КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ РАДІОФІЗИКИ, ЕЛЕКТРОНІКИ ТА КОМП'ЮТЕРНИХ СИСТЕМ

Кафедра радіотехніки та радіоелектронних систем

«На правах рукопису»

Робота допущена до захисту в ЕК
рішенням кафедри радіотехніки та радіоелектронних систем
від _____протокол №
Завідувач кафедри доктор фіз.-мат. наук, професор
_____ Ігор АНІСІМОВ

ДИПЛОМНА РОБОТА МАГІСТРА
на тему:

«ІДЕНТИФІКАЦІЯ СТРУКТУРНИХ ЕЛЕМЕНТІВ БАГАТОАТОМНИХ МОЛЕКУЛ МЕТОДАМИ БІКЛАСТЕРІЗАЦІЇ ЇХ МАС-СПЕКТРІВ»

Виконала:
студентка 2-го курсу магістратури
денної форми навчання
спеціальності 172 – Електронні комунікації та радіотехніка
ОПП «Інформаційна безпека телекомунікаційних систем і мереж»
Шелякіна Богдана-Марія Анатоліївна_____

Науковий керівник:
доктор технічних наук, доцент кафедри радіотехніки
та радіоелектронних систем
Ольшевський Сергій Валентинович_____

Рецензент:
доктор технічних наук, професор,
заслужений діяч науки і техніки України
Литвиненко Володимир Іванович_____

Засвідчую, що у цій дипломній роботі
немає запозичень з праць інших авторів без
відповідних посилань
Студентка _____ Богдана-Марія ШЕЛЯКІНА

Київ-2025

РЕФЕРАТ

Дипломна робота: 45 с., 13 рис., 2 дод., 30 джерел.

Об’єкт дослідження – бікластеризація мас-спектрів поліатомних молекул.

Мета роботи – розробка і вдосконалення машинних методів аналізу мас-спектроскопічних сигналів для створення програмно-апаратних засобів автоматизованої ідентифікації хімічних речовин.

У результаті комплексного аналізу методів попередньої обробки, покращення якості, корекції, класифікації та кластеризації мас-спектрів було реалізовано ефективну систему автоматизованої обробки спектральних даних. Зокрема, була здійснена бікластеризація із застосуванням алгоритму k-means, а також використано дерева рішень для ідентифікації структурних елементів поліатомних органічних молекул. Це дозволило більш точно виділяти ключові фрагменти молекулярних структур та встановлювати взаємозв’язки між різними класами сполук на основі їх мас-спектрометричних характеристик.

Використовуючи мову програмування Python було реалізовано алгоритми бікластеризації. та визначення хімічного складу молекул. Розроблена система забезпечує автоматизовану обробку мас-спектрів, що включає фільтрацію шумів, нормалізацію даних, виявлення піків та їх подальшу інтерпретацію. Впровадження методів машинного навчання дозволяє підвищити точність класифікації сполук та прискорити процес аналізу, що є особливо важливим при роботі з великими обсягами спектральної інформації.

Система може бути ефективно використана в галузях аналітичної хімії, фармацевтики, біотехнологій та екологічного моніторингу для автоматизації процесів аналізу хімічного складу речовин, ідентифікації невідомих зразків, виявлення домішок та контролю якості продукції. Такий підхід сприяє зниженню впливу людського фактора, підвищенню продуктивності та забезпеченню більш надійних результатів у спектральному аналізі.

ЗМІСТ

РЕФЕРАТ 1
ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ 4
ВСТУП 5
АНАЛІЗ СТАНУ ПРОБЛЕМИ 7
1.1. Сучасні методи кластеризації та бікластеризації 9
МАТЕРІАЛИ ТА МЕТОДИ 11
2.1. Мас-спектрометрія 11
2.2. Методи кластеризації 13
2.3. Середовище розробки Jupiter 16
2.4. Бікластерний аналіз 18
2.5. Різновиди класифікацій 23
РЕЗУЛЬТАТИ ЕКСПЕРИМЕНТУ ТА ОБГОВОРЕННЯ 26
3.1. Попередня підготовка мас-спектрів та формування векторного представлення 26
3.2. Оптимізація методів автоматизованого розбиття молекулярних мас-спектрів на класи хімічних речовин. 27
3.3. Кластеризація молекул методом k-means 28
3.4. Класифікація мас-спектрів методом "Дерево рішень" 29
3.5. Попередня підготовка даних для бікластеризації мас-спектрів 30
3.6. Бікластеризація мас-спектрів 33
ВИСНОВКИ 37
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ 38
ДОДАТОК 1 41
ДОДАТОК 2 43

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ

CAS – Chemical Abstracts Service (служба хімічної анотації, що присвоює унікальні номери хімічним речовинам)

Cl – Хлор

Jupyter – інтерактивне середовище для наукових обчислень і візуалізації даних

kNN – алгоритм k найближчих сусідів (k-nearest neighbors)

MS – Mass Spectrometry (мас-спектрометрія)

PCA – Principal Component Analysis (метод головних компонент)

PCB – Поліхлорбіфеніли

QSAR – Quantitative Structure–Activity Relationship (кількісні зв’язки «структура–активність»)

QSPR – Quantitative Structure–Property Relationship (кількісні зв’язки «структура–властивість»)

RMS – Root Mean Square (середньоквадратичне значення)

SDF – Structure Data File (формат файлу структурних даних)

SVM – Support Vector Machine (метод опорних векторів)

ХОП – Хлорорганічні пестициди

r – радіус траєкторії іона в магнітному полі

m – маса іона

q – заряд іона

V – швидкість іона

B – індукція магнітного поля

σ – стандартне відхилення (дисперсія) шуму

α – кут між векторами векторного простору мас-спектрів

R_0, R_x, Rd – радіус-вектори мас-спектрів

k – кількість кластерів у кластеризації

n – кількість об’єктів

m – кількість ознак (характеристик)

ВСТУП

Необхідність у впровадженні точних та ефективних методів виявлення залишкових концентрацій забруднювачів у харчовій сировині, продуктах харчування та кормах для тварин є надзвичайно актуальною. Крім цього, важливою є розробка інноваційних технологій очищення продукції від шкідливих домішок, що дозволить зменшити ризики харчових отруєнь і підвищити безпеку продукції.

Серед сучасних методів, що забезпечують високу надійність у виявленні хлорорганічних сполук, мас-спектрометрія займає провідне місце. Цей метод дозволяє отримати унікальні відомості як про хімічний склад молекули, так і про її просторову конфігурацію. Це особливо важливо у випадках, коли йдеться про контроль залишкових кількостей пестицидів, які часто представлені різними ізомерами. За результатами хроматографії ізомери можуть демонструвати окремі піки, що створює ризик помилкової ідентифікації як різних сполук.

Сучасні методи контролю шкідливих речовин та залишків базуються переважно на мас-спектрометричних дослідженнях, що забезпечують як високу чутливість, так і точність. Проте одним із ключових обмежень є неможливість повної автоматизації процесу для невідомих речовин. Ідентифікація зазвичай здійснюється шляхом порівняння отриманого спектра з базою еталонних мас-спектрів. Якщо досліджувана сполука відсутня у базі, точне визначення її складу чи структури стає проблематичним.

Автоматизація процесів ускладнюється також тим, що багато агрохімікатів є сумішами кількох активних речовин, а не окремими сполуками. У таких випадках застосування технологій машинного навчання може значно підвищити ефективність аналізу, зокрема через використання методів класифікації.

Ідея цього підходу полягає в тому, щоб автоматично віднести мас-спектр невідомої речовини до певного класу шляхом обчислення схожості спектральних характеристик з відомими зразками. Це може бути реалізовано за допомогою

алгоритмів машинної кластеризації, які здатні працювати з великими обсягами складних даних.

На етапі навчання класифікаційної моделі необхідно використовувати великі обсяги попередньо класифікованих мас-спектрів. Це дає змогу алгоритмам виявляти характерні патерни, притаманні хлорорганічним сполукам, і надалі з високою точністю розпізнавати нові зразки.

Інтеграція машинного навчання в процес аналізу хлорорганічних сполук забезпечує низку переваг: оперативну обробку інформації, високу точність класифікації та можливість працювати з масштабними наборами даних. Такий підхід відкриває нові горизонти для вивчення фізико-хімічних властивостей сполук, оцінювання їхньої дії, а також для відкриття нових речовин.

Проте важливо враховувати й деякі труднощі, зокрема варіабельність мас-спектральних даних, яка може залежати від умов проведення аналізу, якості зразків чи характеристик приладів. Тому актуальним завданням є створення алгоритмів, які будуть стійкими до таких змін.

Незважаючи на ці виклики, використання методів машинного навчання для класифікації мас-спектрів залишається потужним і перспективним інструментом у процесі виявлення хлорорганічних забруднень.

АНАЛІЗ СТАНУ ПРОБЛЕМИ

На сьогоднішній день машинне навчання є однією з найактивніше прогресуючих наукових галузей, яка знаходить широке застосування, зокрема у фармацевтичному виробництві [1]. На відміну від класичних фізичних підходів, заснованих на використанні строго визначених рівнянь, таких як методи квантової хімії або моделювання молекулярної динаміки, машинне навчання оперує алгоритмами, що здатні виявляти приховані залежності між експериментальними характеристиками малих молекул і їх фізичними, хімічними чи біологічними властивостями. Це відкриває можливість передбачення властивостей нових молекул. Крім того, у порівнянні з фізичними моделями, машинне навчання забезпечує вищу продуктивність і краще масштабується при роботі з великими обсягами даних, не потребуючи надто великих обчислювальних ресурсів.

Одним із центральних напрямів використання машинного навчання у сфері розробки лікарських засобів є пошук закономірностей між структурною організацією молекул і їхньою біологічною активністю, що позначається терміном SAR [2]. Результати фармакологічного скринінгу можуть бути використані для вдосконалення молекулярної структури потенційних лікарських сполук з метою підвищення їх активності, біодоступності або інших властивостей. У минулому для цього потрібно було проводити численні цикли синтезу та експериментальної перевірки, що було витратним як у часі, так і в ресурсах. Зараз, завдяки можливостям машинного навчання, можна здійснювати моделювання типу QSAR (кількісні зв’язки структура-активність) або QSPR (структура-властивість), що дозволяє оцінити, як зміни в структурі впливатимуть на властивості молекул [3].

QSAR-моделі ефективно застосовуються для передбачення різноманітних параметрів лікарських речовин, включаючи токсичність, поведінку в метаболізмі, взаємодії з іншими ліками і навіть канцерогенний потенціал [3]. Перші підходи, такі як моделі Ханша і Фрі-Вільсона, базувалися на багатофакторному регресійному аналізі, що дозволяв пов’язувати просторові та фізико-хімічні

властивості молекул (наприклад, розчинність або гідрофобність) із їх активністю [4]. Проте, ці методики мали певні обмеження — зокрема, нестачу даних і припущення про лінійний характер залежностей. Сучасні завдання вимагають складніших моделей, які можуть враховувати нелінійні зв’язки у великих обсягах даних, що стало можливим завдяки розвитку хемоінформатики та машинного навчання.

Хімічна подібність слугує ключовим інструментом у віртуальному відборі лігандів [5]. Головна ідея полягає у пошуку молекул, які структурно та функціонально подібні до заданої сполуки [6]. На основі припущення, що молекули зі схожими структурами часто мають подібну біологічну дію, будуються численні алгоритми, що застосовуються у скринінгу, орієнтованому на ліганди [7].

