МАГІСТЕРСЬКА РОБОТА

МР. ІПм - 24.00.00.000 ПЗ

Група ІПм-24-2

Ковальчук Василь

2025

Івано-Франківський національний технічний університет нафти і газу

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Ковальчук Василь Ярославович

(прізвище, ім'я, по батькові)

УДК 004.9

(індекс)

МАГІСТЕРСЬКА РОБОТА

Моделі та методи кластеризації даних на основі властивості щільності

(назва роботи)

Інженерія програмного забезпечення

(назва освітньої програми)

121 - Інженерія програмного забезпечення

(шифр і назва спеціальності)

Ковальчук В.Я.

(підпис, ініціали та прізвище здобувача освітнього ступеня)

Науковий керівник Романишин Тарас Любомирович, к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Допущено до захисту

Завідувач кафедри

доц. Бандура В.В.

(посада) (підпис) (дата) (ініціали та прізвище)

Нормоконтроль

доц. Вовк Р.Б.

(посада) (підпис) (дата) (ініціали та прізвище)

Робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Івано-Франківськ – 2025

Івано-Франківський національний технічний університет нафти і газу

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Освітній рівень магістр

Спеціальність 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:
Зав. кафедрою **ІПЗ**

доц. В.В. Бандура

“ 04 ” вересня 2025 р.

ЗАВДАННЯ

НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

Ковальчуку Василю Ярославовичу

(прізвище, ім’я, по-батькові)

1. Тема магістерської роботи “ Моделі та методи кластеризації даних на основі властивості щільності”

керівник проекту (роботи) Романишин Т.Л., к.т.н., доцент

затверджені наказом закладу вищої освіти від “ 05 ” листопада 2025 р. № 695 /7

2. Строк подання студентом проекту (роботи) 15 грудня 2025 р.

3. Вихідні дані до проекту (роботи) Теоретичні концепції та формальні моделі побудови та функціонування програмних технологій кластеризації даних

4. Зміст розрахунково - пояснювальної записки(перелік питань, які потрібно розробити)

1. Аналіз предметної області кластеризації даних на основі алгоритмів щільності
2. Дослідження методів та методологій кластеризації даних на основі властивості щільності
3. Проєктування, реалізація модифікацій алгоритмів кластеризації даних на основі щільності
4. Критичний аналіз алгоритму кластеризації на основі щільності та його модифікацій

5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)

1. Графічна інтерпретація концепцій DENsity-based CLUstEring (рис. 1.1)
2. Ключові етапи роботи алгоритму CFSFDP (рис. 1.2)
3. Графічна інтерпретація алгоритму кластеризації (рис. 1.3)
4. Графічна ілюстрація роботи алгоритму kNN (рис. 1.4)
5. Графічна інтерпретація алгоритму KDE (рис. 1.5)

6. Консультанти розділів проекту (роботи)

Розділ	Консультант	Підпис, дата
Перевірка на плагіат	доц., к.т.н. Вовк Р.Б.	

7. Дата видачі завдання 04 вересня 2025 р.

Керівник _____
(підпис)

Завдання прийняв до виконання_____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назви етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Підбір і вивчення літератури по темі магістерської роботи	15.09.2025	виконано
2	Аналіз предметної області кластеризації даних на основі алгоритмів щільності	29.09.2025	виконано
3	Дослідження методів та методологій кластеризації даних на основі властивості щільності	15.10.2025	виконано
4	Проєктування, реалізація модифікацій алгоритмів кластеризації даних на основі щільності	08.11.2025	виконано
5	Критичний аналіз алгоритму кластеризації на основі щільності та його модифікацій	20.11.2025	виконано
6	Затвердження пояснювальної записки роботи завідувачем кафедри	16.12.2025	виконано

Студент – магістр _____
(підпис)

Керівник роботи _____
(підпис)

АНОТАЦІЯ

Магістерська робота: 75 с., 29 рис., 4 табл., 41 джерело.

Тема: Моделі та методи кластеризації даних на основі властивості щільності

Мета магістерської роботи: вдосконалення моделей і методів кластеризації даних на основі властивості щільності, які забезпечують підвищення точності, адаптивності та стійкості процесу кластеризації.

Об'єкт дослідження: процес кластеризації багатовимірних даних, що здійснюється з використанням методів машинного навчання без учителя.

Предмет дослідження: моделі, методи та алгоритми кластеризації даних на основі властивості щільності, зокрема модифікації класичних підходів.

Результати дослідження

В роботі розроблено декілька модифікацій алгоритму кластеризації, зокрема ітеративний алгоритм кластеризації з гаусовим ядром та градієнтною оптимізацією, який забезпечує поступове уточнення позицій центрів кластерів і зменшення похибки кластеризації.

Висновок

Нозроблено, обґрунтовано та експериментально підтверджено ефективність удосконалених моделей кластеризації даних на основі властивості щільності, що дозволяють суттєво підвищити якість аналізу складних інформаційних структур.

КЛАСТЕРИЗАЦІЯ ДАНИХ; ЩІЛЬНІСТЬ; K-НАЙБЛИЖЧІ СУСІДИ; ЯДРОВА ОЦІНКА ЩІЛЬНОСТІ; ГРАДІЄНТНА ОПТИМІЗАЦІЯ; МАШИННЕ НАВЧАННЯ; ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ.

ABSTRACT

Master Thesis: 75 pp., 29 fig., 4 tab., 41 sources.

Topic: Models and methods of data clustering based on the density property

The purpose of the master's thesis: improvement of models and methods of data clustering based on the density property, which ensure increased accuracy, adaptability and stability of the clustering process.

Object of research: the process of clustering multidimensional data, carried out using unsupervised machine learning methods.

Subject of research: models, methods and algorithms of data clustering based on the density property, in particular modifications of classical approaches.

Research results

The work developed several modifications of the clustering algorithm, in particular an iterative clustering algorithm with a Gaussian kernel and gradient optimization, which provides a gradual refinement of the positions of cluster centers and a reduction in clustering error.

Conclusion

The effectiveness of improved data clustering models based on the density property has been developed, substantiated and experimentally confirmed, which allow significantly improving the quality of analysis of complex information structures.

DATA CLUSTERING; DENSITY; K-NEAREST NEIGHBORS; KERNEL DENSITY ESTIMATION; GRADIENT OPTIMIZATION; MACHINE LEARNING; INTELLECTUAL DATA ANALYSIS.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

CFSFDP – Clustering by Fast Search and Find of Density Peaks – Кластеризація шляхом швидкого пошуку та знаходження піків щільності

kNN – k-Nearest Neighbors – k-Найближчі сусіди

kNN-CFSFDP – k-Nearest Neighbors Clustering by Fast Search and Find of Density Peaks – Кластеризація за k-Найближчими сусідами шляхом швидкого пошуку та знаходження піків щільності

FLAME – Fuzzy clustering by Local Approximation of MEmberships – Нечітка кластеризація шляхом локальної апроксимації приналежностей

DBSCAN – Density-Based Spatial Clustering of Applications with Noise – Просторова кластеризація програм на основі щільності з шумами

SVM – Support Vector Machine – Метод опорних векторів

RBF – Radial Basis Function – Радіально-базисна функція

MLP – Multilayer Perceptron – Багатошаровий перцептрон

ρ – Density – Локальна щільність

δ – Distance – Мінімальна відстань (до точки з вищою щільністю)

ВСТУП

Актуальність теми.

Сучасний етап розвитку інформаційних технологій характеризується стрімким зростанням обсягів, різноманітності та складності даних, що зумовлює підвищену потребу у створенні ефективних інтелектуальних систем для їх аналізу та обробки. Одним із найважливіших напрямів досліджень у галузі машинного навчання та інтелектуального аналізу даних є кластеризація — процес автоматичного групування об’єктів у множини (кластери) відповідно до їхніх внутрішніх властивостей чи ступеня подібності.

Особливе місце серед методів кластеризації посідають підходи, що ґрунтуються на властивості щільності. Вони дозволяють виявляти кластери довільної форми, ідентифікувати області підвищеної концентрації даних та ефективно відокремлювати шум. Однак класичні алгоритми цього типу (DBSCAN, DENCLUE, CFSFDP) мають обмеження, пов’язані з вибором параметрів, нестійкістю до змін щільності даних та неоднорідністю оцінки локальних структур. Ці недоліки актуалізують необхідність розроблення удосконалених моделей, здатних до адаптації в умовах нерівномірного розподілу даних.

Проблематика щільнісної кластеризації набуває особливого значення у зв’язку з розповсюдженням систем обробки великих даних, аналітичних платформ штучного інтелекту та прогнозних моделей, у яких якість кластеризації безпосередньо впливає на точність подальшої інтерпретації результатів. Саме тому розроблення методів, що забезпечують стабільність, точність і параметричну гнучкість процесу кластеризації, є актуальним науковим завданням.

Актуальність дослідження обумовлена потребою підвищення ефективності кластеризації даних у контексті сучасних інформаційних систем, що функціонують в умовах великих обсягів, шумів і високої

розмірності даних. Традиційні методи, такі як k-means або ієрархічні алгоритми, не завжди здатні адекватно відображати складні топологічні властивості простору ознак і часто потребують попереднього визначення кількості кластерів.

Методи, що базуються на властивості щільності, навпаки, дозволяють автоматично визначати кількість кластерів та працювати зі структурами довільної форми. Однак їх ефективність істотно залежить від вибору гіперпараметрів та коректності оцінки локальної щільності. Зважаючи на це, актуальною науковою задачею є вдосконалення моделей і алгоритмів кластеризації, які поєднують ідеї адаптивного оцінювання щільності, методів найближчих сусідів та ядерних функцій.

Крім того, сучасні застосування у сфері біоінформатики, фінансового аналізу, телекомунікацій, кібербезпеки та розпізнавання образів потребують розроблення універсальних алгоритмів, здатних забезпечувати стабільність кластеризації для даних різної природи. Саме тому проведене дослідження має як наукову, так і прикладну актуальність.

Метою магістерської роботи є вдосконалення моделей і методів кластеризації даних на основі властивості щільності, які забезпечують підвищення точності, адаптивності та стійкості процесу кластеризації.

Об’єктом дослідження є процес кластеризації багатовимірних даних, що здійснюється з використанням методів машинного навчання без учителя.

Предметом дослідження є моделі, методи та алгоритми кластеризації даних на основі властивості щільності, зокрема модифікації класичних підходів.

Завдання дослідження

Для досягнення поставленої мети у роботі необхідно було розв’язати такі основні завдання:

1. Провести аналіз теоретичних основ кластерного аналізу та класифікацію існуючих підходів до кластеризації даних.

2. Дослідити принципи побудови алгоритмів кластеризації на основі властивості щільності та визначити їхні сильні й слабкі сторони.

3. Створити та реалізувати модифікації алгоритмів щільнісної кластеризації з адаптивним вибором параметрів.

4. Провести експериментальні дослідження для оцінювання ефективності розроблених моделей на тестових наборах даних.

5. Виконати порівняльний аналіз результатів роботи базових і модифікованих алгоритмів та сформулювати практичні рекомендації щодо їх застосування.

Методи дослідження

Для досягнення мети використано такі методи:

- математичні методи аналізу та моделювання — для формалізації процесу кластеризації та побудови функцій щільності;

- методи статистичного аналізу — для оцінювання щільності розподілу даних і визначення параметрів кластерів;

- алгоритмічні та обчислювальні методи — для реалізації модифікованих алгоритмів кластеризації та їх оптимізації;

- експериментальні методи — для перевірки працездатності моделей на тестових наборах даних та порівняльної оцінки ефективності;

- візуалізаційні методи аналізу даних — для графічної інтерпретації результатів кластеризації та оцінки її якості.

Наукова новизна отриманих результатів

Наукова новизна магістерської роботи полягає у розробленні нових та вдосконаленні існуючих методів кластеризації даних на основі властивості щільності. Основні результати, що мають наукову новизну:

- Удосконалено модель кластеризації на основі властивості щільності, у якій поєднано механізми локальної оцінки щільності (k-найближчі сусіди) з ядерним підходом до її неперервного оцінювання.

- Запропоновано модифікацію алгоритму CFSFDP, що використовує адаптивну метрику для вибору центрів кластерів, що підвищує точність і стійкість результатів.

Практичне застосування результатів

Практичне значення роботи полягає у можливості використання розроблених моделей і алгоритмів у системах аналітики даних, розпізнавання образів, біоінформатики, фінансовому прогнозуванні, інформаційній безпеці та інших галузях, де необхідне автоматичне виявлення структур у складних багатовимірних просторах.

Структура магістерської роботи. Робота складається зі вступу, трьох розділів та висновків. Загальний обсяг роботи становить 75 сторінок, і містить 29 рисунків, 4 таблиці, список використаних джерел із 41 найменування.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ КЛАСТЕРИЗАЦІЇ ДАНИХ НА ОСНОВІ АЛГОРИТМІВ ЩІЛЬНОСТІ

1.1. Кластерний аналіз як фундаментальний метод аналізу даних

1.1.1. Концептуальні основи та призначення кластеризації

Кластерний аналіз (Cluster Analysis) є ключовим методом некерованого машинного навчання (Unsupervised Learning) та аналізу даних (Data Mining), спрямованим на класифікацію об'єктів у гомогенні підмножини (кластери) на основі метрики їхньої подібності [1]. Основна мета кластеризації полягає у виявленні внутрішньої структури даних шляхом знаходження груп об'єктів, які демонструють високу інтракластерну подібність при мінімізації інтеркластерної подібності. Ефективна кластеризація також передбачає ідентифікацію та відокремлення шумових точок (викидів, outliers).

Об'єкти, інтегровані в один кластер, мають спільні властивості або закономірності, що дозволяє робити індуктивні висновки про характеристики окремого об'єкта на основі атрибутів групи, до якої він належить [2].

1.1.2. Задачі та методологічні аспекти

Кластерний аналіз виконує низку ключових функцій:

- Розробка класифікаційних систем.
- Дослідження концептуальних схем для структуризації сутностей.
- Генерація гіпотез шляхом експлораторного аналізу даних.
- Верифікація гіпотез або підтвердження існування типів, визначених іншими аналітичними методами, у поточному наборі даних.

Кластерний аналіз забезпечує стисле представлення даних та є необхідним інструментом у великомасштабних процесах обробки інформації [2]. Слід зазначити, що універсального алгоритму для всіх завдань кластеризації не існує. Ефективність залежить від конкретного уявлення про кластер, яке може варіюватися від груп з мінімальною внутрішньою

дисперсією (наприклад, k-середніх), до регіонів високої щільності (наприклад, DBSCAN), або ж певних статистичних розподілів.

Кластеризація даних формалізується як багатокритеріальна задача оптимізації. Вибір оптимального алгоритму та його параметрів (включаючи функцію відстані, поріг щільності, або задану кількість кластерів) є ітеративним процесом та залежить від характеристик набору даних і кінцевої мети дослідження [17].

Кластерний аналіз має широке міжгалузеве застосування:

- Біоінформатика / Біологія - визначення популяційної структури шляхом кластеризації генетичних даних людини;

- Комп'ютерні науки - сегментація зображень з метою виділення окремих областей для подальшого розпізнавання об'єктів або виявлення меж.

- Медицина - fналіз профілів резистентності до антибіотиків та класифікація антимікробних сполук на основі механізму дії.

- Астрономія та науки про землю - оцінка властивостей родовищ, включаючи відновлення відсутніх даних свердловинного керна або кривих каротажу.

- Соціальні науки - ідентифікація географічних районів із підвищеною концентрацією певних видів злочинності для цільового реагування.

- Економіка та електронна комерція - групування різноманітних товарних позицій (наприклад, на платформах електронної торгівлі) у набори унікальних продуктів для каталогізації та аналізу ринку.

1.2. Теоретичні аспекти кластерного аналізу та класифікація алгоритмів

1.2.1. Визначення кластера та метрика подібності

У контексті аналізу даних, кластер визначається як сукупність об'єктів, які демонструють високий ступінь подібності (близькості) між собою, тоді як

об'єкти, що належать до різних кластерів, характеризуються низьким ступенем подібності (високою відстанню).

Ключовим методологічним етапом кластерного аналізу є вибір метрики для кількісного визначення близькості (подібності) або відстані (неподібності). Це вимірювання може застосовуватися до пари об'єктів, об'єкта та кластера (центроїда) або пари кластерів. Вибір відповідної функції близькості безпосередньо впливає на топологію та форму результуючих кластерів. Зважаючи на суб'єктивний та проблемно-залежний характер кластерного аналізу, не існує універсального критерію для визначення найкращого підходу до вимірювання близькості або характеристик кластера.

1.2.2. Класифікація алгоритмів кластеризації

Алгоритми кластеризації зазвичай класифікують на кілька категорій на основі їхніх принципів роботи та математичних моделей:

а) Методи, базовані на центроїдах (k-середні та k-медоїди)

Ці методи є ітеративними та підходять переважно для виявлення опуклих (сферичних) кластерів. Кластери формуються шляхом групування даних, які є найближчими до центральної точки (центроїда або медоїда).

Алгоритм спрямований на оптимізацію цільової функції (як правило, мінімізацію суми квадратів відстаней від точок до центроїда).

Механізм k-середніх. Кожен зразок призначається до найближчого середнього значення (центроїда). Центри кластерів ітеративно оновлюються, і процес триває доти, доки центроїди не перестануть змінюватися або не буде перепризначень точок.

Величина k (очікувана кількість кластерів) має бути заздалегідь визначена користувачем.

Недоліки - чутливість до початкової ініціалізації центроїдів, що може призводити до різних локальних оптимумів [5].

б) Алгоритми на Основі Розподілу

Цей підхід розглядає набір даних як суміш попередньо визначених функцій розподілу ймовірностей (наприклад, суміш Гауссових розподілів). Це параметричний підхід, де невідома функція щільності ймовірності вважається такою, що належить до певної параметричної родини. Кластери відповідають компонентам цієї суміші розподілів.

в) Методи кластеризації на основі щільності (Density-Based Methods)

Ці методи є непараметричними і не залежать від форми кластерів, фокусуючись на локальній щільності розташування точок даних. Вони ефективні для ідентифікації кластерів довільної форми та для обробки шуму.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) - визначає кластер як набір густо з'єднаних точок. Використовується поріг щільності для класифікації точок як шум (викиди).

