УДК 004.9

COMPARISON OF DATA CLUSTERING ALGORITHMS

ПОРІВНЯННЯ АЛГОРИТМІВ КЛАСТЕРИЗАЦІЇ ДАНИХ

Doroshenko I.V. / Дорошенко І.В.
c. p.-m.s., as.prof. / к. ф.-м.н., доц.
ORCID: 0000-0001-8729-1768
Chernivtsi National University, Chernivtsi, Kotsyubynskoho 2, 58012
Чернівецький національний університет, Чернівці, вул.Коцюбинського 2, 58012
Knihnitska T.V. / Книгніцька Т.В.
Doctor of Philosophy in Mathematics and Statistics / доктор філософії у галузі математики та статистики
ORCID: 0000-0003-4614-5945
Chernivtsi National University, Chernivtsi, Kotsyubynskoho 2, 58012
Чернівецький національний університет, Чернівці, вул.Коцюбинського 2, 58012
Kreshtanovych M.A. / Крештанович М.А.
magistr / магістр
Chernivtsi National University, Chernivtsi, Kotsyubynskoho 2, 58012
Чернівецький національний університет, Чернівці, вул.Коцюбинського 2, 58012

Анотація. *У статті проведено порівняння алгоритмів кластеризації даних: K-Means, Hierarchical Agglomerative Clustering (HAC), Density-Based Spatial Clustering of Applications with Noise (DBSCAN), Expectation–Maximization clustering using Gaussian Mixture Models (GMM). Порівняння здійснюється завдяки наперед згенерованим наборам даних, які мають різний характер поведінки: концентричні кола (2 кластери), смужки (3), хмари (3), нероздільна множина (1), серпи (2). Для кожного з наборів даних застосовано перелічені методи і визначено найкращий алгоритм кластеризації для певного типу даних. Розглянуті алгоритми кластеризації даних додатково застосовано до трьох наборів реальних даних. Для візуалізації результатів порівняння створено інтерактивний веб-застосунок для інтерактивної кластеризації даних згаданими алгоритмами, який розгорнуто на хмарному сервері shinyapps.io.*

Ключові слова: *кластеризація даних, кластерний аналіз, метод K-середніх, ієрархічна агломеративна кластеризація, просторова кластеризація заснована на щільності, моделі суміші Гауса.*

Вступ.

Проблема кластеризації даних широко вивчається в літературі для аналізу даних та машинного навчання у різних сферах життєдіяльності людини. Кластеризацію можна вважати короткою моделлю даних, яку можна інтерпретувати в сенсі підсумкової або генеративної моделі. Основну задачу кластеризації можна сформулювати так: маючи набір точок даних, розділити їх на набір груп, які максимально відрізняються. У той же час елементи, які відносяться до однієї групи, повинні бути максимально схожими. Міра

подібності між вимірюваннями даних визначається за допомогою Евклідової відстані, відстані Махаланобіса тощо.

Важливо враховувати природу даних та їх властивості при виборі методу кластеризації, оскільки неправильний вибір алгоритму кластеризації може призвести до некоректних результатів. Крім того, конкретний тип даних також має значний вплив на визначення проблеми. Наприклад, для числових даних може бути ефективним метод k-середніх, тоді як для категоріальних або текстових даних використовуються інші методи, такі як ієрархічна кластеризація або методи, що базуються на векторних представленнях.

1. Набір даних для кластеризації

У роботі використано сім наборів даних, п'ять з яких є згенерованими з нормального розподілу, два – з рівномірного розподілу. За допомогою лінійної комбінації згенерованих даних отримано наступні хмари даних, які потрібно розділити на кластери чотирьома методами кластеризації. Нижче показано (рисунок 1) п'ять типів вхідних даних. Наша мета – здійснити кластеризацію цих наборів за допомогою кожного алгоритму кластеризації та встановити, який алгоритм працює найкраще. Кожен із п’яти наборів даних (рисунок 1) названий відповідним чином до розсіювання точок даних: Галактика (Galaxy), Серпи (Sickle), Смужки (Slash), Око (Eye), Рій (Swarm).

Для того, щоб побачити, як на практиці працює кластеризація даних розглянемо ще три набори даних, отримані з платформи Kaggle. Перший набір даних доступний за наступним посиланням https://www.kaggle.com/datasets/arjunbhasin2013/ccdata?datasetId=14701&language=R. Ці дані стосуються сегментації клієнтів для визначення маркетингової стратегії. Даний набір даних узагальнює поведінку використання приблизно 9000 активних власників кредитних карток протягом останніх 6 місяців. Для кожного клієнта файл містить 18 поведінкових змінних. Для здійснення кластеризації використано два стовпці – BALANCE: сума балансу, що залишилася на рахунку клієнта для здійснення покупок та PURCHASES: загальна кількість покупок, здійснених з рахунку.

Другий набір даних https://www.kaggle.com/datasets/harrywang/wine-dataset-for-clustering є результатами хімічного аналізу вин, вирощених в одному регіоні Італії, але отриманих з трьох різних сортів винограду. Всього знайдено 13 типів складових у трьох типах вина. Для здійснення кластеризації даних використано такі складові: Alcohol та Malic_Acid.

Третій набір даних https://www.kaggle.com/datasets/uciml/electric-power-consumption-data-set містить вимірювання споживання електроенергії в одному домогосподарстві з однохвилинною частотою записів протягом майже 4 років. Використано такі змінні: Global_active_power та Global_reactive_power. Кожен із згаданих наборів даних названий відповідним чином – Card, Wine, Elect.