Машинне навчання поділяється на навчання з учителем і без нього [8]. У першому випадку моделі формуються на основі наявних мічених даних, що дозволяє передбачати результати для нових, ще не досліджених об'єктів. До цієї категорії належать такі методи, як лінійна регресія, алгоритм k найближчих сусідів (kNN), байєсівські підходи, метод опорних векторів (SVM), алгоритми випадкових лісів і штучні нейронні мережі. У безучительських підходах алгоритми працюють із неміченими даними, самостійно виявляючи схожість або структури в інформації, наприклад у вигляді молекулярних дескрипторів. Також виділяють напівконтрольоване навчання, що поєднує в собі невелику кількість мічених прикладів із великою кількістю немічених для підвищення ефективності моделей на складних або дисбалансованих вибірках [9].

До безучительських методів належать, зокрема, алгоритми для зменшення розмірності — як-от метод головних компонент (PCA), аналіз незалежних компонент (ICA), а також деякі варіації SVM, імовірнісні моделі та нейронні мережі [10]. Окрім цього, широко використовується кластеризація, яка дозволяє розподіляти дані у групи за принципом подібності, вимірюваної у просторі дескрипторів.

Сучасні алгоритмічні рішення у сфері машинного навчання надають нові інструменти для аналізу складних і нелінійних зв’язків між структурними

особливостями хімічних сполук та їх численними фізико-хімічними параметрами, що відкриває великі перспективи у фармацевтичних дослідженнях.

1.1. Сучасні методи кластеризації та бікластеризації

Кластерний аналіз, відомий також як групування або аналіз скупчень, є методом, що дозволяє розділити сукупність об'єктів на окремі групи — кластери. Основна мета полягає в тому, щоб об'єкти, що потрапляють в один кластер, мали якомога більше спільних ознак, тоді як представники різних кластерів суттєво відрізнялися між собою. Термін «кластер» походить з англійської мови ("cluster") і означає «сукупність» або «група».

Цей підхід дозволяє систематизувати об'єкти за подібністю їх властивостей, навіть без наявності попередніх знань про структуру генеральної сукупності. Об’єкти описуються за допомогою певного набору змінних, які можна уявити як координати в багатовимірному просторі.

Суть кластеризації полягає в об'єднанні подібних об'єктів у компактні групи на основі їх характеристик, з одночасним максимальним відокремленням різних груп одна від одної. Важливо відзначити, що кластеризація не змінює кількість змінних, проте дозволяє зменшити кількість окремих елементів шляхом групування.

Цей підхід широко використовується для аналізу кількісних характеристик об'єктів. Його характерна особливість — відсутність потреби в оцінюванні статистичної значущості, як це притаманно класичній статистиці, що робить кластеризацію гнучким і універсальним інструментом. Вона ефективна у таких сферах, як обробка статистичних даних, розпізнавання образів, класифікація і навіть векторна квантизація.

Однією з переваг цього методу є можливість працювати з різними типами даних і враховувати одразу кілька змінних, що особливо корисно для аналізу складних, різнорідних даних, наприклад у сфері економіки чи соціології.

Формально задача кластеризації визначається як процес поділу набору з n об’єктів, кожен з яких характеризується d ознаками, на декілька підмножин. Як правило, оптимізаційним критерієм при цьому виступає мінімізація функції спотворення, яка найчастіше виражається як сума квадратів відстаней між об’єктами та центрами їхніх кластерів.

Серед найвідоміших методів кластеризації — алгоритм *K*-середніх (K-means), який відноситься до неконтрольованого навчання і активно застосовується у сфері аналізу даних. Його принцип роботи базується на ітеративному процесі: спочатку випадковим чином (або за спеціальною процедурою) вибираються початкові центри кластерів. Потім кожен об'єкт прив’язується до найближчого центроїда за певною метрикою (наприклад, Евклідовою або косинусною відстанню), після чого обчислюються нові центроїди як середні значення координат об'єктів у межах кожного кластера. Цей процес повторюється до досягнення стабільності або до виконання визначених умов зупинки — наприклад, обмеженої кількості ітерацій чи відсутності змін у внутрішньогруповій дисперсії.

Перевагою методу *K*-середніх є його ефективність навіть при обробці великих обсягів даних. Оскільки він належить до некерованого навчання, немає потреби у попередньому маркуванні даних.

Існує чимало варіантів і вдосконалень базового алгоритму. Наприклад, K-means++ оптимізує вибір початкових центрів кластерів, що покращує результати. Інша модифікація — метод *K*-медоїдів — замість середніх використовує реальні об'єкти як представники кластерів.

Цей підхід широко застосовується у найрізноманітніших сферах: від класифікації зображень і досліджень споживацької поведінки до біоінформатики та рекомендаційних систем. Завдяки здатності виявляти приховані закономірності у великих наборах даних кластеризація сприяє більш глибокому розумінню складної інформації.

МАТЕРІАЛИ ТА МЕТОДИ

2.1. Мас-спектрометрія

Мас-спектрометрія — це метод дослідження складу речовин шляхом аналізу мас іонів, які утворюються внаслідок іонізації. Основою методу є вимірювання співвідношення маси іона до його заряду та визначення їх кількості, що в сукупності формує мас-спектр. Наприклад, мас-спектр дельтаметрину (рис2.1.) наочно демонструє характер розподілу таких іонів.

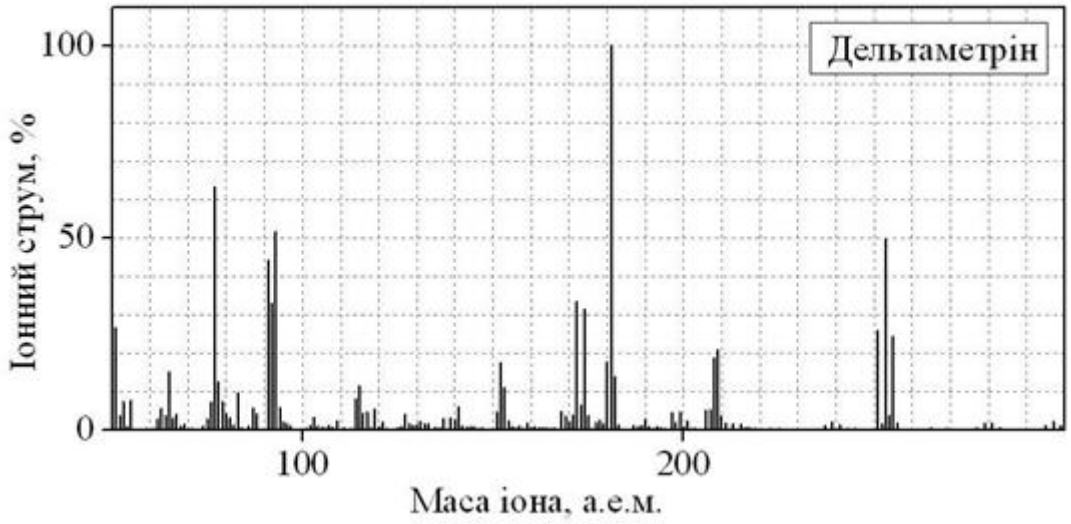


Рис. 2.1. Мас-спектр дельтаметрину

Речовини можуть бути спочатку переведені у газоподібний стан, а потім іонізовані. Іонізація не завжди є обов’язковою, зокрема при дослідженні вже іонізованих газів, як у випадку з іоносферами або електричними розрядами. Найчастіше аналізують позитивні іони, оскільки їх отримання є технічно простішим та ефективнішим, хоча в деяких випадках вивчають і негативні.

Перші мас-спектри були отримані в Англії Дж. Дж. Томсоном у 1910 році, а згодом — Ф. Астоном у 1919 році. Це стало основою для відкриття стабільних ізотопів. Від початку свого розвитку мас-спектроскопія використовувалася для

точного визначення ізотопного складу та атомних мас елементів, і вона зберігає цю функцію донині.

Метод дозволяє з великою точністю виявляти варіації ізотопного складу. Похибка становить близько ±0,01 % для складу ізотопів, і до ±0,00001 % для мас легких ядер. Це забезпечило широке використання методу в таких галузях, як ядерна енергетика, геохімія, геологія, хімія та медицина.

У геологічних дослідженнях мас-спектрометрія використовується для датування порід та мінералів, зокрема шляхом аналізу ізотопів свинцю та аргону. У хімії вона застосовується для аналізу складу речовин і дослідження їх молекулярної структури. Найточніші результати досягаються при іонізації цілих молекул без їхнього попереднього розпаду.

Для аналізу тугоплавких або нестійких речовин застосовують метод іонізації у вакуумній іскрі. Кількісний аналіз може бути ускладнений через однакову масу іонів з різних сполук. Щоб уникнути цього, використовують "м’які" методи іонізації, які зменшують фрагментацію, а також комбінують мас-спектрометрію з іншими методами, наприклад, з газовою хроматографією.

Під час іонізації частина молекул залишається цілою, утворюючи молекулярні іони, а інша частина розпадається на фрагменти. Аналіз мас та концентрацій цих іонів дозволяє встановити як масу, так і будову молекули.

У фізико-хімічних дослідженнях мас-спектрометрія дає змогу вивчати механізми іонізації, енергетичні характеристики молекул, такі як енергія іонізації, теплота випаровування, енергія зв’язку. Її також використовують для аналізу складу атмосфери Землі й інших планет, а в медицині — як метод швидкого аналізу газів.

Метод відзначається високою чутливістю — можна дослідити кількості речовини на рівні трильйонних часток грама. Основною фізичною величиною є питомий заряд іона (співвідношення заряду до маси), який визначають за траєкторією іона у магнітному чи електричному полі. Радіус цієї траєкторії обчислюють за формулою:

r = mV / qB, де r — радіус, m — маса, V — швидкість, q — заряд, B — індукція магнітного поля.

Для молекулярного аналізу часто застосовують іонізацію електронним ударом, яка викликає збудження молекули та її подальший розпад. Хоча ця теорія ще не дозволяє точно передбачити мас-спектри, у практиці використовують порівняння спектрів відомих речовин. Для підвищення точності прилади калібрують за зразками з відомим складом.

Сучасні мас-спектрометри оснащені комп'ютерами для автоматичної обробки та інтерпретації результатів, що значно підвищує ефективність аналізу.

2.2. Методи кластеризації

Для побудови інформативної та якісної моделі генетичної регуляторної мережі, першочергово необхідно згрупувати ознаки, які мають кореляційні зв'язки. Інакше кажучи, потрібно здійснити кластеризацію як ознак, так і об'єктів на основі їхньої схожості. У загальному вигляді задача кластеризації формалізується наступним чином. Вважаємо, що вхідні дані задано у вигляді матриці:

$$\mathbf{A} = \left\{ x_{ij} \right\}, i = 1, \ldots, n;\ j = 1, \ldots, m \tag{2.1}$$

де n — кількість об'єктів (рядків), що аналізуються; m — число характеристик або ознак (стовпців), що описують кожен об'єкт. Завдання кластеризації полягає в поділі множини об'єктів чи ознак на непорожні групи — кластери — які, в ідеалі, не перетинаються. При цьому межі між кластерами можуть мати довільну геометричну форму [11]:

$$\begin{aligned} &\mathbf{K} = \left\{ \mathbf{K}_s \right\}, s = 1, \ldots, k; \mathbf{K}_1 \cup \mathbf{K}_2 \cup \ldots \cup \mathbf{K}_k = \mathbf{A}; \\ &\mathbf{K}_i \cap \mathbf{K}_j = \varnothing, i \neq l, i, j = 1, \ldots, k \end{aligned} \tag{2.2}$$

де k — кількість кластерів.

На сьогодні методи кластерного аналізу детально розглянуті у численних дослідженнях [12]. Стандартна процедура кластеризації передбачає такі основні етапи:

* формулювання задачі кластерного аналізу;

* створення масиву даних, що описують об’єкти, і вибір цих об’єктів;

* вибір метрики для оцінювання близькості між об’єктами;

* визначення кластерного алгоритму та початкових параметрів;

* здійснення процесу кластеризації та побудова кластерної структури;

* розрахунок показників якості групування;

* аналіз та інтерпретація результатів;

* прийняття рішень щодо природи отриманого групування.