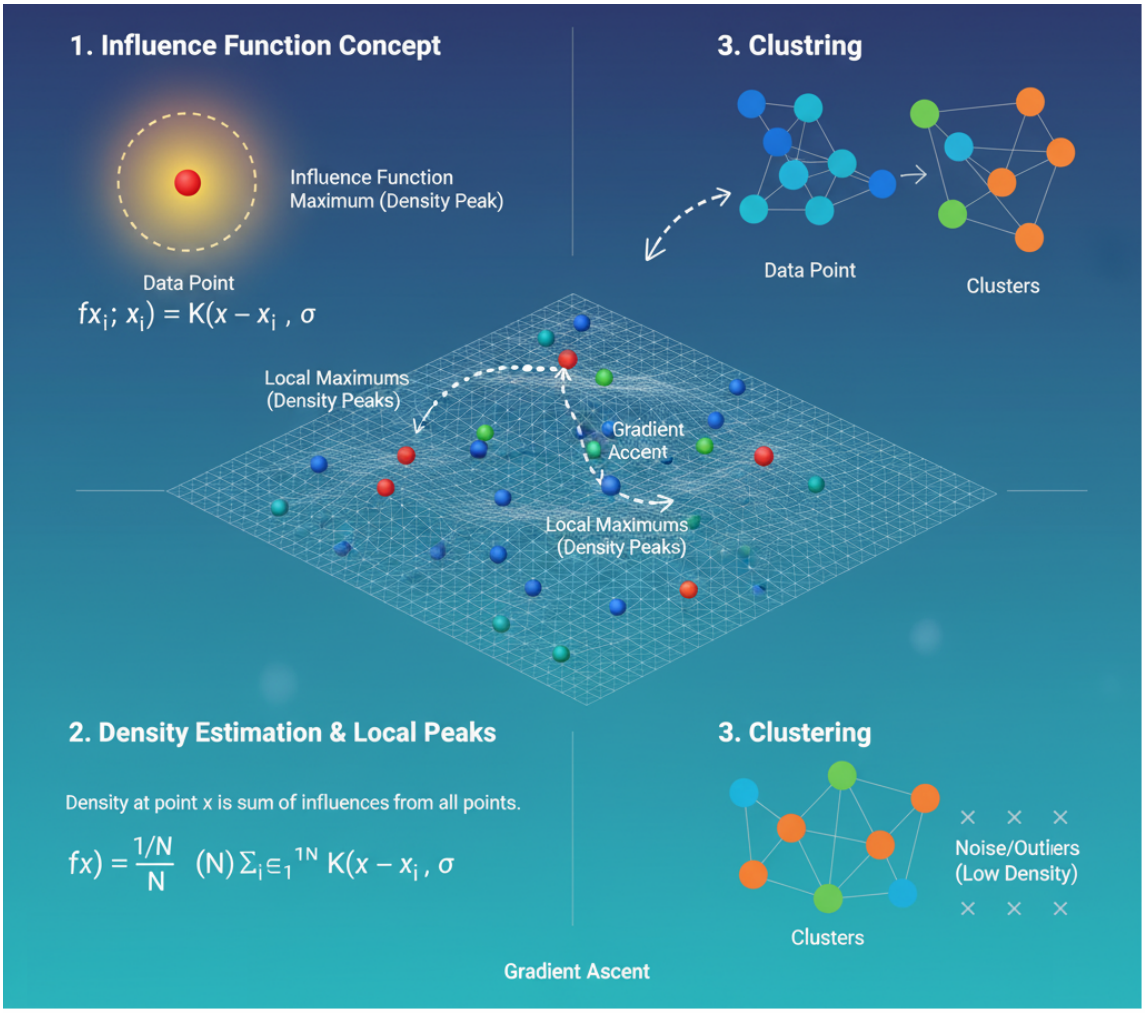


Рис. 1.1. Графічна інтерпретація концепцій DENsity-based CLUstEring

DENCLUE (DENsity-based CLUstEring) - використовує оцінку ядерної щільності (Kernel Density Estimation, KDE) для моделювання кластерів:

- Кластери визначаються локальними максимумами оціненої функції щільності.

- Точки даних призначаються кластерам за допомогою процедури сходження на пагорб (hill-climbing). Точки, що сходяться до одного й того ж локального максимуму функції щільності, формують один кластер.

Недоліком DENCLUE є те, що процедура сходження на пагорб може вимагати надмірно малих кроків на початкових етапах і не завжди гарантує точну конвергенцію до максимуму, лише його наближення.

1.3. Аналіз та удосконалення методів кластеризації даних на основі на щільності

Кластеризація даних являє собою фундаментальний метод аналізу даних (Data Mining), що полягає у групуванні об'єктів (спостережень) у підмножини (кластери) на основі метрики подібності (або відстані). Критерієм якісної кластеризації є висока інтракластерна подібність (гомогенність) та низька інтеркластерна подібність (гетерогенність). Широкий спектр застосувань кластеризації охоплює такі критичні галузі, як біоінформатика (наприклад, аналіз генних експресій), сегментація зображень та маркетингові дослідження (наприклад, сегментація споживачів).

1.3.1. Дослідження алгоритмів кластеризації на основі щільності

Дане дослідження присвячене поглибленому вивченню кластеризації даних, з особливим акцентом на методах, базованих на щільності. Ці методи ефективні для виявлення кластерів довільної форми та стійкі до аномалій (шуму).

Центральним об'єктом аналізу є алгоритм кластеризації на основі щільності (Clustering by Fast Search and Find of Density Peaks - CFSFDP),

відомий також як кластеризація за центрами щільності. Його теоретична основа полягає в припущенні, що центри кластерів характеризуються двома ключовими властивостями:

- Висока локальна щільність, тобто центри кластерів оточені сусідами з нижчою локальною щільністю. Вони є "піками" в ландшафті щільності даних.

- Велика відстань. Центри кластерів знаходяться на відносно великій відстані від будь-яких інших точок, які мають вищу щільність ніж сусідні елементи.

1.3.2. Удосконалення методу кластеризації на основі щільності (CFSFDP)

В рамках проєкту було здійснено експериментальну верифікацію та модифікацію базового алгоритму CFSFDP. Запропоновано та протестовано низку методів удосконалення, спрямованих на більш точне визначення локальної щільності та/або ідентифікацію центрів кластерів:

- Метод на основі k-найближчих сусідів (k-Nearest Neighbors, k-NN): Використовується для більш робастної оцінки локальної щільності.

- Метод на основі гаусового ядра (Gaussian Kernel): застосування функції Гаусового ядра для зваженого обчислення щільності, що забезпечує більш гладке та безперервне представлення щільності даних.

- Ітеративний метод на основі Гаусового ядра: Подальше вдосконалення, яке включає ітераційні процедури для оптимізації параметрів ядра або уточнення центрів кластерів.

Вказані модифікації були застосовані до чотирьох стандартизованих тестових наборів даних.

Проведено детальний порівняльний аналіз отриманих результатів кластеризації, що дозволило кількісно оцінити ефективність та переваги запропонованих удосконалень порівняно з оригінальним алгоритмом CFSFDP.

1.4. Формальне представлення алгоритму кластеризації на основі щільності

1.4.1. Опис основних кроків алгоритму

Алгоритм CFSFDP виконується у три основні кроки:

1. Обчислення характеристик точок

Для кожної точки даних і обчислюються дві ключові характеристики:

а) Локальна щільність (ρ_i)

Локальна щільність ρ_i точки і визначає, наскільки вона оточена іншими точками. Існує два поширені способи її обчислення:

- Метод ядра (Kernel Method) - використовує Гаусове ядро, де $\chi(x)$ — функція ядра (наприклад, Гаусове):

$$\rho_i = \sum_j \chi(d_{ij})$$

де d_{ij} — відстань між точками і та j.

- Метод відсічної відстані (Cut-off Kernel Method) - обчислює кількість точок j, що знаходяться на відстані, меншій за заздалегідь визначений поріг відсікання d_c:

$$\rho_i = \sum_j \chi(d_{ij} - d_c)$$

де $\chi(x)=1$, якщо $x<0$ (тобто $d_{ij}<d_c$), і $\chi(x)=0$ в іншому випадку.

б) Відстань до точки з вищою щільністю (δi)

δi — це мінімальна відстань від точки і до будь-якої іншої точки j, яка має вищу локальну щільність ($\rho_j > \rho_i$):

$$\delta_i = \min_{j:\rho_j>\rho_i} (d_{ij})$$

Для точки i* з найвищою щільністю у всьому наборі даних, її δ_i* встановлюється як максимальна відстань d_{max} до будь-якої іншої точки:

$$\delta_{i^*} = \max_j(d_{i^*j})$$

2. Ідентифікація центрів кластерів (Decision Graph)

Центри кластерів обираються вручну або евристично на основі рішення графіка (Decision Graph). Рішення графік — це двовимірний графік, де по осі X відкладається локальна щільність (ρ), а по осі Y — відстань до вищої щільності (δ).

Центри кластерів — це точки, які мають великі значення ρ (висока щільність) та великі значення δ (значна віддаленість від точок з вищою щільністю). Вони чітко виділяються у верхньому правому куті графіка, оскільки представляють піки щільності, ізольовані від інших піків.

3. Призначення точок до кластерів

Після визначення центрів кластерів, всі інші точки призначаються до кластера їхнього найближчого сусіда, який має вищу щільність (ρ). Цей процес відбувається лише в один прохід, без ітерацій:

- Кожна точка i призначається до того ж кластера, що й точка j (її "батьківська" точка), для якої була обчислена δ_i.

- Цей процес гарантує, що призначення слідує шляхом "градієнта" щільності, рухаючись від менш щільних регіонів до піків щільності (центрів).

На рисунку 1.1 подано ключові етапи роботи алгоритму CFSFDP:

- Density & Distance Calculation (обчислення щільності та відстані): Показує точки даних, кольором кодуючи їхню локальну щільність (ρ), а також стрілками демонструючи відстань (δ) до найближчого сусіда з вищою щільністю.

- Decision Graph & Cluster Head Selection (графік рішень та вибір центрів кластерів): Відображає точки на графіку ρ проти δ, де чітко видно, як обираються "Cluster Heads" (центри кластерів) у верхньому правому куті.

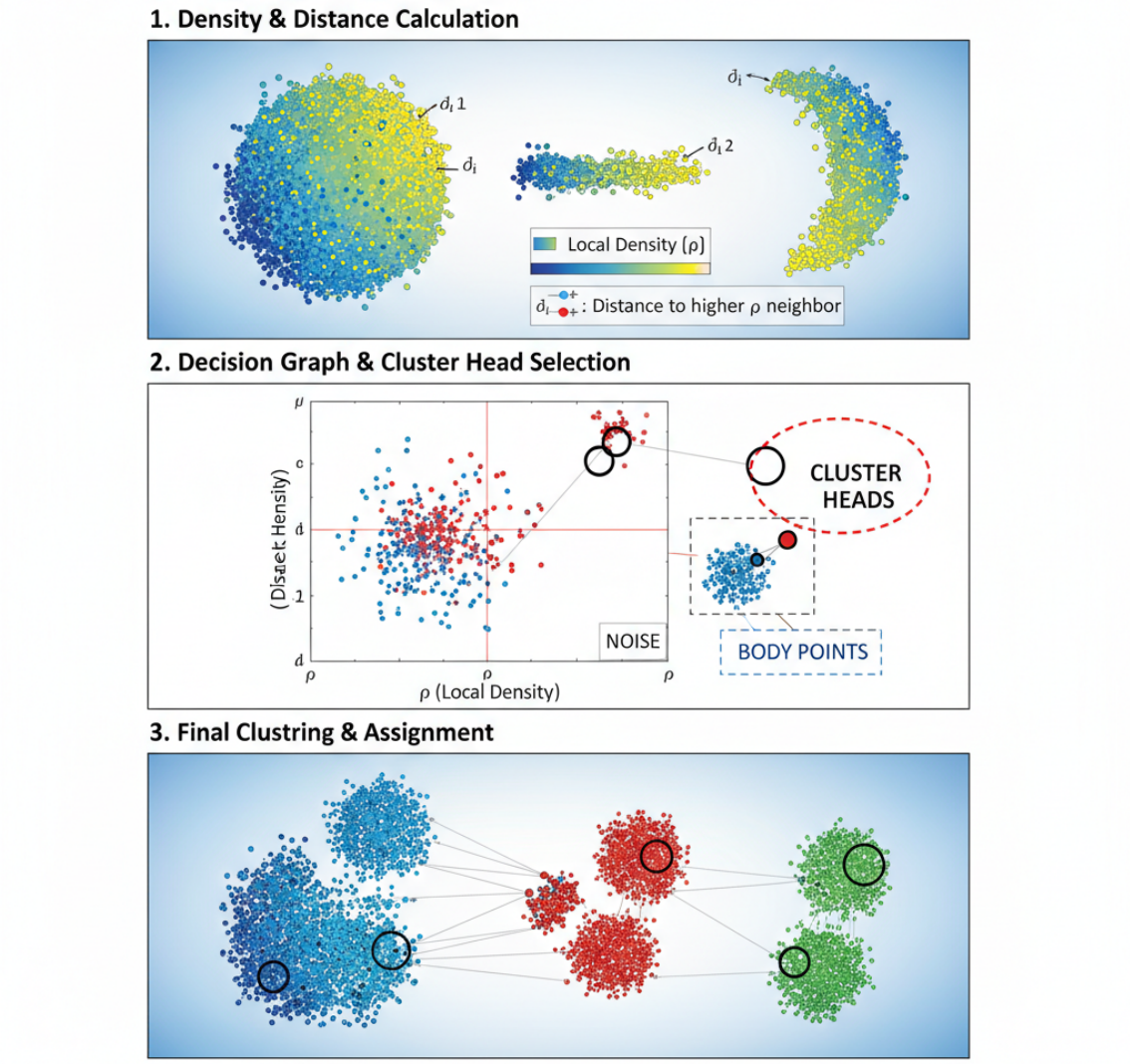


Рис. 1.2. Ключові етапи роботи алгоритму CFSFDP

- Final Clustering & Assignment (фінальна кластеризація та призначення): Демонструє остаточний результат кластеризації, де кожна точка призначена до свого кластера, і видно, як вони групуються навколо обраних центрів.

1.4.2. Графічна інтерпретація алгоритму кластеризації на основі щільності

1. Розподіл точок та характеристики

На рисунку 1.3 нижче точки даних візуалізовані у просторі, а їхні властивості ρ (локальна щільність) і δ (відстань до точки з вищою щільністю) ілюструють механізм алгоритму.

Таблиця 1.1.

Розподіл точок щодо кластерів

Точка	Локальна щільність (ρ)	Відстань до вищої щільності (δ)	Призначення
A	**Висока**	**Висока**	**Центр кластера 1**
B	**Висока**	**Висока**	**Центр кластера 2**
C	Висока	Низька	Належить до кластера 1
D	Середня	Низька	Належить до кластера 2
E	Низька	Середня	Можливий викид/шум

Центри кластерів (A, B) - це точки, що знаходяться на вершинах "пагорбів" щільності. Вони мають багато сусідів (ρ велике) і розташовані далеко від будь-якої іншої, ще вищої вершини (δ велике).

Інші точки (C, D) мають високу ρ, але низьку δ, оскільки розташовані близько до центру кластера з вищою щільністю.

Точки на межі/шум (E): Можуть мати низьку ρ (як шум) або помірну ρ і δ (як точки на межі кластерів).

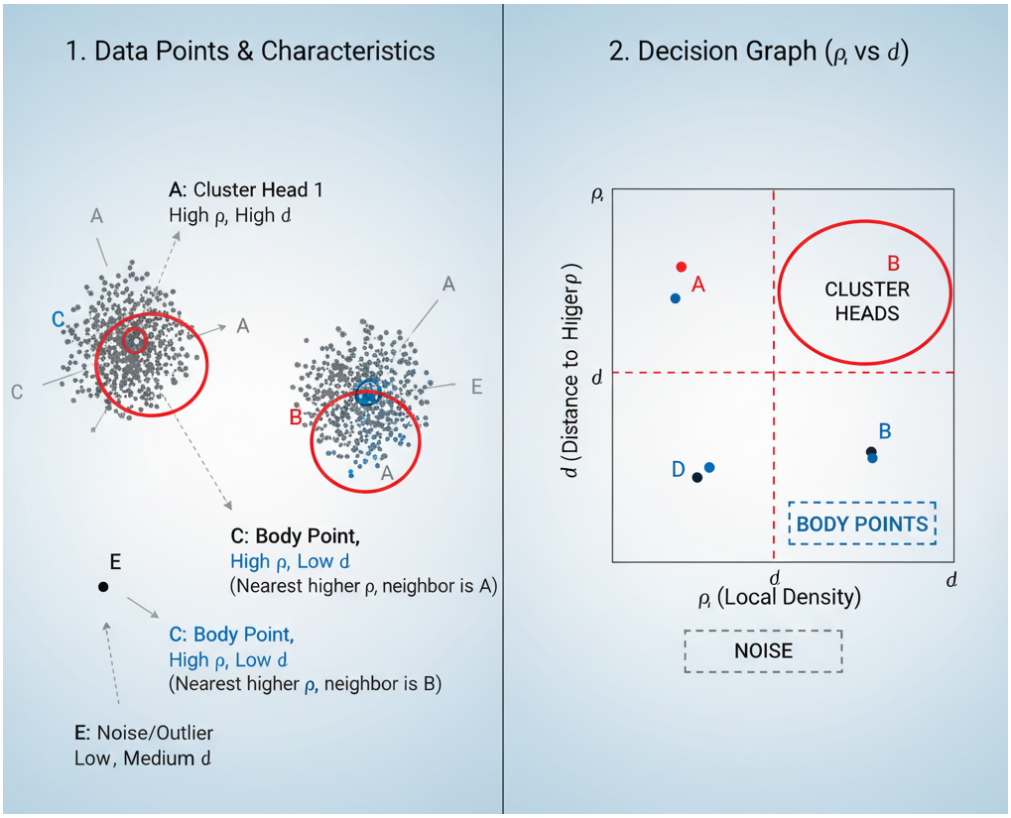


Рис. 1.3. Графічна інтерпретація алгоритму кластеризації

2. Графік рішення (Decision Graph)

Графік Рішення є критичним етапом для візуального вибору центрів кластерів.

- Вісь X (ρ) - локальна щільність.