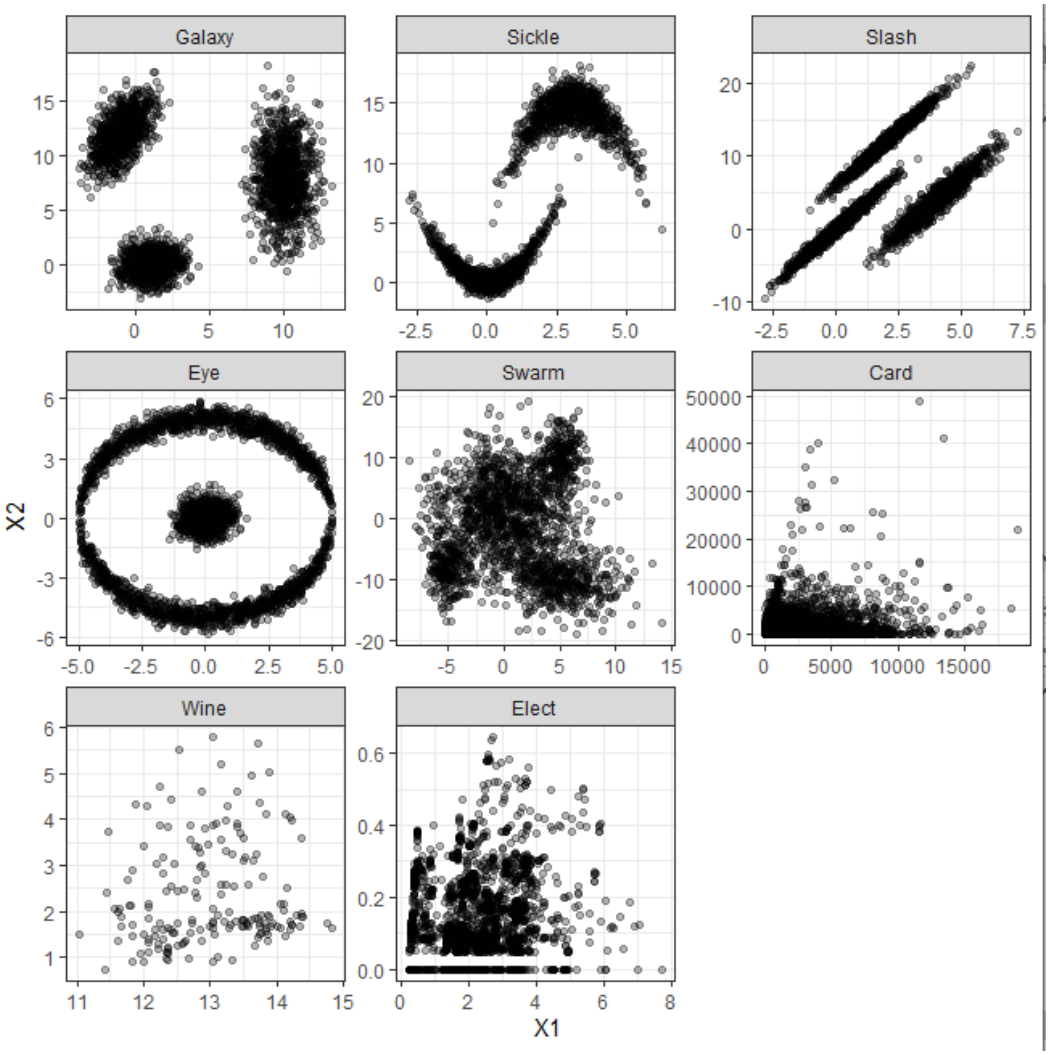

Рисунок 1 - Вхідні дані, які необхідно розділити на кластери

2. Результати кластеризації

У дослідженні згенеровано дані із нормального та рівномірного розподілів. Такого роду симуляції або симуляції за допомогою методу Монте-Карло [1] часто застосовують для перевірки роботи та порівняння алгоритмів. На додаток, використано реальні дані для здійснення кластеризації чотирьом методами.

Кожен із алгоритмів вимагає налаштування параметрів кластеризації для його якісної роботи. Кластеризація сильно залежить від конкретного набору даних і мети аналізу. Отже, розглянемо, як кожен алгоритм працює в кожному випадку. Кожен набір даних має дві функції $X1$ і $X2$, які генерують точки кластерів із наведених вище розподілів, і мітку набору даних (для цілей візуалізації даних).

Функція calc_cluster приймає набір даних і параметри для кожного алгоритму як аргументи, обчислює кластери та додає відповідні мітки до кожного кластера даних. Функція plot_cluster приймає назву набору даних як аргумент і будує графіки кластерів, обчислених кожним алгоритмом.

Для кожного набору даних параметри алгоритму змінюються, щоб уникнути помилкового враження про недостатню продуктивність. Важливо відзначити, що DBSCAN іноді виводить «нульовий» кластер даних, який вказує на аутлаєри, виявлені алгоритмом.

Отже, розглянемо результати кластеризації найпростішого випадку – Галактика (Galaxy). Як видно (рисунок 2) у цьому найпростішому випадку немає проблем (за винятком точки «зловмисника» у синьому кластері, заданих k-Means).