Схематичне зображення цього процесу наведене на рис. 2.2. Варто зазначити, що кожен із етапів, відображених на цьому рисунку, може слугувати предметом окремого дослідження. Існує широкий спектр кластерних алгоритмів, кожен з яких пристосований до певного типу або структури даних. Основні з них узагальнено в структурній схемі на рис. 1 [13].

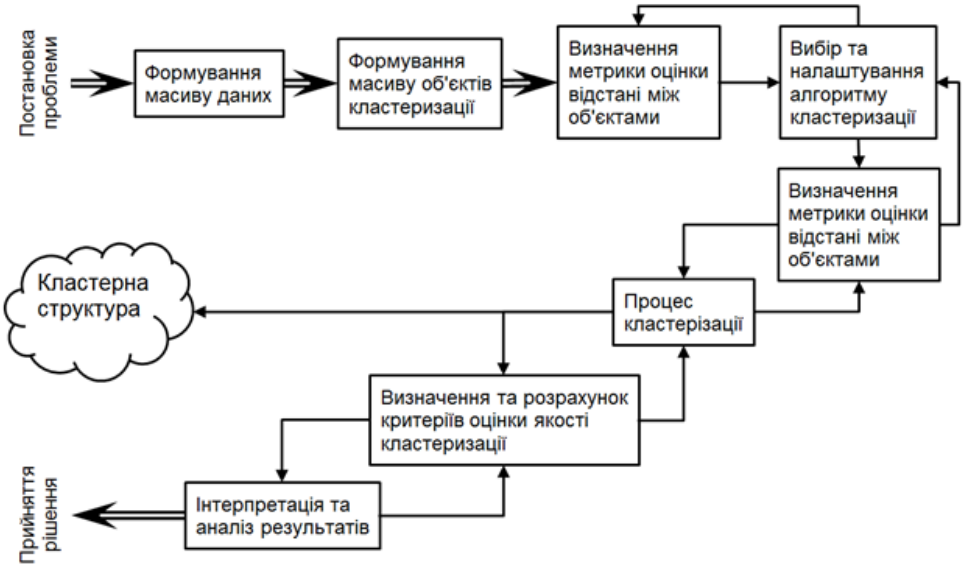


Рис. 2.2. Структурна схема

Ієрархічні методи кластеризації [14] формують дерево рішень щодо групування, однак вимагають визначення рівня ієрархії — об’єднання (агломеративні підходи) або поділу (дивізійні).

Нейромережеві моделі, зокрема карти Кохонена [15] і їхнє продовження — алгоритм SOTA [16], частково вирішують питання ієрархічного рівня кластеризації, проте результат сильно залежить від налаштувань алгоритму, що вимагає додаткової адаптації до конкретних даних.

Індуктивні підходи, базовані на методах групового врахування аргументів [17], мають потенціал для оптимізації процесу самоорганізації кластерів. Проте вони ефективні переважно для даних з малою кількістю ознак, що обмежує їх застосування до мас-спектральних даних, де кількість іонних мас часто перевищує 300.

У [18] розглядається концепція об’єктивної кластеризації, яка передбачає паралельне кластерне групування на двох близьких підмножинах даних і вибір оптимального результату за критерієм балансу. Однак її практична реалізація потребує детального визначення відповідних метрик і критеріїв оцінки якості.

До ітераційних методів класифікації належать алгоритми, які здійснюють багаторазовий перерозподіл об’єктів між кластерами [19], щільнісні [20], еволюційні й генетичні [21], а також методи на основі ґратчастої (grid-based) структури [22] чи підпросторової організації [23]. Перевагою ітеративних алгоритмів є простота реалізації та прозора логіка роботи, проте вони сильно залежать від початкового вибору центрів кластерів та потребують заздалегідь визначеної кількості кластерів, що у випадку ознак невідомо наперед, і це є вагомим обмеженням.

Grid-based алгоритми представляють простір ознак як багатовимірну ґратку, де кластеризація відбувається по комірках. Проте ефективність таких підходів суттєво залежить від вибору розміру комірок, що ускладнює їх використання при роботі з високовимірними даними.

Огляд вищенаведених методів свідчить про наявність як сильних, так і слабких сторін у кожного з них. Більшість підходів краще працює з низькорозмірними даними.

У випадку з профілями ознак вибір ефективного методу кластеризації є складним завданням, що потребує окремих досліджень — зокрема щодо вибору метрик подібності, критеріїв оцінки якості кластерів та підходів до формування груп.

Крім того, більшість алгоритмів мають проблеми з відтворюваністю результатів: успішна кластеризація на одній вибірці не гарантує аналогічний результат на іншій, навіть подібній.

Для необроблених мас-спектральних даних профілем ознак зазвичай є розподіл інтенсивностей іонних піків по масах. Складність таких даних підсилює похибки, що виникають під час кластеризації. Зменшити цю похибку можливо шляхом створення гібридних моделей, які поєднують найбільш адекватні алгоритми кластеризації з методами групування, що враховують об’єктивні критерії оцінки результатів.

2.3. Середовище розробки Jupiter

Jupyter — це інтерактивне середовище розробки, яке широко використовується в наукових дослідженнях, аналізі даних, машинному навчанні та освіті. Його основною особливістю є можливість поєднувати код, текст, формули, візуалізації та інші елементи в єдиному документі, який називається «ноутбук» (notebook). Такий підхід дозволяє дослідникам, аналітикам і студентам не лише писати код, а й документувати свої думки, результати та висновки в зручному й наочному форматі.

Jupyter підтримує багато мов програмування, хоча найчастіше використовується з Python. Назва Jupyter походить від перших літер трьох мов: Julia, Python та R. Основою середовища є веб-інтерфейс, де користувачі можуть виконувати код поетапно, в інтерактивному режимі, переглядати графіки, таблиці, діаграми та інші результати без потреби запускати все з нуля. Це особливо зручно для експериментів, навчання і презентації даних.

Jupyter зручно інтегрується з бібліотеками Python, такими як NumPy, pandas, Matplotlib, seaborn, scikit-learn та TensorFlow. Це робить його надзвичайно корисним для обробки великих обсягів інформації, побудови моделей штучного інтелекту та візуалізації результатів. Завдяки можливості включати текст, оформлений за допомогою Markdown і формули у форматі LaTeX, Jupyter також є чудовим інструментом для підготовки технічної документації та звітів.

Ще однією перевагою є те, що Jupyter працює у звичайному веббраузері, а це означає, що його можна запускати як локально на комп'ютері, так і на віддалених серверах або хмарних платформах, таких як Google Colab. Це розширює можливості для співпраці та доступу до обчислювальних ресурсів без обмежень локального обладнання.

Завдяки своїй простоті, гнучкості та потужності Jupyter став стандартом де-факто в багатьох галузях, де потрібно поєднувати програмування з поясненням результатів. Його активно використовують дослідники, викладачі, студенти, а також фахівці з аналізу даних та розробники моделей штучного інтелекту по всьому світу.

Python у середовищі Jupyter є надзвичайно потужним інструментом для хімічної аналітики, оскільки дозволяє поєднувати обробку даних, математичні розрахунки, візуалізацію результатів та інтерактивність в одному середовищі. Це робить його ідеальним рішенням для сучасного хіміка-аналітика, який працює з великими обсягами експериментальних даних, спектрами, хроматограмами або результатами молекулярного моделювання.

У хімічній аналітиці Python може застосовуватися для обробки даних, отриманих із різних аналітичних приладів — таких як УФ-спектрофотометри, ІЧ-спектрометри, ЯМР, хроматографи. Наприклад, за допомогою бібліотеки NumPy можна виконувати чисельні розрахунки: інтегрування піків, згладжування сигналу, обчислення концентрацій або похідних. З використанням pandas можна зручно зберігати, сортувати й аналізувати дані у вигляді таблиць. Matplotlib або seaborn дозволяють будувати графіки: спектри, калібрувальні криві, залежності інтенсивності сигналу від часу тощо.

Особливу цінність Python у Jupyter представляє для хіміків, які працюють з багатовимірними даними або потребують автоматизації обробки даних. Наприклад, можна легко реалізувати обробку серії спектрів, створити модель калібрування або автоматично виявляти аномалії в результатах аналізу. Для складніших завдань, таких як хемометрія чи машинне навчання в аналітичній хімії, використовують бібліотеки scikit-learn або TensorFlow, які також чудово працюють у Jupyter.

Окрім того, є спеціалізовані бібліотеки для хімії, як-от RDKit для роботи з молекулярними структурами або ChemPy для моделювання хімічних реакцій. Вони дозволяють будувати, аналізувати та візуалізувати молекули, розраховувати фізико-хімічні властивості, або моделювати кінетику реакцій — і все це в інтерактивному режимі.

Таким чином, Python у середовищі Jupyter стає сучасним лабораторним помічником хіміка-аналітика, який не лише виконує обчислення, а й допомагає бачити зв’язки між даними, робити висновки та представляти результати в зручному для наукової комунікації вигляді.

2.4. Бікластерний аналіз

Бікластерний аналіз є однією з найпоширеніших методик для групування профілів ознак з подальшою реконструкцією об'єктів. Під бікластером розуміють множину рядків і стовпців матриці, значення яких виявляють між собою взаємну кореляцію. У процесі виявлення бікластерів матриця профілів ознак трансформується до такого вигляду:

$$\mathbf{A} = \left\{ a_{ij} \right\}, i = 1, \ldots, n;\ j = 1, \ldots, m \tag{2.3}$$

де n — кількість досліджуваних об'єктів (рядків), m — кількість ознак (стовпців), а — це значення j-ї ознаки для і-го об'єкта. Припустимо, що — вектор, який визначає підмножину рядків, а — підмножина відповідних стовпців. Таким чином, початкову матрицю можливо подати як сукупність підматриць, що

відповідають різним бікластерам: , де — підмножина об'єктів, а — підмножина ознак, що разом формують бікластер. Усі елементи бікластера характеризуються високим ступенем внутрішньої кореляції. За аналогією з (2.3), бікластер представлений як окрема матриця ознак:

$$\mathbf{B} = \left\{ b_{ij} \right\}, i = 1,\ldots,k;\ j = 1,\ldots,s \tag{2.4}$$

Аспекти бікластеризації для кількісних даних аналізуються в роботах [23], де наведено огляд методів, їх переваг і недоліків. У [25] проведено порівняльний аналіз різних алгоритмів бікластеризації з метою дослідження профілів кількісних ознак. Автори [26] демонструють використання спектрального алгоритму бікластеризації для моделювання на прикладі синтетичних даних, де ілюструються розподіл об’єктів і їх взаємозв’язки в межах виявлених бікластерів.

На рис. 3 представлено порівняння класичної кластеризації та бікластеризації. У класичному підході (рис. 3а) профілі ознак класифікуються на основі повних векторів ознак об'єктів. При цьому результат значною мірою залежить від вибору метрики подібності, критерію якості кластеризації та самого алгоритму, що може призводити до неоднозначних результатів. У підході бікластеризації (рис. 2.3.) виділяються групи об'єктів і ознак, які демонструють узгоджену поведінку.