- Вісь Y (δ) - мінімальна відстань до точки з вищою щільністю.

На графіку:

- Центри кластерів знаходяться у правому верхньому квадранті (високе ρ і високе δ). Ці точки є піками щільності.

- Точки тіла кластерів знаходяться у правому нижньому квадранті (високе ρ, але низьке δ), оскільки вони знаходяться близько до піка.

- Шум/викиди (Outliers) часто знаходяться у лівому нижньому квадранті (низьке ρ і низьке δ).

Вибір центрів зводиться до візуальної ідентифікації точок-викидів на цьому графіку (точки, які помітно віддалені вгору та вправо).

3. Візуалізація призначення (Assignment Visualization)

Процес призначення точок до кластерів можна візуалізувати як побудову "дерева найближчого сусіда вищої щільності". Кожна точка вказує на свого найближчого сусіда з вищою щільністю. Це створює "потоки" від периферії до центрів. Точки і слідують цим "потоком" доки не досягнуть точки, яка вже є центром кластера, або точки, яка має найбільшу щільність серед усіх сусідів.

1.5. Дослідження та опис алгоритму k-найближчих сусідів (kNN)

Алгоритм k-найближчих сусідів (kNN) — це простий, але потужний метод непараметричного навчання, який використовується як для класифікації, так і для регресії. Його відмінною рисою є те, що це ледачий (lazy) алгоритм, оскільки він не будує жодної моделі на етапі навчання, а просто зберігає весь набір даних. Обчислення відбувається лише на етапі передбачення.

kNN функціонує на основі принципу, що схожі точки даних (сусіди) існують близько одна до одної у просторі ознак.

А. Класифікація kNN

Для класифікації нової (тестової) точки xnew алгоритм виконує такі кроки:

1. Визначення k: Задається ціле число k — кількість найближчих сусідів, які будуть враховуватися.

2. Обчислення Відстані: Розраховується відстань (зазвичай Евклідова відстань) між x_{new} та кожним об'єктом у навчальному наборі даних.

3. Вибір Сусідів: Вибираються k об'єктів із навчального набору, які мають найменшу відстань до x_{new}.

4. Голосування: Клас xnew визначається шляхом голосування більшості серед цих k сусідів. Тестовій точці присвоюється клас, який найчастіше зустрічається серед k найближчих сусідів.

Б. Регресія kNN

Для регресії (передбачення числового значення) замість голосування використовується усереднення значень цільової змінної k найближчих сусідів.

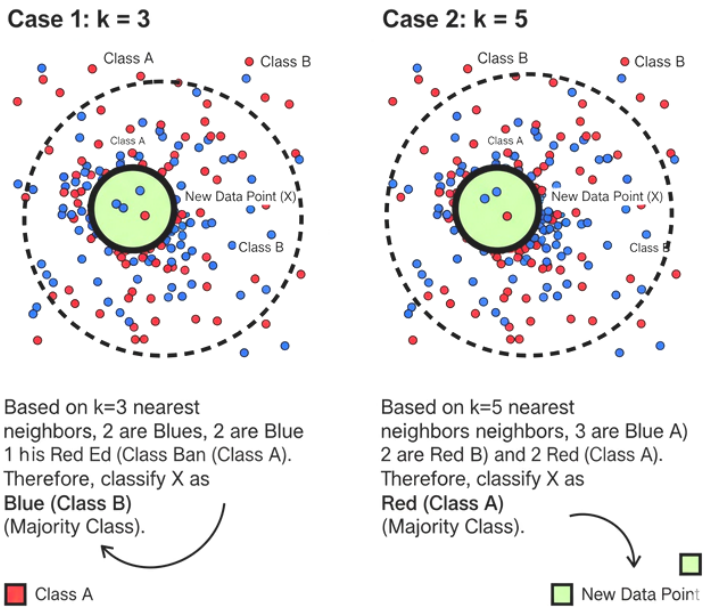


Рис. 1.4. Графічна ілюстрація роботи алгоритму kNN

На рисунку 1.4 показано, як вибір параметра k впливає на класифікацію нової точки.

Таблиця 1.2.

Пояснення роботи алгоритму kNN

Параметр k	**Найближчі сусіди**	**Передбачений клас**	**Пояснення**
k=3	2 сині, 1 червоний	**Синій** (Більшість)	Сині точки переважають, тому новий об'єкт класифікується як синій.
k=5	3 червоні, 2 сині	**Червоний** (Більшість)	Червоні точки переважають, тому новий об'єкт класифікується як червоний.

Також наведемо ключові особливості даного алгоритму:

1. Алгоритм є непараметричний, бо не робить жодних припущень щодо основного розподілу даних.

2. Не вимагає явного етапу навчання; всі обчислення відкладаються до моменту передбачення.

3. Вибір параметра k критично впливає на результат. Мале k робить модель чутливою до шуму (перенавчання), тоді як велике k призводить до розмиття меж класів (недонавчання).

4. Алгоритм дуже чутливий до масштабу ознак, оскільки базується на відстанях; тому дані зазвичай нормалізують або стандартизують перед застосуванням kNN.

5. На етапі передбачення можуть бути високими, особливо для великих наборів даних, оскільки потрібно обчислити відстань до всіх навчальних прикладів.

1.6. Представлення принципу роботи алгоритму оцінки ядерної щільності (Kernel Density Estimation, KDE)

Оцінка ядерної щільності (KDE) — це непараметричний статистичний метод, який використовується для оцінки функції щільності ймовірності

(Probability Density Function, PDF) випадкової величини на основі кінцевого набору даних. KDE є фундаментальною складовою багатьох алгоритмів кластеризації на основі щільності, таких як DENCLUE.

KDE формує гладку функцію щільності, яка є сумою функцій впливу (ядер), центрованих на кожній точці даних. На відміну від гістограми, KDE створює неперервну та диференційовану оцінку щільності, уникаючи залежності від вибору початку та ширини біна.

Для одновимірного набору даних $x_1, x_2, \ldots, x_n$ оцінка щільності $\hat{f}(x)$ у будь-якій точці x визначається за формулою:

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

де:

n — кількість точок даних.

K(·) — ядерна функція (Kernel Function). Зазвичай використовується Гаусове ядро (Gaussian Kernel), яке моделює внесок кожної точки як плавний "пагорб" або "вплив", що зменшується з віддаленням.

h — пропускна здатність (Bandwidth) або параметр згладжування. Це критичний параметр, який контролює ступінь гладкості оцінки.

Ключовим параметром є пропускна здатність (h). Вибір параметра h є вирішальним:

- мале h (низьке згладжування) - оцінка щільності стає надто "гострою" та чутливою до шуму (спостерігається перенавчання).

- велике h (високе згладжування) - оцінка щільності стає надто "розмитою", зливаючи окремі піки (спостерігається недонавчання).

Графічна інтерпретація KDE демонструє, як функція щільності створюється шляхом накладання окремих ядер:

1. Початкові точки на одновимірній осі X розташовані у вигляді синіх точок даних.

2. Над кожною точкою даних будується окрема функція Гаусового ядра (тонка червона крива), центрована в цій точці. Ширина цих ядер контролюється параметром h.

3. Кінцева функція щільності f^(x) (товста чорна крива) є сумою всіх індивідуальних ядерних функцій. Піки цієї сумарної кривої вказують на області високої щільності в даних.

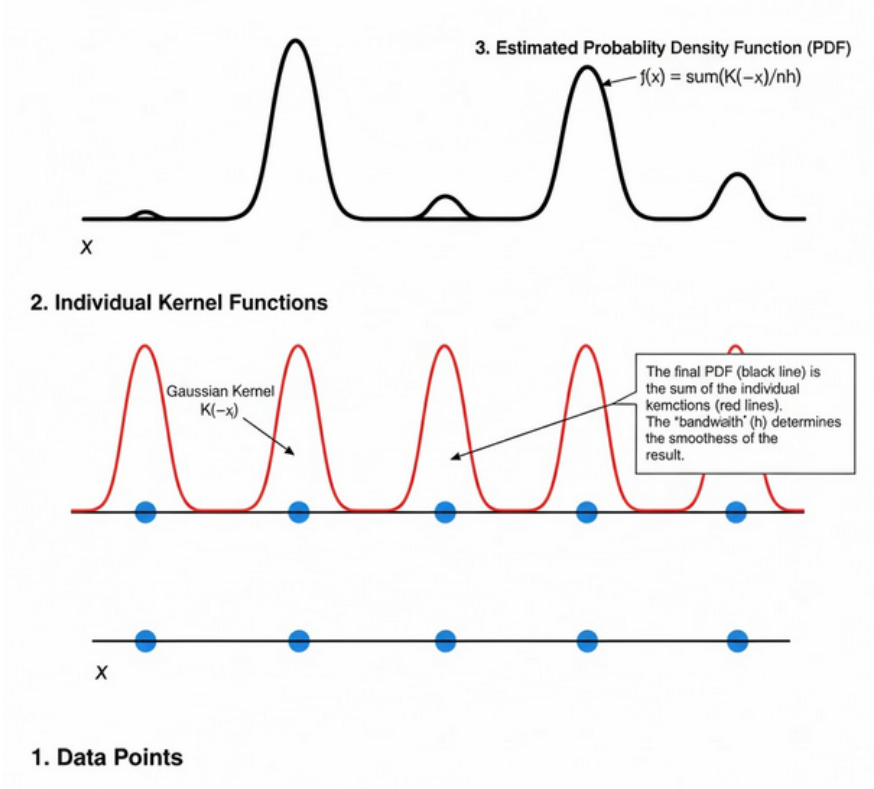


Рис. 1.5. Графічна інтерпретація алгоритму KDE

Таким чином, KDE перетворює дискретний набір даних на неперервний ландшафт щільності, де локальні максимуми цієї функції відповідають потенційним центрам або ядрам кластерів.

Висновки до розділу

У першому розділі здійснено всебічний теоретичний аналіз основ кластерного аналізу як фундаментального напряму інтелектуальної обробки

даних. Розглянуто концептуальні засади кластеризації, її роль у виявленні прихованих закономірностей у даних та формуванні однорідних груп об’єктів без попереднього знання їхньої приналежності. Визначено, що кластеризація є одним із базових методів машинного навчання без учителя, який забезпечує виявлення структур у багатовимірних просторах ознак.

Формалізовано основні етапи алгоритмів на основі щільності — обчислення локальної щільності, визначення відстаней між об’єктами, виявлення центрів кластерів та призначення інших точок до найближчих центрів за певним правилом подібності. Розроблено графічне представлення принципів роботи алгоритму кластеризації на основі щільності, що дало змогу візуалізувати динаміку процесу групування.

РОЗДІЛ 2. ДОСЛІДЖЕННЯ МЕТОДІВ ТА МЕТОДОЛОГІЙ КЛАСТЕРИЗАЦІЇ ДАНИХ НА ОСНОВІ ВЛАСТИВОСТІ ЩІЛЬНОСТІ

2.1. Концептуальні основи та формалізація кластеризації на основі щільності

2.1.1. Визначення та переваги методу

Кластеризація на основі щільності є непараметричним підходом в аналізі даних, де кластери визначаються як області високої щільності в просторі даних. Ключові переваги цього методу полягають у тому, що він не вимагає попереднього задання кількості кластерів і не робить припущень щодо форми або статистичного розподілу набору даних. Це дозволяє алгоритму ефективно виявляти кластери довільної форми, що є значною перевагою порівняно з центроїдними методами (наприклад, k-середні).

2.1.2. Критерії та параметризація

Концептуально, виявлені кластери можна розглядати як результат розрізу (відсікання) функції щільності на певному рівні λ (порогу щільності). Кожен зв'язний регіон, де щільність перевищує λ, відповідає окремому кластеру.

У найпростіших реалізаціях, таких як DBSCAN, для визначення щільності та зв'язності необхідні два вхідні параметри, які задаються користувачем:

1. Мінімальна кількість зразків (MinPts) - необхідна для того, щоб зразок вважався основною (core) точкою (тобто, щільною).

2. Радіус (ϵ або d_c) визначає околицю, в межах якої підраховується MinPts.

Вибір відповідного рівня відсікання λ має вирішальне значення. Надто низький рівень може призвести до злиття різних кластерів в одну область

(over-clustering), тоді як надто високий рівень може спричинити втрату кластерів з нижчою внутрішньою щільністю або фрагментацію кластерів.

2.1.3. Формалізація кластера контуру щільності

Основне припущення методу полягає в тому, що набір даних є вибіркою з деякої невідомої ймовірнісної щільності, а кластери — це високощільні регіони цього розподілу.

Формалізація кластера на основі щільності [6] визначає кластер контуру щільності на рівні λ як максимально зв'язний набір точок xi, для яких локальна щільність $\rho(x_i)$ перевищує поріг λ.

Для реалізації цього припущення необхідно:

- Локальна оцінка щільності - обчислення $\rho(x)$ для кожної точки.

- Визначення зв'язності - встановлення критерію, за яким точки вважаються зв'язаними (наприклад, якщо відстань між ними менша за dc).

Кластери будуються як набори об'єктів, які є зв'язаними з об'єктами, чия щільність перевищує порогове значення λ. Об'єкти, що не належать до таких зв'язних високощільних областей, класифікуються як шум або викиди.

Різні алгоритми кластеризації на основі щільності, розроблені в літературі [7 - 8], відрізняються головним чином методами локальної оцінки щільності, алгоритмами для знаходження зв'язних компонентів та конкретним визначенням зв'язності.

Цей клас кластеризації часто використовується в застосуваннях, де структури даних формуються природними фізичними або географічними процесами (наприклад, визначення доріг, річок, або вулканічних регіонів), і в таких контекстах іноді називається "природними кластерами" [4].

2.2. Алгоритми кластеризації на основі щільності

2.2.1. Алгоритм DBSCAN

Алгоритм DBSCAN (Density-Based Spatial Clustering of Applications with Noise) є одним із найбільш відомих представників методів кластеризації

на основі щільності. Його механізм дозволяє використовувати індексні структури для ефективної оцінки локальної щільності.

Локальна щільність - кількість точок x_j, що знаходяться у є-околиці точки x_i, де є (або d_c) — заданий поріг відстані.

Ядрова точка (Core Point) - точка x_i, локальна щільність якої перевищує поріг щільності MinPts.

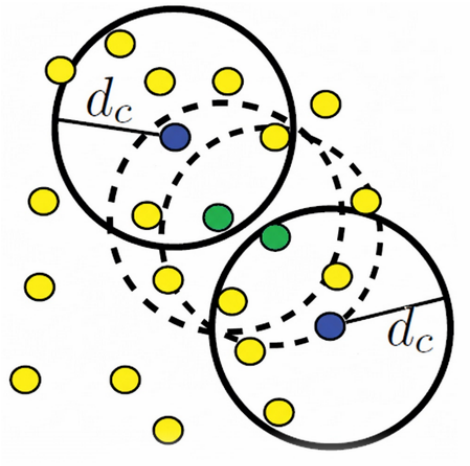


Рис. 2.1. Схематичне зображення зв'язності щільності. Сині точки — ядрові точки, зелені — точки на межах, жовті — інші точки

Визначення зв'язності:

1. Пряма зв'язність за щільністю (Directly Density-Reachable) - точка x_j прямо досяжна за щільністю від ядрової точки x_i, якщо x_j знаходиться у є-околиці x_i.

2. Зв'язність за щільністю (Density-Reachable) - точка x досяжна за щільністю від точки y, якщо існує послідовність ядрових точок (ланцюг) p_1 ,…,p_n = y, де p_1 = x і p_{i+1} прямо досяжна від pi.

3. Зв'язаність за щільністю (Density-Connected) - дві точки x_i та x_j вважаються зв'язаними за щільністю, якщо існує третя ядрова точка x_k, від якої обидві x_i та x_j є досяжними за щільністю.

Структура кластера буде наступною:

- Кластер C визначається як максимально зв'язаний за щільністю набір точок.

- Кластери можуть містити прикордонні точки (Border Points), які є досяжними за щільністю від ядрової точки, але самі не відповідають критерію ядрової точки.

- Об'єкти, які не є частиною жодного кластера, класифікуються як шумові точки (Noise Points).

Процедура кластеризації наступна. DBSCAN ініціює новий кластер C з невизначеної ядрової точки x, послідовно призначаючи до C усі точки, які є зв'язаними з x за щільністю. Кожна ядрова точка "сканується" в радіусі є для визначення нових точок, які слід додати до кластера. При використанні індексних структур для пошуку сусідів, часова складність алгоритму становить O(NlogN) (де N — кількість елементів), хоча за умови послідовного пошуку вона може сягати O(N2).

2.2.2. Алгоритм DENCLUE (Density-Based Clustering)

Алгоритм DENCLUE використовує більш узагальнений підхід, що базується на оцінюванні ядерної щільності (Kernel Density Estimation, KDE).

Щільність даних оцінюється за допомогою функцій ядра. Можуть використовуватися різні ядра, наприклад, Гаусове ядро або рівномірне ядро (SquareWave kernel).

Атрактор щільності (Density Attractor) визначається як локальний максимум оціненої функції щільності. Ці атрактори слугують центрами кластерів. Кожна точка x асоціюється з атрактором щільності, до якого вона сходиться шляхом руху в напрямку максимального градієнта щільності (процедура сходження на пагорб, hill-climbing).

Кластер визначається як зв'язний компонент атракторів щільності та всіх асоційованих з ними точок, за умови, що їхня оцінка щільності перевищує заданий поріг λ.

Реалізація DENCLUE, як правило, спирається на Гаусове ядро та використовує складну структуру даних (наприклад, дерево-R або сітку) для швидкого та ефективного обчислення локальної оцінки щільності, що має вирішальне значення для продуктивності процедури сходження на пагорб.

2.3. Методологія експериментальних досліджень та тестові набори даних

Для емпіричної оцінки та порівняльного аналізу алгоритмів кластеризації було використано чотири стандартні синтетичні та реальні набори даних. Обчислювальні експерименти проводилися в середовищі MATLAB.

2.3.1. Тестові набори даних

У дослідженні було задіяно такі набори даних: Aggregation, Jain, Flame та Spiral опис яких і графічне представлення показано нижче.

Таблиця 2.1.

