)$ змінюється від -1 до 1: високі позитивні значення свідчать про добре відокремлені кластери, значення, близькі до нуля, – про перекриття, а від'ємні – про можливу неправильну класифікацію. Середнє значення по всіх точках використовується як інтегральна міра якості розбиття.

Індекс Калінскі-Харабаза (Calinski–Harabasz Index, CHI) [19] є критерієм, який оцінює співвідношення дисперсії між кластерами до дисперсії всередині кластерів. Він визначається як

$$CHI(k) = \frac{Tr(B_k)/(k-1)}{Tr(W_k)/(n-k)},$$

де B_k – матриця міжкластерної дисперсії, W_k – матриця

внутрішньокластерної дисперсії, n – кількість спостережень, k – кількість кластерів. Операція $Tr()$ означає трасу матриці (trace) — суму елементів головної діагоналі. Чим більше значення CHI, тим кращим вважається розбиття, адже воно означає більшу компактність кластерів та сильніше їх розділення.

Коефіцієнт FPC (Fuzzy Partition Coefficient) [20] застосовується у випадках нечіткої кластеризації, коли кожен об’єкт може належати одночасно до кількох кластерів із різними ступенями належності. FPC обчислюється як

$$FPC = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{k} u_{ij}^2,$$

де u_{ij} – ступінь належності об’єкта i до кластеру j. Значення FPC змінюється від $\frac{1}{k}$ до 1. Високі значення близькі до 1 свідчать про чітке віднесення об’єктів до кластерів, тоді як низькі значення означають сильне перекриття та нечіткість структури. Цей індекс є особливо корисним для визначення оптимального параметра фазифікації m у методі FCM.

1.7. Візуалізація багатовимірних даних

Якщо дані мають дві характеристики, то їх легко відобразити на XY координатній площині. Однак, майже всі реальні випадки описуються за допомогою багатовимірних складних даних, тому необхідно використати метод спрощення для можливості візуалізації такої інформації.

Метод головних компонент (PCA, Principal Component Analysis) [21] — це статистичний підхід для зменшення розмірності багатовимірних даних шляхом проєкції їх у простір меншої кількості змінних (головних компонент). Кожна головна компонента є лінійною комбінацією початкових ознак і обирається так, щоб максимально пояснити варіацію в даних. Таким чином, замість аналізу десятків або сотень вимірів, ми можемо відобразити дані у 2D чи 3D, зберігаючи основну структуру і відносини між об’єктами.

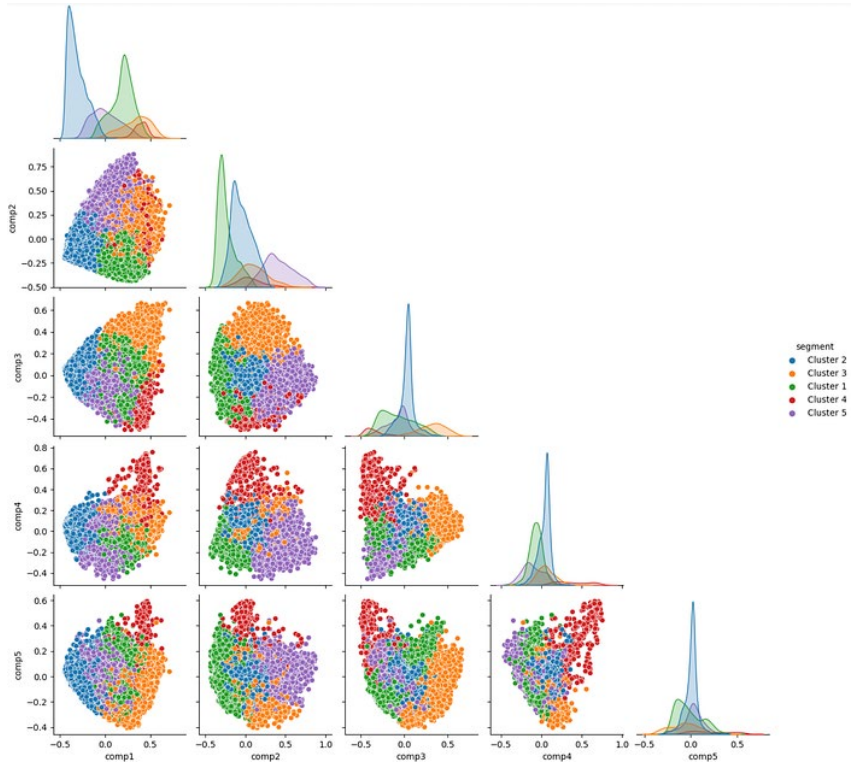

Рис. 1.8. Використання PCA для візуалізації результатів кластеризації даних

У випадку кластеризації PCA особливо корисний для візуалізації результатів (рис. 1.8). Алгоритм працює у багатовимірному просторі, який складно уявити, але за допомогою PCA дані можна спроєктувати на перші дві або три головні компоненти та відобразити на площині чи у просторі. Це дозволяє побачити приблизні межі кластерів, ступінь їхнього перекриття та відстані між ними. Хоч PCA не завжди зберігає всі характеристики кластерів, він дає наочне уявлення про якість і адекватність отриманого групування.

1.8. Аналіз останніх досліджень чітких та нечітких методів кластеризації даних

У роботі індонезійських вчених [22] автори описують використання методів чіткої та нечіткої кластеризації для автоматичного групування вступів (абстрактів) наукових статей. Вони хотіли перевірити чи можна за допомогою методів K-Means та Fuzzy C-Means визначити, до якої наукової галузі відноситься стаття: інформаційних технологій, охорони здоров’я чи економіка.

При цьому враховується, що наукові роботи часто поєднують кілька галузей, тому жорстке віднесення може бути неточним, тоді як нечітка кластеризація дозволяє моделювати міждисциплінарність розглянутих статей.

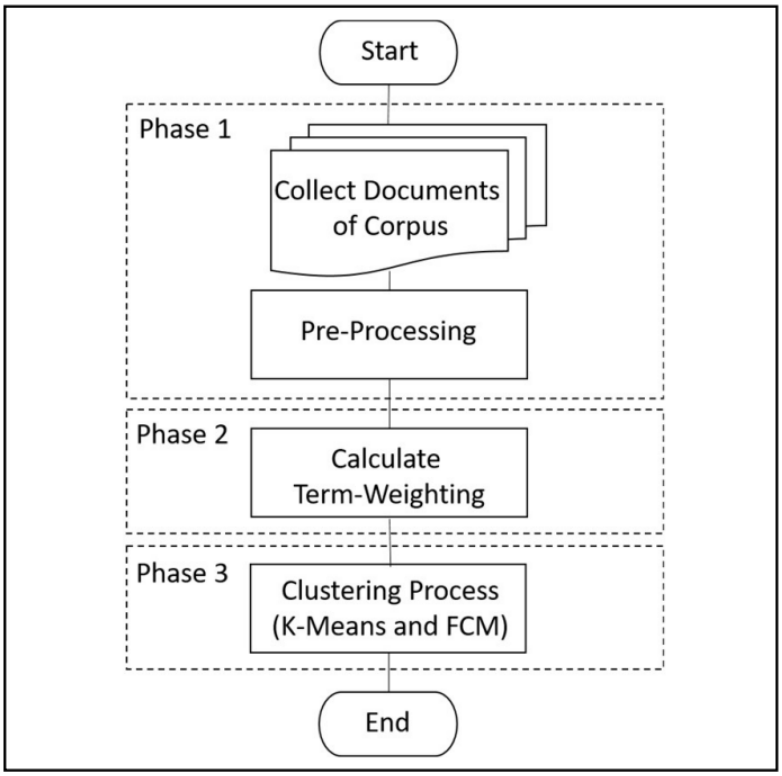

Рис. 1.9. Етапи проведення експерименту. Перший етап – збір та підготовка даних, другий етап – обчислення ваг термів, третій етап – кластеризація

Експеримент (рис. 1.9) проводили з 15 вступами, по 5 з кожної категорії. Для очищення текстів, приведення до нижнього регістру слів, обчислення ваг слів за їхньою частотою використано Python, і для етапу кластеризації – MATLAB. Результати показали, що групи сформовані методом K-Means інтерпретуються важче, оскільки деякі документи могли потрапити в «чужу категорію». Fuzzy C-Means показав більш точне групування: більшість вступів залишилися у своїх початкових категорій, а для граничних випадків виявив їхню міждисциплінарність. Були ситуації, коли K-Means давав завжди однозначний кластер, FCM продемонстрував, що вступ може належати до всіх трьох груп рівнозначно.

У роботі [23] розглядається недолік класичного FCM – випадковий вибір

початкових центрів кластерів, що часто призводить до локальних мінімумів і поганих результатів. Автори пропонують покращений FCM, який обирає початкові центри на основі щільності точок (density-based strategy). Це дозволяє уникнути проблеми випадкових і поганих ініціалізацій на основі метеорологічних даних. Крім того, FCM та його модифікована версія були вперше реалізовані в середовищі WEKA (яке спочатку не мало FCM у своєму наборі алгоритмів). Це розширило можливості платформи і зробило алгоритм доступним користувачам для подальшого аналізу даних.

Експеримент проводили на основі даних про посушливість у провінції Аньхой (Китай) з 463 записами з трьома ознаками – широта, довгота і площа посухи. Вони були попередньо нормалізовані, щоб привести всі атрибути в діапазон [0; 1]. Порівняння здійснювалося у програмному середовищі WEKA між класичним K-Means, FCM та покращеним FCM. WEKA (Waikato Environment for Knowledge Analysis) – це відкрите програмне середовище для інтелектуального аналізу даних (Data Mining) та машинного навчання, розроблене в Університеті Вайкато (Нова Зеландія).

Результати показали, що для збіжності результату FCM і покращений FCM потребують менше ітерацій ніж K-Means. У покращеного FCM помилка менша, ніж у стандартного FCM, але більша за K-Means. З точки зору метеорологів покращений FCM точніше відобразив реальні закономірності – правильніше відніс певні райони до кластерів за рівнем посушливості.

У дослідженні [24] використання алгоритму K-Means, описано моніторинг та прогнозування академічної успішності студентів. Основна ідея полягає у застосуванні методів кластеризації для автоматичного групування студентів за рівнем їхніх результатів (на основі GPA та оцінок з курсів), що дозволяє адміністрації університету швидше приймати рішення та розробляти стратегії підтримки. Для цього вчені організувати матрицю оцінок 79 студентів за 9 курсів (79×9 матриця) та застосувати метод K-Means з метрикою евклідової відстані.

Результати розглядалися для різних вхідних значень кількості кластерів ($k = 3, 4, 5$) та використовувався детермінований підхід – середні значення

оцінок у кластерах, які потім співвідносилися з індексом успішності. Загалом, у всіх випадках алгоритм успішно поділив студентів на групи, що відповідають традиційним рівням академічної оцінки.

Стаття [25] присвячена аналізу впливу вибору метрики відстані на роботу алгоритму K-Means. Автори порівняли результати кластеризації, використовуючи три варіанти: евклідову, манхеттенську відстань та відстань мінковського. Метою роботи є показати, як саме вибір метрики впливає на якість кластеризації та значення спотворення (distortion) — функції, що оцінює ступінь розкиду даних відносно центрів кластерів. Експериметр виконувався на штучних двовимірних даних, що дозволило легко візуалізувати результати.

У результатах було показано наскільки метрика відстані має вирішальне значення для роботи алгоритму K-Means. Евклідова відстань дала найменше спотворення, швидко збігалась та показала найстабільніші кластери, мангеттенська відстань – формує кластери схожої структури але з більшим спотворенням, відстань Мінковського – це узагальнення для обох попередніх метрик (при $p = 1$ і $p = 2$ відповідно), при $p > 2$ спотворення поступово зменшується і результати наближаються до евклідової відстань, однак збіжність повільніша.

Проаналізувавши останні дослідження у сфері чітких і нечітких методів кластеризації, можна сказати, що вони спрямовані на підвищення якості розподілу даних через адаптацію метрик, алгоритмів та областей застосування. Загалом сучасні дослідження об’єднують прагнення зробити кластеризацію більш гнучкою, адаптивною та практично застосовною до реальних задач, де дані мають шум, складну структуру або розмиті межі.

РОЗДІЛ 2.
ЧІТКА ТА НЕЧІТКА КЛАСТЕРИЗАЦІЯ ОСВІТНІХ ДАНИХ НАВЧАННЯ СТУДЕНТІВ

2.1. Постановка задачі

Аналіз освітніх даних є важливим напрямом досліджень у сучасних інформаційних системах, оскільки дозволяє виявляти закономірності у доступі до освіти, її якості та чинниках, що впливають на успішність навчання. Національне опитування щодо доступу та постійності в освіті ENAPE 2021 містить багатовимірні соціально-економічні та демографічні характеристики респондентів, що створює підґрунтя для виявлення прихованих структур у даних.

Ці дані включають інформацію про аудиторію зараховану до різних шкільних циклів та рівнів освіти (дошкільна, початкова, середня, старша школа, вища освіта), показники доступу до освіти, причини відрахування, форми навчання (очне, дистанційне, змішане), технологічні ресурси вдома, методи підтримки студентів та сприяння до навчання школою. Традиційні статистичні методи не завжди забезпечують достатню гнучкість для роботи з такими даними, оскільки вони передбачають наявність чітко визначених класів або лінійних залежностей. Саме тому актуальним є застосування методів кластеризації, що дозволяють групувати дані на основі схожості без попереднього визначення міток.

Класичні алгоритми чіткої кластеризації, зокрема K-Means, дозволяють ефективно знаходити групи в даних, однак їхнім недоліком є жорстке віднесення кожного об’єкта лише до одного кластера. У випадках, коли респондент може належати до кількох груп одночасно (наприклад, через перетин соціальних, економічних чи освітніх факторів), такий підхід обмежує якість аналізу. На відміну від цього, методи нечіткої кластеризації, зокрема Fuzzy C-Means (FCM), надають можливість врахувати ступінь належності об’єкта до різних кластерів, що більш адекватно відображає складність освітніх і соціальних процесів.

Завданням даної роботи є проведення порівняльного аналізу чітких та нечітких методів кластеризації на основі даних ENAPE 2021. Для цього необхідно дослідити ефективність алгоритмів K-Means та FCM у контексті виявлення прихованих закономірностей у вибірці студентів, а також оцінити їх здатність відображати реальні соціально-освітні залежності. Особлива увага приділятиметься порівнянню чіткої та нечіткої кластеризації за кількісними показниками (кількість і структура кластерів, відстані між центрами кластерів, значення функцій похибки) та якісними характеристиками (здатність алгоритмів враховувати розмитість меж між групами, відповідність результатів апріорним знанням про освітнє середовище).

Очікуваним результатом є визначення того, який підхід — чіткий чи нечіткий — забезпечує більш адекватне групування респондентів у контексті багатовимірних освітніх даних. Це дозволить обґрунтувати можливість використання нечіткої кластеризації як більш гнучкого інструмента для моделювання складних соціально-освітніх явищ та створить підґрунтя для подальшого застосування результатів у системах підтримки прийняття рішень у сфері освіти.