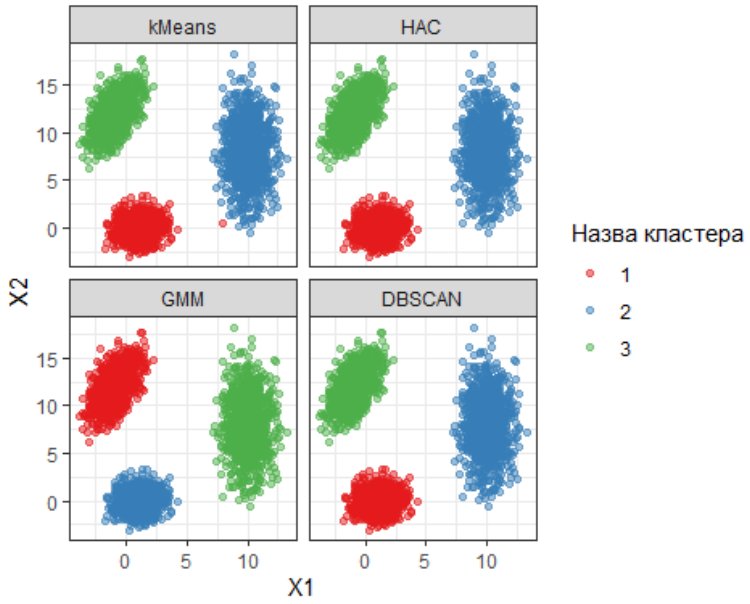

Рисунок 2 - Результати кластеризації набору даних Galaxy

Зрозуміло (рисунок 3), що найкраще з кластеризацією Sickle впоралися алгоритми DBSCAN та HAC. K-Means та GMM містять зелені точки даних у нижньому серпі. Це пов'язано з тим, що центр зеленого кластера знаходиться ближче до крайніх точок з нижнього кластера. Оце і є величезним недоліком вказаних двох підходів. Червона крапка у методі DBSCAN позначає нульовий кластер, тобто аутлаєр (викид). Алгоритм DBSCAN вважає червоне вимірювання аутлаєром в даних.

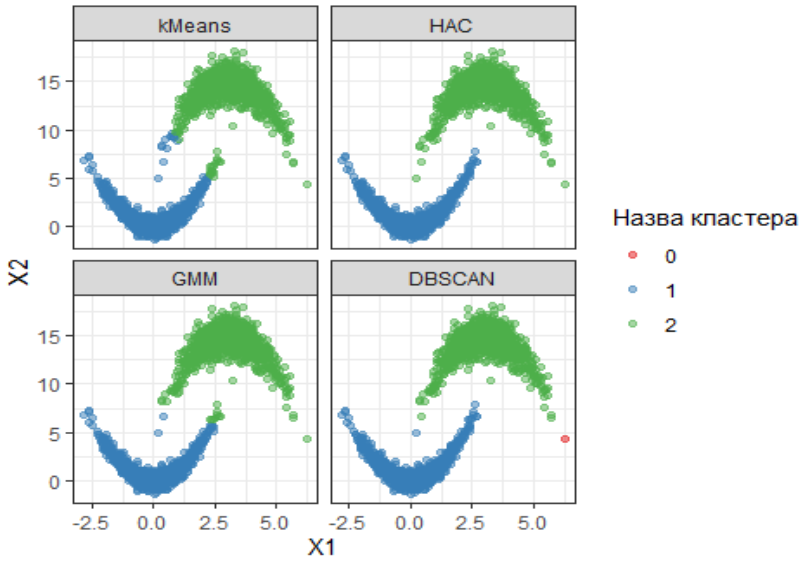

Рисунок 3 - Результати кластеризації набору даних Sickle

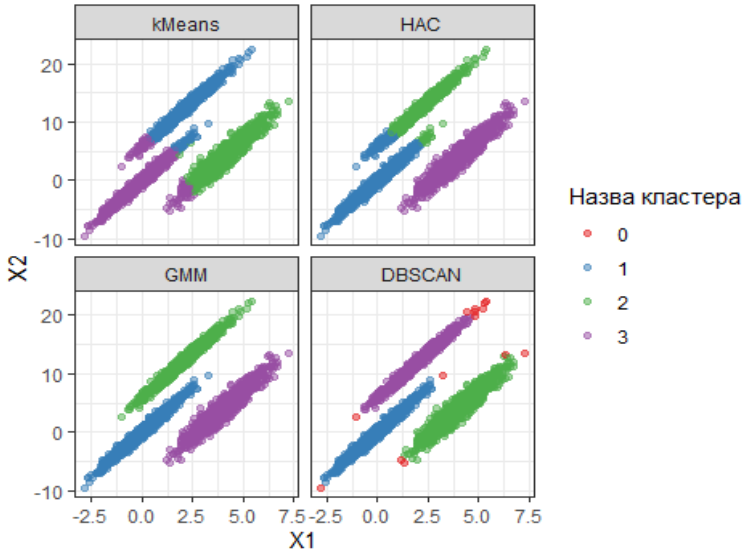

Рисунок 4 - Результати кластеризації набору даних Slash

Зрозуміло (рисунок 4), що лідерами у даному випадку є методи DBSCAN та GMM. У цьому випадку пальму першості віддаємо методу GMM, так як DBSCAN вказав на наявність аутлаєрів. K-Means та HAC (рисунок 4) не впоралися із кластеризацією у даному випадку.