Опис тестових наборів даних

Набір даних	**Призначення та контекст використання**
Aggregation	Використовувався як еталонний набір для задачі агрегаційної кластеризації, демонструючи складну структуру з чітко розділеними, але тісно розташованими кластерами
Jain	Часто застосовується у контексті генетичних алгоритмів для кластеризації, характеризується наявністю нелінійно розділених кластерів
Flame	Походить із домену аналізу даних мікрочипів, використовується для оцінки здатності алгоритмів обробляти кластери складної форми з високим рівнем перекриття
Spiral	Слугує основним тестовим прикладом для спектральної кластеризації та методів кластеризації на основі шляхів, оскільки містить чітко розділені, але високо нелінійні (спіральні) кластери

Набір даних Aggregation (Aggregation Dataset) у контексті кластеризації зазвичай позначає еталонний тестовий набір даних зі складною, специфічною структурою, який використовується для оцінки здатності алгоритмів виявляти численні, тісно розташовані, але чітко розділені кластери.

Цей набір даних був розроблений для тестування алгоритмів кластеризації, особливо тих, які здатні ідентифікувати кластери довільної форми, і часто використовується для оцінки методів агрегаційної кластеризації та метакластеризації.

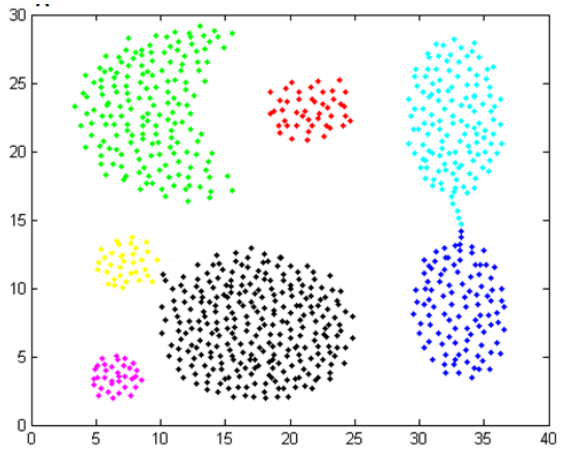


Рис. 2.2. Набір даних Aggregation для тестування різних алгоритмів кластеризації даних

Набір складається з 7 чітких кластерів.

Кластери мають нетривіальну геометрію і розташовані досить близько один до одного. Це створює виклик для традиційних алгоритмів (наприклад, k-середніх), які можуть зливати ці групи, оскільки вони віддають перевагу сферичним кластерам.

Успішна кластеризація на цьому наборі вимагає від алгоритму чітко розрізняти межі між кластерами, які є тісно прилеглими, але структурно незалежними.

Набір Aggregation є стандартним бенчмарком для демонстрації переваг методів, базованих на щільності (як-от DBSCAN та CFSFDP), оскільки вони

можуть ефективно виявляти ці складні, нелінійно розділені структури без попереднього задання кількості кластерів.

Таким чином, Aggregation Dataset слугує критично важливим інструментом для оцінки робастності та точності кластерних алгоритмів у складних умовах.

Набір даних Jain — це класичний, синтетично згенерований бенчмарк для оцінки ефективності алгоритмів кластеризації. Він спеціально розроблений, щоб продемонструвати обмеження традиційних методів і підкреслити переваги методів, які можуть працювати з кластерами нетривіальної форми.

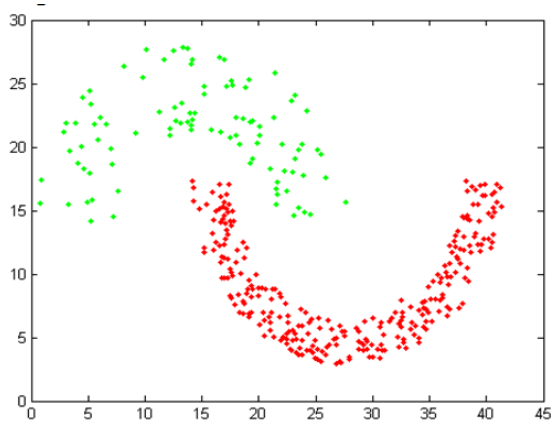


Рис. 2.3. Набір даних Jain для тестування різних алгоритмів кластеризації даних

Ключові характеристики:

1. Набір складається лише з двох основних кластерів.

2. Кластери мають нелінійно розділену та складну геометричну форму. Вони часто візуалізуються як дві "півмісяцеві" або вигнуті структури.

Складність кластеризації полягає в наступному:

- Традиційні методи (наприклад, k-середні). Алгоритми, що припускають сферичну (опуклу) форму кластерів, майже завжди невдало кластеризують цей набір. Вони зазвичай проводять пряму лінію, що розділяє

дані на дві частини, але ця лінія проходить через один із "півмісяців", змішуючи точки, які належать до одного кластера.

- Методи на основі щільності (CFSFDP, DBSCAN, Spectral Clustering). Ці методи є ідеальними для набору Jain, оскільки вони можуть ідентифікувати кластери, ґрунтуючись на щільності або зв'язності, а не на відстані до центроїда. Вони успішно виявляють вигнуту структуру, що розділяє два "півмісяці".

Набір Jain є фундаментальним інструментом для візуальної та кількісної демонстрації того, наскільки добре алгоритм справляється зі структурами даних, які не є опуклими.

На графіку (рис. 2.3) видно дві вигнуті, переплетені структури (дві групи точок), які, хоча і є єдиним кластером, не можуть бути розділені прямою лінією.

У галузі непараметричної кластеризації та оцінки алгоритмів на основі щільності, "Flame" зазвичай посилається на синтетичний 2D-набір даних, створений для демонстрації роботи алгоритму FLAME (Fuzzy clustering by Local Approximation of MEmberships).

Ключові характеристики:

- містить два чіткі кластери.

- кластери мають вигнуту, S-подібну форму та тісно переплітаються по центру.

Як і набір Jain, набір Flame використовується для тестування алгоритмів, які можуть ідентифікувати кластери довільної форми та правильно розділяти їх у точках мінімальної щільності, де кластери проходять дуже близько.

Складність полягає в тому, що алгоритми, які використовують лише центроїди (наприклад, k-середні), або ті, що недостатньо чутливі до локальної щільності, можуть помилково злити ці дві структури або розділити їх неправильною прямою лінією.

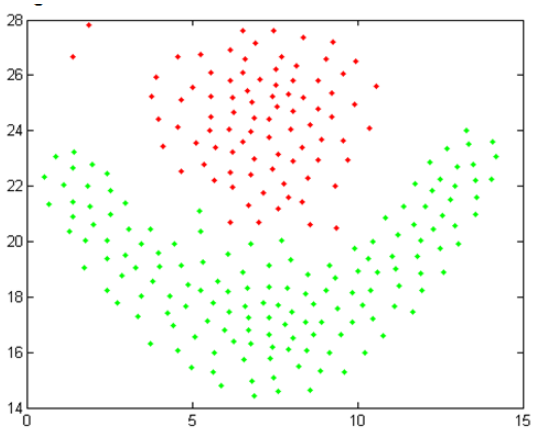


Рис. 2.4. Набір даних Flame для тестування різних алгоритмів кластеризації даних

Набір даних Spiral (Two Spirals або Three Spirals) — це відомий синтетичний бенчмарк для завдань класифікації та кластеризації, розроблений для демонстрації слабкості лінійних моделей.

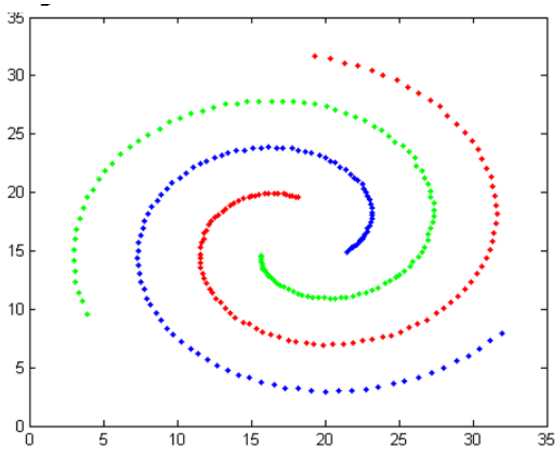


Рис. 2.5. Набір даних Spiral для тестування різних алгоритмів кластеризації даних

Кількість Кластерів/Класів:

- Two Spirals - найпоширеніший варіант, що містить два класи точок.

- Three Spirals - варіант, що містить три класи точок, які спірально віддаляються від центру.

Дані організовані у вигляді кількох спіральних рукавів, що виходять із спільного центру і обертаються навколо нього. Кластери не є лінійно роздільними (non-linearly separable). Неможливо провести одну або навіть кілька прямих ліній, щоб ідеально відокремити один спіральний рукав від іншого.

Набір Spiral є головним тестом для оцінки здатності алгоритму знаходити складні, нелінійні межі рішень. Традиційні лінійні моделі, такі як перцептрон або лінійний SVM, повністю провалюються на цьому наборі. Для успішної класифікації або кластеризації необхідні нелінійні методи, такі як:

- багатошарові нейронні мережі (MLP) з прихованими шарами та нелінійними функціями активації (наприклад, ReLU, Tanh).

- метод опорних векторів (SVM) з нелінійним ядром, таким як RBF-ядро.

- спектральна кластеризація (Spectral Clustering), яка ефективно працює з нелінійно зв'язаними структурами.

2.3.2. Процедура тестування

Експериментальне дослідження включало застосування алгоритмів кластеризації, базованих на щільності, до вказаних наборів даних. Для оцінки чутливості та робастності алгоритмів було проведено тестування з варіацією критичного параметра d_c (поріг відстані). Порівняльний аналіз результатів, отриманих при різних значеннях d_c, дозволив визначити оптимальні параметри та оцінити ефективність кожного методу.

2.4. Методологія кластеризації шляхом швидкого пошуку та знаходження піків щільності

У цьому розділі представлено детальний опис алгоритму кластеризації шляхом швидкого пошуку та знаходження піків щільності (Clustering by Fast Search and Find of Density Peaks, CFSFDP). Цей метод є альтернативним

підходом до традиційних стратегій кластеризації, що демонструє здатність виявляти не сферичні кластери та автоматично визначати кількість кластерів на основі локальних максимумів щільності.

2.4.1. Теоретична основа та визначення

CFSFDP базується на фундаментальному припущенні, що центри кластерів характеризуються двома взаємопов'язаними властивостями:

1. Висока локальна щільність - вони оточені більшою кількістю сусідів порівняно з периферійними точками.

2. Велика відносна відстань - вони знаходяться на значній відстані від будь-якої іншої точки, що має вищу локальну щільність.

Для кожного об'єкта i у наборі даних обчислюються дві величини, що залежать виключно від матриці відстаней між об'єктами: локальна щільність (ρ_i) та мінімальна відстань до точок з вищою щільністю (δ_i).

Локальна щільність ρ_i визначається як кількість об'єктів j, які розташовані в межах відстані відсічення d_c від об'єкта i:

$$\rho_i = \sum_{j=1}^{n-1} \chi\left(d_{ij} - d_c\right)$$

де:

n — загальна кількість точок даних.

d_{ij} — відстань між об'єктами i та j.

$\chi(x) = 1$, якщо $x < 0$ (тобто $d_{ij} < d_c$), і $\chi(x)=0$ в іншому випадку.

Таким чином, ρ_i є простим лічильником сусідів у є-околиці з радіусом d_c.

Відстань до точки з вищою щільністю (δi) δ_i розраховується як мінімальна відстань від об'єкта i до будь-якого іншого об'єкта $_j$, що має вищу локальну щільність ($\rho_j > \rho_i$):

$$\delta_{i^*} = \max_j \left(d_{i^* j}\right)$$

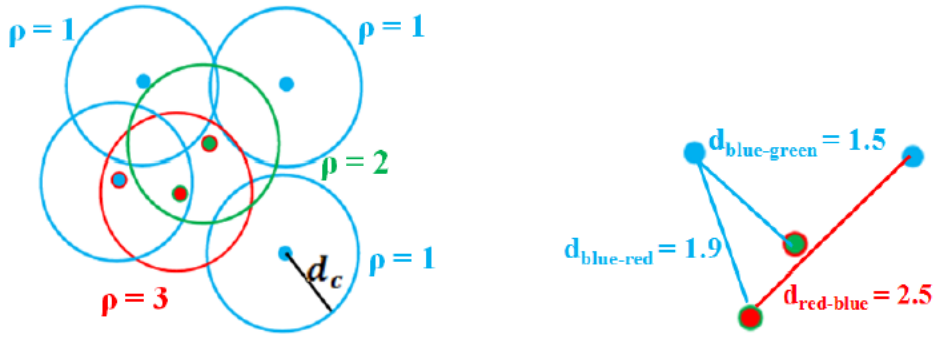
Для об'єкта i*, що має глобально найвищу щільність у наборі даних, δ_i* встановлюється як максимальна відстань до будь-якого іншого об'єкта:

$$\delta_{i^*} = \max_j (d_{i^* j})$$

Це гарантує, що для потенційного центру кластера δ буде значно більшим, ніж відстань до його найближчого сусіда.

2.4.2. Ідентифікація центрів кластерів (графік рішення)

Ідентифікація центрів кластерів ґрунтується на аналізі простору (ρ,δ). Центри кластерів відповідають об'єктам, які демонструють високі значення як ρ, так і δ.



а) б)

Рис. 2.6. Схематичне зображення розрахунку ρі а), схематичне зображення розрахунку δі б)

На рис. 2.6. δ_{red} (для точки з найвищою щільністю) визначається як відстань до найвіддаленішої точки, тоді як δ_{blue} (для точки з нижчою щільністю) визначається як мінімальна відстань до точки з вищою щільністю.

Графік рішення (δ проти ρ) є ключовим інструментом для візуального та евристичного відбору центрів кластерів:

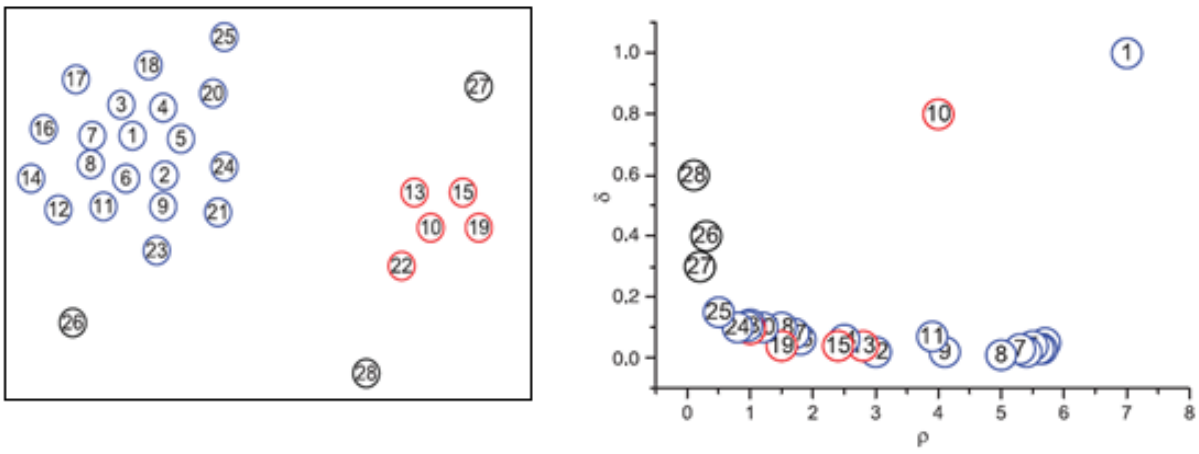
- Центри кластерів - об'єкти, що мають найбільші значення ρ та δ, чітко відокремлюються у верхньому правому квадранті графіка (наприклад, точки 1 та 10 на рис. 2.7 б).

- Точки тіла кластерів - об'єкти з високою ρ та низькою δ (наприклад, точки 7, 8, 9, 13, 15, 22). Вони знаходяться близько до локальних максимумів (центрів).

- Викиди/шум - об'єкти з низькою ρ та високою δ (якщо вони знаходяться далеко від будь-яких інших точок) або низькою ρ та низькою δ (шум).

Очевидно, що ефективність цього методу критично залежить від вибору відстані відсічення d_c.

З викладеного вище зрозуміло, як цей метод сильно залежить від вибраної відстані відсічення, але це дозволяє відрізнити кластер за вищими значеннями як ρ, так і δ. На рис. 2.7 показано набір точок даних та їхній графік рішення, δ проти ρ. З графіка рішення легко відрізнити, які точки є центрами кластера через їхні вищі δ та ρ.



а) б)

Рис. 2.7. а) Розподіл набору даних. б) Графік рішення щільності (δ проти ρ), що демонструє чітке відділення центрів кластерів (точки 1 та 10) від точок тіла кластерів та викидів

Точки з високим ρ та низьким δ належать до існуючих кластерів. Точки 7, 8 та 9 належать до кластера з центром у точці 1 (червоного кольору). Точки 13, 15 та 22 належать до кластера з центром у точці 10 (синього кольору). Точки з низьким ρ та високим δ вважаються викидами (чорного кольору).

2.5. Налаштування параметрів та емпіричне застосування алгоритму кластеризації

2.5.1. Визначення оптимального параметра

Критичним етапом у застосуванні алгоритму CFSFDP є вибір відстані відсічення d_c. Емпіричні дослідження демонструють, що оптимальне значення d_c відповідає умові, за якої середня кількість сусідів (тобто середня ρ_i) становить приблизно 1–2% від загальної кількості об'єктів у наборі даних. Дотримання цієї евристики допомагає забезпечити надійну оцінку локальної щільності.

Слід враховувати, що для малих наборів даних оцінка ρі може мати високу статистичну похибку, що ускладнює точну ідентифікацію піків щільності.

2.5.2. Інтерактивний вибір центрів кластерів

Після обчислення характеристик ρ та δ для всіх точок, ідентифікація центрів кластерів вимагає інтерактивної участі користувача та аналізу графіка рішення (δ проти ρ).

Процедура вибору наступна.

У середовищі, наприклад, MATLAB, користувач має графічно визначити прямокутну область на графіку рішення (рисунок 2.8). Ця область встановлює порогові значення $\rho_{minimum}$ та $\delta_{minimum.}$

Потім усі точки, що потрапляють у визначену область (тобто мають $\rho \geq \rho_{minimum}$ та $\delta \geq \delta_{minimum}$), обираються як центри кластерів.