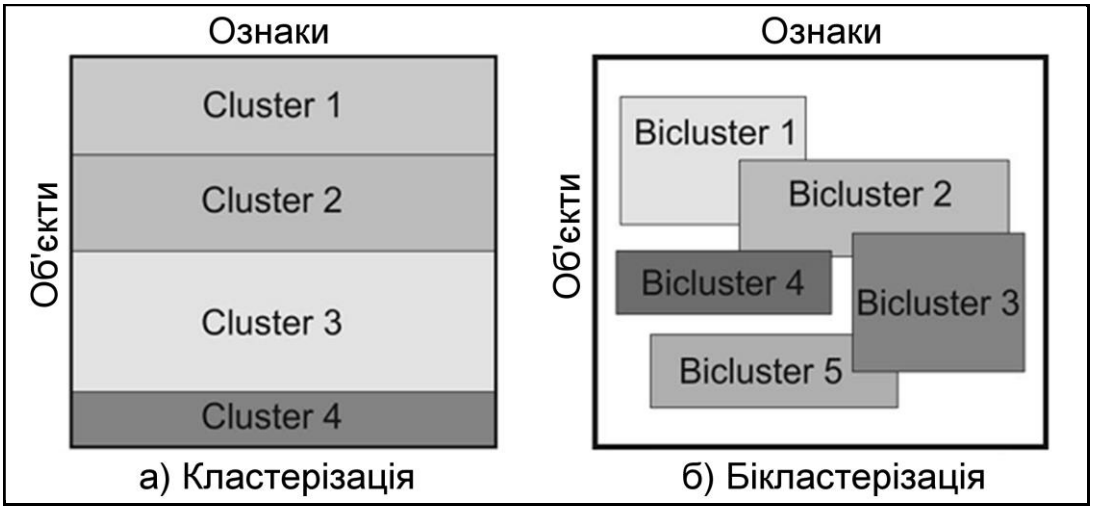


Рис. 2.3. Схема бікластеризації

Такі групи можуть бути меншими за розміром у порівнянні з повною матрицею, а кількість бікластерів регулюється параметрами вибраного алгоритму. Окрім того, бікластери можуть перекриватися, що дозволяє одному елементу бути частиною кількох бікластерів — особливо актуально для біоінформатики, де, наприклад, один ген може брати участь у кількох функціональних процесах.

[27]. Нехай — константа, що відповідає значенню бікластера , тоді кількісні значення і-го об'єкта та j-ї ознаки позначаються відповідно та . Середні значення елементів по рядках, стовпцях та в межах бікластера обчислюються так:

$$\bar{b}_{i\mathbf{j}} = \frac{1}{s}\sum_{j=1}^{s} b_{ij} \tag{2.5}$$

$$\bar{b}_{\mathbf{i}j} = \frac{1}{k}\sum_{i=1}^{s} b_{ij} \tag{2.6}$$

$$\bar{b}_{\mathbf{ij}} = \frac{1}{k}\sum_{i=1}^{k} \bar{b}_{i\mathbf{j}} = \frac{1}{s}\sum_{j=1}^{s} \bar{b}_{\mathbf{i}j} \tag{2.7}$$

Бікластер із постійними значеннями має вигляд матриці , де всі елементи однакові:

$$b_{ij} = \delta \tag{2.8}$$

Хоча така структура є теоретичною і рідко спостерігається в практиці, вона може з’являтися у вигляді шумової компоненти та використовуватися для фільтрації даних. Інші типи — це бікластери зі сталими значеннями у рядках або стовпцях, які описуються наступним виразом:

$$\begin{aligned} b_{ij} &= \delta + \beta_i, \quad b_{ij} = \delta \times \beta_i \\ b_{ij} &= \delta + \beta_j, \quad b_{ij} = \delta \times \beta_j \end{aligned} \tag{2.9}$$

Алгоритми для виявлення структур [28] реалізація яких зазвичай починається з нормалізації даних по рядках або стовпцях із використанням відповідних середніх значень. Для оцінки подібності між рядками чи стовпцями застосовується евклідова відстань, яка і визначає порядок групування.

Когерентні бікластери — ще один важливий тип, де значення елементів обчислюються за наступною формулою:

$$b_{ij} = \delta + \beta_i + \beta_j,$$
$$b_{ij} = \delta \times \beta_i \times \beta_j \qquad (2.10)$$

[29]. Алгоритм використовує параметр дельта , який задає граничну величину середньоквадратичної похибки. Час виконання алгоритму обмежується умовою , де — міра розподілу ознак у бікластері. Значення безпосередньо впливає на структуру і кількість знайдених бікластерів.

Порівняння та типологію наявних алгоритмів бікластеризації показано на структурній схемі на (рис. 2.4.)

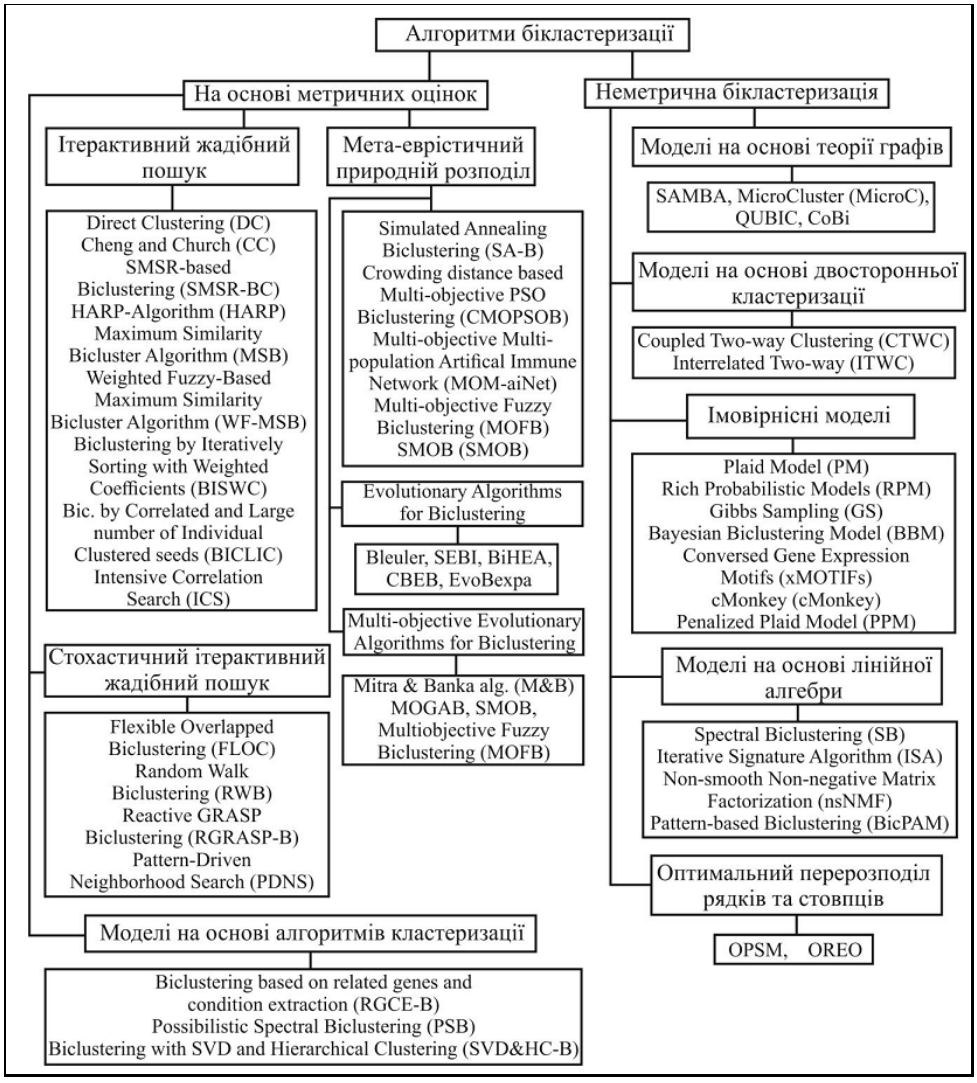


Рис. 2.4. Структрурна схема наявних алгоритмів бікластеризації

Метрики, які часто застосовуються в таких алгоритмах, включають: дисперсію ознак у межах бікластера (алгоритм DC):

$$VAR\left(\vec{\mathbf{i}},\vec{\mathbf{j}}\right)=\sum_{i=1}^{k}\sum_{j=1}^{s}\left(b_{ij}-\bar{b}_{\vec{\mathbf{ij}}}\right)^2 \tag{2.11}$$

середньоквадратичну похибку (алгоритм CC) [54]:

$$MSR\left(\vec{\mathbf{i}},\vec{\mathbf{j}}\right)=\frac{1}{k\times s}\sum_{i=1}^{k}\sum_{j=1}^{s}\left(b_{ij}-\bar{b}_{i\vec{\mathbf{j}}}-\bar{b}_{\vec{\mathbf{i}}j}+\bar{b}_{\vec{\mathbf{ij}}}\right)^2; \tag{2.12}$$

модифіковану середньоквадратичну похибку (SMSR-CC):

$$SMSR\left(\vec{\mathbf{i}},\vec{\mathbf{j}}\right)=\frac{1}{k\times s}\sum_{i=1}^{k}\sum_{j=1}^{s}\frac{\left(\bar{b}_{i\vec{\mathbf{j}}}\times\bar{b}_{\vec{i}j}-b_{ij}\times\bar{b}_{\vec{\mathbf{ij}}}\right)^2}{\bar{b}_{i\vec{\mathbf{j}}}{}^2\times\bar{b}_{\vec{i}j}{}^2};\tag{2.13}$$

індекс релевантності (HARP), що враховує локальну та глобальну дисперсію матриці:

$$RI\left(\vec{\mathbf{i}},\vec{\mathbf{j}}\right)=\sum_{j=1}^{s}\left(1-\frac{\sigma_{\vec{i}j}^2}{\sigma_{\vec{\mathbf{ij}}}^2}\right);\tag{2.14}$$

Окрім того, використовуються і додаткові метрики. Інша група методів бікластеризації базується на графовій теорії, ймовірнісних моделях , методах лінійної алгебри, а також оптимізаційних алгоритмах для перерозподілу рядків і стовпців.

Велика кількість існуючих алгоритмів бікластеризації зумовлює значну варіативність отриманих результатів. До того ж, деякі з методів орієнтовані виключно на бінаризовані дані, що потребує попередньої трансформації профілів ознак у бінарний формат із використанням медіани, середнього значення або інших статистичних параметрів. Така передобробка може призвести до втрати частини інформації, що впливає на точність подальшого аналізу.

2.5. Різновиди класифікацій

Класифікація — це процес впорядкування об’єктів, явищ або процесів на основі їхніх спільних характеристик для полегшення аналізу та дослідження. Вона дозволяє групувати схожі об'єкти у категорії відповідно до заданих критеріїв та проводити висновки про властивості окремих класів. Основним принципом класифікації є виявлення ознак, які дають змогу розрізняти або поєднувати елементи за схожістю.

У рамках задач з навчанням під наглядом класифікація передбачає використання попередньо маркованого набору даних, на основі якого створюється навчена модель. Накопичені спектральні дані, збережені в базах, можуть служити як для виявлення окремих речовин, так і для формування математичних моделей, які вирішують завдання розпізнавання.

Одним із добре відомих підходів у класифікаційних задачах є метод опорних векторів (SVM). Його суть полягає у побудові гіперплощини, яка найкращим чином розділяє об'єкти різних класів у багатовимірному просторі ознак. Така гіперплощина обирається так, щоб максимізувати відстань між нею та найближчими об'єктами кожного класу — так званими опорними векторами. Цей метод ефективний навіть у випадках, коли класи складно розділити лінійно.

Ще одним розповсюдженим підходом є байєсівський класифікатор, що оцінює ймовірність належності об'єкта до певного класу, обираючи найвірогідніший варіант. Спрощеною формою цього методу є наївний байєсівський класифікатор, який передбачає незалежність між усіма ознаками. Хоча це припущення часто не відповідає реальності, метод все ж забезпечує доволі точні результати. Його обмеженням є нездатність безпосередньо працювати з безперервними змінними, які потребують попереднього переведення у дискретну форму, що іноді призводить до втрати даних.

Більш гнучким підходом, що враховує залежності між ознаками, є використання байєсівських мереж. Вони дозволяють інтегрувати апріорну інформацію та взаємозв’язки між ознаками, що підвищує точність моделювання в складних системах.