Далі представлено результати (рисунок 5) кластеризації набору даних Eye чотирьома алгоритмами. Серед лідерів знову алгоритм DBSCAN та алгоритм HAC. K-Means та GMM здійснили неправильну кластеризацію.

Здається, що лідером є алгоритм DBSCAN. Подивимося на результати кластеризації. У даному випадку бачимо, що K-Means, GMM, HAC чудово впоралися із задачею класифікації, DBSCAN – не впорався.

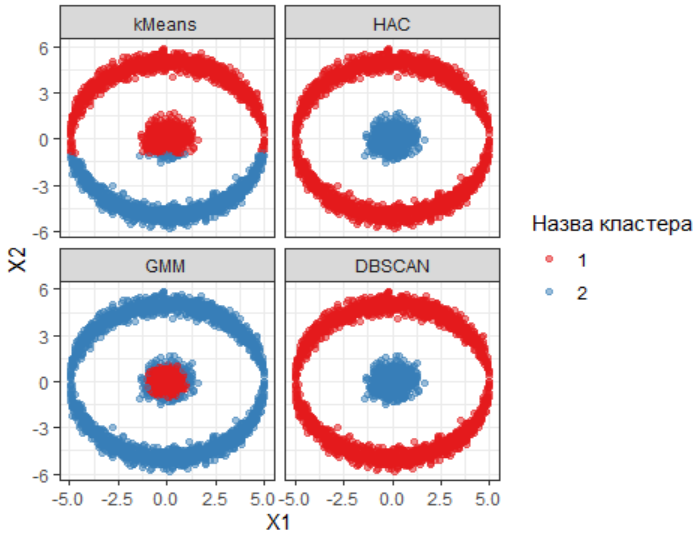

Рисунок 5 - Результати кластеризації набору даних Eye

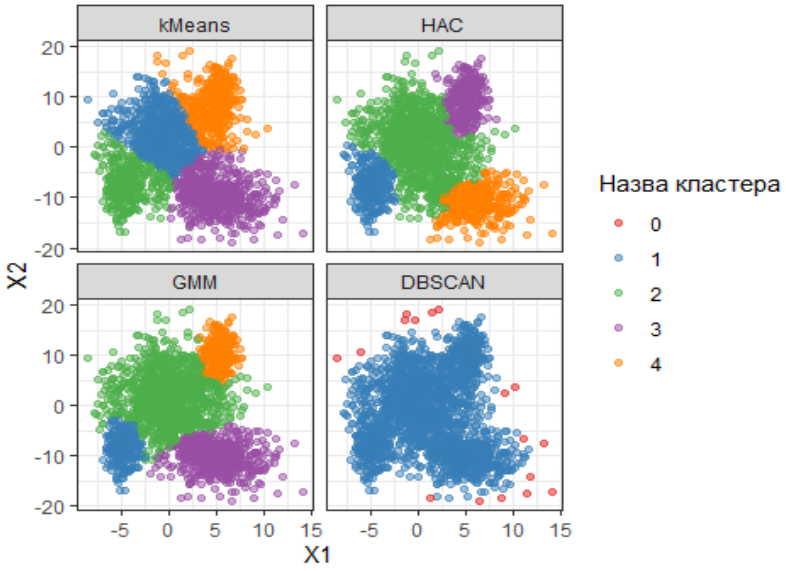

Рисунок 6 - Результати кластеризації набору даних Swarm

Для набору даних Swarm (рисунок 6) немає чітких меж між кластерами. Саме у цьому випадку важко використовувати DBSCAN, тому що просто неможливо вибрати параметри для розділення даних на певну кількість кластерів (у нашому випадку від 3 до 6). Наприклад, якщо значення ε зменшити, щоб визначити менший окіл, отримаємо більше 15 кластерів.

Отже, розглянемо результати кластеризації. Маємо двох лідерів – DBSCAN (не впорався з набором Swarm) та HAC (не впорався з набором Slash), які правильно кластеризували 4 із 5 наборів даних. На другому місці – алгоритм GMM (не впорався з наборами Eye та Sickle), який правильно кластеризував 3 із 5 наборів даних. Алгоритм K-Means правильно кластеризував лише один набір даних. K-Means – це інтуїтивно зрозумілий швидкий алгоритм, але він не в змозі обробляти випадки, коли кластери погано розділені або перекриваються, оскільки центр кластера визначається середнім значенням його точок.

Розглянемо тепер результати кластеризації наборів даних Card, Wine, Elect. На рисунках 8-10 показано кластеризовані набори даних, відповідно. Здійснено кластеризацію даних Card (рисунок 8). Як бачимо, метод DBSCAN кластеризує дані на основі їх щільності. Дані є щільними біля точки початку координат. Тому цей алгоритм зобразив це скупчення одним кластером. Червоні крапочки вважаються аутлайєрами даних. Зелений кластер містить невелику кількість

елементів. Інші три алгоритми кластеризації підсумовують, що дані Card варто розділити на 3 кластери. Проте метод поділу абсолютно відрізняються.