Щодо впливу користувача, то оскільки вибір цих порогових значень є суб'єктивним, розмір і кількість виявлених кластерів безпосередньо залежать від рішення оператора.

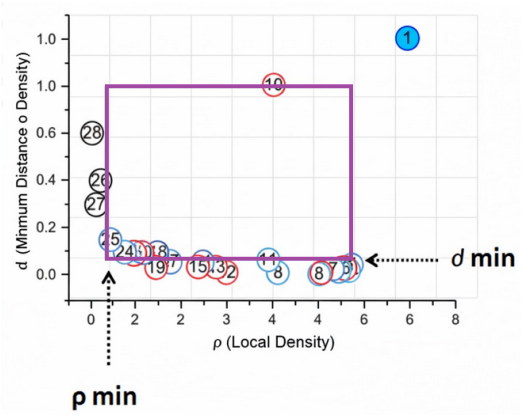


Рис. 2.8. Прямокутник, обраний користувачем для визначення $\rho_{minimum}$ та $\delta_{minimum}$.

2.5.3. Результати застосування адгоритму CFSFDP до тестових наборів даних

Метод CFSFDP був застосований до чотирьох тестових наборів даних для демонстрації його ефективності у виявленні кластерів складної форми.

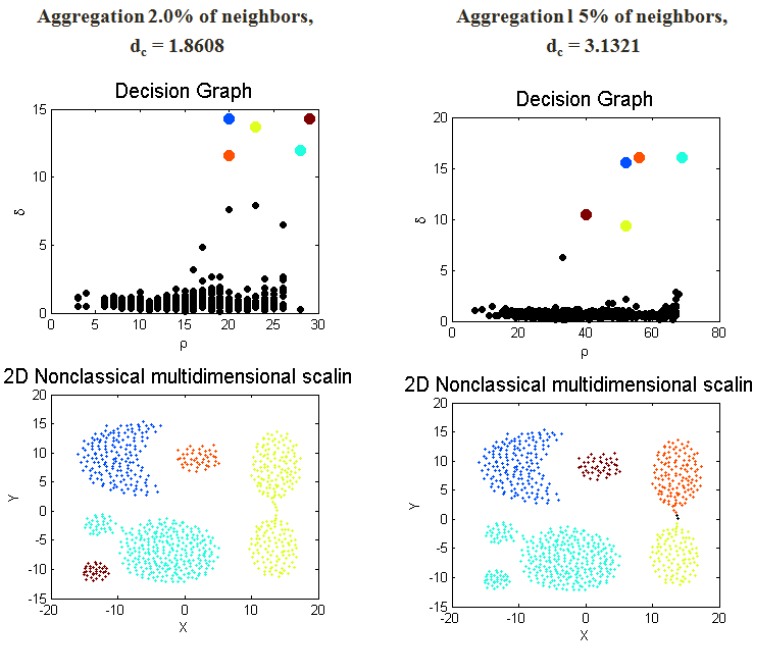


Рис. 2.9. Набір даних Aggregation, застосовано метод CFSFDP

На наборі Aggregation (рис. 2.9) метод успішно виявив усі кластери, які тісно розташовані.

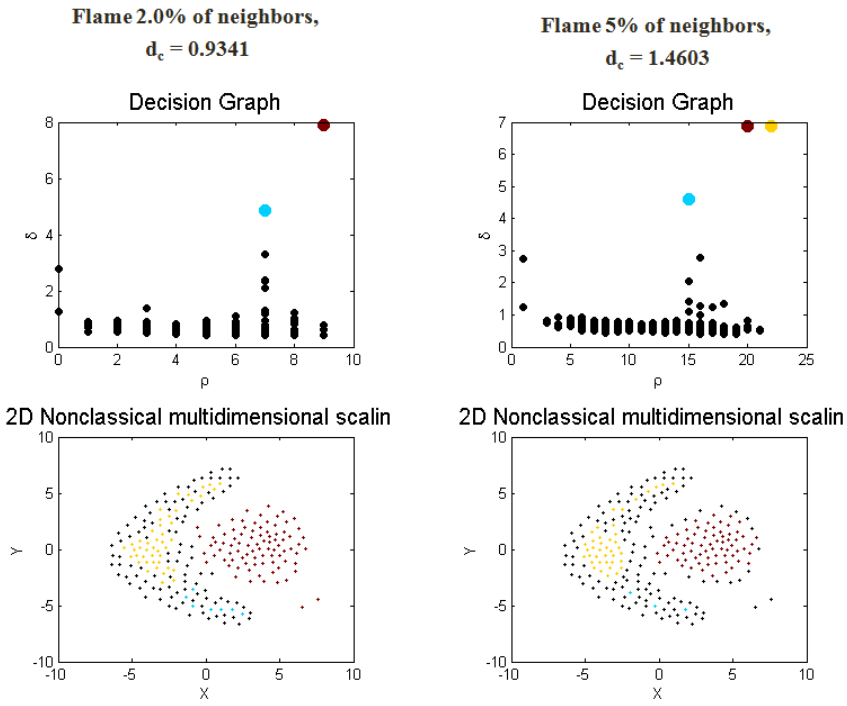


Рис. 2.10. Набір даних Flame, застосовано метод CFSFDP

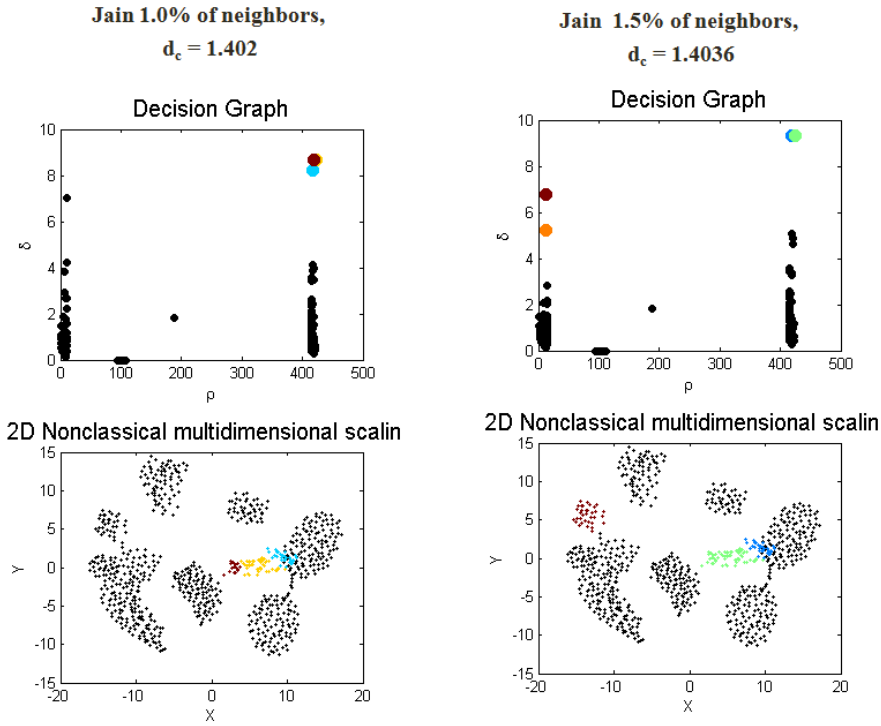


Рис. 2.11. Набір даних Jain, застосовано метод CFSFDP

Набір Flame (рис. 2.10) продемонстрував здатність розрізняти два перекриваючихся, але чітко розділених кластери, а при застосуванні набору Jain (рис. 2.11) - алгоритм ефективно ідентифікував два нелінійно розділені кластери.

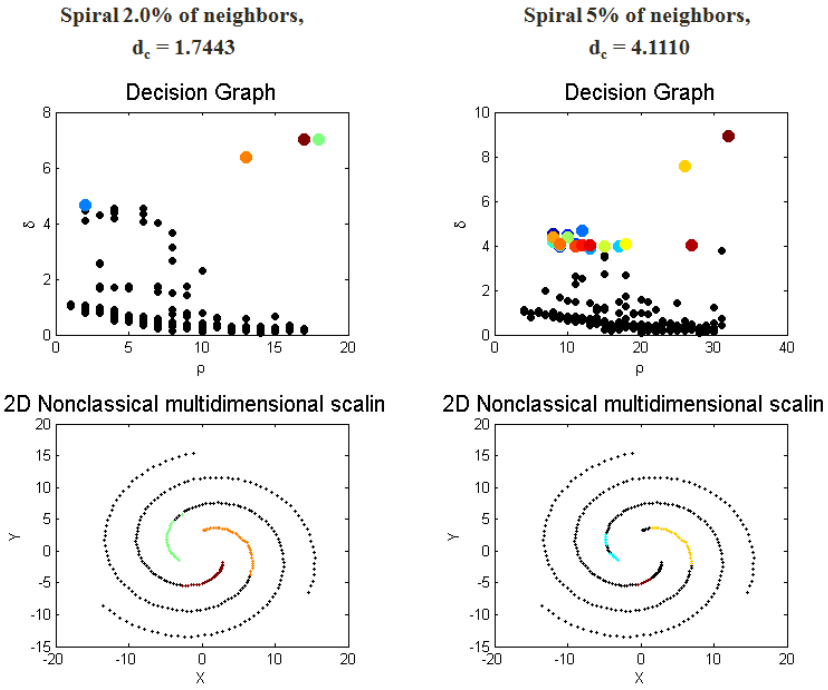


Рис. 2.12. Набір даних Spiral, застосовано метод CFSFDP

Щодо набору Spiral (рис. 2.12), то показано здатність CFSFDP виявляти кластери довільної, не опуклої форми (спіральні кластери), що є перевагою над традиційними центроїдними методами.

Висновки до розділу

Другий розділ присвячено глибокому дослідженню математичних моделей і методів кластеризації даних, що базуються на властивості щільності розподілу. Проведено формалізацію основних понять і критеріїв, які описують щільність у багатовимірному просторі, та розглянуто параметризацію, що визначає якість виділення кластерів.

Поглиблено аналіз відомих алгоритмів — DBSCAN та DENCLUE. Визначено їхні переваги у виявленні кластерів довільної форми та обмеження, пов’язані з вибором параметрів ε (радіус сусідства) та MinPts (мінімальна кількість точок у кластері). Для DENCLUE показано потенціал використання ядерних функцій для неперервного оцінювання щільності та підвищення стабільності результатів. У результаті проведених досліджень сформовано теоретичну основу для розроблення удосконалених моделей кластеризації на основі властивості щільності, що враховують локальні особливості розподілу даних та параметричну гнучкість у виборі метрик.

РОЗДІЛ 3. ПРОЄКТУВАННЯ, РЕАЛІЗАЦІЯ ТА АНАЛІЗ МОДИФІКАЦІЙ АЛГОРИТМІВ КЛАСТЕРИЗАЦІЇ ДАНИХ НА ОСНОВІ ВЛАСТИВОСТІ ЩІЛЬНОСТІ

Алгоритм кластеризації шляхом швидкого пошуку та знаходження піків щільності (CFSFDP) був реалізований та модифікований у програмному середовищі MATLAB з метою підвищення його робастності та незалежності від критичних параметрів. Оригінальна концепція CFSFDP була збережена, але були внесені зміни до методів оцінки локальної щільності.

3.1. Модифікація алгоритму кластеризації на Основі k-найближчих сусідів

3.1.1. Методологія модифікації

Однією з основних проблем класичного CFSFDP є чутливість до вибору жорсткого порогу відстані відсічення d_c. Для подолання цієї залежності було адаптовано підхід, який замінює жорстке підрахування сусідів у межах d_c на метрику, засновану на k-найближчих сусідах (kNN).

Модифікована оцінка локальної щільності (ρi) визначається не як кількість сусідів, а як середня відстань до M найближчих сусідів:

$$\rho_i = \frac{1}{M} \sum_{j=1}^{M} (d_{ij})$$

де:

M — кількість найближчих сусідів, обраних для оцінки щільності.

d_{ij} — відстань до j-го найближчого сусіда.

Кількість сусідів M встановлюється як фіксований відсоток (m_{ratio}) від загальної кількості точок N у наборі даних:

$$M = m_{\text{ratio}} \times N$$

Ця модифікація ефективно перетворює метрику щільності з лічильника фіксованого радіусу на метрику, базовану на адаптивній середній відстані, роблячи оцінку ρі більш стабільною та менш залежною від одного жорсткого глобального параметра. При цьому визначення δ_i (відстань до точки з вищою щільністю) залишається незмінним, зберігаючи основний принцип CFSFDP.

3.1.2. Експериментальні результати

Застосування kNN-CFSFDP до тестових наборів даних продемонструвало ефективність модифікації, особливо у випадках кластерів складної форми.

На наборі даних Aggregation (рисунок 3.1) спостерігається успішна ідентифікація всіх відомих кластерів.

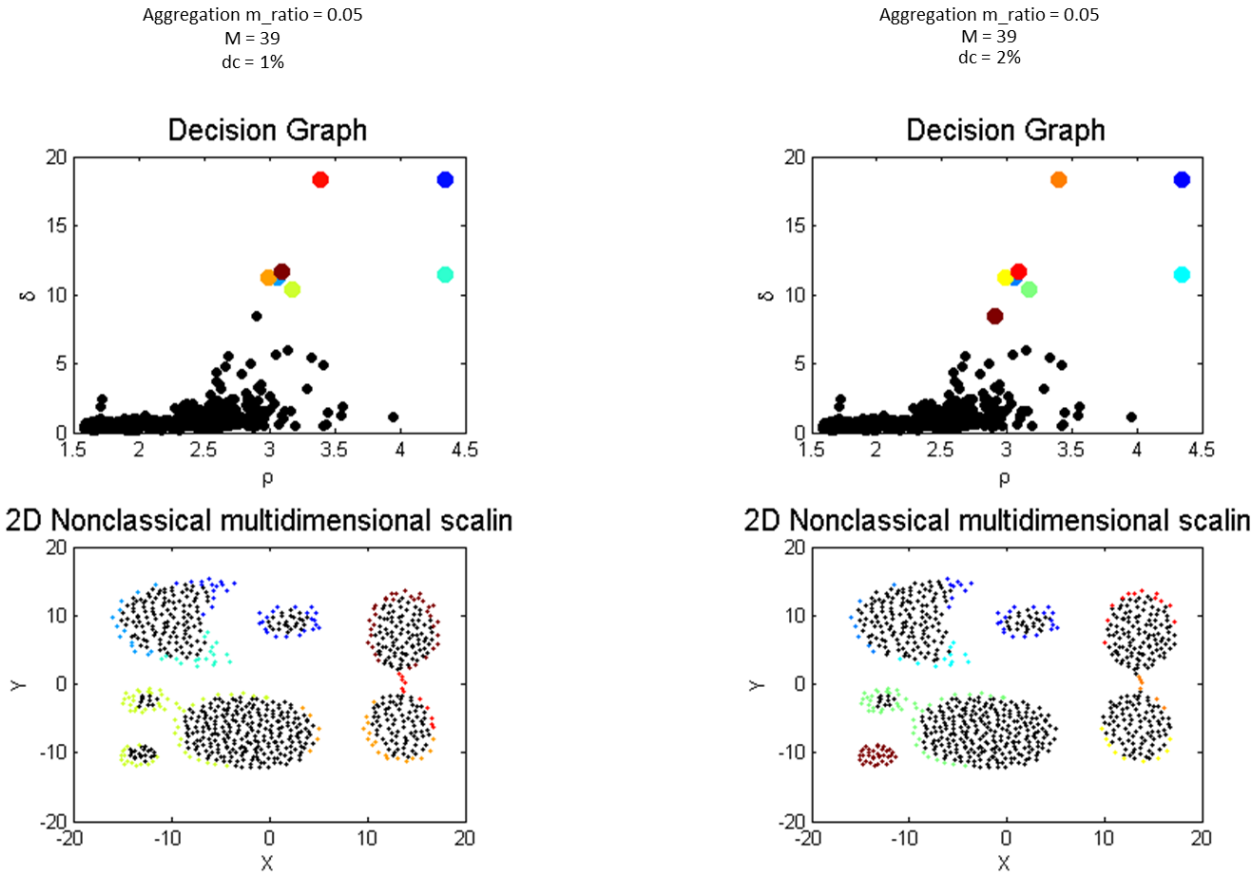


Рис. 3.1. Набір даних Aggregation, застосовано метод kNN-CFSFDP

Верхні графіки представляють простори рішень, де кожна точка даних відображається за двома ключовими показниками, які визначають її приналежність до кластера:

- Локальна щільність (ρ) обчислена на основі кількості сусідів у межах радіусу d_c (або k найближчих сусідів).

- Мінімальна відстань (δ) - відстань до найближчої точки з вищою щільністю ρ.

Зміна параметра d_c з 1% на 2% (лівий vs. правий графік) демонструє наступне:

- Лівий графік (d_c=1%) - менший радіус d_c призводить до більш локалізованого обчислення щільності. Це формує більш витягнуту хмару точок. Сім центрів кластерів (точки з високими ρ та δ) є чітко відокремленими від основної маси точок даних, що свідчить про їхню виняткову позицію як центрів.

- Правий графік (d_c=2%): Збільшення радіусу d_c призводить до того, що більше точок вважаються сусідами. Це збільшує загальну локальну щільність ρ для багатьох точок, і, як наслідок, зменшує мінімальну відстань δ для центрів кластерів. Хоча сім центрів залишаються ідентифікованими, вони стають менш віддаленими від хмари шумів, що знижує контрастність Графіка Рішень.

Нижні графіки (рис. 3.) ілюструють кінцевий результат присвоєння міток кластерів у двовимірному просторі (X та Y) після визначення центрів і призначення решти точок.

Лівий результат (d_c = 1%) - метод успішно розділяє набір даних на сім коректних кластерів (позначених різними кольорами). Це підтверджує, що для цього значення dc алгоритм точно ідентифікував локальні піки щільності, здатні розділити складну геометрію даних.

Правий результат (d_c = 2%) - результат кластеризації залишається високоточним, також відокремлюючи сім кластерів.

Таким чином, візуалізація демонструє надійність методу CFSFDP/kNN-CFSFDP. Навіть при зміні параметра d_c у вузькому діапазоні (1% до 2%), що впливає на метрики ρ та δ, кінцева топологічна структура (кількість та форма кластерів) була коректно відновлена, підтверджуючи ефективність підходу для кластеризації даних із довільною формою.