Серед інших методів варто згадати алгоритм *k*-найближчих сусідів (k-NN). Цей метод відносять до "ледачих" алгоритмів, оскільки він не передбачає фази навчання. Замість цього новий об'єкт класифікується на основі схожості з найближчими прикладами з навчального набору. Це непараметричний метод, що не вимагає припущень про розподіл даних, проте має певні недоліки — зокрема, значне навантаження на ресурси при великих обсягах інформації та чутливість до вибору метрики відстані. Значення параметра *k* також істотно впливає на результат:

малі значення можуть зробити модель надто чутливою до шумів, тоді як великі — надто узагальнюють класи. Оптимальний k часто визначають за допомогою перехресної перевірки leave-one-out.

Ефективним інструментом також вважається метод випадкового лісу (Random Forest). Це ансамблевий алгоритм, що поєднує багато дерев рішень, які навчаються незалежно одне від одного на різних підмножинах даних. Остаточне рішення приймається шляхом голосування дерев у задачах класифікації або усереднення їх прогнозів у задачах регресії. Такий підхід дозволяє зменшити ймовірність перенавчання, але потребує чималих обчислювальних ресурсів. Для досягнення оптимальних результатів слід правильно підібрати параметри, зокрема кількість дерев та їх глибину — чим більше дерев, тим точніше модель, але це збільшує час навчання.

Окремо варто згадати дерева рішень — алгоритм, який має просту й інтерпретовану структуру, що дозволяє його застосовувати як у задачах класифікації, так і регресії. Дерево складається з вузлів, де перевіряються певні значення ознак, і гілок, що ведуть до різних підмножин даних або виходів. Основна перевага — наочність і здатність працювати з різними типами змінних. Однак, якщо дерево має надмірну кількість гілок і вузлів, зростає ризик перенавчання.

У підсумку, вибір методу класифікації залежить від цілей дослідження, обсягів даних, їх природи та вимог до точності. Кожен з алгоритмів має свої сильні й слабкі сторони, і для досягнення найкращого результату часто використовують комбіновані або гібридні підходи.

РЕЗУЛЬТАТИ ЕКСПЕРИМЕНТУ ТА ОБГОВОРЕННЯ

3.1. Попередня підготовка мас-спектрів та формування векторного представлення

На етапі попередньої обробки мас-спектроскопічних даних основною метою є перетворення мас-спектрів досліджуваних речовин у числову форму, придатну для подальшого аналізу методами машинного навчання, зокрема кластеризації.

Кожен мас-спектр представляє собою набір іонних піків, кожен з яких характеризується значенням маси/заряду (m/z) та відповідною інтенсивністю сигналу. Зважаючи на наявність апаратного шуму в системі детекції, до аналізу включаються лише ті піки, інтенсивність яких принаймні втричі перевищує рівень шуму приладу. Це дозволяє виключити випадкові коливання та хибні сигнали, що не мають хімічної значущості.

Для формалізації мас-спектру використовувалося векторне представлення. Кожен спектр описується у вигляді вектора фіксованої розмірності, де координатами є значення інтенсивностей сигналів для кожного цілочисельного значення m/z у заданому діапазоні. В проведених вимірюваннях верхня межа сканування становила 450 одиниць m/z, тому вектор ознак мав 450 компонент. Кожна координата вектора відповідала конкретному значенню маси/заряду, а її числове значення — інтенсивності відповідного піку. У випадках, коли в спектрі конкретної речовини пік за відповідним значенням m/z був відсутній, інтенсивність вважалася рівною рівню шуму приладу.

При цьому передбачалося, що всі зафіксовані іони є однозарядними, оскільки режим іонізації був налаштований таким чином, що ймовірність утворення багатозарядних іонів залишалася вкрай низькою. Це дозволяло трактувати індекси координат вектора як масу іона без додаткових перерахунків.

Таким чином, кожен мас-спектр було приведено до уніфікованого векторного формату, де:

Розмірність вектора відповідала максимальному значенню m/z у межах сканування;

Кожна координата вектора містила інтенсивність піку або рівень шуму при його відсутності.

Отримане векторне представлення мас-спектрів стало базисом для подальшого розрахунку дескрипторів та застосування методів кластерного аналізу, що дозволяє групувати молекули за схожістю їхніх спектральних характеристик.

3.2. Оптимізація методів автоматизованого розбиття молекулярних мас-спектрів на класи хімічних речовин.

Розглядаємо дані мас-спектрів хлорогранічних молекул, що складаються з 450 вимірювань, де є дві ознаки — маса іона та інтенсивність відповідного піка. Вибираємо координати вектора ознак, що відповідають інтенсивності піків. Тому наш вектор має 450 координат. Далі обираємо опорний радіус-вектор Ro (відносно якого порівнюватимемо спектри). В даному випадку це спектр C12H10 (біфеніл).

Порівнюємо його з іншими спектрами. Для цього беремо мас-спектр CI (3-хлорбіфеніл) і визначаємо його радіус-вектор Rx. Для порівняння обчислюємо різницю між векторами Ro та Rx — отримуємо вектор Rd. Також знаходимо довжину цього вектора.

Для порівняння також потрібно знайти кут між векторами Ro та Rd, використовуючи формули:

$$\cos = Ro*RdRo*Rx \quad (3.1)$$

$$\alpha = \arccos f_0 (Ro*RdRo*Rd) \quad (3.2)$$

Отримані вектори відображаємо в полярній системі координат (рис. 3.1.).

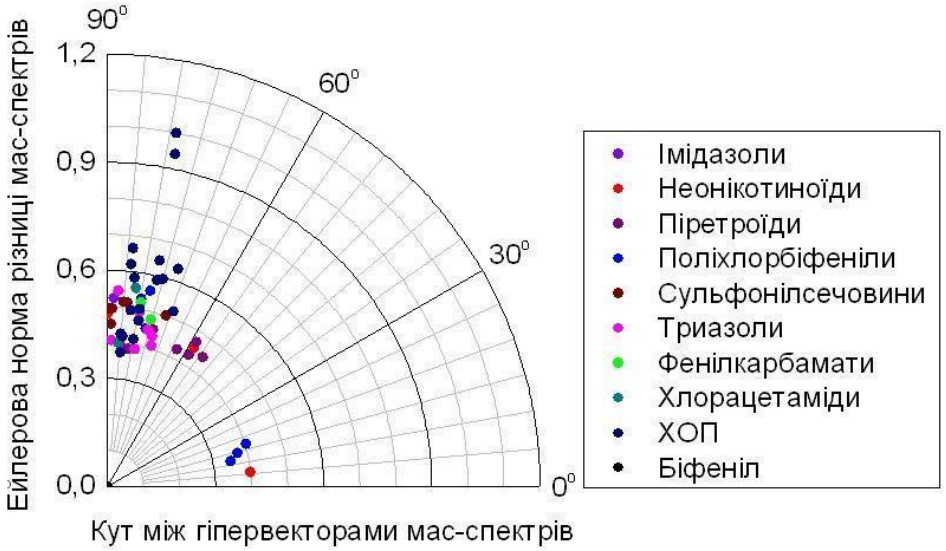


Рис. 3.1. Отримані кластери в полярній системі координат

3.3. Кластеризація молекул методом k-means

На рис. 3.2. показано результати кластеризації за допомогою методу k-means.

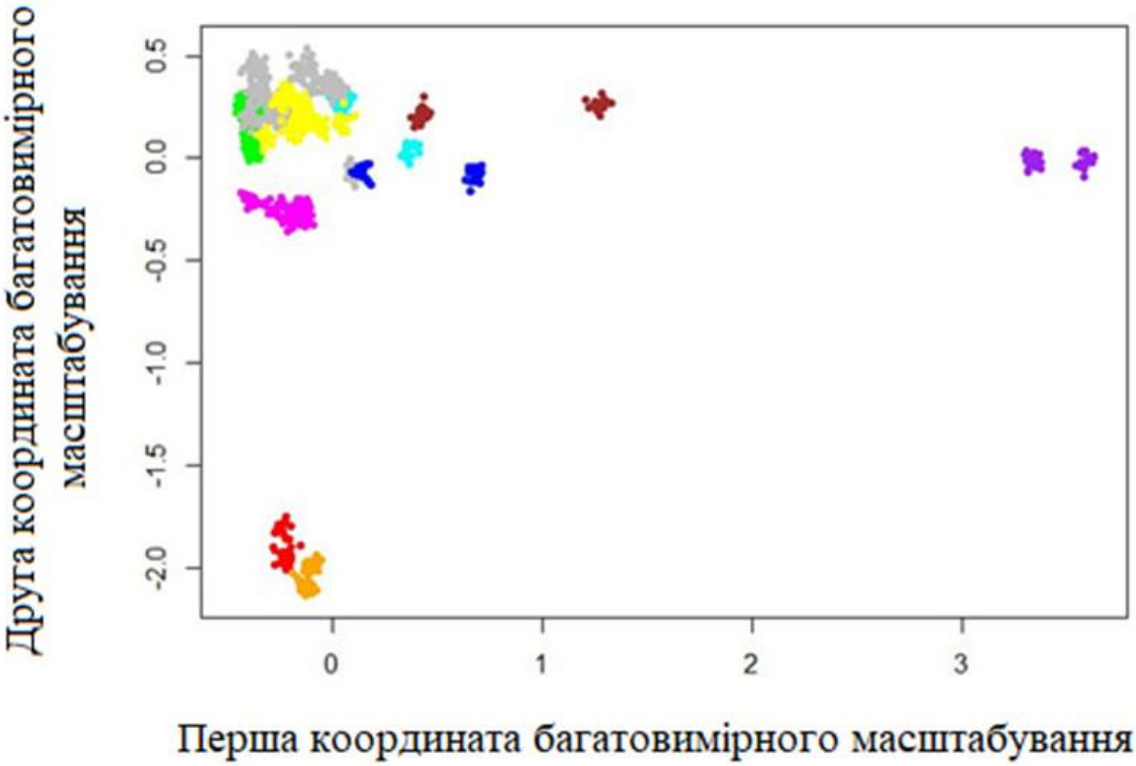


Рис. 3.2. Результати кластеризації за допомогою методу k-means

У цьому випадку були використані такі класи: червоний — біфеніл, синій — імідазол, зелений — неонікотиноїд, оранжевий — піретроїд, пурпурний — поліхлорбіфеніл, жовтий — сульфонілсечовина, блакитний — триазол, пурпурний — фенілкарбамат, коричневий — хлорацетамід, сірий — ХОП.

Цей метод використано в середовищі RStudio для кластеризації. Кожна змінна була шкалована, щоб уникнути викидів і забезпечити рівні вимірювання.

3.4. Класифікація мас-спектрів методом "Дерево рішень"

Для класифікації та прогнозування молекул було застосовано метод "Дерево рішень". Усього було надано 1059 спостережень і 457 змінних.

Після побудови моделі, багато змінних виявилися незначущими. Для побудови моделі була використана тренувальна вибірка з 1000 спостережень, а тестова містила 59 спостережень. Точність моделі становить 100%. Результат зображено на рис. 3.3.