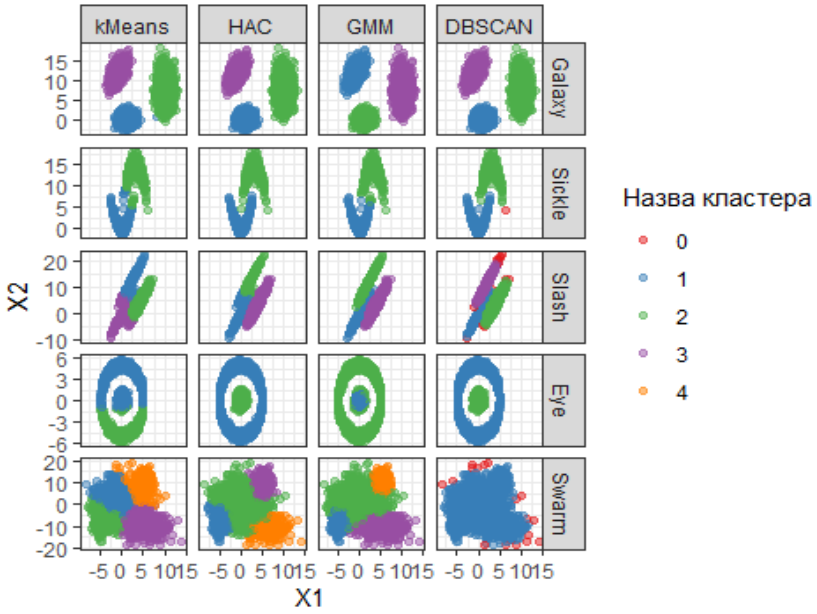

Рисунок 7 - Результати кластеризації усіх згенерованих наборів даних

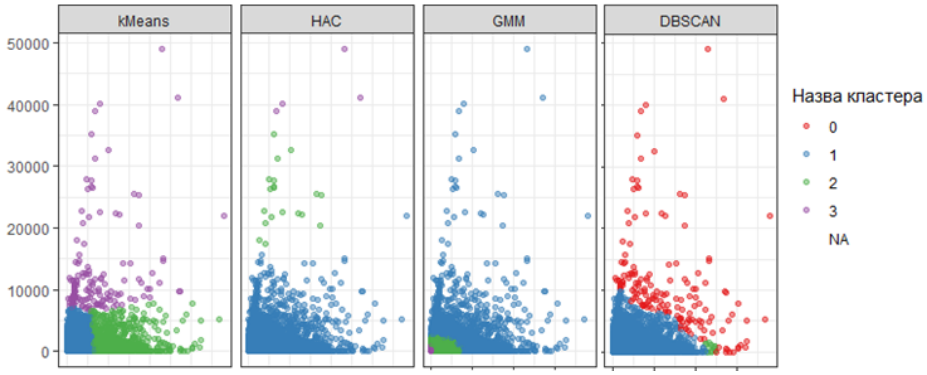

Рисунок 8 - Кластеризація набору даних Card

Показано кластеризацію даних Wine (рисунок 9). Як бачимо, алгоритми кластеризації kMeans, HAC, GMM здійснили схоже розбиття на 3 групи вхідних даних. Метод DBSCAN кластеризував дані на два кластери. Знову причиною є щільність точок даних зі зростанням Y координати.

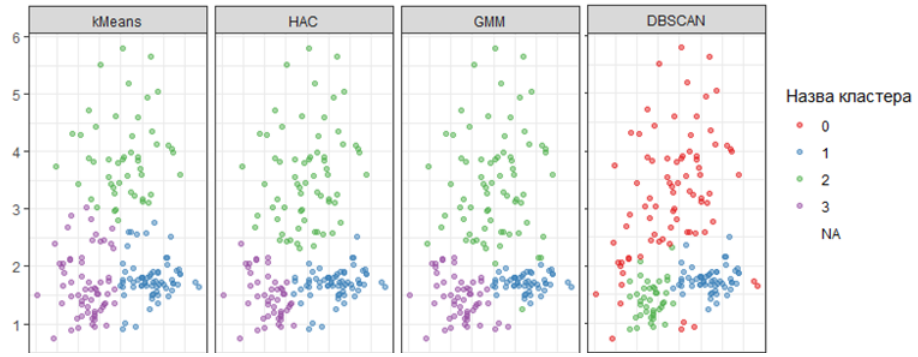

Рисунок 9 - Кластеризація набору даних Wine

Показано кластеризацію даних для набору Elect (рисунок 10). Тут знову результати кластеризації даних алгоритмами kMeans, HAC, GMM є схожими, а результати кластеризації за допомогою алгоритму DBSCAN відрізняються.

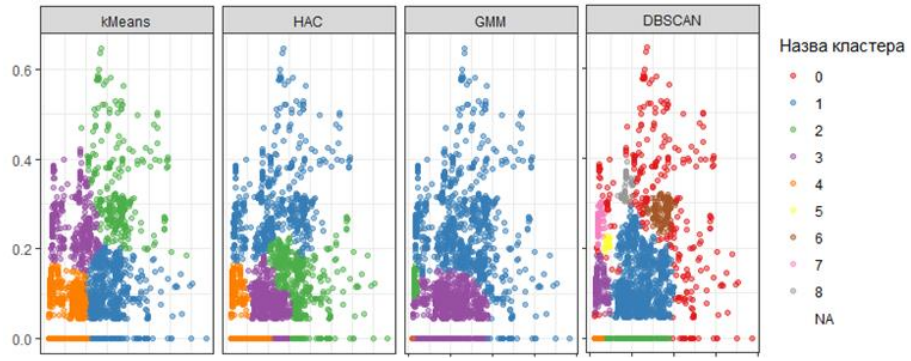

Рисунок 10 - Кластеризація набору даних Elect

3. Створення Shiny App

У середовищі R Programming за допомогою пакету RStudio створено Shiny App (інтерактивна веб-сторінка). Shiny сторінку розміщено на безкоштовному хмарному сервері shinyapps.io. Меню веб-додатку складається із вигляду вхідних даних, знаходження оптимальної кількості кластерів за допомогою чотирьох підходів та демонстрації або інтерактивної кластеризації даних на вибрану кількість кластерів. Веб-сторінка доступна за наступним посиланням: https://wc7rar-brainshturm-math0statistics0science.shinyapps.io/Nick/.