2.2. Методологія дослідження

Для проведення дослідження чітких і нечітких методів кластеризації даних буде використано масив реальних соціально-освітніх даних з опитування ENAPE 2021 [26]. Цей набір містить широкий спектр показників, які характеризують доступ, утримання та прогрес студентів у системі освіти в Мексиці. Дані включають 3 пов’язані датасети у різних файлах (df01, df02, df03). Файл «df01» містить дані з характеристиками допомогосподарств, для оцінки умов та спроможностей студентів навчатися вдома, «df02» та «df03» – дані кожного студента окремо у відповідності з їхнім ідентифікатором домогосподарства з файлу «df01».

Першим етапом є підготовка даних: очищення від пропусків, нормалізація

числових значень, а також відбір релевантних ознак, що впливають на якість результатів кластеризації. Для цього застосовуються методи попередньої обробки, включаючи стандартизацію даних та приведення категоріальних змінних до числової форми, якщо такі є.

Другим етапом є застосування алгоритмів кластеризації. Для чіткої кластеризації використовується метод K-Means, що передбачає фіксовану кількість кластерів і жорсткий розподіл об’єктів. Для нечіткої кластеризації використовується алгоритм FCM, який дозволяє об’єкту одночасно належати до кількох кластерів із певними значеннями ступеня приналежності. Це дозволить порівняти, як жорсткі та нечіткі методи відображають приховані закономірності у даних, та оцінити їх здатність до роботи з неоднорідними і багатовимірними вибірками.

Третім етапом є проведення експериментів із різною кількістю кластерів, що дозволяє дослідити вплив цього параметра на якість кластеризації. Для обох методів будуть побудовані візуалізації розподілу даних у кластери та матриці результатів, які демонструють відмінності у підходах. Для нечіткої кластеризації додатково буде розглянуто матрицю приналежності, яка показує розподіл ступенів належності об’єктів до кожного кластера. Окрім цього, необхідно застосувати метрики оцінки якості кластеризації – силуетний коефіцієнт.

Заключним етапом є аналіз отриманих результатів та формулювання висновків. Буде проведено узагальнення переваг і недоліків чіткої та нечіткої кластеризації на даних ENAPE 2021, визначено умови, за яких застосування кожного з методів є найбільш доцільним. Це дозволить зробити практичні рекомендації щодо вибору методів кластеризації для дослідження великих соціально-освітніх даних.

2.3. Обґрунтування вибору інструментальних засобів

Для реалізації дослідження чіткої та нечіткої кластеризації даних було обрано мову програмування R та інтегроване середовище розробки RStudio

2024.09.0. Такий вибір обумовлений як науковою орієнтацією інструментів, так і наявністю розвиненої екосистеми бібліотек для обробки, аналізу та візуалізації даних.

Мова R [27-28] є однією з найпопулярніших мов програмування у сфері статистики, машинного навчання та аналізу даних. Вона спеціально створена для обробки великих обсягів статистичної інформації та надає широкі можливості для роботи з таблицями даних, матрицями та багатовимірними масивами. Однією з ключових переваг R є її орієнтація на математичні обчислення та статистичні моделі, що робить її надзвичайно зручною для досліджень у соціально-економічних та освітніх даних, до яких належить і набір ENAPE 2021.

Крім базового функціоналу, R має велику кількість пакетів для проведення як класичної (наприклад, k-means, hierarchical clustering), так і сучасної кластеризації, включно з нечіткими методами (fuzzy c-means, possibilistic clustering тощо). Для задачі кластеризації R забезпечує просту роботу з матрицями належності та метриками нечіткої кластеризації, вбудовані функції для обчислення показників якості кластеризації (індекс Данна, силует-метрика тощо), гнучкі засоби для візуалізації результатів, що дозволяють представити дані у вигляді графіків, діаграм та теплових карт.

Ще однією вагомою перевагою R є активна спільнота користувачів та розробників, яка постійно підтримує оновлення пакетів і пропонує нові інструменти для аналізу. Це дозволяє швидко адаптувати найновіші алгоритми кластеризації без необхідності реалізовувати їх з нуля.

Мова Python [29] сьогодні теж є однією з найпоширеніших у світі завдяки своїй універсальності та простоті синтаксису. Вона активно застосовується у сфері веброзробки, створення програмного забезпечення, аналізу даних, машинного навчання та штучного інтелекту. Python має потужну екосистему бібліотек (NumPy, pandas, scikit-learn, TensorFlow, PyTorch), що робить її зручним інструментом для виконання різноманітних завдань, у тому числі й статистичного аналізу. Гнучкість мови дозволяє поєднувати різні підходи до роботи з даними, а інтеграція з іншими мовами та технологіями робить Python

універсальним вибором для прикладних і наукових задач.

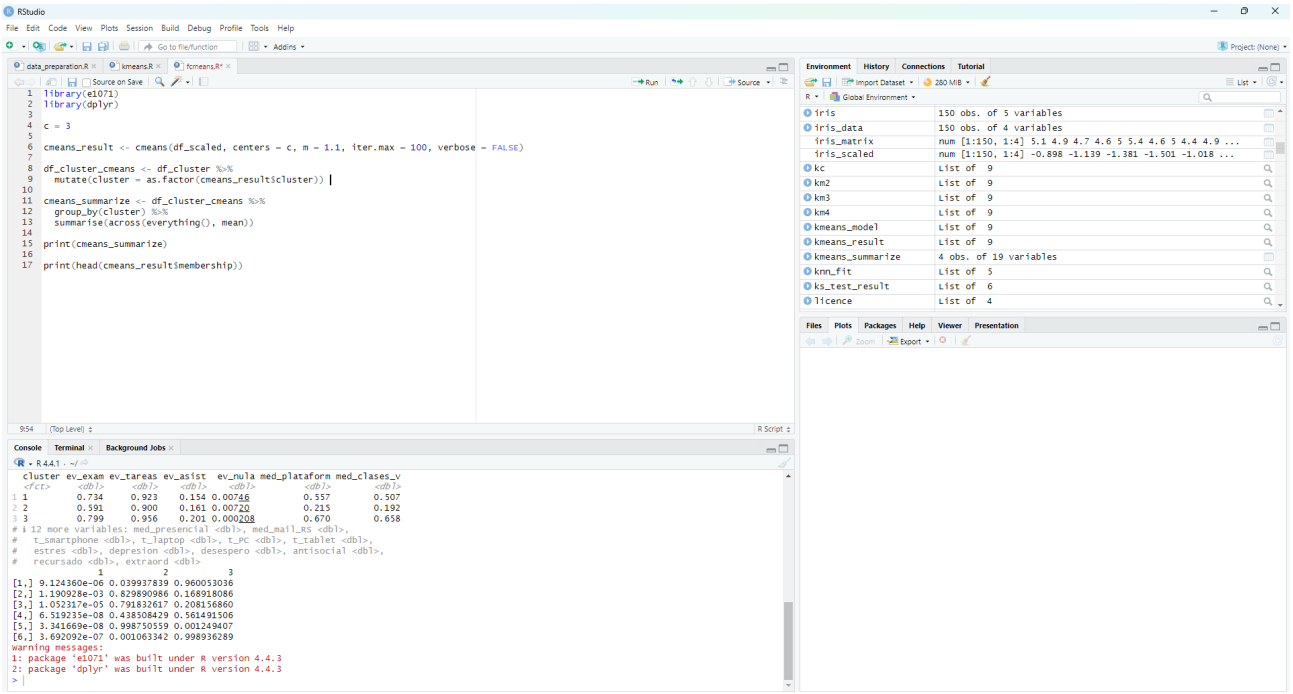

Рис. 2.1 – Інтерфейси середовища RStudio 2024.09.0

Разом із тим, порівняно з Python, мова R є більш вузькоспеціалізованою і розробленою саме для статистики та аналізу даних. Якщо Python потребує додаткових бібліотек для виконання навіть базових статистичних розрахунків, то R має їх інтегрованими «за замовчуванням». Крім того, статистичні пакети в R, зокрема для кластеризації, регресійного аналізу та візуалізації, часто є більш глибоко розробленими і спеціально адаптованими під потреби дослідників. Це робить R більш зручним інструментом для академічних і наукових робіт, де важлива точність і наявність готових методів. Висновок полягає в тому, що хоча Python є універсальною мовою програмування, R краще підходить для завдань кластеризації та статистичного аналізу, оскільки надає досліднику більш природне й ефективне середовище для проведення експериментів без необхідності додаткового налаштування.

Таким чином, використання R дає можливість реалізувати експериментальну частину дослідження максимально ефективно, спираючись на готові перевірені бібліотеки, а також забезпечити відтворюваність результатів

завдяки відкритості й поширеності цієї мови програмування в академічному середовищі.

Для зручної роботи з мовою R у даній роботі використовується інтегроване середовище розробки RStudio [30] у версії 2024.09.0 (рис. 2.1). Це середовище є найбільш поширеним інструментом серед користувачів R завдяки своїй зручності, стабільності та широкому набору можливостей. RStudio надає користувачу зручний редактор коду з підсвічуванням синтаксису та автоматичним доповненням, інтегрований термінал для виконання команд, інструменти для покрокового налагодження коду, панель для візуалізації результатів (графіків, таблиць, інтерактивних елементів), вбудоване управління пакетами та середовищами проектів.

Версія 2024.09.0 [31] є оновленою та містить покращення в продуктивності, сумісності з новими версіями R та розширені можливості роботи з великими наборами даних. Вона дозволяє створювати RMarkdown документи для поєднання коду, пояснень та результатів у єдиному звіті, що є важливою складовою академічних досліджень.

2.4. Підготовка даних до аналізу

Для проведення дослідження з використанням методів чіткої та нечіткої кластеризації необхідно здійснити повноцінну підготовку даних, оскільки якість результатів безпосередньо залежить від правильності обробки вихідної інформації. В основі даної роботи використовується набір даних ENAPE 2021 (Encuesta Nacional sobre Acceso y Permanencia en la Educación), проведений мексиканським інститутом статистики (INEGI). Він містить велику кількість інформації про освітній доступ, рівень залученості та соціально-економічні характеристики населення.

Цільова сукупність ENAPE 2021 охоплює осіб у віці від 0 до 29 років, що дозволяє проаналізувати повний освітній цикл — від дошкільної освіти до вищої школи. Особливістю дослідження є багатовимірність даних: інформація

збирається як на рівні домогосподарства, так і на рівні окремої особи, що забезпечує можливість дослідження взаємозв'язків між соціальними умовами, доступом до технологій та індивідуальною освітньою траєкторією.

У роботі використовуються три взаємопов'язані частини датасету.

«df01_Dataset_ENAPE.csv» — містить соціально-демографічні характеристики та інформацію про матеріально-технічне забезпечення домогосподарств. Сюди входять такі змінні, як кількість мешканців, стать і вік членів сім'ї, наявність комп'ютерів, ноутбуків, планшетів, смартфонів, доступу до інтернету, а також суб'єктивні оцінки ролі освіти у підвищенні рівня життя та працевлаштуванні. Цей файл дозволяє вивчити загальні умови, у яких формується освітній доступ.

«df02_Dataset_ENAPE.csv» — деталізує інформацію про кожну особу, яка навчається. Тут зібрані відомості про вік, стать, рівень освіти, факт завершення попереднього навчального року, причини вибуття або пропуску навчання, використання додаткових освітніх послуг (репетиторство, перездачі, повторні курси), форми оцінювання у школах (екзамени, проєкти, участь у класі, відвідування), а також способи організації навчання — дистанційно, очно чи змішано. Крім того, у файлі містяться дані про використання цифрових платформ та навчальних матеріалів, залучення технічних засобів (смартфони, ноутбуки, телевізори тощо) та рівень підтримки з боку сім'ї чи викладачів.

«df03_Dataset_ENAPE.csv» — є результатом інтеграції попередніх двох файлів на основі унікального ідентифікатора домогосподарства (FOLIO). У цій таблиці розміщені 32 343 записи та 67 ознак, що об'єднують характеристики домогосподарств із характеристиками окремих осіб. Такий формат зручний для кластеризації, оскільки дозволяє комплексно враховувати вплив умов життя на навчальний процес та результати студентів.

Для роботи використовувався набір бібліотек readr і dplyr, які забезпечують зручне завантаження, обробку та трансформацію даних. Основним джерелом був файл «df03_Dataset_ENAPE.csv» (рис. 2.2), що містить інтегрований набір даних із дослідження ENAPE 2021, у якому інформація про

респондентів поєднана з характеристиками домогосподарств. Саме цей файл дозволив працювати з кожним індивідом як основною одиницею аналізу.

Для побудови моделі кластеризації було здійснено відбір релевантних змінних, що найбільш повно відображають освітні процеси та психоемоційний стан учнів. До переліку відібраних характеристик увійшли:

- способи оцінювання, атрибути "ev_exam", "ev_tareas", "ev_asist", "ev_nula" (оцінювання через іспити, виконання домашніх завдань, відвідуваність чи відсутність оцінювання);
- формати навчання, атрибути "med_plataform", "med_clases_v", "med_presencial", "med_mail_RS" (традиційні, дистанційні, змішані, із застосуванням цифрових платформ або електронної пошти);
- технології, використані для навчання, атрибути "t_smartphone", "t_laptop", "t_PC", "t_tablet", "t_TV" (смартфони, ноутбуки, стаціонарні ПК, планшети, телевізори);
- психоемоційні стани учнів, атрибути "estres", "depresion", "desespero", "antisocial" (стрес, депресія, відчуття відчаю, труднощі у соціалізації);
- результативність та участь у навчанні, атрибути "recursado", "extraord", "t_horas" (факт повторного проходження предметів, складання додаткових іспитів, кількість годин, витрачених на навчальні завдання).

Наступним етапом стала фільтрація даних, метою якої було усунення некоректних або неінформативних значень. У випадку більшості змінних кодові позначення 0 та 9 використовувалися як службові — вони не відображали фактичної поведінки чи стану учня, а тому були видалені. Після цього для всіх бінарних змінних було проведено уніфікацію: значення кодувалися у форматі 0/1, де 1 означало наявність певної характеристики (наприклад, використання ноутбука чи наявність депресії), а 0 — її відсутність.