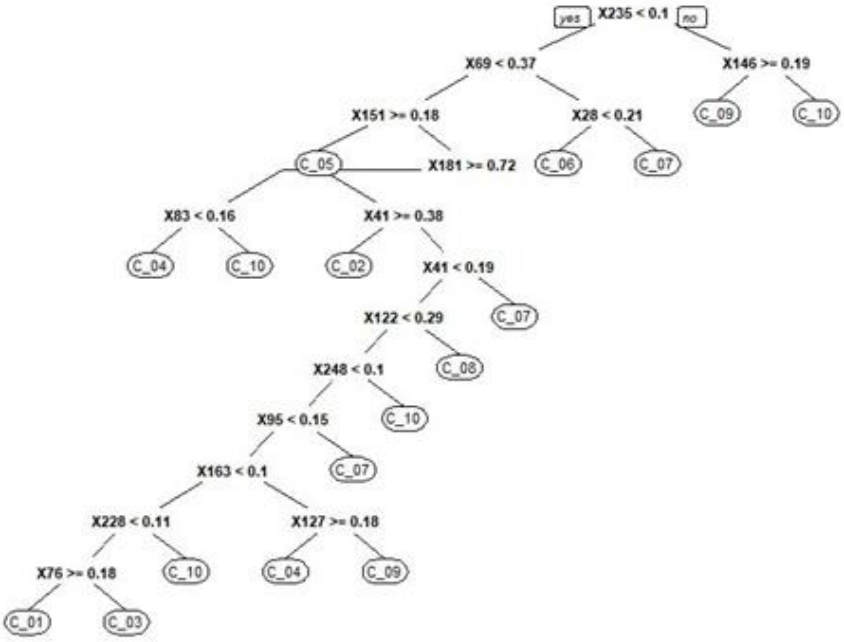


Рис. 3.3. Результат класифікації

Для прикладу застосування класифікатора розглядаємо молекули піретроїдів (λ-цигалотрин, дельтаметрин, циперметрин). Структури цих молекул наведені на рис. 3.4.

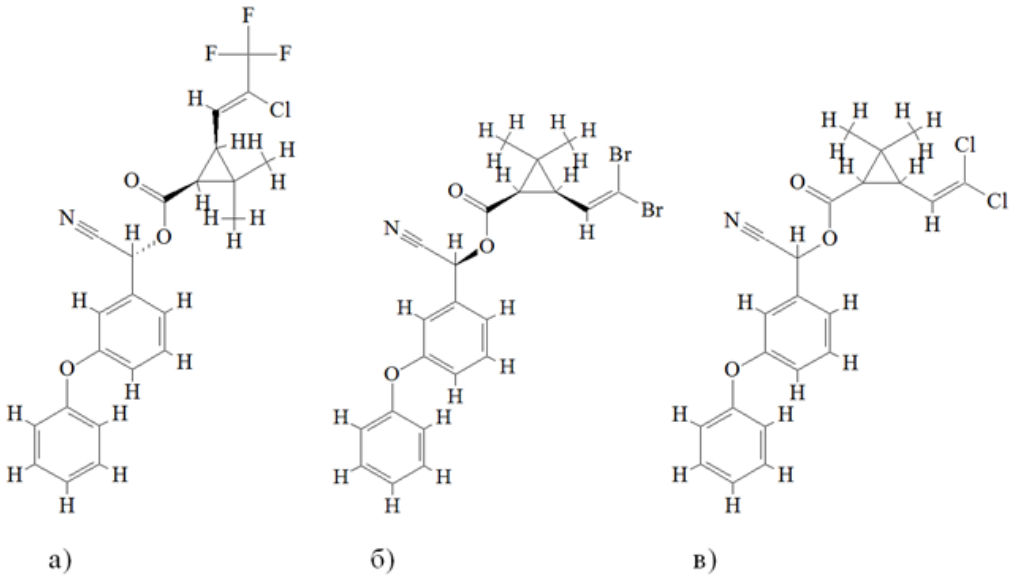


Рис. 3.4. Структури молекули піретроїдів

Ці молекули мають спільний структурний компонент, а саме α-циано-3-фіноксібензильний фрагмент (рис.3.5.).

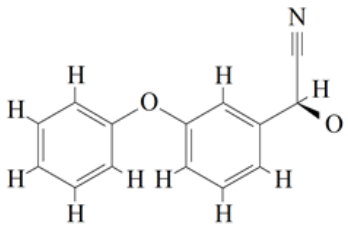


Рис. 3.5. α-циано-3-фіноксібензильний фрагмент

3.5. Попередня підготовка даних для бікластеризації мас-спектрів.

Для виконання підготовчих дій було встановлено програмне забезпечення ChemDraw Pro — редактор молекулярних структурних формул, сумісний з усіма сучасними версіями операційної системи Windows. Після встановлення і активації

програми за допомогою відповідного серійного номера було розпочато заповнення таблиці, де програма автоматично відтворювала структурну формулу молекули. Після цього структура копіювалась до відповідної клітинки таблиці у Excel без необхідності ручного малювання.

Подальша робота виконувалась для мас-спектрів поліхлорбіфенілів (ПХБ). У кожному з файлів графічні зображення структурних формул замінювалися на об’єкти, створені за допомогою ChemDraw. Це дозволило забезпечити інтегровану роботу з хімічними структурами без потреби у сторонньому програмному забезпеченні. Ця частина слугувала підготовкою до основного завдання в межах роботи. Основною метою є проведення внутрішньої кластеризації мас-спектрів молекул, що мають спільні структурні елементи. Для цього необхідно сформувати збалансований простір векторів. Починаючи з одного з мас-спектрів ПХБ, обирається піковий сигнал, що відповідає молекулярному іону, інтенсивність якого перевищує рівень шуму. Рівень шуму визначається за формулою:

$$\sigma = 0.06 * \sqrt{\sum I^2} \tag{3.3}$$

де I — інтенсивності всіх іонів у спектрі.

Далі, використовуючи редактор структурних формул, із фрагментів вихідної молекули формуються всі можливі структурні варіанти, чия молекулярна маса не перевищує масу обраного піка. В процесі побудови ідентифікуються такі комбінації атомів, видалення яких з молекули дозволяє досягти бажаної маси, відповідної певному піку.

У мас-спектрі всі піки з масами, що перевищують знайдені фрагменти, замінюються випадковими значеннями, що імітують шум. Ці значення моделюються як гаусівський шум з дисперсією σ і математичним сподіванням 0.5σ. Для кожної ітерації формується окремий шумовий мас-спектр. Процедура повторюється для всіх піків у спектрі та для кожного з мас-спектрів ПХБ. На (рис. 3.6, 3.7) зображено залежність інтенсивності від маси молекули.

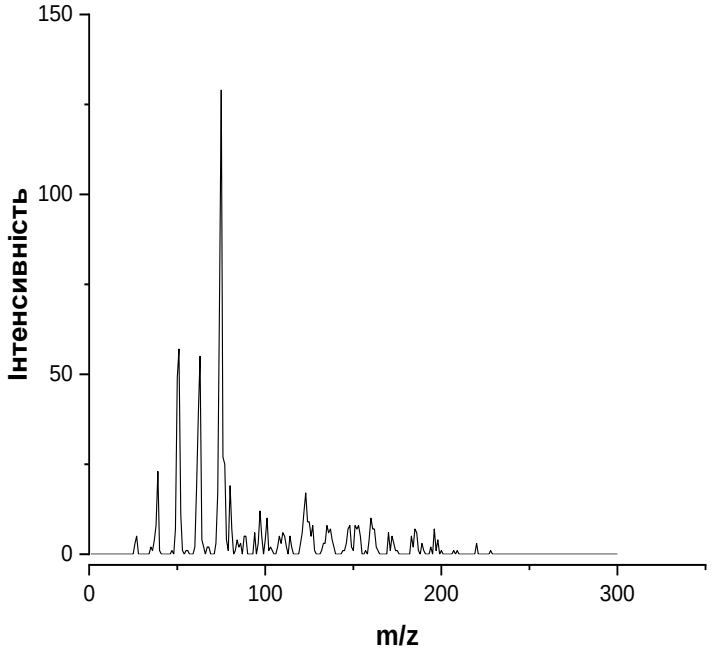


Рис. 3.6. Модифікований мас-спектр для піку з порядковим номером 75

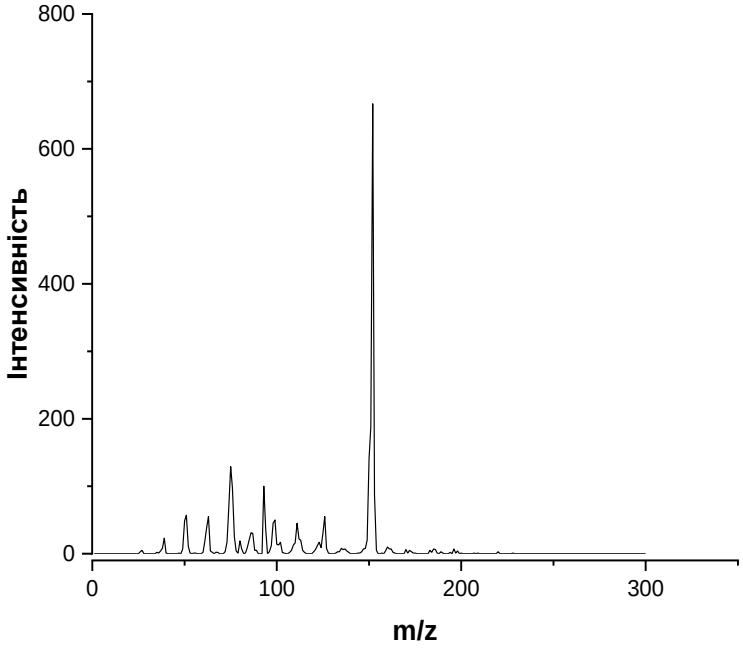


Рис. 3.7. Модифікований мас-спектр для піку з порядковим номером 152

Це є етапом розмноження мас-спектрів, тобто штучне створення нових варіацій шляхом додавання до оригінального спектра шуму, згенерованого згідно з розрахованою дисперсією σ. Це дозволяє підвищити репрезентативність даних та забезпечити можливість коректного проведення внутрішньої кластеризації фрагментів за спільними структурними характеристиками.

Після формування масивів структурних формул та відповідних мас-спектрів, для кожного піку створюється кілька варіантів з урахуванням шумових впливів. Це допомагає отримати розширені вибірки для подальшого аналізу.

3.6. Бікластеризація мас-спектрів

Отже, запропонований підхід базується на двоетапній кластеризації (бікластеризації) в просторі векторних ознак мас-спектрів.

На першому етапі – так званій вертикальній кластеризації – виконується групування мас-спектрів різних зразків за схожістю їхніх пікових профілів. Кожен отриманий кластер у цьому випадку корелює з певним хімічним класом сполук (наприклад, поліхлорбіфеніли чи різні класи пестицидів) та відображає наявність характерного набору фрагментів структури молекули, що проявляється у вигляді стійкого поєднання m/z-піків. Такий поділ дає змогу провести попередню ідентифікацію невідомої речовини на рівні хімічного класу та обмежити подальший аналіз відповідним підмножинами даних.

На другому етапі – горизонтальній кластеризації – усередині кожного вертикального кластера здійснюється подальше групування окремих іонів (пік-співвідношень), що належать до одного мас-спектра. Ця внутрішньокластерна кластеризація дозволяє виділити структурні фрагменти молекули, які характеризуються набором споріднених інтенсивностей і мас-іонів і, відповідно, відбитків їхньої фрагментації. (Рис. 3.8.) ілюструє розподіл m/z-піків одного спектра на окремі кластери (методом k-means з використанням метрики косинусна подібність), кожен з яких може асоціюватися з певною частиною молекулярної структури.