На наборі Flame (рисунок 3.2) відбувається чітке розділення кластерів складної форми.

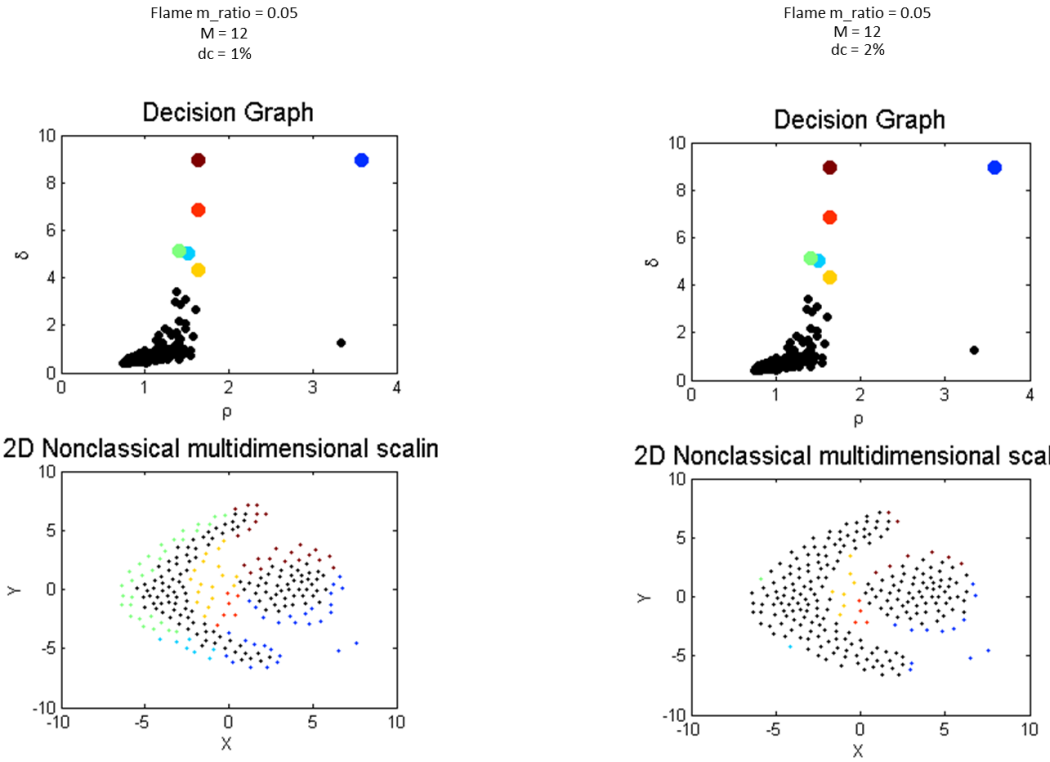


Рис. 3.2. Набір даних Flame, застосовано метод kNN-CFSFDP

Обидва верхні графіки показують графіки рішень, які використовуються для ідентифікації центрів кластерів.

Лівий графік (dc=1%).

Чітко виділяються дві домінантні точки (темно-синій та темно-бордовий/коричневий) з найвищими значеннями δ (мінімальна відстань до точок з вищою щільністю), які відповідають двом справжнім кластерам у наборі Flame.

Нижче вздовж осі δ видно кілька менш значущих піків (червоний, помаранчевий, жовтий, блакитний, зелений). Їхня присутність вказує на те, що два основні кластери складаються з кількох високощільних суб-регіонів або що dc=1% є занадто малим і ідентифікує локальні піки всередині справжніх кластерів.

Правий графік (d_c=2%).

Збільшення параметра d_c до 2% дещо зменшує контраст між піками (центрами), зсуваючи їх ближче до основи, але дві основні домінантні точки все ще залишаються на вершині, підтверджуючи два істинні кластери.

Нижні графіки показують розподіл точок після застосування алгоритму:

Лівий результат (d_c=1%).

Набір Flame містить лише два істинні кластери. Однак, на цьому графіку дані розділені на шість або сім окремих кольорових груп (кластерів), що відображає надмірну кластеризацію (over-clustering). Це прямий наслідок того, що метод вибрав у центри не лише два основні піки, а й вторинні (нижчі) піки щільності, ідентифіковані на графіку рішень.

Через низьке d_c=1% алгоритм розглядає вузькі "горловини" в структурі Flame як області низької щільності, що дозволяє йому розділяти кластери на окремі сегменти.

Правий результат (d_c=2%):

Хоча на графіку все ще присутні кілька кольорів, переважають дві основні кольорові групи (зелена/ жовта/ блакитна та червона/ бордова/ помаранчева), які формують два великі спіральні кластери.

Збільшення d_c до 2% зробило обчислення щільності менш локальним і допомогло з'єднати деякі суб-регіони, що призвело до кращого наближення до істинних двох кластерів набору Flame.

Отже, застосування методу до набору Flame демонструє його чутливість до вибору параметрів:

- Надто мале d_c призводить до фрагментації справжніх кластерів на кілька підгруп через надмірну чутливість до локальних варіацій щільності.

- Покращене розділення (d_c=2%): Збільшення d_c до 2% покращує результат, дозволяючи алгоритму відновити складну, вигнуту форму двох істинних кластерів, ігноруючи деякі внутрішні варіації щільності.

Це підкреслює, що для набору Flame, як і для більшості наборів даних із довільною формою, вибір оптимального параметра щільності (d_c) є критично важливим для отримання коректної структури кластеризації.

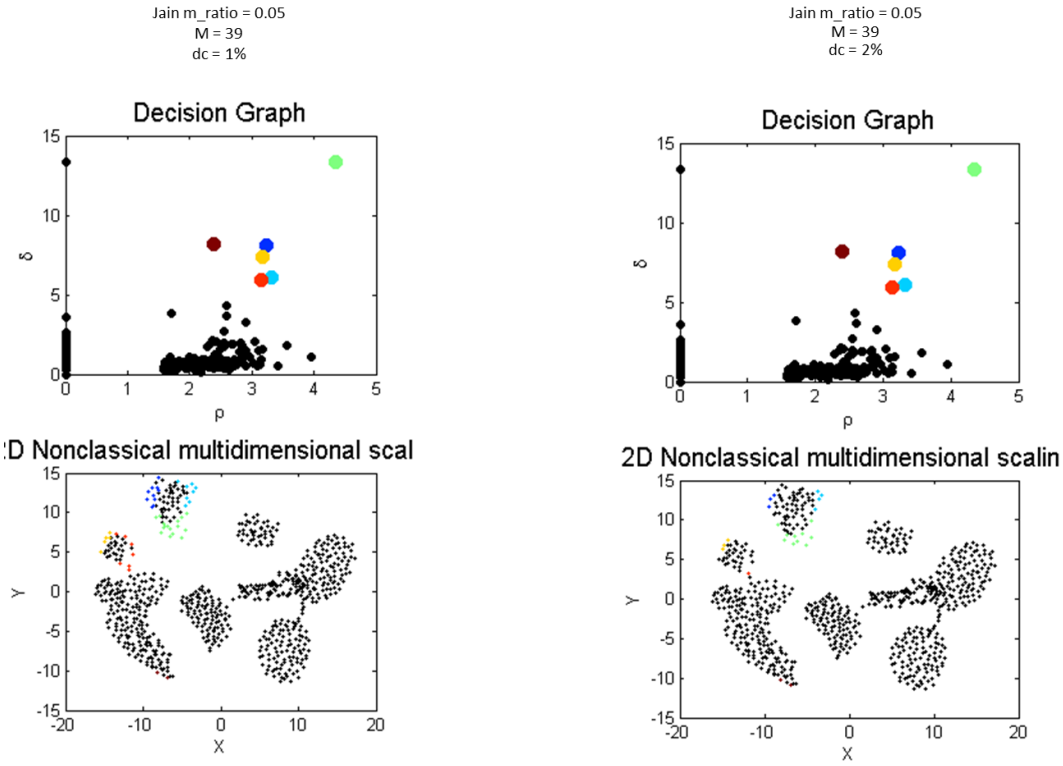


Рис. 3.3. Набір даних Jain, застосовано метод kNN-CFSFDP

Для набору Jain (рис. 3.3) - модифікований метод коректно виявив два нелінійні кластери.

Обидва верхні графіки показують розподіл точок за їхньою локальною щільністю (ρ) та мінімальною відстанню до точок з вищою щільністю (δ).

На лівому графіку (d_c=1%): Чітко виділяється один домінантний пік (світло-зелений) із найвищими значеннями ρ та δ. Це може свідчити про те,

що для цього параметра d_c алгоритм ідентифікує лише один справжній центр, тоді як інші піки (бордовий, синій, червоний) мають значно нижчі δ. Це вказує на те, що два істинні кластери Jain на цій візуалізації, ймовірно, фрагментовані на кілька підкластерів.

На правому графіку (d_c=2%) відбувається збільшення параметра d_c до 2% (збільшуючи радіус, що використовується для обчислення щільності) не призводить до значної зміни вигляду графіка рішень порівняно з d_c=1%. Це підтверджує, що топологічна структура даних у цьому вузькому діапазоні dc зберігається. Домінантний пік (світло-зелений) залишається головним кандидатом на центр.

По результатах кластеризації (2D-візуалізація), то нижні графіки показують, як алгоритм розділив набір даних Jain.

Лівий результат (d_c=1%).

На графіку видно, що два істинні, великі, вигнуті кластери Jain були надмірно кластеризовані (over-clustering), розділившись на багато дрібніших груп (близько 6-7).

Алгоритм не зміг з'єднати вигнуті частини кожного "спірального" кластера в єдине ціле. Натомість він ідентифікував кожен невеликий, щільний сегмент спіралі як окремий кластер (наприклад, червоний, оранжевий, жовтий сегменти в лівій частині). Цей результат свідчить про те, що параметр d_c=1% є занадто малим і робить алгоритм надмірно чутливим до локальних варіацій щільності.

Правий результат (d_c=2%) свідчить, що збільшення d_c до 2% не вирішило проблему надмірної кластеризації. Набір Jain все ще залишається розділеним на 6-7 сегментів, не відображаючи істинні дві вигнуті структури.

Отже, застосування методу до набору Jain, як показано на рисунку 3.3, демонструє некоректне відновлення істинної структури даних для обраного діапазону параметрів d_c. Замість того, щоб ідентифікувати два істинні, вигнуті кластери, алгоритм ідентифікує численні локальні піки щільності вздовж кривих, що призводить до фрагментації даних.

Це підтверджує, що для набору Jain потрібен ретельний підбір параметрів dc (можливо, значно більших, ніж 2%) або використання альтернативних методів, таких як спектральна кластеризація або методи, що краще враховують зв'язність між точками, для успішного з'єднання вигнутих структур.

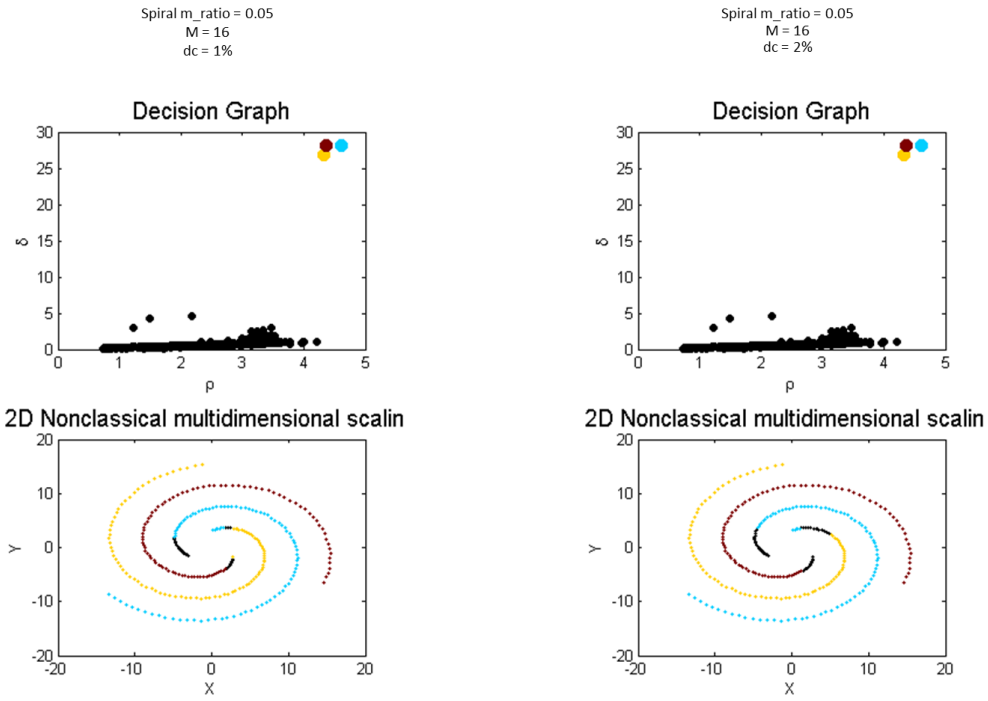


Рис. 3.4. Набір даних Spiral, застосовано метод kNN-CFSFDP.

Для набору Spiral (рисунок 3.4) метод підтвердив здатність виявляти складні, переплетені спіральні кластери, що свідчить про високу робастність модифікації до не опуклих структур даних.

Обидва верхні графіки показують, як алгоритм ідентифікує центри кластерів на основі щільності (ρ) та мінімальної відстані до точок з вищою щільністю (δ). На обох графіках (dc=1% і dc=2%) чітко виділяються три точки-кандидати на центри кластерів (темно-коричневий, синій та червоний) із найвищим значенням δ та відносно високим ρ.

Незначна зміна параметра d_c з 1% на 2% практично не впливає на позиції трьох точок-центрів на Графіку Рішень, що свідчить про стійкість

ідентифікації цих піків щільності незалежно від обраного радіусу d_c у цьому діапазоні.

Нижні графіки ілюструють, як метод розділив спіральний набір даних.

Лівий результат (d_c=1%).

Алгоритм розділив набір даних на чотири кластери (темно-коричневий, синій, червоний та жовтий/помаранчевий). Це є некоректним результатом для класичного набору Two Spirals або Three Spirals. Замість того, щоб з'єднати дві або три вигнуті структури, метод:

- розпізнав внутрішні витки спіралей як окремі кластери (наприклад, жовтий і блакитний сегменти).

- розділив спіральні рукави на окремі сегменти, фрагментуючи істинну структуру даних.

Правий результат (d_c=2%) свідчить про те, що збільшення d_c також призводить до некоректного розділення на чотири кластери, яке ідентичне результату при d_c=1%. Це підтверджує, що для набору Spiral обраний діапазон параметрів dc не дозволяє алгоритму адекватно з'єднати точки вздовж вигнутих "рукавів", щоб сформувати цілісні спіральні кластери.

Таким чином, застосування методу kNN-CFSFDP до набору Spiral на цій візуалізації демонструє його обмеження у випадку ідентифікації сильно витягнутих, переплетених структур. Алгоритм успішно знаходить кілька локальних піків щільності (три на графіку рішень), але через їхнє розташування вздовж спіралі він фрагментує спіральні структури на кілька сегментів (4 кластери), замість того, щоб об'єднати їх у 2 або 3 істинні спіралі.

3.2. Модифікація алгоритму кластеризації на основі гаусового ядра

3.2.1. Методологія модифікації

Друга модифікація алгоритму CFSFDP спрямована на впровадження методів оцінки ядерної щільності (Kernel Density Estimation, KDE) для більш

тонкого та зваженого визначення локальної щільності ρ_i. Цей підхід забезпечує визначення ймовірнісної щільності у кожній точці, базуючись на так званій функції впливу (influence function), яка моделює внесок кожної точки даних у щільність свого оточення.

Методи, що використовують оцінку щільності ядра, зокрема DBSCAN, можна розглядати як особливі випадки цього загального підходу. Ця модифікація CFSFDP може бути класифікована як алгоритм, що належить до родини DENCLUE, оскільки використовує ту ж методологію оцінки щільності. На відміну від підрахунку сусідів у межах жорсткого порогу d_c, локальна щільність ρ_i розраховується як сума функцій впливу (гаусового ядра) від усіх сусідніх точок:

$$\rho_i = \sum_{j=1}^{n-1} e^{-\left(\frac{d_{ij}}{d_c}\right)^2}$$

де:

d_{ij} — відстань між об'єктами i та j.

Використання Гаусового розподілу забезпечує зважений внесок кожного сусіда: чим ближча точка j до i (менше d_{ij}), тим більший її внесок у ρ_i.

Параметр відстані відсічення d_c тут функціонально відповідає стандартному відхиленню (σ) Гаусового ядра, описуючи ступінь згладжування або дисперсію даних, що впливає на форму функції щільності.

Як і в DENCLUE, атрактори щільності (центри кластерів) визначаються як локальні максимуми отриманої загальної функції щільності.

3.2.2. Експериментальні результати

Застосування модифікації CFSFDP на основі Гаусового ядра (GK-CFSFDP) до тестових наборів даних продемонструвало високу ефективність у виявленні кластерів складної геометрії.

Щодо набору Aggregation, який показано на рисунку 3.5, то модифікований метод на основі гаусового ядра успішно виявив усі агреговані

кластери, підтверджуючи його здатність до розділення тісно розташованих груп.