Останнім кроком стало масштабування даних за допомогою функції scale(). Це перетворення привело всі числові ознаки до однакового масштабу (нульове середнє та одинична дисперсія). Така процедура є необхідною умовою для алгоритмів кластеризації, оскільки вона запобігає ситуації, коли змінні з

більшими числовими діапазонами (наприклад, t_horas) непропорційно впливають на процес формування кластерів.

data_preparation.R

Source on Save

```r
library(readr)
library(dplyr)

df03 <- read.csv("df03_Dataset_ENAPE.csv")
selected_vars <- c(
  # Способи оцінювання
  "ev_exam", "ev_tareas", "ev_asist", "ev_nula",

  # Формати навчання
  "med_plataform", "med_clases_v", "med_presencial", "med_mail_RS",

  # Технології, використані для навчання
  "t_smartphone", "t_laptop", "t_PC", "t_tablet", "t_TV",

  # Психоемоційні стани
  "estres", "depresion", "desespero", "antisocial",

  # Результати та участь
  "recursado", "extraord", "t_horas"
)

df_cluster <- df03 %>%
  select(all_of(selected_vars)) %>%
  filter(across(-t_horas, ~ !.x %in% c(0, 9))) %>%
  mutate(across(-t_horas, ~ ifelse(.x == 1, 1, 0)))

df_scaled <- scale(df_cluster)
```

Рис. 2.2 – Скрипт на R для підготовки даних до кластеризації

У випадку кластеризації важливо, щоб обрані змінні були якомога менш корельовані між собою, адже сильна кореляція свідчить про надлишковість інформації: дві змінні фактично описують один і той самий аспект явища. Це може призвести до спотворення результатів, коли кластери формуватимуться не за реальними відмінностями у даних, а через повторювані фактори. Тому додатковим етапом підготовки даних став аналіз кореляційної залежності між атрибутами. Для цього було обчислено матрицю парних коефіцієнтів кореляції Пірсона за допомогою функції cor(). Використовувалася опція use = "complete.obs", яка забезпечує коректний підрахунок у випадку пропущених

значень, а також параметр method = "pearson", що дозволяє оцінити лінійний взаємозв'язок між змінними. Виконаний код зображений на рисунку 2.3. Для візуалізації результатів застосовано бібліотеку corrplot, яка забезпечує зручне графічне представлення кореляційної структури даних (рис. 2.4).

data_preparation.R | data_correlation.R

Source on Save

```
1  library(corrplot)
2
3
4  cor_matrix <- cor(df_scaled, use = "complete.obs", method = "pearson")
5
6
7  print(round(cor_matrix, 2))
8
9  corrplot(cor_matrix, method = "color", type = "upper",
10         tl.col = "black", tl.srt = 45,
11         col = colorRampPalette(c("blue", "white", "red"))(200),
12         title = "Матриця кореляцій", mar = c(0, 0, 2, 0))
```

Рис. 2.3 – Скрипт на R для аналізу кореляції відібраних даних

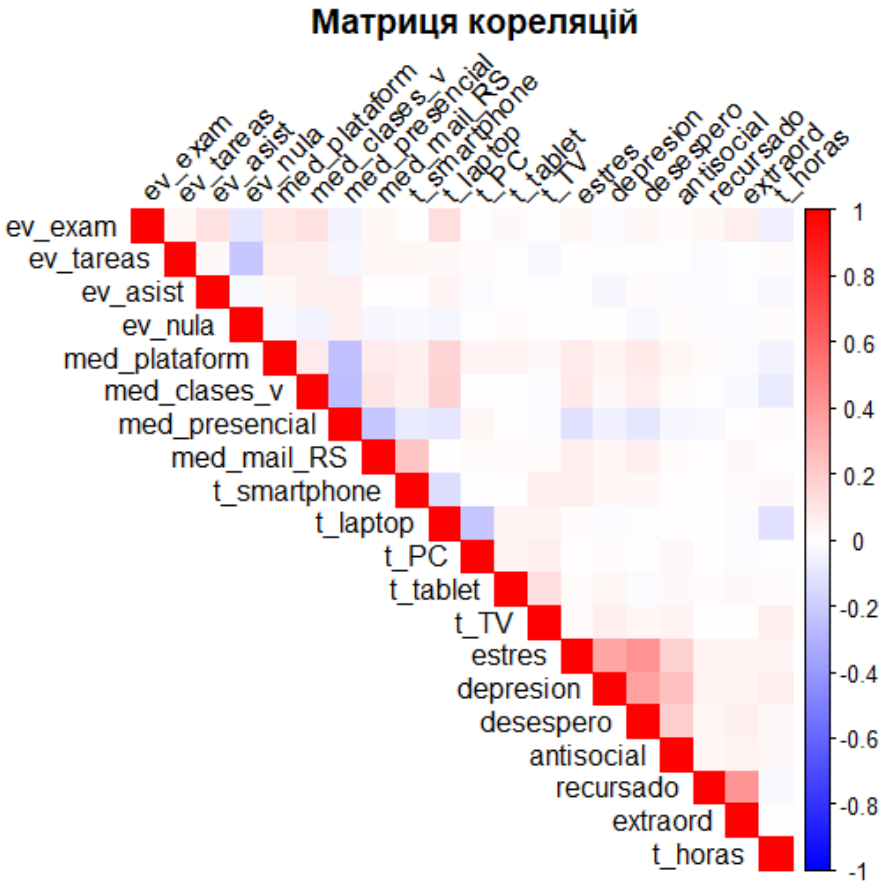

Рис. 2.4 – Візуалізація матриці кореляції відібраних атрибутів

Червоний колір відображає позитивну кореляцію, синій — негативну, а білий відповідає відсутності суттєвого зв’язку. Інтенсивність кольору відповідає силі кореляції. Отримані результати показали такі тенденції – між змінними, що відображають способи оцінювання (ev_exam, ev_tareas, ev_asist) спостерігається мінімальний позитивний взаємозв’язок, що вказує на певну схожість у використанні цих атрибутів. Показники, пов’язані з цифровими технологіями (t_smartphone, t_laptop, t_PC, t_tablet, t_TV), утворюють групу з мінімальним кореляційним зв’язком: половина мають мінімальний позитивний зв’язок, а інша половина – негативний. Психоемоційні характеристики (estres, depresion, desespero, antisocial) виявили між собою позитивну кореляцію, що може свідчити про наявність комплексного впливу стресових факторів на учнів. Вони описують єдину латентну характеристику — психологічний стан, тому їх одночасне використання виправдане для точнішої сегментації. Змінна t_horas (кількість годин, витрачених на навчання) не виявила сильних кореляцій із більшістю інших атрибутів, що робить її незалежним чинником у подальшій кластеризації.

Аналіз кореляцій дав змогу виявити групи взаємопов’язаних змінних та підтвердити доцільність їх використання у подальшому моделюванні. Це також дозволяє зменшити ризик надлишковості даних та уникнути дублювання інформації під час побудови кластерної моделі.

Таким чином, на виході було сформовано чистий та збалансований набір даних df_scaled, готовий до застосування методів кластерного аналізу. Ця підготовка гарантувала коректність подальших експериментів і забезпечила підґрунтя для об’єктивного порівняння чітких та нечітких алгоритмів кластеризації.

2.5. Реалізація методів кластеризації

У цьому підпункті описано практичну реалізацію k-means (рис. 2.5) та FCM у R для сформованого та стандартизованого набору ознак df_scaled.

data_preparation.R | data_correlation.R | kmeans.R | kmeans_vizualization.R

Source on Save

```r
set.seed(2)
kmeans_result <- kmeans(df_scaled, centers = 3, nstart = 25)

df_cluster_kmeans <- df_cluster %>%
  mutate(cluster = as.factor(kmeans_result$cluster))

kmeans_summarize <- df_cluster_kmeans %>%
  group_by(cluster) %>%
  summarise(across(everything(), mean))

print(kmeans_summarize)
print(kmeans_result$tot.withinss)
```

Рис. 2.5 – Скрипт на R для виконання методу k-means

Насамперед необхідно зафіксувати випадковість алгоритму. Set.seed() забезпечує відтворюваність: k-means стартує з випадкових центрів, тому без фіксації зерна різні запуски даватимуть різні розбиття. Для перевірки стабільності моделі радять прогнати кілька фіксованих сідів (напр., 1-20) і порівняти метрики. kmeans(df_scaled, centers = 3, nstart = 25) запускає базову реалізацію алгоритму Hartigan–Wong (за замовчуванням у stats::kmeans). Вхід — масив стандартизованих ознак (df_scaled). Стандартизація потрібна, щоб ознака t_horas (години/тиждень) не «перетягувала» відстані на себе й не домінувала над бінарними 0/1-ознаками. Кількість центрів описується за допомогою параметра centers. Аргумент nstart = 25 означає, що алгоритм запуститься 25 разів із різними випадковими ініціалізаціями центрів і поверне розв’язок з найменшою сумою внутрішньокластерних квадратів (WSS). Це простий спосіб уникнути «невдалих» локальних мінімумів. Для складніших даних варто збільшувати це значення до 50–100. За потреби можна явно задати iter.max — максимальну кількість ітерацій, типово 10. Однак, якщо алгоритм не встигає збігатися можна підвищити це значення, скажімо, до 100. Також можна змінити algorithm на "Lloyd" чи "MacQueen", але в більшості прикладів Hartigan–Wong сходиться швидше.

Для наочного представлення результатів кластеризації методом k-means було використано підхід із застосуванням головних компонент (PCA). Цей метод

дозволяє зменшити розмірність даних, які складаються з великої кількості змінних, і візуалізувати їх у двовимірному або тривимірному просторі. Оскільки алгоритм k-means працює у багатовимірному просторі, пряме зображення розподілу об’єктів є неможливим. Тому застосування PCA дало можливість відобразити найсуттєвіші відмінності між об’єктами через перші головні компоненти, які пояснюють найбільшу частку дисперсії в даних.

Функція prcomp() була використана для обчислення головних компонент. Вона здійснила обертання координатної системи, так що нові осі (PC1, PC2, PC3, PC4 тощо) максимально пояснюють розкид даних. Важливо, що вхідні дані були вже стандартизовані (df_scaled), тому окремі змінні не домінували у процесі розрахунків. Результати PCA збережено у датафреймі df_pca, куди також додано інформацію про належність кожного об’єкта до кластера, отриману з результатів k-means.

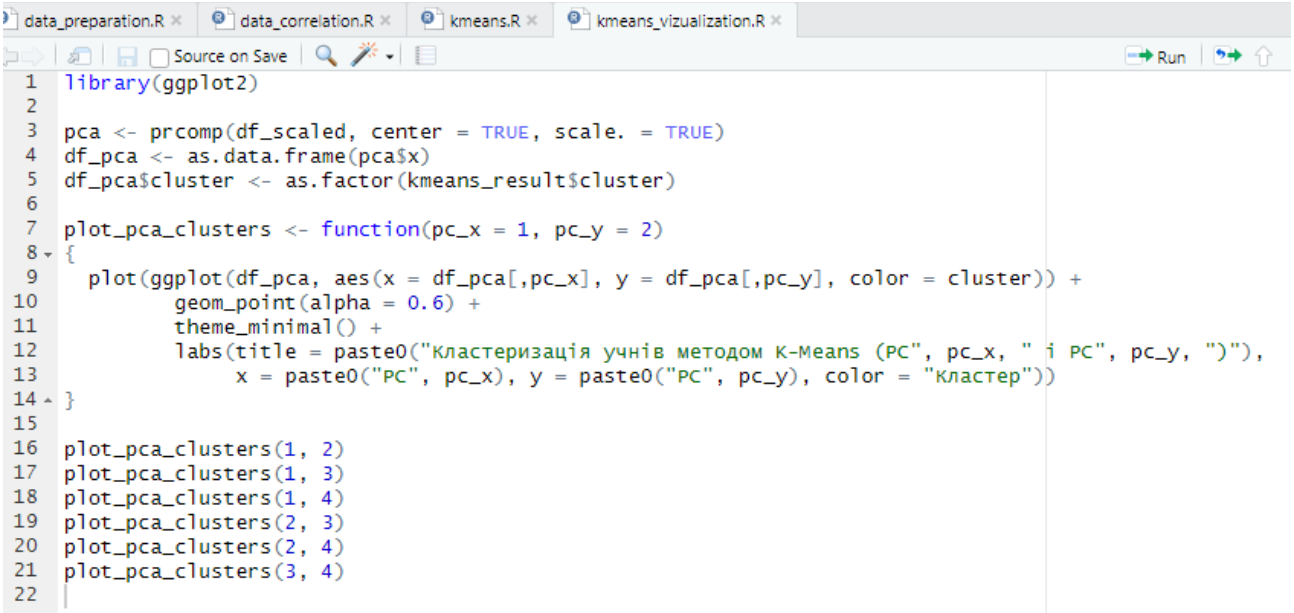

Рис. 2.6 – Скрипт на R для візуалізації результатів методу k-means

Побудова графіків здійснювалася з використанням пакету ggplot2 (рис. 2.6). Кожна точка на площині відповідала окремому студентові з набору даних, а її колір визначав кластер, до якого його відніс алгоритм. Завдяки цьому можна побачити, наскільки чітко розділяються кластери в просторі головних

компонент. Задано прозорість точок (alpha = 0.6), що дозволило уникнути сильного перекриття у випадку щільного розташування спостережень.

Було реалізовано функцію plot_pca_clusters, яка будує графіки для будь-якої пари головних компонент. Таким чином, вдалося отримати кілька проекцій даних: PC1–PC2, PC1–PC3, PC1–PC4, PC2–PC3, PC2–PC4 та PC3–PC4. Кожна з цих проекцій дає змогу оцінити розподіл об’єктів і виявити, чи залишаються кластери відносно чіткими за різних поєднань координат. У деяких випадках кластери можуть перекриватися, що свідчить про схожість поведінки частини студентів за обраними ознаками. Водночас у проекціях, де спостерігається чітке відокремлення груп, можна зробити висновок про стійкість кластеризації.

Важливим є те, що PCA дає змогу не лише візуалізувати результати, а й краще зрозуміти структуру даних. Якщо більша частина дисперсії пояснюється першими двома або трьома компонентами, це свідчить про високу інформативність отриманих графіків. Завдяки поєднанню методів PCA і k-means кластеризація стає більш зрозумілою: можна інтерпретувати не лише числові середні значення у кластерах, а й безпосередньо побачити, як об’єкти групуються у просторі даних.

На рисунку 2.7 зображений один з графіків візуалізації результатів методу k-means. Детальніше експериментальні сценарії описуються у підпункті 2.7.

Реалізація методу нечіткої кластеризації (FCM) здійснювалась за допомогою функції cmeans() із пакету e1071 (рис. 2.8). У коді було встановлено кількість кластерів c = 3, що відповідає кількості груп, визначених у попередньому методі k-means. Параметр m = 1.1 відповідає за рівень розмитості кластерів (fuzzification parameter). Чим ближче значення m до 1, тим чіткіше відбувається розподіл об’єктів, а чим воно більше — тим більш рівномірно розподіляються вагові коефіцієнти між кластерами. У даному випадку було обрано значення, наближене до 1, що дозволяє зберегти певну гнучкість, але водночас уникнути надмірної розмитості. Параметр iter.max = 100 обмежує кількість ітерацій, які виконує алгоритм для пошуку стійкого розв’язку.