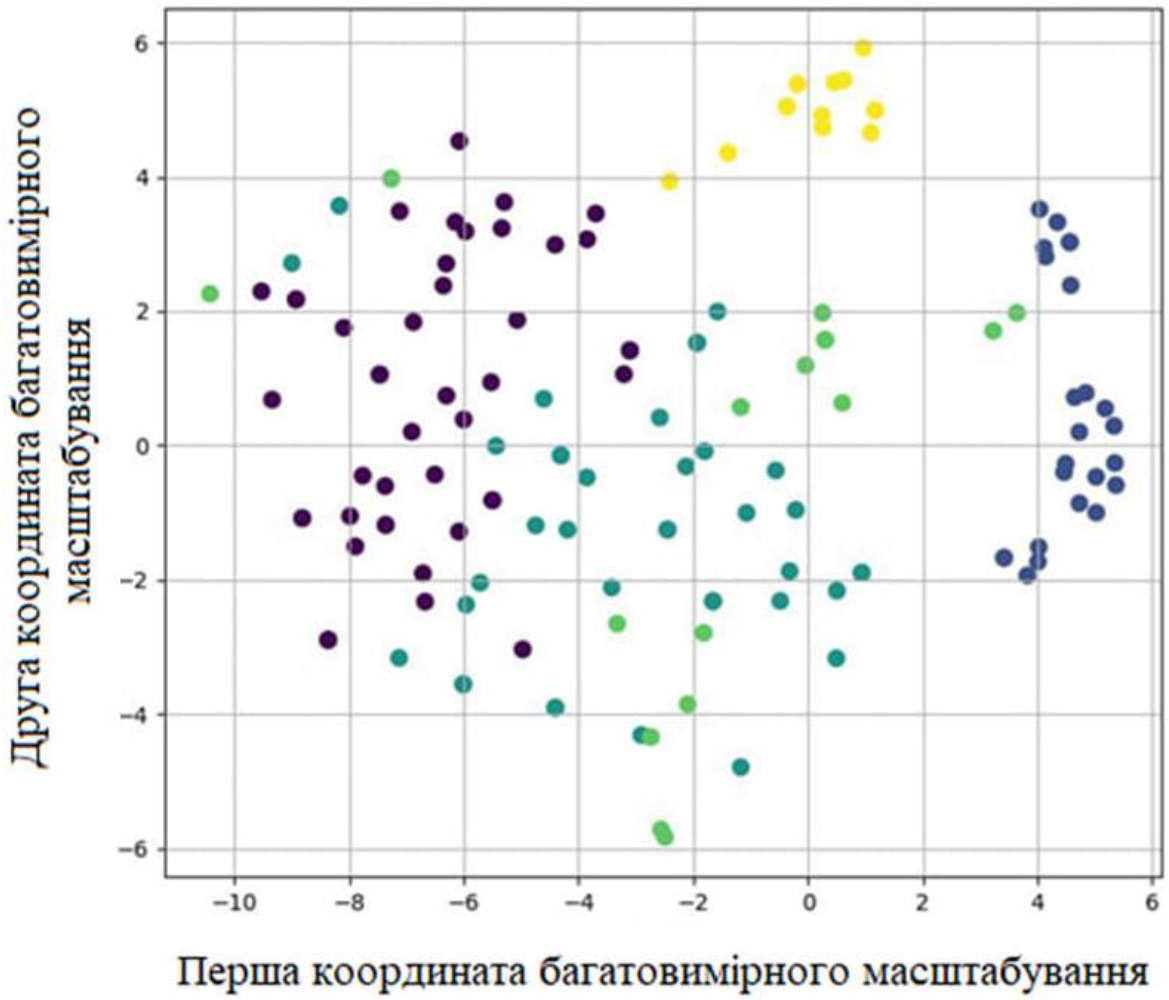
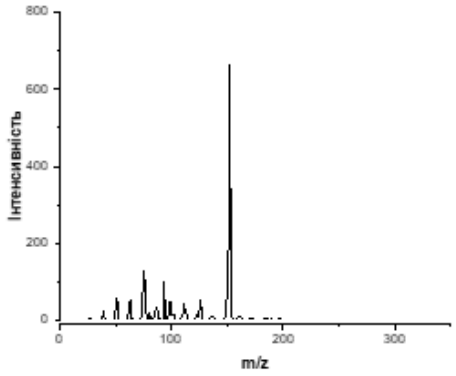
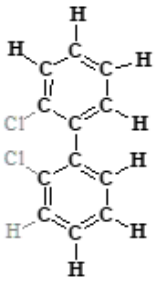
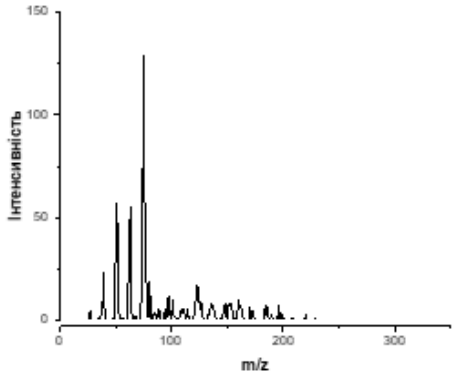
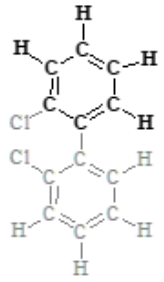
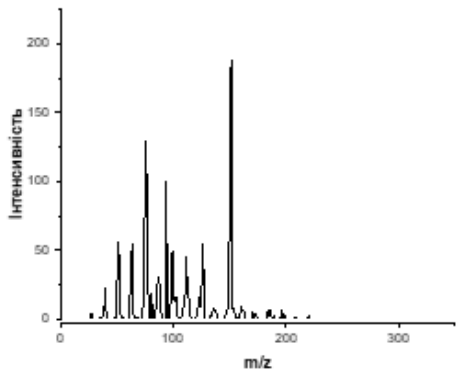
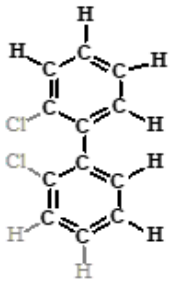
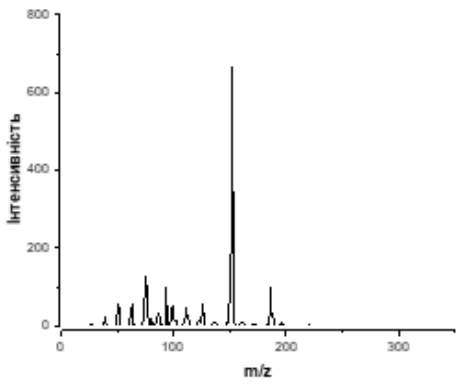
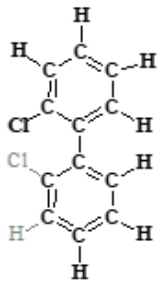


Рис. 3.8. Кластеризація інтенсивностей методом k-means, де (фіолетовий clust0- маса 152, синій clust1- маса 75, блакитний clust2- маса 151, зелений clust3- 186, жовтий clust4- маса 222)

Такий підхід забезпечує глибоку структурну ідентифікацію: за наявності мас-спектра досліджуваної речовини в навчальній множині можна досить точно відтворити як її хімічний клас, так і часткову молекулярну структуру, що показано на (рис. 3.9.)

Позначення бікластера	Масспектр	Структура молекули
BICLUSTER 0		
BICLUSTER 1		
BICLUSTER 2		
BICLUSTER 3		

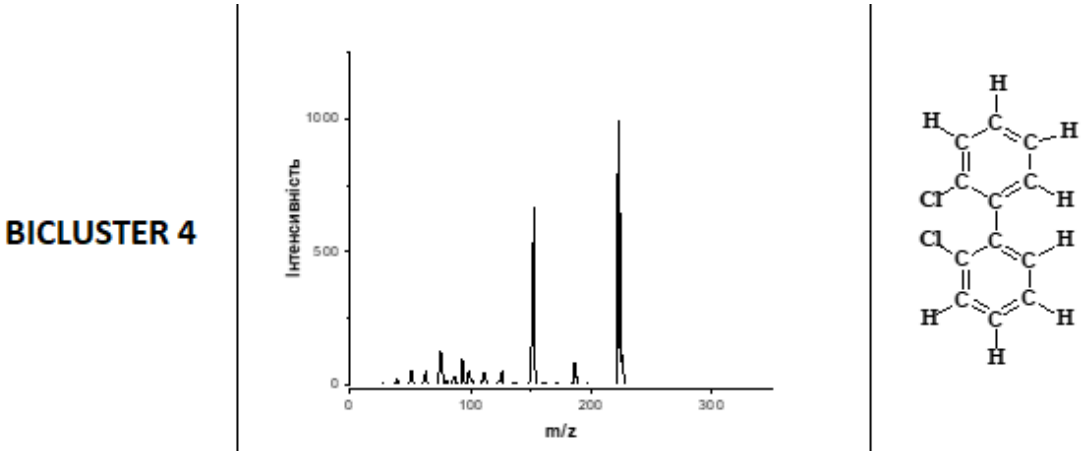


Рис. 3.9. Відповідність структурних фрагментів за центральною масою у кластерах.

Перевагою бікластеризації є можливість багаторівневого аналізу: спочатку – гранулярного розподілу спектрів за віднесенням до великих груп речовин, а потім – детального виділення внутрішніх характеристик кожного спектра. У контексті поліхлорбіфенілів це дозволяє, наприклад, відрізняти ізомери та класи конгенерів за характерними піковими наборами; для пестицидів – виділяти функціональні фрагменти та групи атомів, що забезпечує не лише класову, а й молекулярно-структурну ідентифікацію. Такий поетапний метод упорядковує простір векторних ознак і значно підвищує надійність та точність аналітичних висновків.

ВИСНОВКИ

В результаті виконаної роботи :

Проведено аналіз машинних методів ідентифікації об’єктів, який показав, що комбунівання алгоритмів бікластеризації та класифікації мас-спектрів багатоатомних молекул є перспективним підходом для ідентифікації молекулярної структури невстановлених речовин.

Розроблено та реалізовано на експериментально отриманих масивах мас-спектрів хлорорганічних сполук оригінальні простори ознак для представлення мас-спектрів багатовимірними векторами з використанням метрик евклідового простору та косинусної подібності, як критеріїв схожості мас-спектрів.

Розроблено та реалізовано на платформі Jupiter мовою Python оригінальний програмний модуль, який втілив автоматизацію процедур попередньої підготовки експериментальних даних, кластеризації, бікластеризації та класифікації мас-спектрів багатоатомних молекул.

Проведено дослідження, які показали, що кластеризація алгоритмом k-means мас-спектрів молекул різних класів хімічних сполук дає збіг кількості кластерів мас-спектрів і кількості досліджених хімічних класів молекул. Додатковими статистичними дослідженнями була встановлена однозначна відповідність між кластерами мас-спектрів і хімічними класами молекул.

Проведено дослідження, які показали, що процедура бікластеризації виділяє в мас-спектрах молекул певного хімічного класу окремі групи іонів, які пов’язані з відповідними фрагментами молекулярної структури. Сформульовано правила встановлення однозначної відповідності між бікластерами, які інтерпретуються як відокремлені групи іонів, і фрагментами молекулярної структури. Це дозволяє з достовірністю методу бікластерізації частково ідентифікувати молекулярну структуру невстановлених речовин за їхніми мас-спектрами.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. M. Altarawneh, B.Z. Dlugogorski, E.M.Kennedy and J.C. Mackie, MECHANISMS FOR PCDF AND PCB FORMATION FROM FIRES: PATHWAYS FROM OXIDATION OF CHLOROBENZENES//Proceedings of 7th AOFST Symposium, Hong Kong SAR, China, 2007 p.p.130-142.

2. Scott G., Huling and Bruce E.P ivetz In-Situ Chemical Oxidation//Engineering Issue, EPA/600/R-06/072 August 2006 P.60.

3. Cherkasov, et al. QSAR modeling: where have you been? Where are you going to? J. Med. Chem., 57 (2014), pp. 4977-5010

4. H. Kubinyi Free Wilson analysis. Theory, applications and its relationship to Hansch analysis Quant. Struct. Act. Relat., 7 (1988), pp. 121-133

5. R.P. Sheridan, S.K. Kearsley Why do we need so many chemical similarity search methods? Drug Discov. Today, 7 (2002), pp. 903-911

6. A.G. Maldonado, et al. Molecular similarity and diversity in chemoinformatics: from theory to applications Mol. Divers., 10 (2006), pp. 39-79

7. J. Bajorath Molecular similarity concepts for informatics applications Methods Mol. Biol., 1526 (2017), pp. 231-245

8. N.M. Nasrabadi Pattern recognition and machine learning J. Electron. Imag., 16 (2007), p. 049901

9. E. Kondratovich, et al. Transductive support vector machines: promising approach to model small and unbalanced datasets Mol. Inf., 32 (2013), pp. 261-266

10. Hyvarinen, E. Oja Independent component analysis: algorithms and applications Neural Netw., 13 (2000), pp. 411-430

11. J.D. MacCuish, N.E. MacCuish Chemoinformatics applications of cluster analysis Comput. Mol. Sci., 4 (2014), pp. 34-48

12. L.B. Akella, D. DeCaprio Cheminformatics approaches to analyze diversity in compound screening libraries Curr. Opin. Chem. Biol., 14 (2010), pp. 325-330.