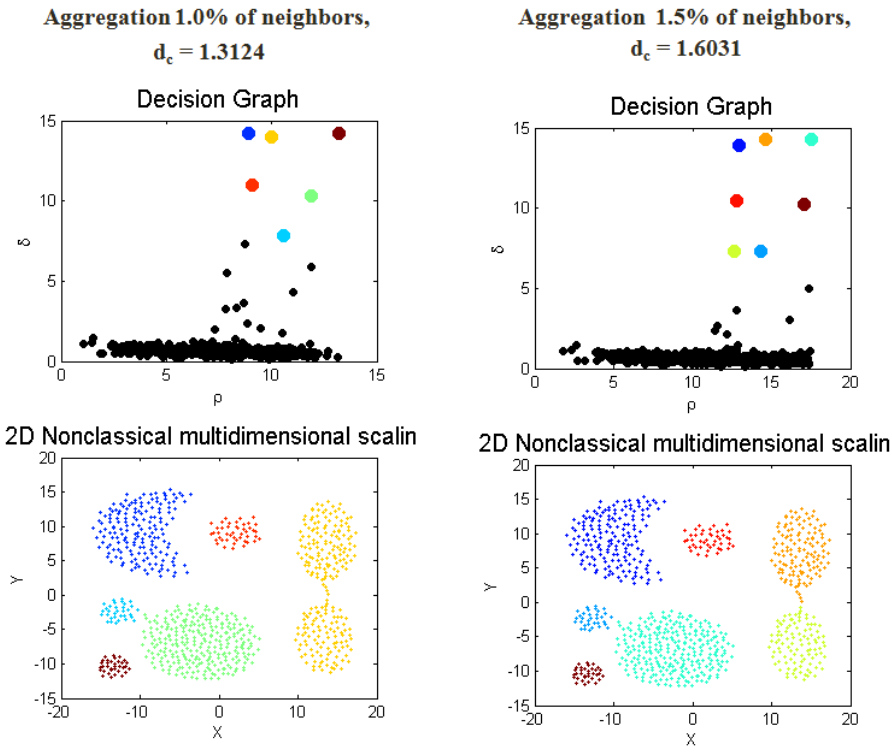


Рис. 3.5. Набір даних Aggregation, застосовано метод Гаусового ядра

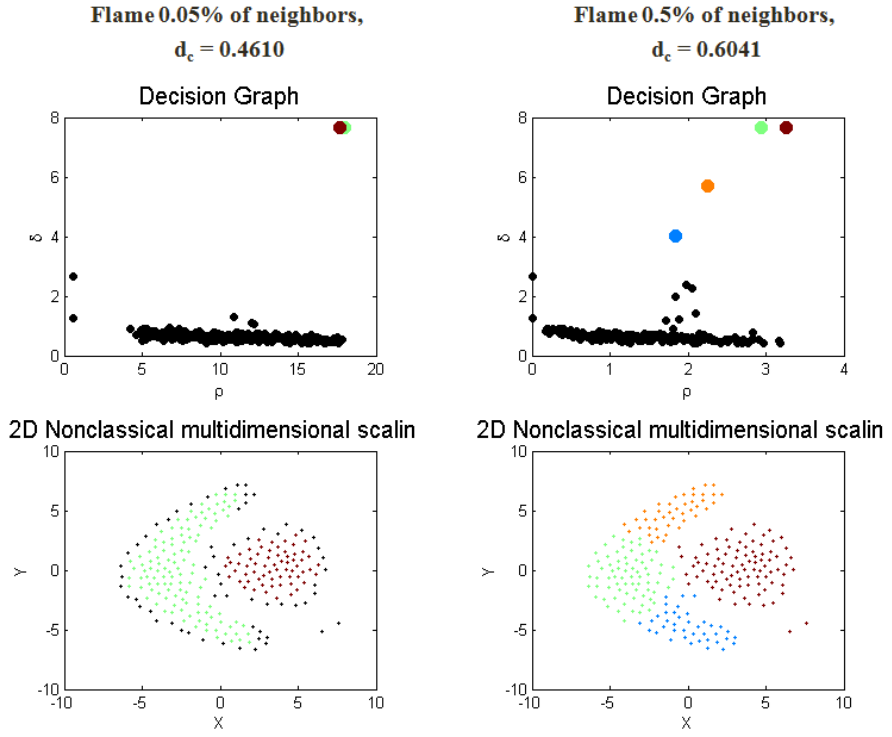


Рис. 3.6. Набір даних Flame, застосовано метод Гаусового ядра

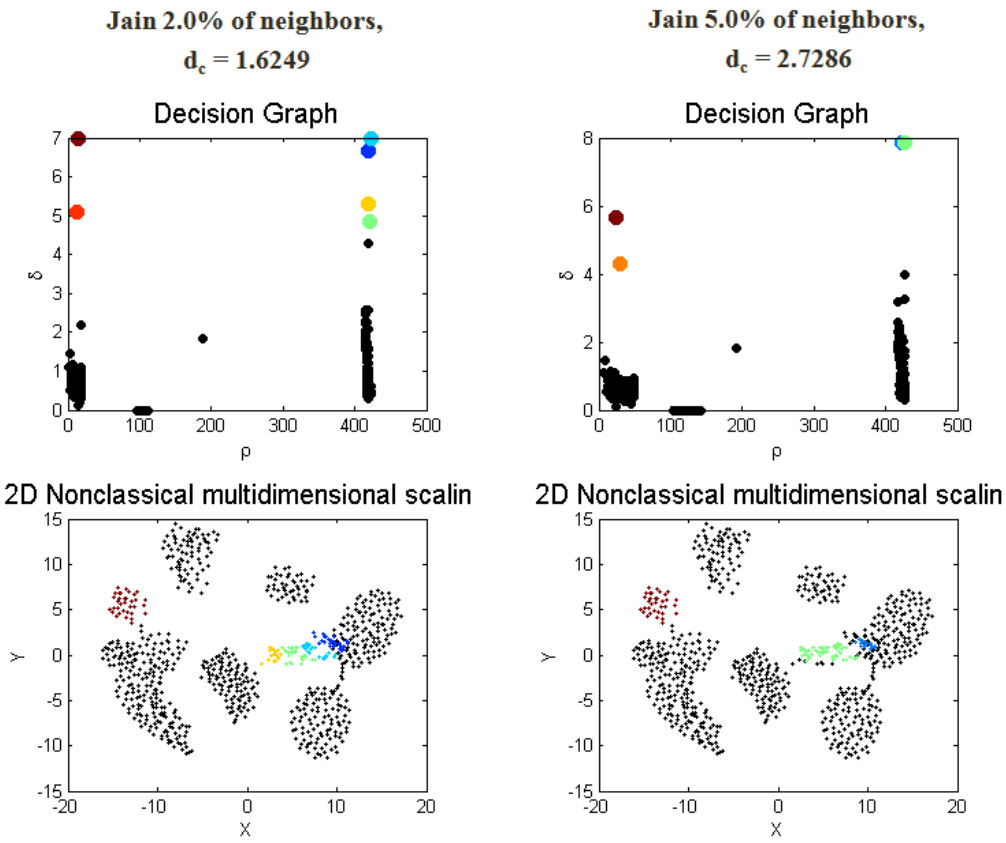


Рис. 3.7. Набір даних Jain, застосовано метод Гаусового ядра

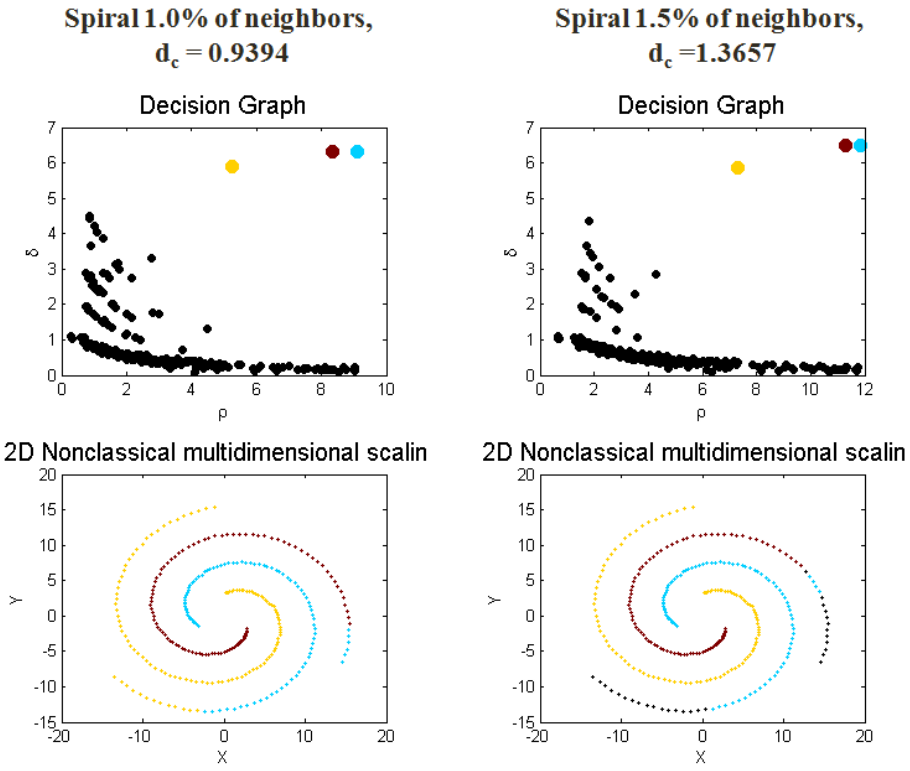


Рис. 3.8. Набір даних Spiral, застосовано метод Гаусового ядра

Для набору Flame (рисунок 3.6) відбулось чітке розділення кластерів складної форми, що свідчить про робастність до складних структур. В наборі Jain (рисунок 3.7) метод коректно ідентифікував два нелінійно розділені кластери. Для набору Spiral (рисунок 3.8) GK-CFSFDP продемонстрував відмінну здатність до виявлення кластерів довільної, вигнутої форми (спіральні кластери), що є ключовою перевагою методів, базованих на щільності.

3.3. Ітеративний алгоритм кластеризації щільності на основі гаусового ядра

3.3.1. Методологія модифікації та градієнтна оптимізація

Проведений аналіз оцінки щільності на основі ядра продемонстрував обнадійливі результати, що стало підставою для впровадження ітеративного підходу, заснованого на градієнтній оптимізації, з метою подальшого підвищення точності та стабільності кластеризації. Ця модифікація поєднує статистичний підхід KDE з динамічним пошуком піків щільності.

У рамках KDE, атрактор щільності x* визначається як локальний максимум функції щільності ймовірності f*(X). Для знаходження цих максимумів використовується градієнт щільності ∇f*(X), який є похідною функції щільності.

Процедура притягування за щільністю (Density Attractor Search) полягає в наступному: точка x вважається притягнутою за щільністю до атрактора x* (що належить до її кластера), якщо її положення ітеративно оновлюється за правилом градієнтного підйому (Gradient Ascent):

$$x_{t+1} = x_t + \delta \cdot \nabla \mathbf{f}^* (x_t)$$

де:

x_t — поточна позиція точки на ітерації t.

δ — розмір кроку (у цій реалізації прийнято δ=1).

$\nabla f^*(x_t)$ — градієнт щільності у точці x_t.

Щодо визначення градієнта локальної щільності ($\nabla\rho i$), то нова функція впливу, що використовується для обчислення градієнта $\nabla\rho i$, виводиться з похідної Гаусового визначення ρ:

$$\nabla\rho_i = \sum_{j=1}^{n-1} d_{i,j} e^{-\left(\frac{d_{ij}}{d_c}\right)^2}$$

Ця величина $\nabla\rho i$ відображає напрямок максимального зростання щільності. На кожній ітерації точка і зміщується у цей напрямок, наближаючись до найближчого атрактора щільності. Хоча ρ_i у класичному CFSFDP враховує відстані, в ітеративному контексті її похідна (градієнт) керує пошуком атракторів. З часом, коли точки сходяться до своїх атракторів, їхній ітеративний рух припиняється (градієнт прямує до 0).

3.3.2. Експериментальні результати

Застосування ітеративної модифікації CFSFDP на основі Гаусового ядра (Iterative GK-CFSFDP) до тестових наборів даних продемонструвало покращену якість кластеризації та більш чітке виділення меж кластерів, порівняно з неітеративними версіями.

Для набору Aggregation (рисунок 3.9), метод успішно виділив усі агреговані кластери, мінімізуючи вплив фонового шуму. В наборі Flame (рисунок 3.10) спостерігається чітке розділення двох складних кластерів, підтверджуючи ефективність градієнтного підйому.

Для набору Jain (рисунок 3.11) відбувається коректне виявлення нелінійно розділених кластерів, а для набору Spiral (рисунок 3.12), ітеративний підхід особливо ефективно відокремив переплетені спіральні структури, що є складним завданням для більшості алгоритмів.