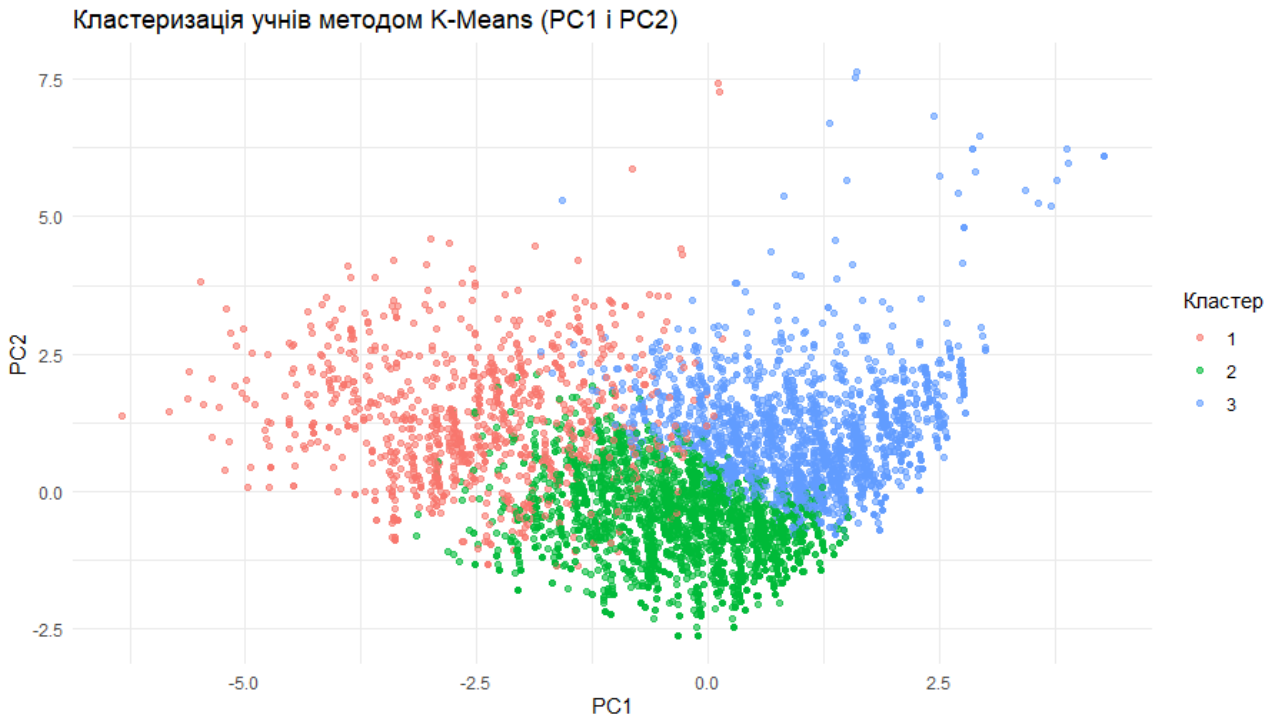

Рис. 2.7 – Візуалізація результатів методу k-means

```
data_preparation.R | data_correlation.R | fcmeans.R | kmeans.R | kme
Source on Save
1  library(e1071)
2  library(dplyr)
3
4  c = 3
5
6  cmeans_result <- cmeans(df_scaled, centers = c, m = 1.1,
7                          iter.max = 100, verbose = FALSE)
8
9  df_cluster_cmeans <- df_cluster %>%
10   mutate(cluster = as.factor(cmeans_result$cluster))
11
12 cmeans_summarize <- df_cluster_cmeans %>%
13   group_by(cluster) %>%
14   summarise(across(everything(), mean))
15
16 print(cmeans_summarize)
17
18 print(head(cmeans_result$membership))
```

Рис. 2.8 – Скрипт на R для виконання методу FCM

Результати кластеризації зберігаються у змінній cmeans_result. Подібно до випадку з k-means, створюється новий датафрейм df_cluster_cmeans, куди додається стовпець із номерами кластерів для кожного спостереження. Після

цього було виконано агрегацію з використанням функції summarise(), яка дозволила підрахувати середні значення змінних у кожному кластері. Це дає змогу описати узагальнений профіль кожної групи студентів — наприклад, наскільки часто вони користуються певними технологіями, які методи оцінювання переважають, чи є в них проблеми психоемоційного характеру.

Ключовою відмінністю FCM є матриця ступенів належності (cmeans_result$membership), де кожен рядок відповідає окремому студенту, а кожен стовпець — кластеру. Значення в цій матриці лежать у діапазоні від 0 до 1 і відображають, з якою ймовірністю або мірою об’єкт належить до певного кластера. Наприклад, студент може мати 0.7 у першому кластері, 0.2 у другому та 0.1 у третьому. На відміну від k-means, де такий студент потрапив би лише до одного кластера, FCM відображає його проміжне положення між кількома групами.

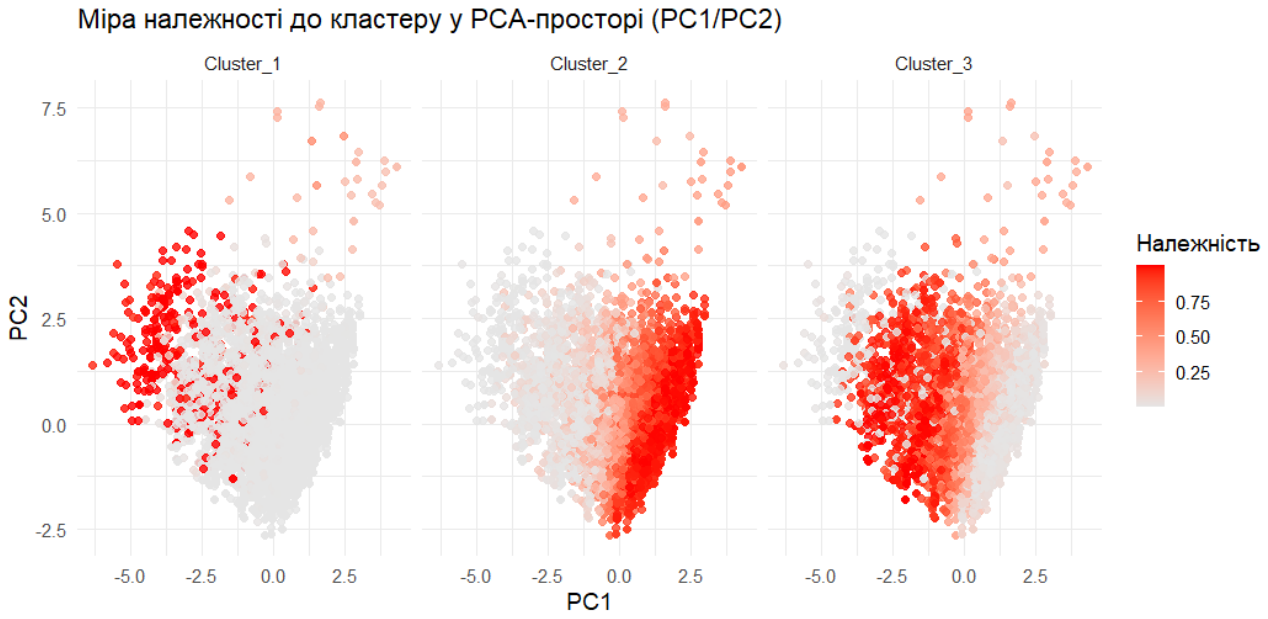

Рис. 2.9 – Візуалізація результатів методу FCM

FCM візуалізація реалізується аналогічно до методу k-means, однак через відображення ступеня належності до кожного з кластерів (рис. 2.9): точки фарбуються градієнтом від світлого до насиченого червоного залежно від сили членства. Для кожного кластера будується окрема панель (facet), що дозволяє

простежити, наскільки чітко або розмито розподілені дані у PCA-просторі. Таким чином, k-means демонструє дискретний розподіл, тоді як FCM надає гнучку картину, що враховує перехідні випадки і дозволяє побачити зони перетину між кластерами.

Реалізація k-means є більш компактною та менш вимогливою в обчисленнях, що робить її зручною для початкового аналізу. Натомість реалізація FCM дещо складніша й вимагає додаткових параметрів, але надає глибшу інформацію, зокрема через матрицю ступенів належності, що є суттєвою перевагою при роботі з реальними даними. Порівняльний аналіз реалізації методів на R показано на таблиці 1.

Таблиця 1. Порівняння реалізації методів k-means та FCM на R

Критерій	**K-means**	**Fuzzy c-means**
Пакети R	Використано базову функцію kmeans() (належить до стандартної бібліотеки R, додаткових пакетів не потрібно)	Використано функцію cmeans() з пакету e1071, що вимагає попереднього встановлення і підключення
Кількість рядків коду	Близько 10–12 рядків: ініціалізація, кластеризація, додавання кластерів у датафрейм, обчислення середніх значень	Близько 12–14 рядків: додатково потрібно вказати параметр fuzziness (m), а також проаналізувати матрицю належностей
Налаштовувані параметри	Основні: centers (кількість кластерів), nstart (кількість запусків для кращого результату), iter.max (максимальна кількість ітерацій)	Основні: centers (кількість кластерів), m (ступінь розмитості), iter.max (максимальна кількість ітерацій), verbose (вивід інформації під час роботи)
Тип результатів	Чіткий розподіл: кожен об’єкт має лише один кластер	Нечіткий розподіл: кожен об’єкт має вагові коефіцієнти належності до всіх кластерів
Вивід додаткових даних	tot.withinss – показник якості (сумарна відстань всередині кластерів)	membership – матриця належності (ступені приналежності кожного об’єкта до всіх кластерів)
Обчислювальні витрати	Менші: алгоритм швидший через простішу оптимізацію	Трохи більші: через роботу з матрицею ваг та розмитим

		розподілом
Придатність до практичного застосування	Добре підходить для швидкої кластеризації та базового аналізу	Краще підходить для складних випадків, коли об’єкти не належать строго лише до однієї групи

2.6. Оцінювання якості кластеризації

Правильний вибір кількості кластерів та перевірка отриманих результатів є ключовими для достовірності висновків. У наведеному коді на рисунку 2.10 для методу k-means використано один із найпоширеніших підходів — силуетний коефіцієнт (Silhouette). Значення коефіцієнта варіюється від -1 до 1: високі значення (близькі до 1) вказують на добре відокремлені та чітко сформовані кластери, значення біля 0 — на об’єкти, які знаходяться на межі кластерів, а від’ємні значення сигналізують, що об’єкт, ймовірно, віднесений не до того кластера.

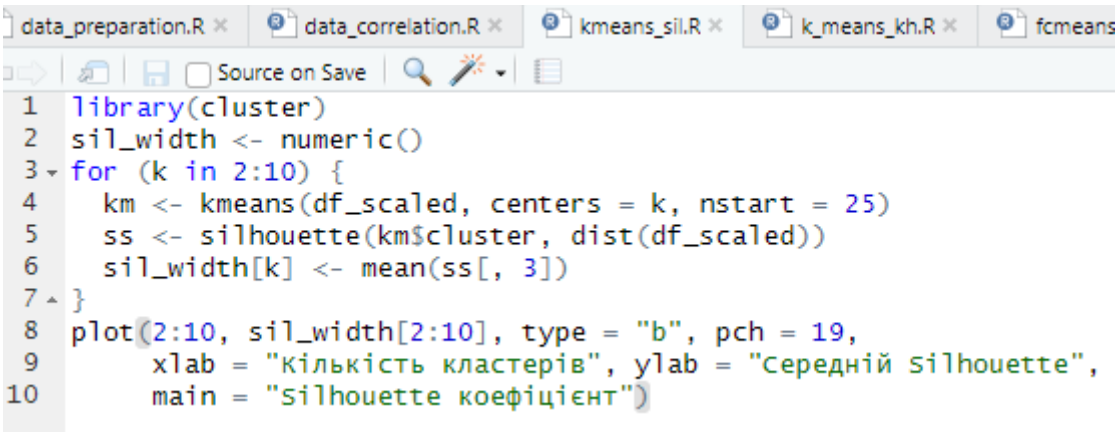

Рис. 2.10 – Скрипт на R для оцінювання якості кластеризації методу k-means за допомогою силуетного коефіцієнта

У коді використано цикл for, який перебирає кількість кластерів від 2 до 10. Для кожного значення виконується кластеризація методом k-means з фіксованою кількістю центрів (centers = k) та з багатьма випадковими ініціалізаціями (nstart = 25) для підвищення стабільності результату. Далі обчислюється силуетний коефіцієнт за допомогою функції silhouette(), яка

приймає як аргументи номери кластерів для кожної точки (km$cluster) та матрицю відстаней між об’єктами (dist(df_scaled)). З отриманого масиву даних силуету вибирається третій стовпець (ss[, 3]), що містить числове значення коефіцієнта для кожного об’єкта, і далі обчислюється середнє значення цього показника для всього розбиття. Результати зберігаються у векторі sil_width, а після завершення циклу будується графік залежності середнього силуетного коефіцієнта від кількості кластерів. Таким чином, можна візуально визначити оптимальну кількість кластерів — це точка, де значення Silhouette досягає максимуму.

Як видно з рисунка 2.11, середній силуетний коефіцієнт показав найбільше значення для трьох кластерів, однак сам показник вийшов доволі низьким (0.15). Це означає, що всередині кластерів спостерігається значна внутрішня різнорідність, а відстані між кластерами не надто великі. Така ситуація є типовою для реальних соціально-освітніх даних на кшталт ENAPE 2021, де характеристики учнів і домогосподарств не утворюють чітко відокремлених груп, а значно перекриваються між собою. На відміну від синтетичних даних чи технічних задач, де часто можна отримати добре розділені кластери з високим силуетним коефіцієнтом (>0.5), у складних емпіричних наборах значення 0.1–0.2 вважається прийнятним і вказує на наявність латентних структур, що відображають лише часткові закономірності у вибірці. Таким чином, навіть відносно низький коефіцієнт не означає, що кластеризація є некорисною, а лише підкреслює складність та багатовимірність даних. Тому значення $\boldsymbol{k = 3}$ є прийнятним в такій ситуації.

Також було проведена оцінка якості кластеризації за допомогою індексу Калінскі-Харабаза (рис. 2.12). Спочатку задається початкове зерно генератора випадкових чисел для відтворюваності результатів, а далі в циклі так само перебирається кількість кластерів від 2 до 10. Отримані розбиття передаються у функцію intCriteria() з пакету clusterCrit, яка обчислює різні внутрішні індекси кластеризації, зокрема індекс Калінскі-Харабаза. Його значення зберігаються у векторі ch_values, після чого будується графік залежності цього індексу від

кількості кластерів. Індекс Калінскі-Харабаза оцінює співвідношення між дисперсією між кластерами та всередині кластерів, і чим більше його значення, тим краще кластери розділені. Таким чином, графік на рисунку 2.13 допомагає вибрати оптимальне число кластерів, орієнтуючись на пікове значення індексу.