Використовуючи веб-додатки Shiny, кластеризація даних стає набагато зрозумілішою задачею. Використовуємо набір даних про кредитні картки клієнтів банку Card.

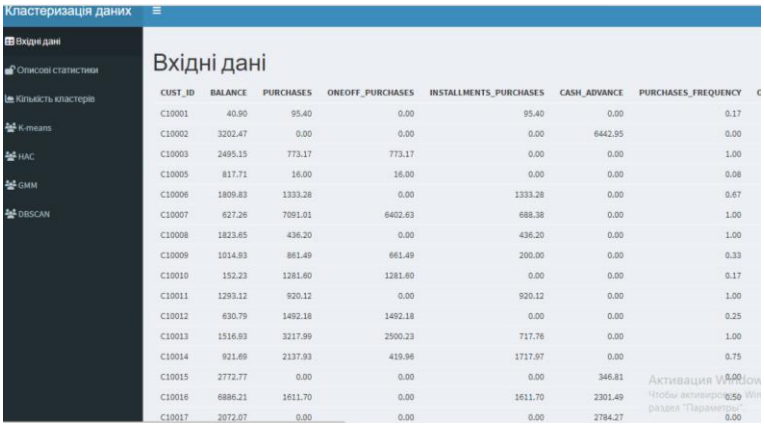

Рисунок 11 - Вигляд вхідних даних

Для кластеризації використовуємо стовпці 2 та 3. Наступні рисунки 12-13 показують результат роботи методів визначення оптимальної кількості кластерів – NbClust, Метод ліктя, Метод силуету, Gap Statistic Method. Кожен із цих методів показує різну оптимальну кількість кластерів. Це ще один доказ того, що немає універсальних підходів та методів. Задача оптимальної кластеризації даних залишається відкритою.

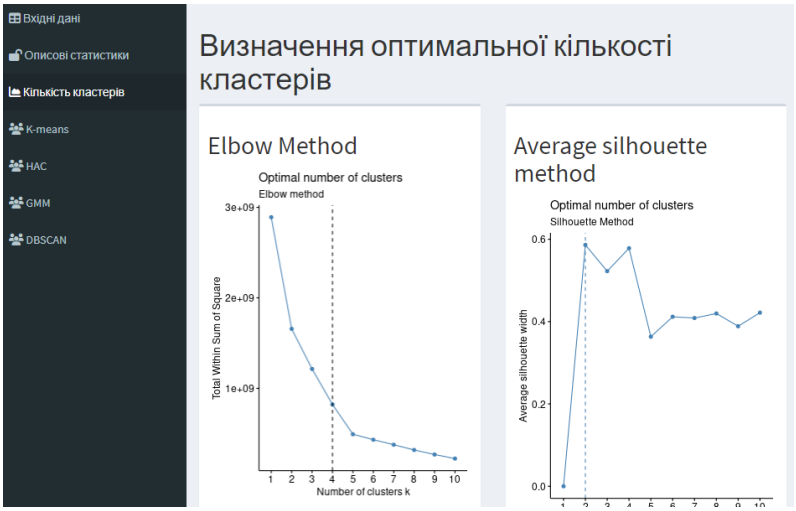

Рисунок 12 - Визначення оптимальної кількості кластерів

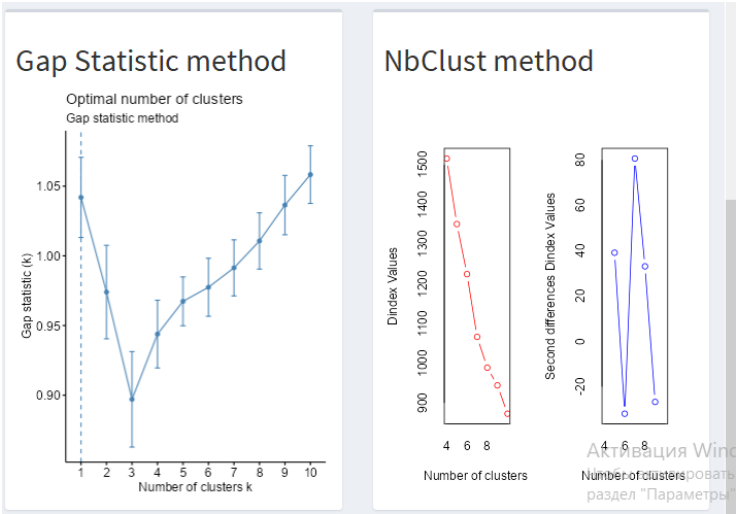

Рисунок 13 - Визначення оптимальної кількості кластерів

Наступні рисунки показують статистику кластеризації та сам процес кластеризації за допомогою чотирьох підходів.