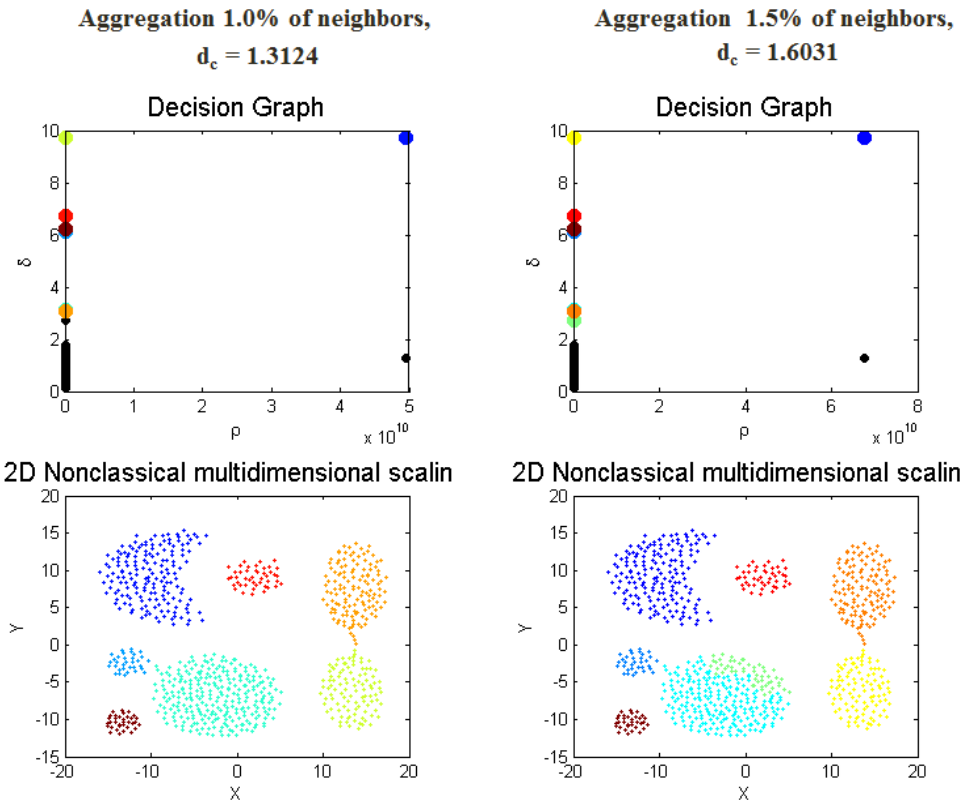


Рис. 3.9. Набір даних Aggregation, застосовано ітеративний метод гаусового ядра

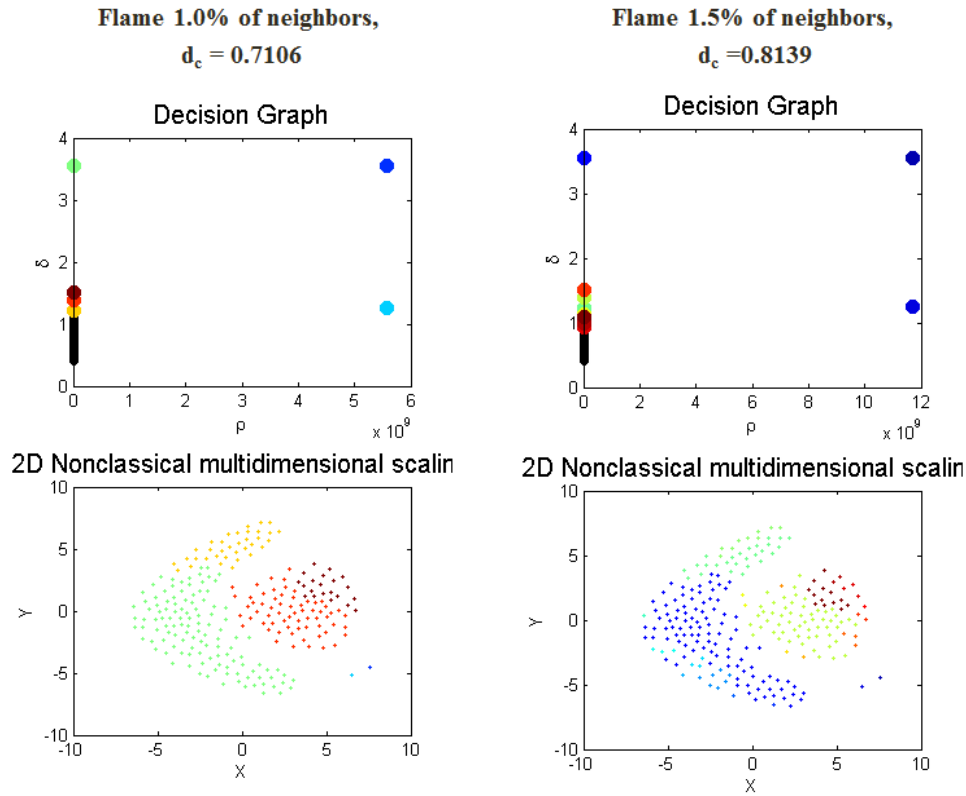


Рис. 3.10. Набір даних Flame, застосовано ітеративний метод Гаусового ядра

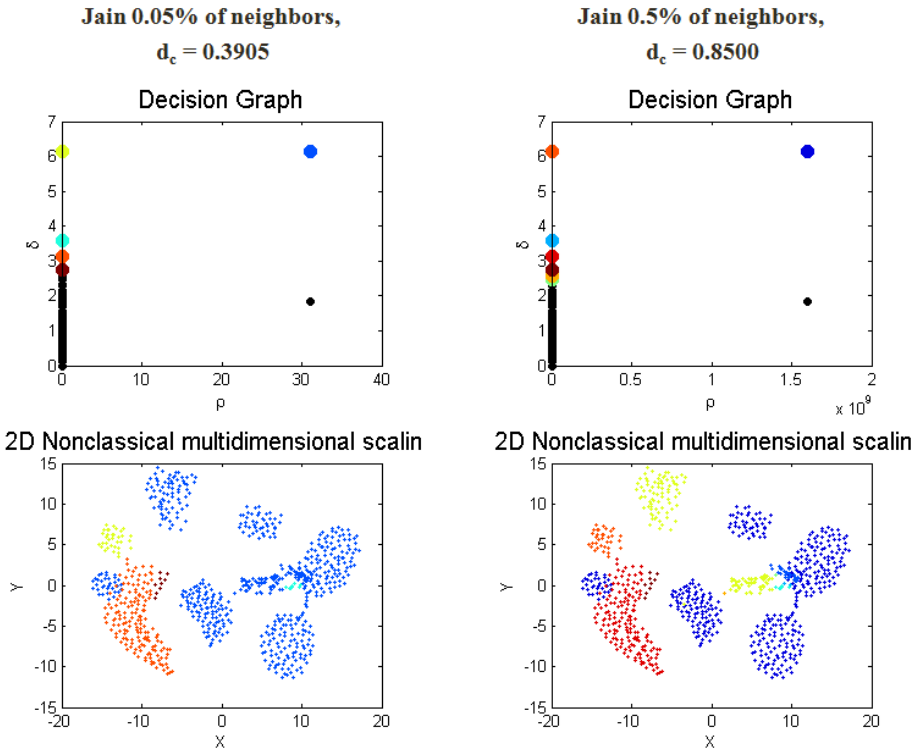


Рис. 3.11. Набір даних Jain, застосовано ітеративний метод Гаусового ядра

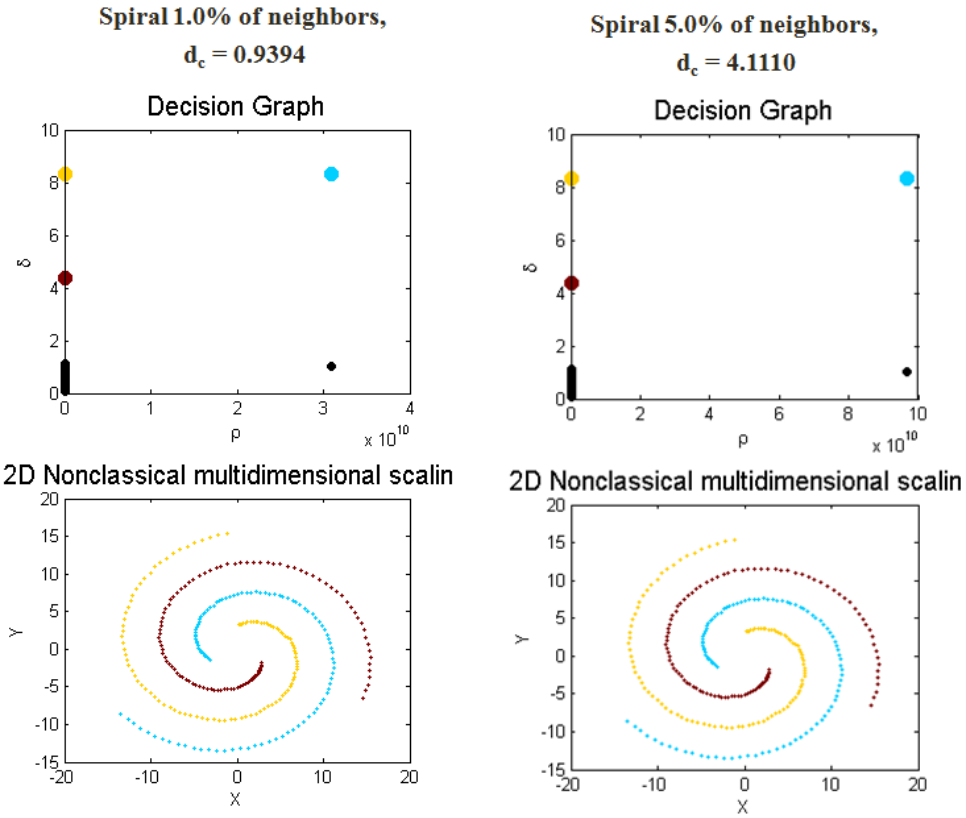


Рис. 3.12. Набір даних Spiral, застосовано ітеративний метод Гаусового ядра

3.4. Критичний аналіз та висновки щодо алгоритму кластеризації на основі щільності та його модифікацій

3.4.1. Оцінка оригінального алгоритму кластеризації

Алгоритм кластеризації шляхом швидкого пошуку та знаходження піків щільності (CFSFDP) довів свою високу ефективність у кластерному аналізі [1], зокрема завдяки здатності ідентифікувати центри кластерів як точки з одночасно високою локальною щільністю (ρ) та великою мінімальною відстанню до об'єктів з вищою щільністю (δ).

Ключові недоліки оригінального CFSFDP:

1. Чутливість до d_c, оскільки існує сильна залежність результатів кластеризації від вибору параметра відстані відсічення (d_c).

2. Евристична залежність, яка полягає в тому, що користувач повинен евристично визначити середню кількість сусідів (у відсотках), що формують кластер, яка прямо корелює з d_c. Хоча рекомендовано, щоб середня кількість сусідів відповідала 1–2% від загальної кількості точок, цей вибір залишається суб'єктивним.

На відміну від DBSCAN, який вимагає двох попередньо заданих параметрів (є та MinPts), CFSFDP, незважаючи на залежність від d_c, пропонує графік рішення (ρ проти δ), що полегшує пост-фактум ідентифікацію центрів кластерів.

3.4.2. Аналіз модифікованих методів

У цій роботі були досліджені модифікації CFSFDP, спрямовані на підвищення робастності та автоматизації.

а) CFSFDP на основі k-найближчих сусідів (kNN-CFSFDP)

Цей підхід замінює жорстку залежність від dc на оцінку щільності на основі середньої відстані до M найближчих сусідів.

Як перевага, то спостерігається зниження чутливості до d_c, оскільки оцінка щільності стає адаптивною до локальної структури даних. Хоча цей

метод впливає на абсолютні значення оцінок ρ, основний механізм розділення центрів зберігається.

Отже, метод успішно використовує оцінку розділення, забезпечуючи більш стабільні результати.

б) CFSFDP на основі гаусового ядра (GK-CFSFDP)

Метод використовує Гаусове ядро для оцінки ρ, де d_c виступає як стандартне відхилення ядра.

Використання Гаусової апроксимації забезпечує "згладженіше" визначення кластера, роблячи метод менш чутливим до статистичної похибки, спричиненої невеликою кількістю точок або шумом. Загальні результати схожі на оригінальний CFSFDP, зберігаючи чітке розділення кластерів на графіку рішення.

в) Ітеративний CFSFDP на основі гаусового ядра (Iterative GK-CFSFDP)

Цей метод поєднує KDE з градієнтним підйомом для динамічного пошуку атракторів щільності.

Досягнуті дуже високі показники якості кластеризації, з майже ідеальним віднесенням точок до кластерів, що значно зменшує проблему викидів.

Щодо недоліків, то існує утруднення з інтерпретацією графіка рішення, оскільки внаслідок ітеративного наближення до локальних максимумів, значення локальної щільності (ρ) для центрів кластерів прямує до нуля. Це ускладнює графічне розрізнення центрів кластерів (вибір прямокутника для $\rho_{minimum}$ та $\delta_{minimum}$), що є ключовим евристичним етапом аналізу.

Таблиця 3.1.

Переваги алгоритму кластеризації CFSFDP

Характеристика	**ρ** **Локальна щільність**	**δ** **Відстань до точки з вищою щільністю**
Центри кластерів	**Висока**	**Висока**
Точки тіла кластерів	Висока	Низька
Викиди (Outliers)	Низька	Низька / Висока

Головна перевага CFSFDP полягає в його здатності візуалізувати та автоматично визначати центри кластерів через представлення даних у просторі (ρ,δ).

Отже, Iterative GK-CFSFDP показав найкращі результати з точки зору якості кластеризації, мінімізуючи викиди, kNN-CFSFDP успішно подолав основну слабкість оригінального CFSFDP — сильну залежність від ручного вибору d_c.

Майбутні напрямки досліджень можуть бути спрямовані в автоматизації вибору параметра d_c (або еквівалентного параметра σ чи M). Розробка механізму, який дозволяє алгоритму обчислювати оптимальне значення d_c без ручного втручання або використання евристики 1-2%, значно підвищить об'єктивність та практичну застосовність методу CFSFDP.

Висновки до розділу

У третьому розділі виконано практичну реалізацію та аналітичне порівняння модифікацій алгоритмів кластеризації, розроблених на основі властивості щільності. Основну увагу зосереджено на вдосконаленні методів шляхом інтеграції підходів kNN та KDE у структуру щільнісних алгоритмів.

Запропоновано модифікацію алгоритму кластеризації на основі k-найближчих сусідів, у якій адаптивно обчислюється радіус локальної області з урахуванням щільності розподілу. Такий підхід підвищує точність визначення меж кластерів і зменшує чутливість до вибору фіксованого параметра ε. Результати експериментів показали підвищення якості кластеризації для даних зі змінною густотою.

Розроблено модифікацію алгоритму на основі гаусового ядра, яка використовує ядерну функцію для оцінки локальної щільності з урахуванням вагових коефіцієнтів. Виявлено, що такий підхід забезпечує більш плавну оцінку меж кластерів та стійкість до шумових точок.

Крім того, запропоновано ітеративний алгоритм кластеризації на основі гаусового ядра з градієнтною оптимізацією, який дозволяє поступово уточнювати позиції центрів кластерів, мінімізуючи похибку функції щільності. Такий метод поєднує ідеї щільнісної оцінки та оптимізаційного навчання, що підвищує точність розбиття на кластери у випадках складних або перекривних структур даних.

ВИСНОВКИ

У магістерській роботі розв’язано науково-прикладну задачу підвищення ефективності процесу кластеризації даних шляхом удосконалення моделей та методів, що ґрунтуються на властивості щільності. Проведене дослідження має комплексний характер і поєднує теоретичний аналіз, математичне моделювання, алгоритмічну формалізацію та експериментальну перевірку результатів.

У процесі роботи здійснено всебічне вивчення предметної області кластеризації даних. Визначено основні поняття, цілі та принципи кластерного аналізу як складової інтелектуальної аналітики. Обґрунтовано, що методи кластеризації на основі щільності становлять особливий клас алгоритмів, орієнтованих на виявлення структур даних довільної форми без необхідності апріорного задання кількості кластерів. Їхньою ключовою властивістю є здатність відокремлювати області високої щільності від зон шуму, що робить їх придатними для роботи з великими, неоднорідними або зашумленими наборами даних.

Проведено аналіз класичних алгоритмів щільнісної кластеризації — DBSCAN, DENCLUE та CFSFDP. Визначено їхні переваги та недоліки, зокрема, чутливість до вибору параметрів, неоднорідність оцінки локальної щільності та обмежену стійкість до нерівномірного розподілу даних. На основі цього сформульовано наукову проблему — підвищення точності та стабільності кластеризаційного процесу через модифікацію механізмів оцінювання щільності та вибору центрів кластерів.

Розроблено низку модифікацій алгоритмів кластеризації на основі властивості щільності. Зокрема, запропоновано підхід, що інтегрує метод k-найближчих сусідів (kNN) з локально-адаптивним вибором радіуса сусідства, що забезпечує більш точне визначення меж кластерів у неоднорідних множинах даних. Окрім того, створено алгоритм, заснований на гаусовій

ядерній функції (KDE), який дозволяє плавно оцінювати розподіл щільності та підвищує стійкість до викидів.

Особливу увагу приділено розробленню ітеративного алгоритму кластеризації на основі гаусового ядра із застосуванням градієнтної оптимізації. Такий підхід забезпечує поступове уточнення позицій центрів кластерів, що сприяє зменшенню похибки оцінки щільності та покращенню узгодженості результатів кластеризації. Результати експериментів на тестових наборах даних підтвердили ефективність розроблених моделей, які демонструють вищі показники якості у порівнянні з базовими алгоритмами DBSCAN та CFSFDP.

Отримані результати мають наукову новизну, яка полягає у вдосконаленні методів кластеризації шляхом поєднання принципів локальної щільності, ядерної оцінки та адаптивної метричної параметризації. Запропоновані підходи формують основу для подальшого розвитку гібридних алгоритмів машинного навчання, здатних самостійно адаптуватися до характеристик оброблюваних даних.

Таким чином, у межах виконаної магістерської роботи розроблено, обґрунтовано та експериментально підтверджено ефективність удосконалених моделей кластеризації даних на основі властивості щільності, що дозволяють суттєво підвищити якість аналізу складних інформаційних структур і забезпечують перспективи для подальших досліджень у сфері інтелектуальної обробки даних.

ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Ester, M., Kriegel, H.-P., Sander, J., Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD'96), pp. 226-231.
2. Ankerst, M., Breunig, M. M., Kriegel, H.-P., Sander, J. (1999). OPTICS: ordering points to identify the clustering structure. In: Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data, pp. 49-60.
3. Hinneburg, A., Keim, D. A. (1998). An efficient approach to clustering in large multimedia databases with noise. In: Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining (KDD'98), pp. 58-65.
4. Kriegel, H.-P., Kröger, P., Sander, J., Zimek, A. (2011). Density-based clustering. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 1(3), 231-240.
5. Campello, R. J. G., Moulavi, D., Sander, J. (2013). Density-based clustering based on hierarchical density estimates. In: Advances in Knowledge Discovery and Data Mining – PAKDD 2013, Lecture Notes in Computer Science Vol. 7819, pp. 160-172.
6. Bhuyan, R., Borah, S. (2023). A Survey of Some Density Based Clustering Techniques. arXiv preprint arXiv:2306.09256.
7. Wu, J., Kumar, V., Quinlan, J. R., et al. (2008). Top 10 algorithms in data mining. Knowledge Information Systems, 14(1), 1-37.
8. Zimek, A., Schubert, E., Kriegel, H.-P. (2012). A survey on unsupervised outlier detection in high-dimensional numerical data. Statistical Analysis and Data Mining, 5(5), 363-387.

9. Chaudhuri, K., Dasgupta, A., Freund, Y., et al. (2014). The sample complexity of clustering with neighborhood information. Journal of Machine Learning Research, 15, 331-369.
10. Chaudhuri, K., Dasgupta, A. (2010). Rates of convergence for the cluster tree. In: Advances in Neural Information Processing Systems (NIPS) 23, pp. 343-351.
11. Singh, L., Scott, C., Nowak, R. (2009). Adaptive hierarchical clustering using noisy queries. In: Proceedings of the 26th Annual International Conference on Machine Learning (ICML'09).
12. Jang, K., Baraniuk, R. (2019). DBSCAN++: Towards fast and scalable density clustering. In: Proceedings of the 36th International Conference on Machine Learning (ICML'19).
13. Gionis, A., Mannila, H., Tsaparas, P. (2007). Clustering aggregation. ACM Transactions on Knowledge Discovery from Data, 1(1), 1.
14. Hinneburg, A., Keim, D. A. (2002). An efficient approach to clustering in large multimedia databases with noise. The VLDB Journal, 10(3), 103-123.
15. Xu, R., Wunsch, D. (2005). Survey of clustering algorithms. IEEE Transactions on Neural Networks, 16(3), 645-678.
16. Jain, A. K., Dubes, R. C. (1988). Algorithms for Clustering Data. Prentice-Hall.
17. Rokach, L., Maimon, O. (2005). Clustering Methods. In: Data Mining and Knowledge Discovery Handbook. Springer, pp. 321-352.
18. Hastie, T., Tibshirani, R., Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd ed.). Springer.
19. Chaudhuri, K., Dasgupta, A., Kalogerakis, A., Singh, S. (2010). On the convergence rates of kernel density estimate cluster tree. In: Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS'10).

20. Chen, J., Yu, P. S. (2019). A Domain Adaptive Density Clustering Algorithm for Data with Varying Density Distribution. arXiv preprint arXiv:1911.10293.
21. Wang, Y., Wang, D., Zhou, Y., Zhang, X., Quek, C. (2022). VDPC: Variational Density Peak Clustering Algorithm. arXiv preprint arXiv:2201.00641.
22. Tang, Y., He, Y., Tan, H., et al. (2011). MR-DBSCAN: An Efficient Parallel Density-based Clustering Algorithm using MapReduce. In: IEEE 17th International Conference on Parallel and Distributed Systems (ICPADS'11).
23. Ding, C., He, X. (2004). K-means clustering via principal component analysis. In: Proceedings of the 21st International Conference on Machine Learning (ICML'04), pp. 29-36.
24. Berkhin, P. (2006). A survey of clustering data mining techniques. In: Grouping Multidimensional Data, Springer, pp. 25-71.
25. Scikit-learn developers. (2024). DBSCAN — scikit-learn 1.7.1 documentation. Retrieved from https://scikit-learn.org
26. Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In KDD '96: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (pp. 226–231).
27. Ng, A. Y., Jordan, M. I., & Weiss, Y. (2002). On spectral clustering: Analysis and an algorithm. In Advances in neural information processing systems (pp. 849–856).
28. Schölkopf, B., Smola, A. J., & Müller, K. R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. Neural computation, 10(5), 1299–1319.
29. Chang, C. C., & Lin, C. J. (2011). LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology (TIST), 2(3), 27:1–27:27.

30. Campello, R. J. G. B., Moulavi, D., & Sander, J. (2015). Density-based clustering: A survey. Data Mining and Knowledge Discovery, 29(5), 1287–1314.
31. MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability (Vol. 1, pp. 281–297).
32. Fraley, C., & Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. Journal of the American Statistical Association, 97(458), 611–631.
33. Gath, I., & Geva, A. B. (1989). Unsupervised optimal fuzzy clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 11(7), 773–781.
34. Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20, 53–65.
35. Liu, W., Li, M., Zhu, X., & Liu, G. (2018). Adaptive density peaks clustering based on k-nearest neighbors and neighborhood search. Knowledge-Based Systems, 141, 151–162.
36. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794).
37. Agrawal, R., Gehrke, J., Gunopulos, D., & Raghavan, P. (1998). Automatic subspace clustering of high dimensional data for data mining applications. In SIGMOD '98: Proceedings of the 1998 ACM SIGMOD international conference on Management of data (pp. 94–105).
38. Pei, J., Han, J., & Wang, W. (2002). Constraint-based spatial clustering: Issues and techniques. In Geographic Data Mining and Knowledge Discovery (pp. 209–226).

39. Tibshirani, R., Walther, G., & Hastie, T. (2001). Estimating the number of clusters in a dataset via the Gap statistic. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 63(2), 411–423.
40. Kriegel, H.-P., Kröger, P., & Zimek, A. (2009). Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. ACM Transactions on Knowledge Discovery from Data (TKDD), 3(1), 1:1–1:58.
41. Chen, Y., Lu, R., & Yu, Y. (2019). Research on the selection of optimal parameters for the density peak clustering algorithm. Applied Sciences, 9(12), 2490.