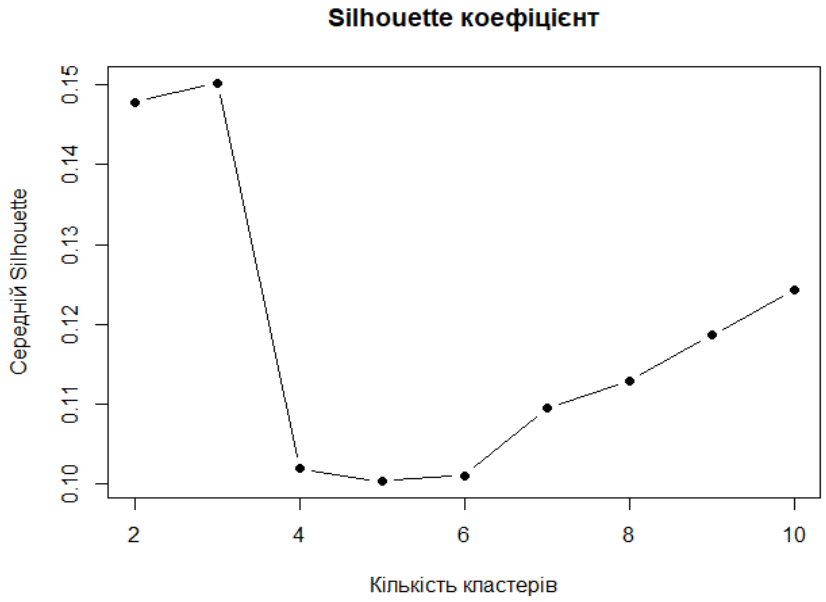

Рис. 2.11 – Значення силуетного коефіцієнта зі зміною кількості кластерів у методі k-means

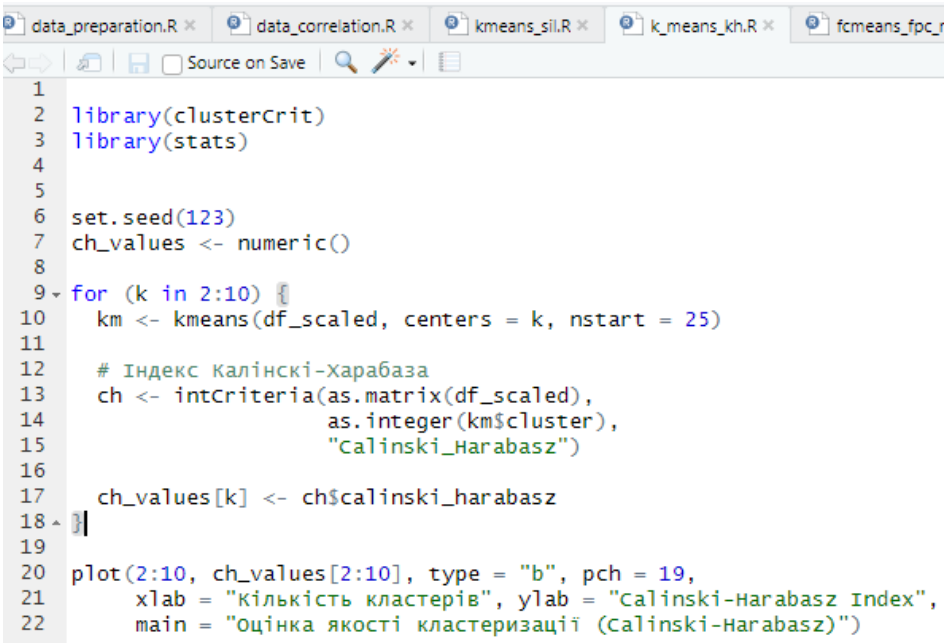

Рис. 2.12 – Скрипт на R для оцінювання якості кластеризації методу k-means за допомогою індексу Калінскі-Харабаза

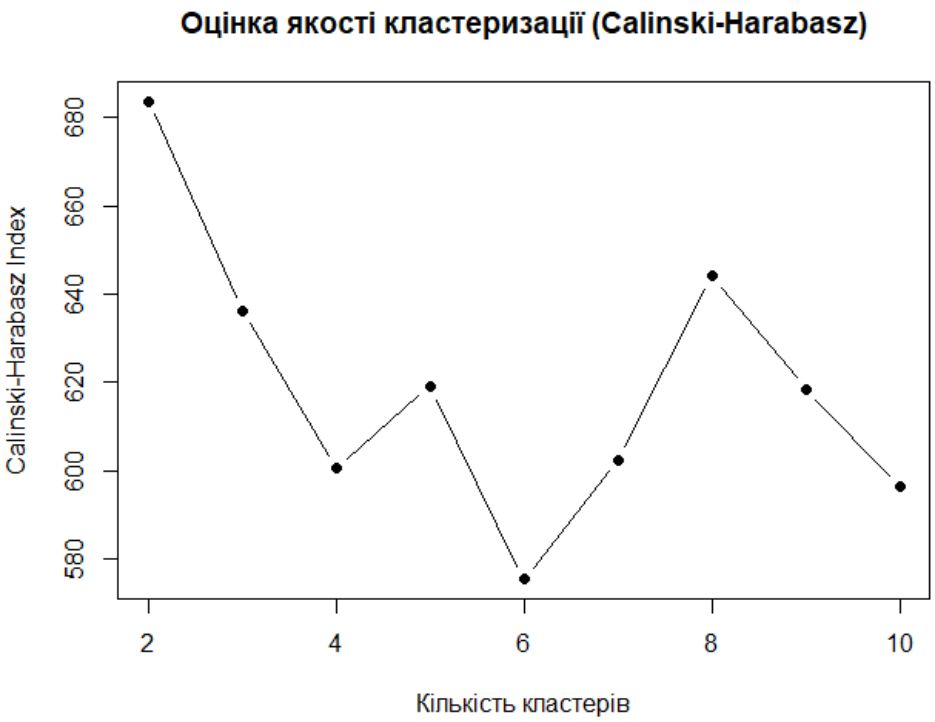

Рис. 2.13 – Значення індексу Калінскі-Харабаза зі зміною кількості кластерів у методі k-means

Індекс показав найбільше значення для 2 кластерів (680), що формально вказує на кращу роздільність даних саме за цією кількістю. Однак результати силуетного коефіцієнта демонструють, що найбільш оптимальною кількістю кластерів є 3, навіть попри те, що саме значення силуету є відносно низьким. Це пояснюється тим, що індекс Калінскі-Харабаза більше враховує глобальну компактність та віддаленість кластерів, тоді як силуетний коефіцієнт краще відображає локальну структуру та якість поділу кожної точки. Тому, враховуючи обидва критерії, можна зробити висновок, що розбиття на 3 кластери є більш адекватним для цих даних: воно забезпечує баланс між якістю групування та інтерпретованістю результатів, тоді як поділ на 2 кластери надто спрощує структуру вибірки.

Для оцінки якості кластеризації алгоритму FCM необхідно проаналізувати результати не тільки з різною заданою кількістю кластерів, а й з різними значеннями параметра фазифікації. У коді на рисунку 2.14 здійснюється оцінка

якості за допомогою індексу нечіткості кластеризації (Fuzzy Partition Coefficient, FPC).

data_preparation.R | data_correlation.R | kmeans_sil.R | k_means_kh.R | fcmeans_fpc_m.R

Source on Save

```r
library(e1071)


m_values <- seq(1.1, 2, by = 0.1)
fpcs <- numeric(length(m_values))

k <- 3

for (i in seq_along(m_values)) {
  m <- m_values[i]
  result <- cmeans(df_scaled, centers = k, m = m)
  fpcs[i] <- sum(result$membership^2) / nrow(df_scaled)
}

plot(m_values, fpcs, type = "b", lwd = 2, col = "steelblue",
     xlab = "Параметр нечіткості (m)", ylab = "FPC",
     main = "FPC при змінному параметрі m (кількість кластерів = 3)")
```

Рис. 2.14 – Скрипт на R для підбору параметра фазифікації методу FCM

Спочатку задається діапазон значень параметра нечіткості m від 1.1 до 2 з кроком 0.1. Далі для кожного значення m виконується кластеризація з фіксованою кількістю кластерів (k = 3), після чого обчислюється значення FPC, яке вимірює рівень визначеності належності точок до кластерів: чим ближче FPC до 1, тим чіткіше сформовані кластери, а значення, близькі до 1/k, свідчать про майже випадковий розподіл. Результати зберігаються у векторі й візуалізуються на графіку (рис. 2.15), що дозволяє побачити, при якому значенні параметра m досягається найвища якість кластеризації для обраної кількості кластерів.

У цьому випадку результати показують, що при параметрі фазифікації m = 1.1 значення FPC дорівнює 0.7, тоді як уже при m = 1.2 воно падає до 0.59 і далі стрімко зменшується нижче 0.4. Це означає, що при невеликому рівні нечіткості (наближеному до класичного k-means) кластери формуються досить чітко, і об’єкти мають відносно високу належність до конкретних груп. Коли ж m зростає, алгоритм починає «розмазувати» точки між кількома кластерами, що в

твоїх даних призводить до втрати чіткості та зниження інформативності результатів. Тому вибір m = 1.1 є найкращим для цієї задачі: він забезпечує баланс між невеликою нечіткістю (яка дозволяє врахувати складні випадки на межах кластерів) та збереженням високої якості кластеризації, яку підтверджує максимальне значення FPC.

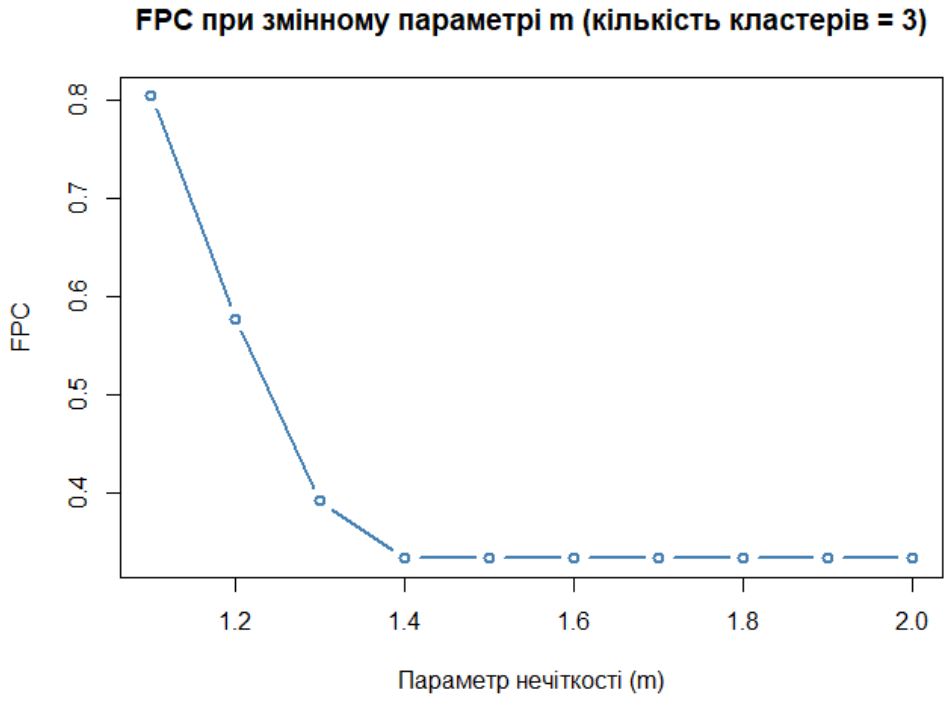

Рис. 2.15 – Графік зміни показника FPC залежно від параметру фазифікації m

Для оцінки кількості кластерів у FCM можна застосувати силуетний коефіцієнт, але з деякими змінами в роботі алгоритму. У класичному випадку силуетний коефіцієнт використовується для жорсткої кластеризації, де кожен об’єкт має однозначну належність до кластера. В алгоритмі FCM об’єкти належать до кількох кластерів з різними мірами належності. Тому для обчислення силуета зазвичай роблять так (рис. 2.16):

Отримують жорсткі мітки кластерів із FCM, тобто призначають кожну точку до того кластера, де вона має максимальну належність (apply(result$membership, 1, which.max)).

Використовують стандартний метод silhouette() з пакету cluster для цих

жорстких міток.

Отже, так і зроблено для даних ENAPE 2021. Результат дуже добре ілюструє ключову проблему силуетного коефіцієнта у випадку FCM. Оскільки цей метод у класичній формі працює лише з жорсткими кластерами, під час його застосування до нечіткої кластеризації доводиться штучно «зводити» належності до одного найбільшого значення (hard assignment). У результаті значна частина інформації про ступінь належності елементів втрачається. Наприклад, об’єкт, що має приблизно рівні належності до двох кластерів (0.51 і 0.49), у силуетному підході буде повністю віднесений до одного кластера, ніби друга належність зовсім не існувала. Це призводить до занадто «спрощеної» оцінки, яка не відображає реальної природи нечітких розподілів.

data_correlation.R | kmeans_sil.R | k_means_kh.R | fcmeans_fpc_m.R | fcmeans_sil.R

Source on Save

```
1  library(e1071)
2  library(cluster)
3
4  sil_width <- numeric()
5  for (k in 2:10) {
6    cmeans_result <- cmeans(df_scaled, centers = k, m = 1.2)
7    hard_clusters <- apply(cmeans_result$membership, 1, which.max)
8    ss <- silhouette(hard_clusters, dist(df_scaled))
9    sil_width[k] <- mean(ss[, 3])
10 }
11 plot(2:10, sil_width[2:10], type = "b", pch = 19,
12      xlab = "Кількість кластерів", ylab = "Середній Silhouette",
13      main = "Silhouette коефіцієнт")
14
```

Рис. 2.16 – Скрипт на R для оцінювання якості кластеризації методу FCM за допомогою силуетного коефіцієнта

Низькі значення коефіцієнта (0.075–0.1) показують, що кластери перетинаються і чітко відділити їх складно. Однак така оцінка не є однозначною, оскільки метод не враховує силу нечіткої належності, яка могла б показати, що навіть при сильному перекритті кластери все одно мають певну структуру. Тому силует у FCM можна використовувати лише як допоміжний інструмент. Однак, у цьому випадку він показав хороший результат для $\boldsymbol{k = 3}$ (рис. 2.17).

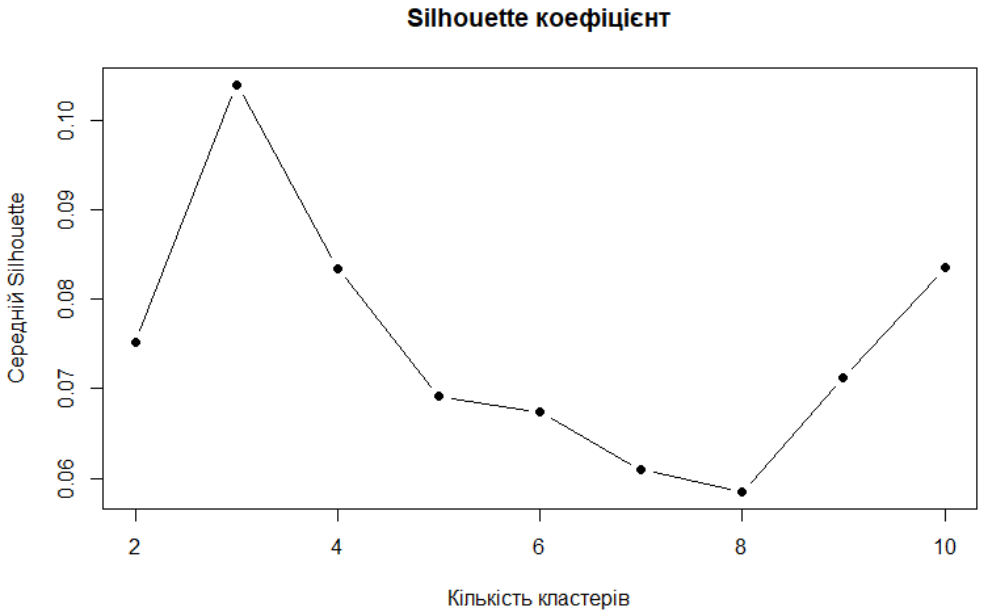

Рис. 2.17 – значення силуетного індексу зі зміною кількості кластерів для «жорсткого» FCM

У результаті оцінки якості кластеризації було вирішено, що варто покласти $\boldsymbol{k = 3}$ і $\boldsymbol{m = 1.1}$. Порівняльний аналіз способів оцінки якості кластеризації наведено на таблиці 2.

