`

СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Навчально-науковий інститут (факультет) інформаційних технологій та електроніки

Кафедра програмування та математики

Пояснювальна записка

До магістерської дипломної роботи

Магістр

(освітньо-кваліфікаційний рівень)

на тему «Кластерний аналіз текстових даних web-форуму на основі алгоритму c-means

Виконав: студент _2_ курсу, групи ІСТ-20дм спеціальності

126 «Інформаційні системи та технології»

(шифр і назва спеціальності)

Сітченко О.В.

(прізвище та ініціали)

Керівник Захожай О.І.

(прізвище та ініціали)

Рецензент Кряжич О.О.

(прізвище та ініціали)

Сєвєродонецьк – 2021 рік

`

СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Навчально-науковий інститут (факультет) інформаційних технологій та електроніки
Кафедра програмування та математики
Освітньо-кваліфікаційний рівень магістр
спеціальність 126 «Інформаційні системи та технології»
(шифр і назва спеціальності)

ЗАТВЕРДЖУЮ
Завідувач кафедри ПМ,
д.т.н., доцент
_____ д.т.н., доц. Лифар В.О.
(підпис)
«___»_____2021 р.

ЗАВДАННЯ
на магістерську дипломну роботу студенту

_____Сітченко Олександр Валерійович_____
(прізвище, ім'я, по батькові)

1. Тема роботи Кластерний аналіз текстових даних web-форуму на основі алгоритму c-means._____
керівник роботи доцент, к.т.н Захожай Олег Ігорович_____,
(вчене звання, науковий ступінь, прізвище, ім'я, по батькові)

затверджений наказом університету від «_30_»_11____2021 року №182/15.16

2. Строк подання студентом роботи 20 грудня 2021 р.
3. Вихідні дані до роботи Матеріали науково-дослідної практики, науково-методична література; дані інтернет-мережі; ._____
_____
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
   4.1 Вступ
   4.2 Аналітичний огляд питання (огляд публічних джерел інформації
   4.3 Основна частина, в якої висвітлити методи, які будуть використовуватися для реалізації проєкту та алгоритму c-means.
   4.4 Практична частина – огляд технологій, які використовуються під час реалізації проєкту.
   4.4 Висновки
   4.5 Перелік використаних джерел
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
_____

`

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання 8 листопада 2021 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1	Одержання завдання на виконання роботи	8.11.2021	
2	Укладання і погодження з керівником плану і етапів виконання роботи	10.11.2021	
3	Узагальнення даних літературних джерел	13.11.2021	
4	Аналіз шляхів виконання завдання. Вибір і погодження з керівником оптимального шляху виконання завдання	17.11.2021	
5	Аналіз технічних засобів та існуючих систем	25.11.2021	
6	Реалізація практичної частини завдання	01.12.2021	
7	Укладання, оформлення та погодження пояснювальної записки з керівником	08.12.2021	
8	Здача пояснювальної записки на кафедру	20.12.2021	
9	Підготовка доповіді та презентації	27.12.2021	

Студент _____ Сітченко О.В._____
(підпис) (прізвище та ініціали)

Керівник роботи _____ _Захожай О.І._____
(підпис) (прізвище та ініціали)

`

РЕФЕРАТ

Робота містить: 32 сторінки основного тексту, 4 сторінки додатків, 3 рисунка, 15 використаних посилань.

Метою магістерської дипломної роботи є вивчення методів кластерного аналізу в Data Mining та їх практичне застосування на прикладі аналізу текстових даних веб-форуму.

Був проведений детальний аналіз питання та методів його вирішення. Багато часу було приділено аналізам алгоритмів кластерного аналізу та їх використання в аналізі текстових даних.

В результаті виконаної роботи було розглянуто алгоритм кластерного аналізу c-means.

Результати даного проєкту можуть використовуватися в подальшій роботі в двох напрямках: побудові інформаційної системи автоматичної обробки текстових даних веб-форумів та в навчанні нейромережі без вчителя.

Система реалізована відповідно всім вимогам технічного завдання.

Зроблено детальний опис алгоритму веб-парсеру та алгоритму нечіткої кластеризації c-means.

`

ЗМІСТ

ВСТУП

Сучасний комп'ютерний термін Data Mining переводиться як «Добування даних». Нерідко разом з Data Mining зустрічаються терміни Knowledge Discovery («Виявлення знань») і Data Warehouse («сховище даних»). Виникнення зазначених термінів, які є невід'ємною частиною Data Mining, пов'язане з новим витком у розробці засобів та методів з обробки та зберігання даних. Отже, мета Data Mining полягає у виявленні прихованих правил і закономірностей у великих (Дуже великих) обсягах даних. Зазвичай Data Mining поділяють на задачі класифікації, моделювання та прогнозування.