13. Soni N., Ganatra A. Categorization of Several Clustering Algorithms from Different Perspective: A Review. International Journal of Advanced Research in Computer Science and Software Engineering, 2012. Vol. 2, issue 8. P. 63–68.

14. Murtag F., Contreras P. Algorithms for hierarchical clustering: an overview. Data Mining and Knowledge Discovery, 2012. ol. 2, issue 1. P. 86–97.

15. Bodyanskiy Y. V., Deineko A. O., Kutsenko Y. V., Zayika O. O. Data streams fast EM-fuzzy clustering based on Kohonen's self-learning. Proceedings of the 2016 IEEE 1st International Conference on Data Stream Mining and Processing, DSMP 2016. P. 309–313.

16. Dorazo J., Carazo J. M. Phylogenetic reconstruction using an unsupervised growing neural network that adopts the topology of a phylogenetic tree. Journal of Molecular Evolution, 1997. № 44(2). P. 226–234.

17. Івахненко О. Г. Метод групового урахування аргументів конкурент методу стохастичної апроксимації. Автоматика, 1968. №3. С. 58–72.

18. Madala H. R., Ivakhnenko A. G. Inductive Learning Algorithms for Complex Systems Modeling. CRC Press, 1994. 365 p.

19. Abonyi J. Feil B. Cluster Analysis for data Mining and System Identification. Birkhäuser Verlag AG, 2007.

20. Dunn J. A fuzzy relative of the ISODATA process and its use in detecting compact well separated clusters. Journal of Cybernetics, 1974. P. 32–57.

21. Ester M., Kriegel H. Sander J. A density-based algorithm for discovering clusters in large spatial databases. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 1996. P. 226–231.

22. Kriegel H.-P., Kröger P., Sander J, Zimek A. "Density-based clustering". Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2011. № 1(3). P. 231–240.

23. Grosan C., Abraham A. Hybrid Evolutionary Algorithms: Methodologies, Architectures, and Reviews. Studies in Computational Intelligence (SCI), 2007. Vol. 75. P. 1–17.

24. Berkhin P. A Survey of Clustering Data Mining Techniques. Grouping Multidimensional Data. Recent Advances in Clustering. Springer-Verlag Berlin Heidelberg, 2006. P. 25–72.

25. Pontes B., Giráldez R., Aguilar-Ruiz J. S. Biclustering on expression data: A review. Journal of Biomedical Informatics, 2015. №57. P. 163–180.

26. Eren K., Deveci M., Kucuktunc O., Catalyurek U. V. A comparative analysis of biclustering algorithms for gene expression data. Briefings in Bioinformatics, 2012. Vol. 14, №3. P. 279–292.

27. Kluger Y., Basry R., Chang J. T., Gerstein M. Spectral biclustering of microarray data: co-clustering genes and conditions. Genome Resources, 2003. №13(4). P. 703–716.

28. Mukhopadhyay A., Maulik U., Bandyopadhyay S. On biclustering of gene expression data. Current Bioinformatics, 2010. №5. P. 204–216.

29. Califano A., Stolovitzky G., Tu Y. Analysis of gene expression microarrays for phenotype classification. In Proceedings of the International Conference on Computational Molecular Biology, 2000. P. 75–85.

30. Cheng Y., Church G. M. Biclustering of expression data // Proceedings of the 8th International Conference on Intelligen t Systems for Molecular Biology (ISMB’00), 2000. P. 93–103.

ДОДАТОК 1

```
# Робоча директорія
setwd("C:/Users/Робочий стіл/")
# Підключення даних
list.files()
library(readxl)
data <- read_excel("data.xlsx")
View(data)
# Подивимся датасет
dim(data)
sum(is.na(data))
data[1:20,456]
data[1:20,457]
# Видалим пропущенні значення
data <- na.omit(data)
# Видалим переміну name для дерева рішень
new_data <- data[,-457]
# Розіб'єбо вибірку на тренивочну та тестову
new_data_train <- new_data[1:1000,]
new_data_test <- new_data[-(1:1000),]

# Будуємо дерево рішень по тренивочній вибірці
library(rpart)
library(rpart.plot)
fit <- rpart(Clases ~., data = new_data_train)
prp(fit)
# Зроби прогноз на тестову вибірку
predict <- predict(fit, newdata = new_data_test, type = "class")
# Перевіримо точність моделі
correct_predictions <- sum(predict == new_data_test$Clases)
accuracy <- (correct_predictions / nrow(new_data_test)) * 100
accuracy
library(caret)
library(ggplot2)
library(ggthemes)
# Важливість переміних для дерева рішень
importance <- varImp(fit)
# Візуалізації важливості
imp_df <- data.frame(var = rownames(varImp(fit)),
   importance = varImp(fit)$Overall)
plotlY <- ggplot(imp_df, aes(x = var, y = importance, fill = var)) +
  geom_col(show.legend = FALSE) +
  scale_y_continuous(expand = c(0,0)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust =
0.5),
        plot.title = element_text(hjust = 0.5, size = 16),
        axis.title.x = element_text(size = 10)) +
  labs(title = "Графік важливості переміних",
       x = "Переміна",
       y = "Важливість")
```

```
plotlY

#####################################################################
############
################################ Кластеризація
#################################
#####################################################################
############
set.seed(123)
# Графік локтя
wss <- sapply(1:10, function(k) sum(kmeans(data[,1:455],
centers=k)$withinss))
df <- data.frame(k=1:10, WSS=wss)
ggplot(df, aes(x=k, y=WSS)) +
  geom_point() +
  geom_line() +
  labs(title = "Графік локтя",
       x = "Кількість кластерів", y = "WSS") +
  theme_bw()
# Кластеризація kmeans
colors <- c("red", "blue", "green", "orange", "purple", "yellow",
"cyan", "magenta", "brown", "gray")
fit_clust <- kmeans(data[, 1:455], 10, iter.max = 200)
dist.data <- dist(data[, 1:455])
mds.data <- cmdscale(dist.data)
plot(mds.data, col = colors[fit_clust$cluster], pch = 20)
```

ДОДАТОК 2

```
import pandas as pd
import numpy as np
from sklearn.metrics.pairwise import cosine_similarity
from sklearn.cluster import KMeans
from sklearn.manifold import TSNE
import matplotlib.pyplot as plt

# 1. Завантаження даних
file_path = r"c:\Users\lilya\OneDrive\Desktop\new\PSB.xlsx"
df = pd.read_excel(file_path, sheet_name="Лист1")

# 2. Попередня обробка: відокремлення назв та інтенсивностей
samples = df['mass']
intensity_data = df.drop(columns=['mass'])

# 3. Ігнорування нульових значень (замінюємо їх на NaN)
filtered_data = intensity_data.replace(0, np.nan)

# Видалення повністю порожніх рядків
valid_rows = filtered_data.dropna(how='all')

# Заповнення NaN значень нулями для розрахунку косинусної подібності
filled_data = valid_rows.fillna(0)

# 4. Косинусна подібність та перетворення у відстань
similarity_matrix = cosine_similarity(filled_data)
distance_matrix = 1 - similarity_matrix
distance_matrix[distance_matrix < 0] = 0  # усунення чисельних
похибок

# 5. Кластеризація (KMeans)
n_clusters = 5
kmeans = KMeans(n_clusters=n_clusters, random_state=42)
clusters = kmeans.fit_predict(similarity_matrix)

# 6. Зниження розмірності (t-SNE)
tsne = TSNE(n_components=2, random_state=42, perplexity=30,
metric='precomputed', init='random')
tsne_result = tsne.fit_transform(distance_matrix)

# 6.1 Центри мас у просторі t-SNE
tsne_df = pd.DataFrame(tsne_result, columns=['tsne_1', 'tsne_2'])
tsne_df['cluster'] = clusters
centroids = tsne_df.groupby('cluster')[['tsne_1', 'tsne_2']].mean()

# 7. Візуалізація з центрами мас
plt.figure(figsize=(10, 7))
scatter = plt.scatter(tsne_result[:, 0], tsne_result[:, 1],
c=clusters, cmap='viridis', label='Зразки')
```

```
# Наносимо центри мас
for cluster_id, row in centroids.iterrows():
    plt.scatter(row['tsne_1'], row['tsne_2'], marker='X',
color='red', s=200, edgecolor='black', label=f'Центр мас
{cluster_id}')

plt.title('Кластеризація інтенсивностей (KMeans + косинусна
подібність)')
plt.xlabel('t-SNE 1')
plt.ylabel('t-SNE 2')
plt.colorbar(scatter, label='Кластер')
plt.grid(True)
plt.legend()

# 8. Збереження зображення
output_path = "cluster_visualization.png"
plt.savefig(output_path)
plt.show()

# 9. Додаємо інформацію про кластер до таблиці з даними
clustered_df = valid_rows.copy()
clustered_df['cluster'] = clusters

# 10. Середні інтенсивності по кожному кластеру
cluster_summary = clustered_df.groupby('cluster').mean()

# 11. Пошук спільних мас (ознаки, що зустрічаються у >10% зразків
кластеру)
common_features = {}
threshold = 0.1  # 10%

for cluster_id in sorted(clustered_df['cluster'].unique()):
    cluster_data = valid_rows[clusters == cluster_id]
    nonzero_fraction = (cluster_data > 0).sum() /
cluster_data.shape[0]
    common_features[cluster_id] = nonzero_fraction[nonzero_fraction
> threshold].index.tolist()

# 12. Виведення результатів
print("\n=== Загальна характеристика кожного кластера ===\n")
for cluster_id in sorted(common_features.keys()):
    print(f"Кластер {cluster_id}:")
    print(f"- Кількість зразків: {(clustered_df['cluster'] ==
cluster_id).sum()}")
    print(f"- Спільні маси (інтенсивності > 0 у >{int(threshold *
100)}% зразків):")
    print(f"  {common_features[cluster_id]}\n")

# 13. Виведення центрів мас
print("\n=== Центри мас у просторі t-SNE ===\n")
print(centroids)
```

```
# 14. Пошук центрального зразка кожного кластера та його а.о.м.
clustered_df['mass'] = samples.loc[valid_rows.index].values  # 
додаємо назви зразків
central_samples = {}

for cluster_id in sorted(clustered_df['cluster'].unique()):
    cluster_indices = clustered_df.index[clustered_df['cluster'] == 
cluster_id]
    cluster_distances = distance_matrix[np.ix_(cluster_indices, 
cluster_indices)]

    mean_distances = cluster_distances.mean(axis=1)
    central_idx = cluster_indices[np.argmin(mean_distances)]
    central_sample_name = clustered_df.loc[central_idx, 'mass']

    intensity_vector = filled_data.loc[central_idx]
    present_masses = intensity_vector[intensity_vector > 
0].index.astype(str).tolist()

    central_samples[cluster_id] = {
        'name': central_sample_name,
        'aom': present_masses
    }

print("\n=== Центральні зразки кластерів та їх а.о.м. ===\n")
for cluster_id, info in central_samples.items():
    print(f"Кластер {cluster_id}:")
    print(f"- Центральний зразок: {info['name']}")
    print(f"- А.о.м.: {info['aom']}\n")
```