Таблиця 2.

Порівняння оцінки якості кластеризації для методів K-Means та FCM

Критерій	**K-Means**	**Fuzzy C-Means**
Силуетний коефіцієнт	0.15 (максимум серед усіх k)	≈0.1 (низький через втрату нечіткої інформації)
Індекс Калінскі-Харабаза	≈640 (стабільність при 2-3-ох кластерах)	Не застосовується безпосередньо
Fuzzy Partition Coefficient (FPC)	Не застосовується	0.7 (найвище при $\boldsymbol{m = 1.1}$)
Оптимальна кількість кластерів	3	3
Параметр нечіткості m	-	1.1

2.7. Експериментальні сценарії

Експериментальні сценарії включають візуалізації перебору різного значення параметру $\boldsymbol{k} \in \{\mathbf{2}, \ldots, \mathbf{5}\}$ та різні метрики відстаней (евклідова / мангеттенська). Для FCM найбільше уваги сконцентровано на параметрі фазифікації $\boldsymbol{m} \in [\mathbf{1.1}; \mathbf{2.0}]$.

Графік для трьох кластерів у просторі PC1/PC2 зображено на рисунку 2.5.3. Для решти просторів (PC1/PC3, PC1/PC4, PC2/PC3, PC2/PC4, PC3/PC4) графіки зображені на рисунках 2.18 – 2.22.

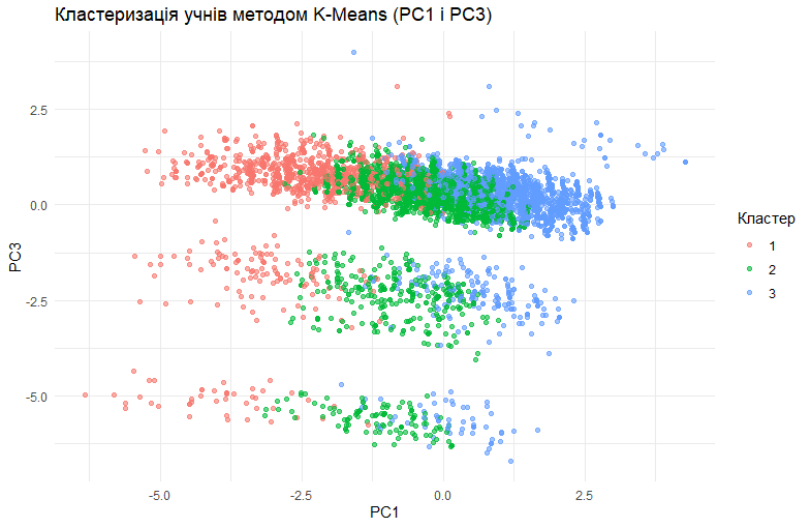

Рис. 2.18 – Результати кластеризації методом K-Means у просторі PC1/PC3

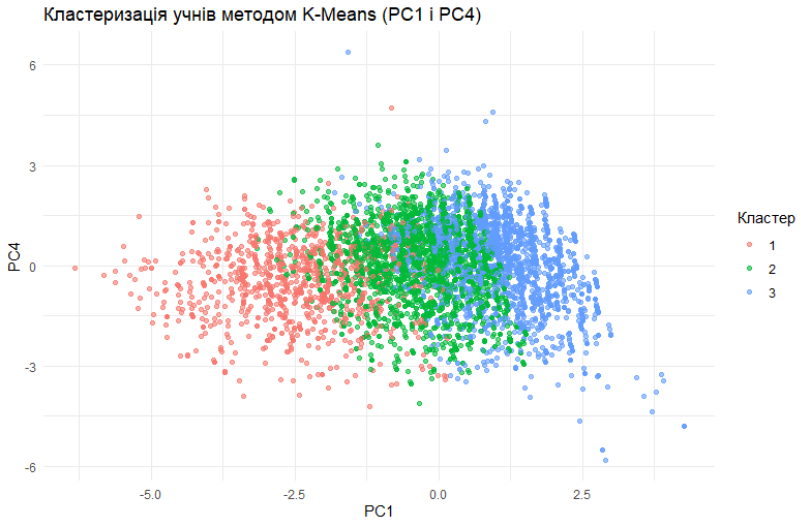

Рис. 2.19 – Результати кластеризації методом K-Means у просторі PC1/PC4

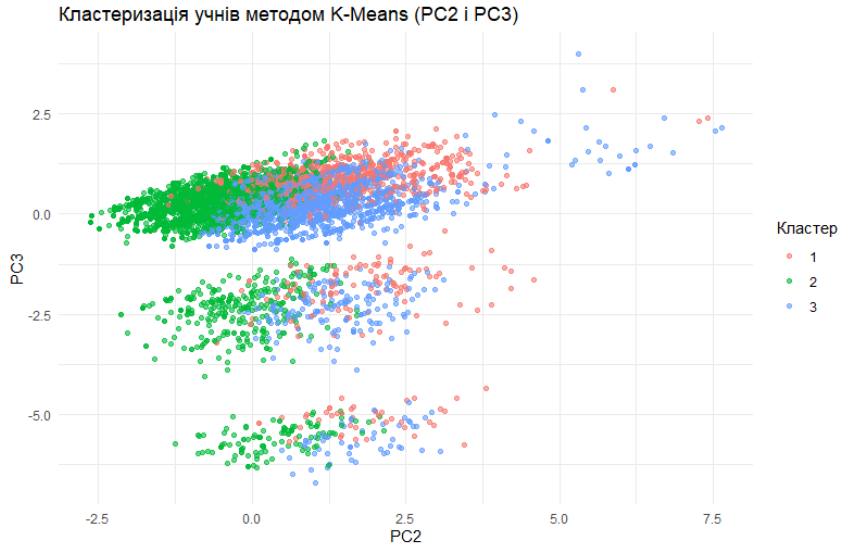

Рис. 2.20 – Результати кластеризації методом K-Means у просторі PC2/PC3

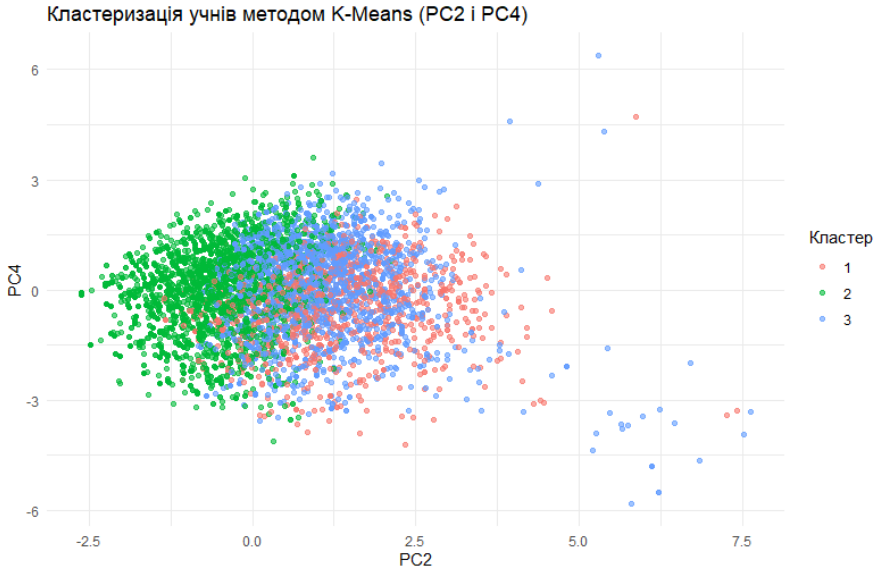

Рис. 2.21 – Результати кластеризації методом K-Means у просторі PC2/PC4

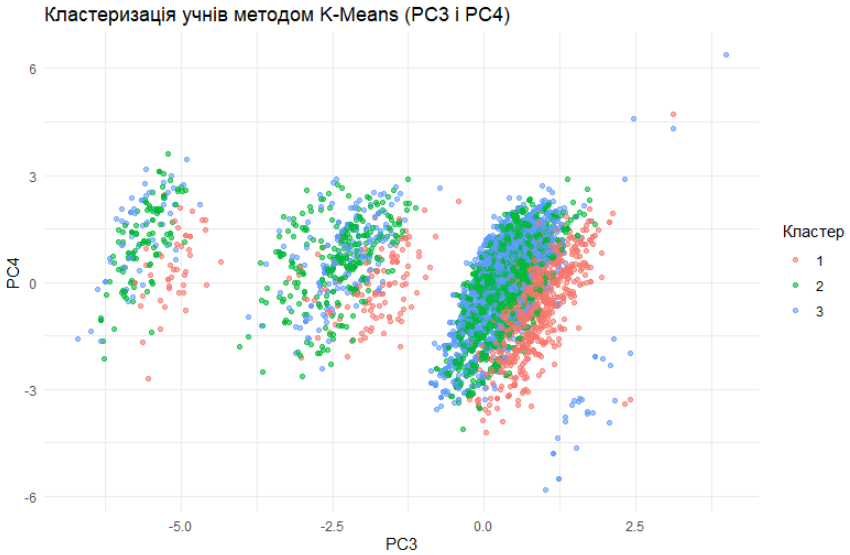

Рис. 2.22 – Результати кластеризації методом K-Means у просторі

PC3/PC4

Для чотирьох і п’яти кластерів у просторі PC1/PC2 наведені графіки на рисунках 2.23 – 2.24.

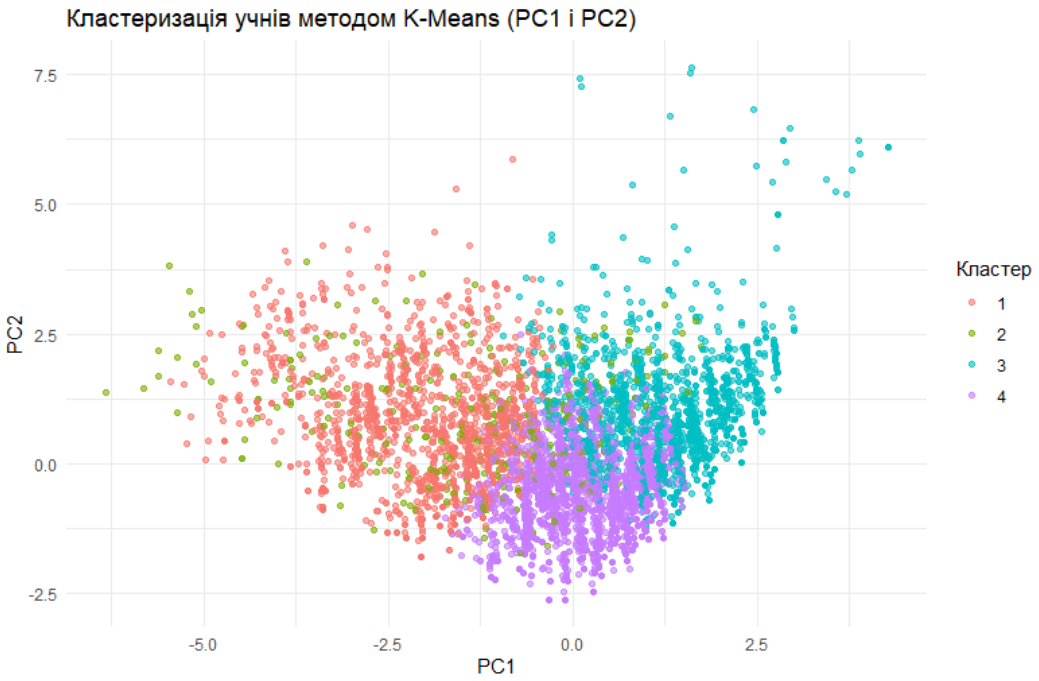

Рис. 2.23 – Результати кластеризації методом K-Means для 4-ох кластерів у просторі PC1/PC2

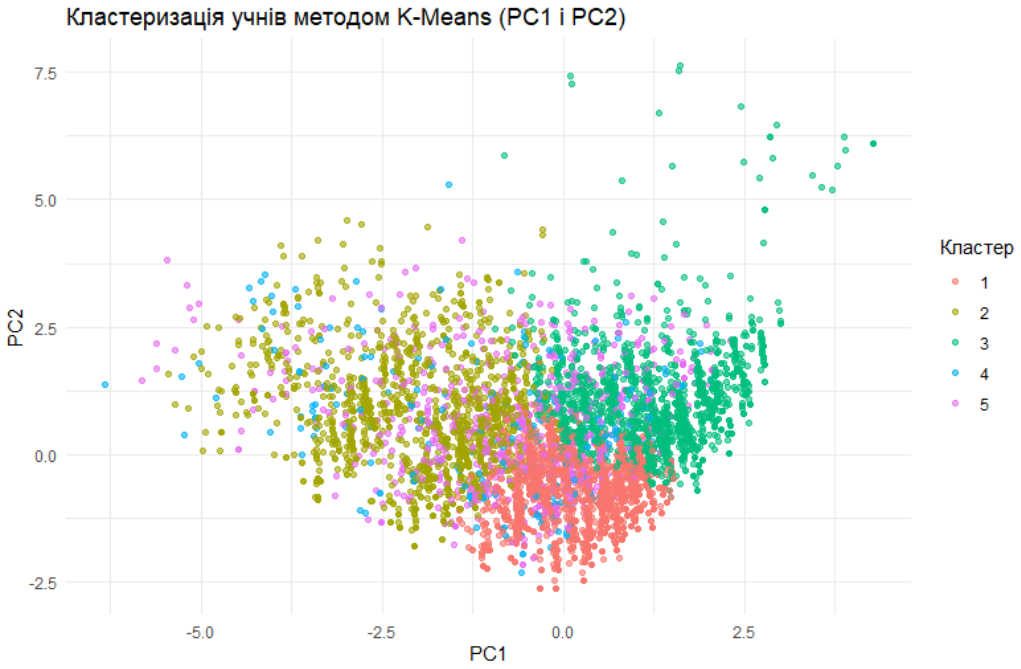

Рис. 2.24 – Результати кластеризації методом K-Means для 5-ох кластерів у просторі PC1/PC2

Для значення $\boldsymbol{m = 1.1}$ наведений рисунок 2.9. На рисунка 2.25. – 2.29 наведені графіки результатів кластеризації методом FCM із рештою значеннями параметру $\boldsymbol{m}$ у просторі PC1/PC2.