Так трапилося, що людський розум не зовсім пристосований для обробки та сприйняття величезних масивів різнорідної інформації. В середньому людина, за деяким винятком, не здатна виявляти більше декількох взаємозв'язків у невеликих вибірках, що й казати про великі об’єми даних. Але і традиційна статистика, довгий час претендувала на роль основного інструмента аналізу даних, так само нерідко пасує при рішенні задач з реального життя. Статистика оперує усередненими характеристиками вибірки даних, які часто є фіктивними величинами (середньої платоспроможністю клієнта, коли в залежності від функції ризику або функції втрат вам необхідно вміти прогнозувати спроможність і наміри клієнта; середньої інтенсивністю сигналу, тоді як вам цікаві характерні особливості та передумови піків сигналу і т. д.).

Тому математичну статистику головним чином використовують для перевірки заздалегідь сформульованих гіпотез що дає більш кращий результат, тоді як визначення гіпотези іноді буває досить складною і трудомісткою задачею. Сучасні технології в Data Mining сконцентровані на переробці інформацію з метою автоматизації пошуку шаблонів (патернів) даних, які будуть характерні для будь-яких фрагментів в неоднорідних багатомірних даних.

Одним з методів класифікації в Data Mining є кластерний аналіз, мета якого полягає в знаходження груп схожих об’єктів у вибірці даних. Цей метод дозволяє

спростити подальший аналіз даних за рахунок того що отримані кластери набагато легше проаналізувати, ніж початкову вибірку об’єктів, або використовувати його для завдань навчання нейронних мереж без вчителя, надаючи вибірку необхідної інформації.

Об’єктом дослідження даного проєкту є процеси циркуляції інформації.

Предметом дослідження виступає модель та алгоритм інформаційної технології кластерного аналізу текстових даних.

Метою дослідження є вивчення методів кластерного аналізу в Data Mining та їх практичне застосування на прикладі аналізу текстових даних форуму.

РОЗДІЛ 1. АНАЛІТИЧНИЙ ОГЛЯД

1.1 КЛАСТЕРНИЙ АНАЛІЗ

Одним з етапів аналізу даних є кластерний аналіз. Він полягає в розбитті заданої вибірки об'єктів (ситуацій) на підмножини (кластери) так, щоб в кожному кластері містилися схожі об'єкти, а об'єкти різних кластерів відрізнялися між собою. В подальшому отриманні кластери використовуються або для подальшого аналізу інформації, через те що отримані кластери набагато легше проаналізувати, ніж початкову вибірку об’єктів, або для завдань навчання нейронних мереж без вчителя.[1]

Кластерний аналіз має на меті наступні завдання:

- Розробити типологію або класифікацію даних вибірки.
- Дослідити корисні концептуальні схемиз групування об'єктів.
- Створення гіпотез відносно даних на основі дослідження.
- Перевірити гіпотези або дослідження для визначення, чи дійсно кластери, отримані тим чи іншим способом, є в наявних даних.

Оскільки поняття «кластеру» не може бути точно визначено, то це є однією з причин чому існує так багато різних методів кластеризації. Але є і спільна риса — це об'єднання схожих об'єктів у групи. Однак, різні дослідники використовують різні моделі кластерів і для кожної з цих моделей можуть бути застосовані різні алгоритми. Поняття кластера, які отримуються у різних алгоритмах, різняться властивостями. Розуміння цих «кластерних моделей» є ключовим для розуміння відмінностей між різними алгоритмами. Типовими кластерними моделями [2]:

- Моделі зв'язності. Наприклад, ієрархічна кластеризація або таксономія будуються на основі відстані між вузлами.
- Центроїдні моделі. Наприклад, метод K-середніх (K-means) представляє кожен кластер єдиним усередненим вектором.

- Статистичні моделі. Кластери будуються ґрунтуючись на статистичних розподілах. Таких як багатовимірний нормальний розподіл з допомогою EM-алгоритму.
- Моделі засновані на щільності. Наприклад, в DBSCAN і в OPTICS кластери визначаються як зв'язані області відповідної щільності у просторі даних.
- Групові моделі. Деякі алгоритми не забезпечують вдосконалену модель для своїх результатів, а просто описують групування об'єктів.
- Графові моделі. Поняття кліки (така підмножина вершин, в якій кожна пара вершин з'єднана ребром) у графі слугує прототипом кластеру. Пом'якшення вимоги до повної зв'язності (тобто, частина ребер може бути відсутня) призводить до поняття відомого як квазі-кліка. Вони будуються алгоритмом HCS.
- Нейронні моделі. Найбільш відомою моделлю нейронної мережі з навчанням без учителя є нейронна мережа Кохонена. Ці моделі, як правило, можна охарактеризувати як подібні на одну або схожі з якимись з наведених вище моделей, включаючи моделі у підпросторах, коли нейронні мережі реалізують метод головних компонент або аналіз незалежних компонент.