Рисунок 14 - Кластеризація даних методом DBSCAN на 6 кластерів

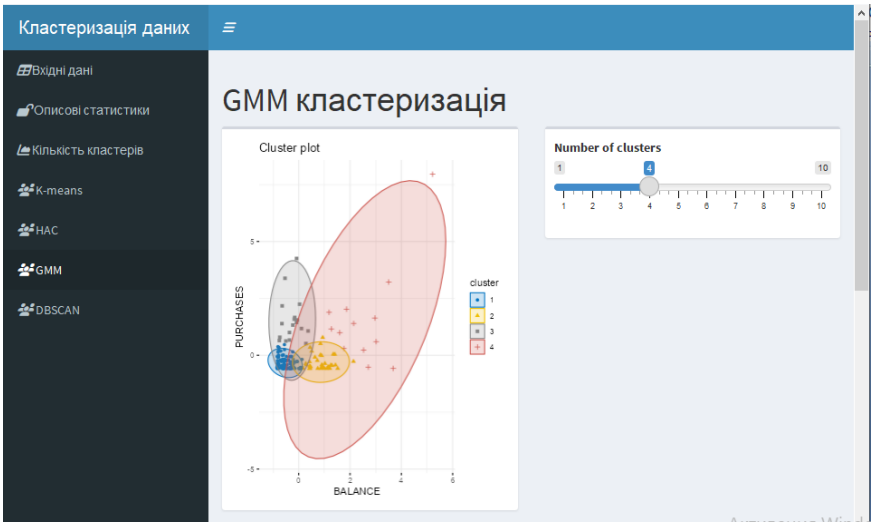

Рисунок 15 - Кластеризація даних методом GMM на 4 кластери

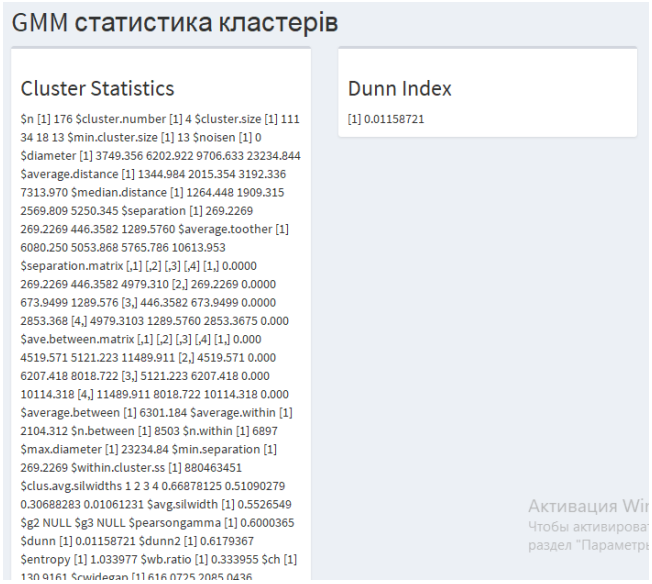

Рисунок 16 - Статистика GMM кластеризації

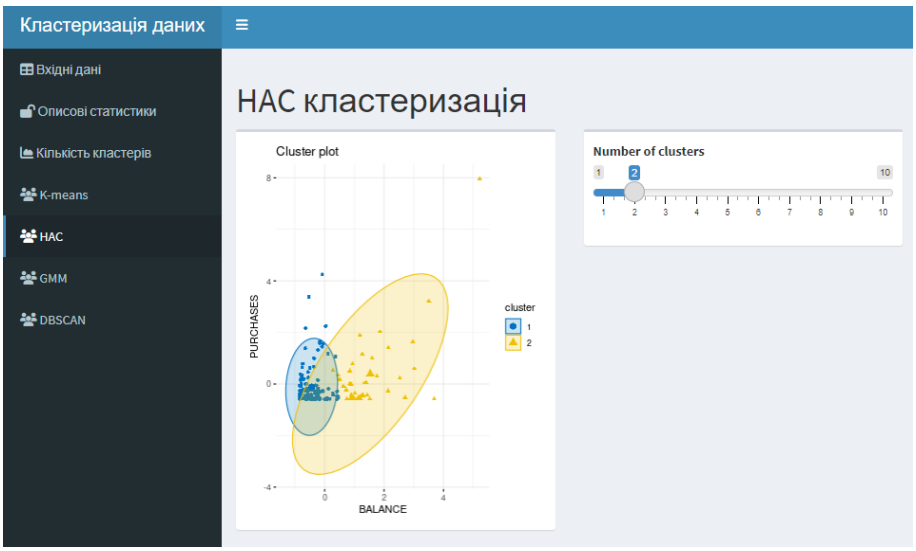

Рисунок 17 - Кластеризація даних методом HAC на 2 кластери

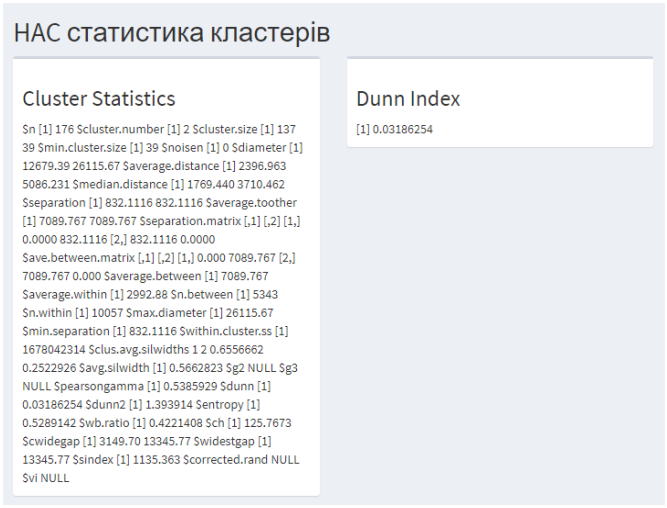

Рисунок 18 - Статистика HAC кластеризації

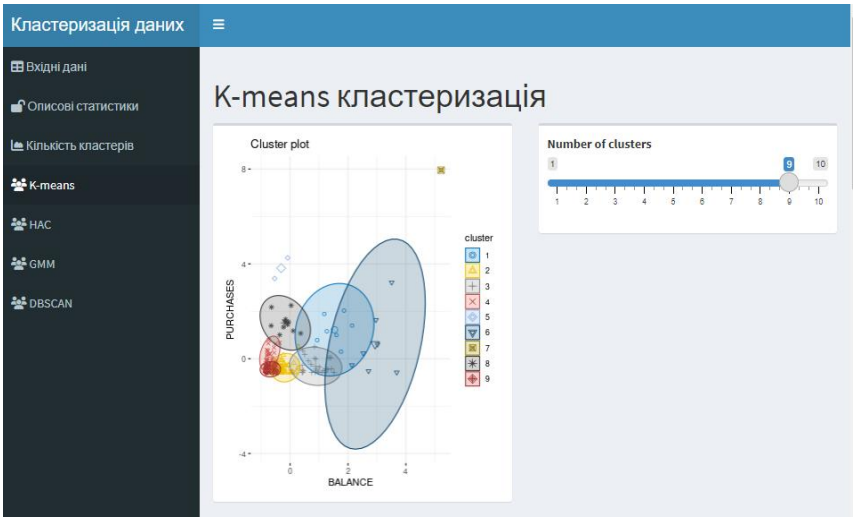

Рисунок 19 - Кластеризація даних методом k-Means на 9 кластерів

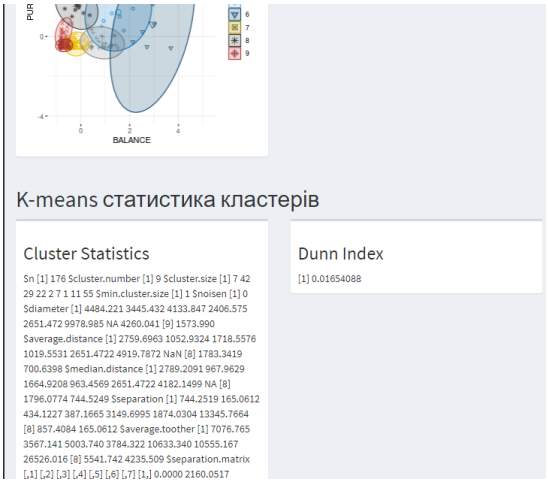

Рисунок 20 - Статистика k-Means кластеризації

Індекс Dunn є мірою валідності кластерів, яка використовується для оцінки якості рішень кластеризації. Чим меншим є індекс Dunn, тим точніше здійснено кластеризацію даних.

Висновки.

Кластеризація є однією з найбільш фундаментальних проблем інтелектуального аналізу даних через її численні застосування для сегментації клієнтів, цільового маркетингу та узагальнення даних.

У даній статті проведено порівняння алгоритмів кластеризації даних. Набори даних були попередньо згенерованими за допомогою нормального та рівномірного розподілів. Для кожного з отриманих наборів даних застосовано перелічені алгоритми кластеризації та визначено кращий алгоритм за результатом усіх кластеризацій. Показано, що серед розглянутих алгоритмів найкраще з задачею кластеризації впоралися алгоритми DBSCAN та HAC. За допомогою трьох наборів даних, отриманих з платформи Kaggle, здійснено кластеризацію реальних даних. Демонстрацію роботи алгоритмів кластеризації даних здійснено з використанням пакету Shiny у створеному веб-додатку. Отриманий веб-додаток розміщено на хмарному сервері shinyapps.io.