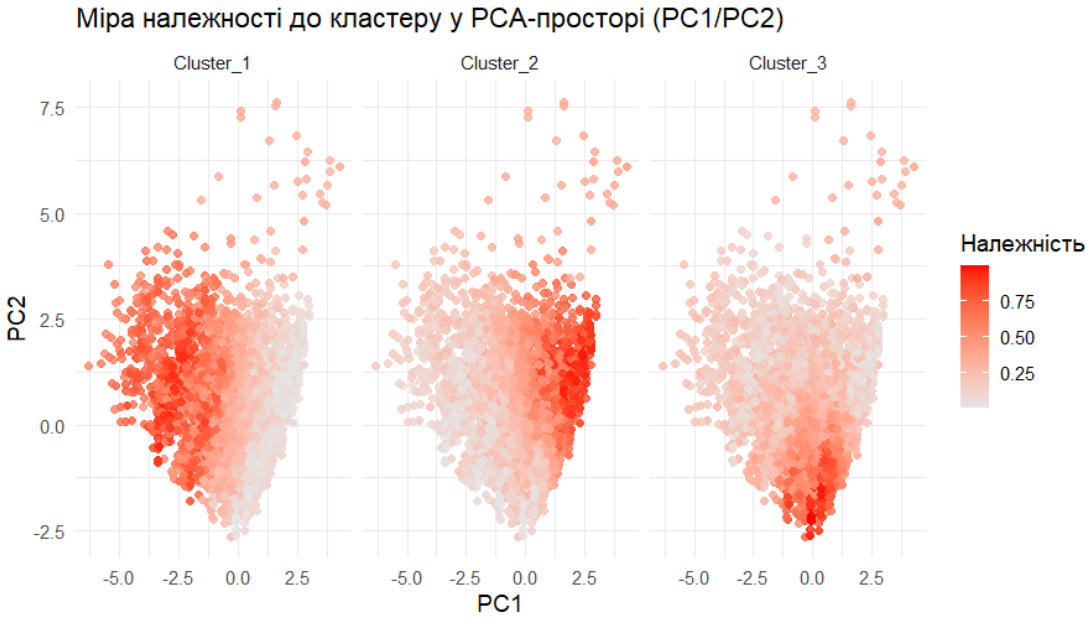

Рис. 2.25 – Результати кластеризації методом FCM при $\boldsymbol{m = 1.2}$

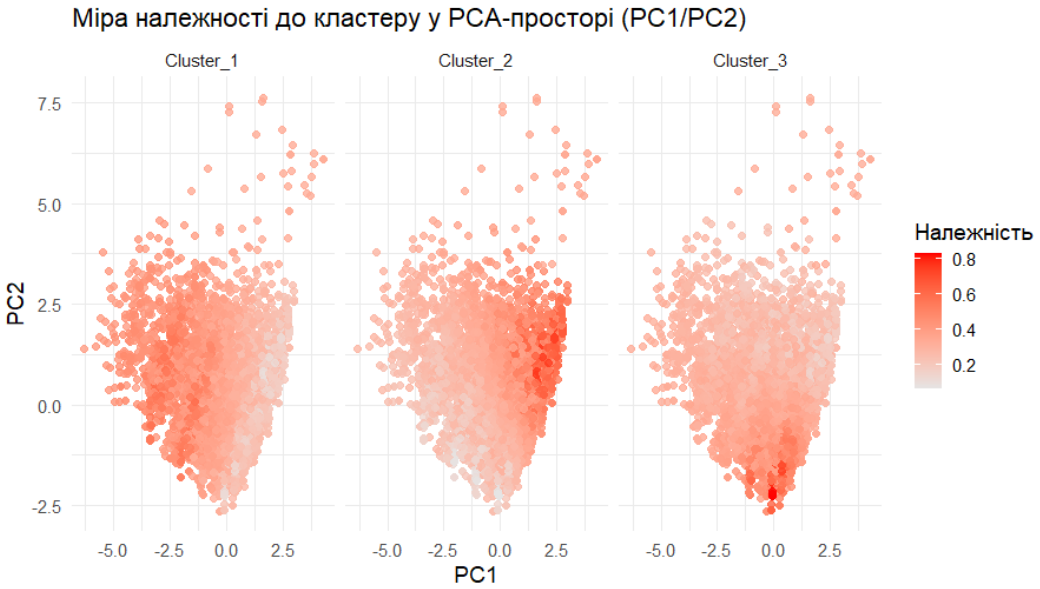

Рис. 2.26 – Результати кластеризації методом FCM при $\boldsymbol{m = 1.4}$

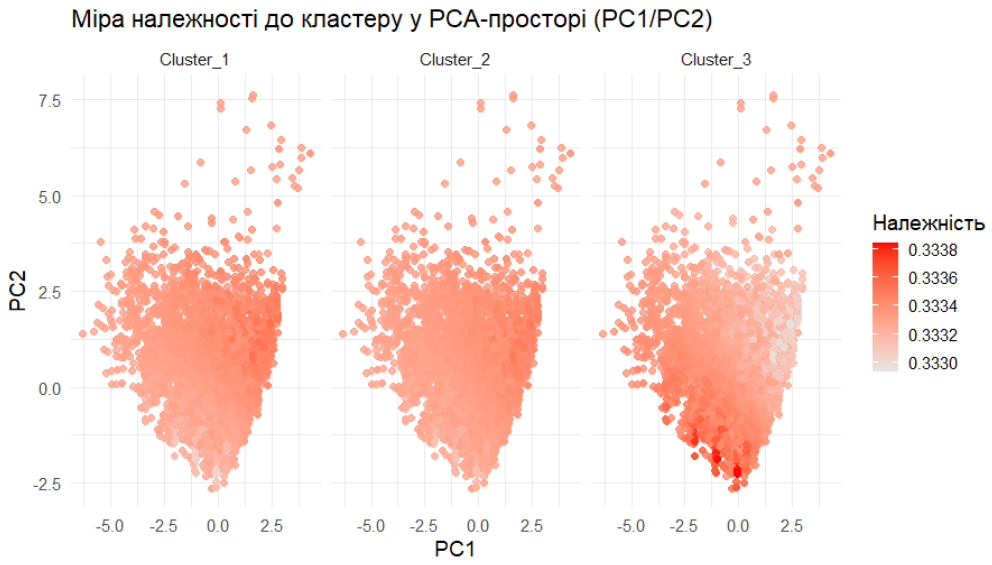

Рис. 2.27 – Результати кластеризації методом FCM при $\boldsymbol{m = 1.5}$

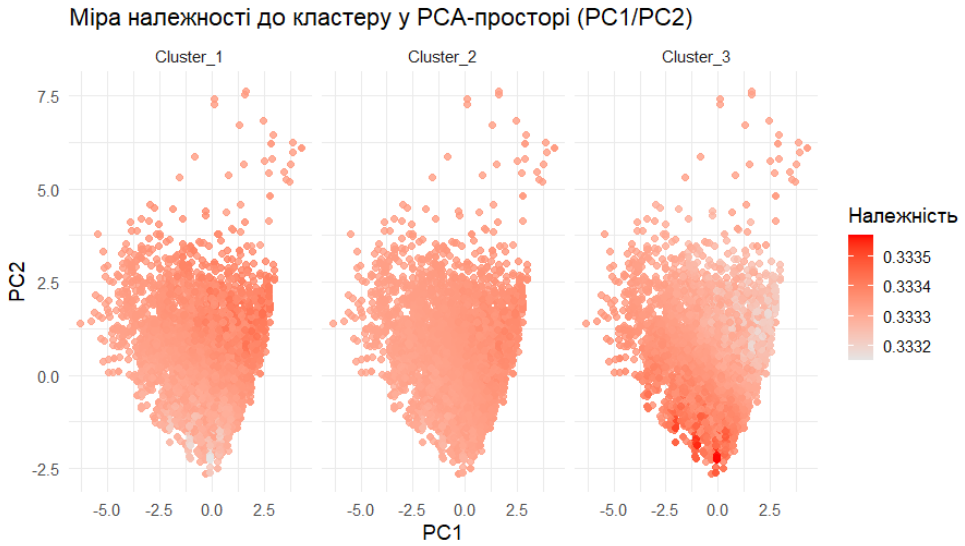

Рис. 2.28 – Результати кластеризації методом FCM при $\boldsymbol{m = 1.7}$

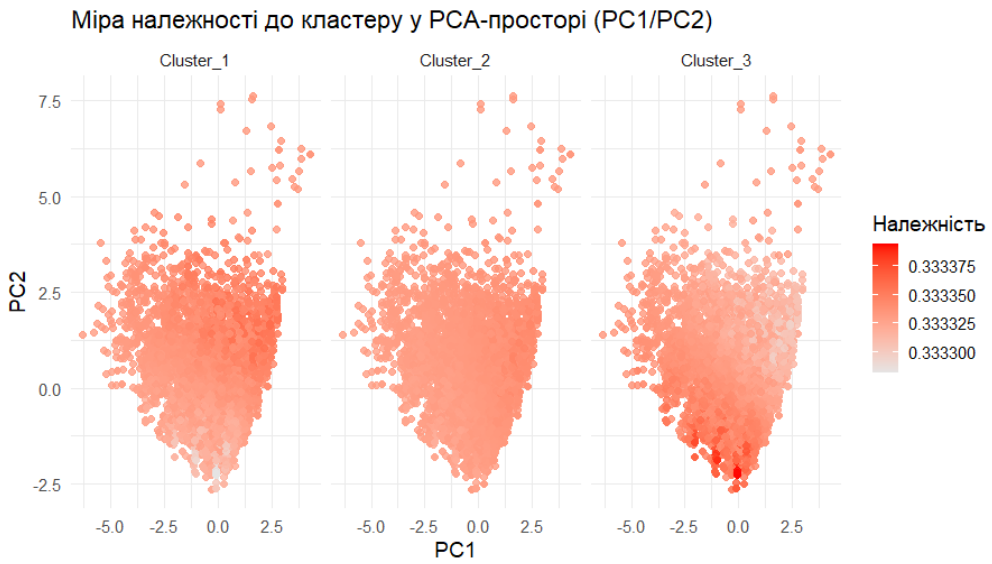

Рис. 2.29 – Результати кластеризації методом FCM при $\boldsymbol{m = 2.0}$

2.8. Аналіз та інтерпретація результатів

Інтерпретація кластерів, отриманих за допомогою методів K-Means та FCM, загалом виявилася однаковою: в обох випадках простежуються ті самі групи студентів. Водночас результати FCM дещо відрізняються за числовими значеннями належності, оскільки цей метод дозволяє врахувати «нечіткість» розподілу, тобто студенти можуть мати часткову приналежність до кількох кластерів. Таким чином, хоча FCM дає більш гнучке представлення структури даних, змістовна інтерпретація кластерів залишається близькою до результатів K-Means, що свідчить про їхню узгодженість і підтверджує стабільність знайдених груп.

Отже, інтерпретовані кластери такі.

Кластер №1 – онлайн відмінники під стресом. Вони мають високі оцінки за завдання та іспити (ev_exam = 0.72, ev_tareas = 0.94), низьку відвідуваність занять (med_presencial = 0.22), проте активну роботу онлайн (med_plataform = 0.57, med_clases_v = 0.53, med_mail_RS = 0.82). Вони ефективно використовують смартфони (0.88), ноутбуки (0.52), але трохи рідше ПК (0.20). Окрім цього вони емоційно виснажені – високий рівень стресу (0.86), депресії (0.84) та відчуття відчаю (0.79). Ці студенти досягають високих результатів у навчанні, але платять за це високим рівнем психологічного навантаження.

Кластер №2 – спокійні та ефективні відмінники. Вони мають найвищі оцінки серед усіх кластерів (ev_exam = 0.81, ev_tareas = 0.96). Відвідуваність низька (med_presencial = 0.11), акцент також на онлайн-навчанні (med_plataform = 0.67, med_clases_v = 0.66). Вони користуються здебільшого ноутбуками (0.68) та смартфонами (0.85). Показники емоційного стану близькі до нуля – у них майже немає стресу, депресії чи відчаю. Це стабільні та спокійні відмінники, які досягають високих результатів при мінімальних зусиллях і психологічних витратах. Вони ефективні у навчанні й добре пристосовані до дистанційних інструментів.

Кластер №3 – традиційні відвідувачі занять із нижчими результатами. Вони мають нижчі академічні результати (ev_exam = 0.56, ev_tareas = 0.89) та основний акцент на відвідування очних занять (med_presencial = 0.69). Використання онлайн-інструментів значно нижче (med_plataform = 0.17, med_clases_v = 0.15). Вони менше використовують ноутбуки (0.28), однак смартфони мають досі доволі високий показник (0.79). Емоційний стан стабільний ($\approx \mathbf{0.1}$) – низькі значення стресу, депресії та відчаю. Це традиційно орієнтовані студенти, які віддають перевагу класичним очним лекціям. Вони витрачають більше часу на навчання, проте їхні результати нижчі за студентів із кластерів 1 і 2.

Отримані результати кластеризації дають змогу загалом охарактеризувати вибірку студентів як таку, що поділяється на три відмінні групи з різними навчальними практиками, технологічними вподобаннями та емоційним станом. Студенти з кластеру №1 продемонстрували високі результати оцінювання, активно користуються сучасними технологіями та онлайн-форматами навчання, проте мають виражені емоційні проблеми — високий рівень стресу, депресії та відчуття безнадійності. У кластері №2 зібрані також успішні студенти, які активно застосовують технології, але відчутно відрізняються від першої групи тим, що мають набагато кращий психоемоційний стан, фактично без ознак депресії чи стресу. Це може свідчити про те, що ця група більш адаптована до гнучких форматів навчання й здатна ефективно працювати в таких умовах.

Кластер №3 відрізняється від інших тим, що студенти тут частіше навчаються в традиційному очному форматі, рідше користуються сучасними онлайн-інструментами і мають дещо нижчі середні показники успішності. Можна припустити, що ці результати не обов’язково свідчать про гірший рівень знань, а радше про специфіку контролю: студенти, які більше часу проводять на очних заняттях, можуть бути «краще контрольовані» викладачами або навіть вдаватися до списування, що нівелює справжній рівень засвоєння матеріалу. Водночас саме ця група має нижчий рівень емоційних проблем, що може пояснюватися більшою соціалізацією, особистим контактом з викладачами та

однокурсниками. Таким чином, можна зробити висновок, що формат навчання справді впливає на емоційний стан студентів: дистанційні та онлайн-орієнтовані формати створюють більше навантаження на психіку, тоді як очні заняття знижують рівень стресу, але не завжди забезпечують найвищі академічні результати.

Алгоритм Fuzzy C-Means (FCM) дав змогу глибше дослідити структуру даних і показав, що студенти насправді є доволі різними навіть у межах одного кластеру. Він дозволив виявити, що частина студентів має змішані характеристики — наприклад, вони можуть демонструвати високі академічні результати, як у кластері №2, але водночас мати схильність до підвищеного рівня стресу, притаманного кластеру №1. Таким чином, FCM підтвердив складність і багатовимірність студентської поведінки, показавши, що поділ на три групи є умовним, а реальна ситуація більш розмита, з багатьма проміжними станами між виділеними кластерами.

ВИСНОВКИ

У дипломному дослідженні було проведено комплексний аналіз методів чіткої та нечіткої кластеризації даних з використанням сучасних інструментів обробки та аналізу інформації. Основною базою для експериментів став набір даних ENAPE 2021, що містить важливу інформацію про освітні практики та психоемоційні стани студентів. Для реалізації поставлених завдань застосовувалися мова програмування R та інтегроване середовище розробки RStudio, які надали можливість поєднати зручність роботи з великими даними та широкий спектр бібліотек для статистичного аналізу й візуалізації результатів.

У роботі було розглянуто та реалізовано два основні підходи до кластеризації: метод K-Means як представник чіткої кластеризації та алгоритм Fuzzy C-Means (FCM) як приклад нечіткої кластеризації. Виконані експерименти дали змогу порівняти ці методи між собою, дослідити їхні сильні та слабкі сторони, а також зрозуміти, як різні алгоритми відображають структуру даних. Зокрема, було показано, що метод K-Means забезпечує простий і зрозумілий поділ об’єктів на кластери, тоді як FCM дозволяє виявити розмитість меж між групами та продемонструвати, що частина студентів має характеристики одразу кількох кластерів. Це надзвичайно важливо при аналізі соціальних чи освітніх даних, де поведінкові патерни не завжди мають чіткі границі.

Для оцінювання якості кластеризації було використано силуетний коефіцієнт та індекс Калінскі–Харабаза, що дозволило кількісно визначити, наскільки адекватним є обране число кластерів. Окрім цього, для нечіткого підходу було застосовано показник Fuzzy Partition Coefficient (FPC), який дав змогу визначити оптимальний параметр фазифікації. Отримані результати підтвердили, що найбільш доцільним є поділ даних на три кластери при значенні параметра m = 1.1 для методу FCM. Такий вибір було обґрунтовано не лише кількісними показниками, а й логічною інтерпретацією результатів.

За результатами аналізу вдалося виділити три основні групи студентів, які

відрізняються за навчальними практиками, використанням технологій та психоемоційними характеристиками. Однією з цікавих інтерпретацій стало те, що студенти з третього кластеру можуть мати нижчі академічні результати через те, що їх легше контролювати під час очних занять. Водночас виявилося, що формат навчання може впливати на емоційний стан студентів, зокрема рівень стресу та депресії. Це дозволяє зробити висновок, що освітні підходи та технологічні умови безпосередньо відображаються на психоемоційному благополуччі. Загальне порівняння використаних методів і результатів наведено в додатку А.

У перспективі подібні підходи можуть бути використані для аналізу даних українських освітніх досліджень, що матиме велике практичне значення. Використання кластеризації дасть змогу виділити основні групи студентів за їхніми навчальними та психоемоційними характеристиками, що може стати корисним для побудови ефективних освітніх стратегій, оптимізації навчального процесу та підтримки студентів із підвищеним ризиком дезадаптації. Таким чином, результати цього дослідження можуть слугувати не лише науковою основою, але й практичним інструментом для удосконалення освітнього середовища в Україні.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ланде Д., Субач І., Бояринова Ю. Основи теорії і практики інтелектуального аналізу даних у сфері кібербезпеки. Київ : ІСЗЗІ КПІ ім. Ігоря Сікорського, 2018. 300 с. URL: https://ela.kpi.ua/server/api/core/bitstreams/5eb4cb50-3ef0-44d1-b35f-a80eade1554b/content.
2. Cluster Analysis / B. S. Everitt et al. 5th ed. London : John Wiley & Sons Ltd, 2011. 330 p. URL: https://cicerocq.wordpress.com/wp-content/uploads/2019/05/cluster-analysis_5ed_everitt.pdf.
3. Aggarwal C. C. Data Mining - The Textbook. New York : Springer Cham, 2015. 734 p. URL: https://doi.org/10.1007/978-3-319-14142-8.
4. Харченко В. О. Основи машинного навчання - Навчальний посібник. Суми : Суми : Сум. держ. університет, 2023. 264 с. URL: https://essuir.sumdu.edu.ua/bitstream-download/123456789/92711/3/Kharchenko_osnovy.pdf;jsessionid=22C26A0E8CAB4B8296158615C8D1DBD5.
5. Cluster Analysis - two examples. *iChrome*. URL: http://ichrome.com/blogs/archives/221.
6. Noble J. What is Hierarchical Clustering?. *IBM*. URL: https://www.ibm.com/think/topics/hierarchical-clustering.
7. Chawla A. The Limitations of DBSCAN Clustering Which Many Often Overlook. *Daily Dose of Data Science | Avi Chawla | Substack*. URL: https://blog.dailydoseofds.com/p/the-limitations-of-dbscan-clustering.
8. DBSCAN Clustering in ML - Density based clustering. *GeeksforGeeks*. URL: https://www.geeksforgeeks.org/machine-learning/dbscan-clustering-in-ml-density-based-clustering/ (date of access: 25.08.2025).
9. Piech C. K Means. *Stanford University CS221*. URL: https://stanford.edu/~cpiech/cs221/handouts/kmeans.html.
10. Fuzzy Clustering - MATLAB & Simulink. *Access Denied*.

URL: https://www.mathworks.com/help/fuzzy/fuzzy-clustering.html.

11. Klir G. J., Yuan B. Fuzzy sets and fuzzy logic: theory and applications. 1995. 573 p. URL: http://www.pzs.dstu.dp.ua/logic/bibl/yuan.pdf.
12. Желдак Т., Коряшкіна Л., Ус С. Нечіткі множини в системах управління та прийняття рішень. Дніпро : Дніпро : НТУ «ДП», 2020. 387 с.
13. Carter T. An introduction to information theory and entropy. Santa Fe : Complex Systems Summer School, 2014. 139 p. URL: https://csustan.csustan.edu/~tom/Lecture-Notes/Information-Theory/info-lec.pdf.
14. Fu L., Medico E. FLAME, a novel fuzzy clustering method for the analysis of DNA microarray data. *BMC Bioinformatics*. 2007. Vol. 8, no. 1. URL: https://doi.org/10.1186/1471-2105-8-3.
15. Schulzch. schulzch/fuzzy_dbscan: An implementation of the FuzzyDBSCAN algorithm. *GitHub*. URL: https://github.com/schulzch/fuzzy_dbscan.
16. Kaymak U., Setnes M. Extended Fuzzy Clustering Algorithms. Rotterdam, 2000. 24 p. URL: https://web.archive.org/web/20110724152254/http://publishing.eur.nl/ir/repub/asset/57/erimrs20001123094510.pdf.
17. GeeksforGeeks. Image Segmentation Using Fuzzy C-Means Clustering. *GeeksforGeeks*. URL: https://www.geeksforgeeks.org/computer-vision/image-segmentation-using-fuzzy-c-means-clustering/.
18. Yadav S. Silhouette Coefficient Explained with a Practical Example: Assessing Cluster Fit". *Medium*. URL: https://medium.com/@Suraj_Yadav/silhouette-coefficient-explained-with-a-practical-example-assessing-cluster-fit-c0bb3fdef719.
19. GeeksforGeeks. Calinski-Harabasz Index – Cluster Validity indices | Set 3. *GeeksforGeeks*. URL: https://www.geeksforgeeks.org/machine-learning/calinski-harabasz-index-cluster-validity-indices-set-3/.
20. Ramezani N. Fuzzy C-means Clustering. *Medium*. URL: https://medium.com/@nafiseramezani1985/fuzzy-c-means-clustering-

f9e047e4e458.

21. Jecksom. Clustering and Principal Component Analysis (PCA) from Sklearn. *Medium*. URL: https://medium.com/@jackiee.jecksom/clustering-and-principal-component-analysis-pca-from-sklearn-c8ea5fed6648.
22. Petrus J., Ermatita, Sukemi. Soft and Hard Clustering for Abstract Scientific Paper in Indonesian. *2019 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS)*. 2019. URL: https://repository.unsri.ac.id/60868/1/Proceeding-ICIMCIS.pdf.
23. Implementation of the Fuzzy C-Means Clustering Algorithm in Meteorological Data / Y. Lu et al. *International Journal of Database Theory and Application*. 2013. P. 1–18. URL: https://scispace.com/pdf/implementation-of-the-fuzzy-c-means-clustering-algorithm-in-1kv7jkhrsb.pdf.
24. Oyelade J. O., Oladipupo O., Obagbuwa I. C. Application of k Means Clustering algorithm for prediction of Students Academic Performance. *International Journal of Computer Science and Information Security*. 2010. Vol. 7, no. 1. URL: https://arxiv.org/abs/1002.2425.
25. Singh A., Yadav A., Rana A. K-means with Three different Distance Metrics. *International Journal of Computer Applications*. 2013. Vol. 67, no. 10. P. 13–17.
    URL: https://research.ijcaonline.org/volume67/number10/pxc3886785.pdf.
26. Cardoso R. National Survey on School Dropout. Kaggle. URL: https://www.kaggle.com/datasets/renecardoso/national-survey-on-school-dropout/data.
27. R: What is R?. *R: The R Project for Statistical Computing*. URL: https://www.r-project.org/about.html.
28. Wickham H., Çetinkaya-Rundel M., Grolemund G. R for Data Science: Import, Tidy, Transform, Visualize, and Model Data. 2nd ed. O'Reilly, 2023. 576 p.
29. McKinney W. Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter 3rd Edition Wes McKinney. 3rd ed. O'Reilly, 2022. 580 p.
30. Dykiel H. An Overview of the RStudio IDE. *Medium*.

URL: https://medium.com/@HadrienD/an-overview-of-the-rstudio-ide-4031a46ad6b0.

31. RStudio IDE and Posit Workbench 2024.09.0: What's New. *Posit*. URL: https://posit.co/blog/rstudio-2024-09-0-whats-new/.

ДОДАТКИ

Додаток А

Таблиця А.1. Загальне порівняння методів K-Means та Fuzzy C-Means

Критерій	K-Means	Fuzzy C-Means (FCM)
Тип кластеризації	Чітка – кожен об’єкт належить лише одному кластеру	Нечітка – кожен об’єкт має ступінь належності до всіх кластерів
Реалізація в R	Вбудований пакет stats, функція kmeans()	Пакет e1071, функція cmeans()
Ключові параметри	centers – кількість кластерів; nstart – кількість початкових запусків; iter.max – максимальна кількість ітерацій	centers – кількість кластерів; m – параметр фазифікації; iter.max – максимальна кількість ітерацій
Складність коду	Простий синтаксис, короткі виклики	Дещо складніший код через додатковий параметр m і роботу з матрицею належностей
Результат кластеризації	Повертає вектор належностей і центи кластерів	Повертає матрицю мір належностей і центри кластерів
Інтерпретація проаналізованих даних	Студент відноситься чітко до одного кластеру, що спрощує аналіз	Студент може одночасно належати до кількох кластерів з різною мірою, що краще відображає реальність
Оцінка якості	Силуетний коефіцієнт та індекс Калінскі-Харабаза	Частково силуетний коефіцієнт та Fuzzy Partition Coefficient (FPC)
Оптимальні параметри проаналізованих даних	Найкраще k = 3 (силуетний коефіцієнт ≈ 0.15; CHI ≈ 640)	Найкраще k = 3, при m = 1.1 (FPC ≈ 0.7)

Продовження таблиці А.1

Візуалізація	PCA + кольорові кластери; чіткі межі	PCA + градієнти; «розмиті» межі
Сильні сторони	Швидкість, простота інтерпретації, наочні результати	Гнучкість, здатність враховувати неоднозначність у даних, більш реалістична модель поведінки
Слабкі сторони	Не враховує випадків, коли об’єкт може належати до кількох груп; чутливий до вибору початкових центрів	Вимагає додаткових параметрів (m), іноді важче інтерпретувати результати, вища обчислювальна складність
Результати для ENAPE 2021	Виявлено три основні групи студентів з різними навчальними та психоемоційними характеристиками	Показано, що студенти всередині кластерів є досить різними, межі між групами не завжди чіткі

Abstract.

Matviychuk O. M. – Analysis of crisp and fuzzy clustering methods based on student learning educational data. Manuscript.

Qualification work for the degree of "Master" in the field of Computer Science, educational program "Computer Science and Information Technologies." – Lesya Ukrainka Volyn National University. – 2025.

This paper presents a comprehensive analysis of crisp and fuzzy clustering methods applied to educational data using the ENAPE 2021 survey, which covers socio-demographic characteristics of households and students' educational trajectories. To achieve this goal, the R programming language and the RStudio environment were employed, enabling preprocessing and preparation of data, including cleaning, variable selection, and normalization. The research implemented the K-Means and Fuzzy C-Means (FCM) algorithms, which allowed for a comparison between crisp and fuzzy clustering methods and for identifying their respective advantages and limitations. The quality of the obtained clusters was evaluated using the Silhouette coefficient, the Calinski–Harabasz index, and the Fuzzy Partition Coefficient (FPC). For visualization of high-dimensional data, the Principal Component Analysis (PCA) method was applied, which made it possible to interpret clustering results in a clear two-dimensional representation. The analysis revealed that the optimal solution was to divide the dataset into three clusters, which highlighted groups of students with different academic and psycho-emotional characteristics. The study showed that fuzzy methods, unlike crisp ones, provide a broader understanding of student diversity by reflecting different degrees of membership in clusters.

Keywords: clusterization, k-means, fuzzy c-means (FCM), R, fuzzy logic.

Анотація

Матвійчук О. М. – Аналіз методів чіткої та нечіткої кластеризації на основі освітніх даних навчання студентів. – Рукопис.

Кваліфікаційна робота на здобуття освітнього ступеня «магістр» за спеціальністю 122 Комп'ютерні науки, освітньої програми Комп'ютерні науки та інформаційні технології. – Волинський національний університет імені Лесі Українки. – 2025 р.

У роботі проведено комплексне дослідження методів чіткої та нечіткої кластеризації освітніх даних на прикладі опитування ENAPE 2021, що охоплює соціально-демографічні характеристики домогосподарств та освітні траєкторії студентів. Для досягнення мети було використано мову програмування R та середовище RStudio, що забезпечили можливість попередньої обробки та підготовки даних, зокрема очищення, відбору змінних і нормалізації. У процесі дослідження було реалізовано алгоритми K-Means та Fuzzy C-Means (FCM), що дозволило порівняти роботу чітких і нечітких методів кластеризації та визначити їхні переваги й обмеження. Якість отриманих кластерів оцінювалась за допомогою таких показників, як силуетний коефіцієнт, індекс Калінскі-Харабаза та коефіцієнт нечіткості (FPC). Для візуалізації багатовимірних даних застосовано метод головних компонент (PCA), який дав змогу інтерпретувати результати кластеризації у зручній двовимірній формі. За результатами аналізу оптимальним виявився поділ на три кластери, що дозволило виділити групи студентів з різними навчальними та психоемоційними характеристиками. Дослідження показало, що нечіткі методи, на відміну від чітких, дають ширше розуміння різноманітності студентів, адже відображають різні ступені належності до кластерів.

Ключові слова: кластеризація, k-means, fuzzy c-means (FCM), R, нечітка логіка.