Література:

[1] . Simulations, Of & Zaidi, Habib & Labb, Claire & Morel, Christian. (1999). Improvement of the performance and accuracy of PET Monte Carlo simulations. Proc. SPIE. 3659. 10.1117/12.349537

[2] Doroshenko I.V., Knihnitska T.V., Deretorska T.I. Comparison of machine learning algorithms for predicting mortality from Covid-19 virus // Sworld Jornal Issue No11, Part 2 January 2022 – P. 72-77 (https://www.sworldjournal.com/index.php/swj/article/view/swj11-02-045).

Abstract. *The article compares the comparison of data clustering algorithms: K-Means, Hierarchical Agglomerative Clustering (HAC), Density-Based Spatial Clustering of Applications with Noise (DBSCAN), Expectation–Maximization clustering using Gaussian Mixture Models (GMM). The comparison is made thanks to pre-generated data sets that have different behavior: concentric circles (2 clusters), stripes (3), clouds (3), inseparable set (1), crescents (2). For each of the data sets, the listed methods are applied and the best clustering algorithm for a certain type of data is determined. Data clustering algorithms were applied to three sets of real data. An interactive web application for interactive data clustering using the mentioned algorithms has been created, which is deployed on the shinyapps.io cloud server.*

Keywords: *data clustering, cluster analysis, K-Means, Hierarchical Agglomerative Clustering (HAC), Density-Based Spatial Clustering of Applications with Noise (DBSCAN), Expectation–Maximization clustering using Gaussian Mixture Models (GMM).*

Статья отправлена: 19.01.2024 р.