В свою чергу кожну з цих моделей можливо поділити на дві групи:

1. Жорстку – об’єкт має чітке відношення тільки до одного визначеного кластеру.
2. М’яку (або нечітку) – об’єкт може належати до кожного з кластерів до певної міри.

Теорія нечітких множин була запропонована Zdzislaw'ом Pawlak'ом на початку 1980-х. В основі цієї теорії простий факт: здатність описати безліч об'єктів обмежуючись нашими можливостями в розрізненні окремих його представників. Зазвичай, різняться лише класи об'єктів, а не самі об'єкти. Деякі елементарні класи такого відношення нерозрізненості можуть бути несумісні, тобто включати об'єкти, що мають однаковий опис, але віднесені до різних категорій.[3]

Внаслідок описаної вище нерозрізненості неможливо точно описати безліч об'єктів у термінах елементарних множин нерозрізнених об'єктів. Для того щоб

вирішити зазначену проблему було введено поняття неточної множини, як пари двох множин - нижнього та верхнього наближення, побудованих з елементарних множин об'єктів. Ця ідея є ключовою для вирішення багатьох інших завдань, зокрема задач класифікації, оцінки залежності між ознаками і класифікацією об'єктів, визначення ступеня такої залежності, обчислення важливості ознак, скорочення кількості ознак і породження вирішальних правил за вихідними даними.

Для визначення відповідності об’єктів до кластерів використовуються ступінь схожості, тобто «відстань» між об’єктами. Існує велика кількість метрик за якими відбувається порівняння об’єктів. Основні з них:

- Євклідова відстань - Найбільш поширена функція відстані. Є відображенням геометричної відстані у багатовимірному просторі.
- Квадрат евклідової відстані - застосовується для надання більшої ваги віддаленим один від одного об'єктам.
- Відстань міських кварталів (манхеттенська відстань) - ця відстань є середньою різницею за координатами. З де більш ця міра відстані призводить до тих же результатів, що і вимірювання звичайної відстані Евкліда. Проте для цього методу вплив окремих великих різниць (викидів) знижується (бо вони не зводяться в квадрат).
- Відстань Чебишева - ця відстань використовується, коли треба визначити два об'єкти як «різні» за умови, що вони відрізняються за якоюсь однією координатою.
- Ступінна відстань - застосовується у разі, коли необхідно збільшити або зменшити вагу, що стосується розмірності, для якої відповідні об'єкти сильно відрізняються.

1.2 ІЄРАРХІЧНІ КЛАСТЕРІЗАЦІЯ

Ієрархічна кластеризація (англ. hierarchical cluster analysis, HCA) в Data Maning та статистиці — метод кластерного аналізу, який намагається побудувати ієрархію кластерів. Стратегії побудови ієрархічної кластеризації діляться на два типи[4]:

- агломератові (об'єднувальні). Це підхід «знизу-вгору». Спочатку кожна точка має власний кластер, а далі пари кластерів об'єднуються при підйомі по ієрархії.
- розділювальні. Це підхід «згори-вниз». Спочатку всі точки знаходяться у єдиному кластері, потім відбувається рекурсивне розбиття при русі вниз по ієрархії.

Отриману ієрархію типово зображають як дендрограму (Додаток А).

Єрархічні методи кластерного аналізу призначені для виявлення неоднорідностей, що існують у просторі змінних. Однак такі неоднорідності зовсім не зобов'язані в реальності існувати, тоді як будь-який метод кластерного аналізу дає результат завжди. Якщо реальних неоднорідностей в експериментальному матеріалі немає, метод виявить маленькі випадкові нерівномірності і на основі цих флуктуацій розділить об'єкти на кластери.

Щоб зрозуміти, чи вдалося виявити реально існуючу структуру об'єктів або метод закінчив працювати, тому що іншого виходу у нього не було, рекомендується слідувати чотирьом рекомендаціям.

- 1. Виконати кластерний аналіз з використанням різних способів вимірювання відстаней. Порівняти, наскільки збігаються отримані результати.
- 2. Виконати кластерний аналіз з використанням різних методів об'єднання кластерів. Порівняти результати.
- 3. Якщо дозволяє розмір матриці даних, розбити набір класифікуються об'єктів на дві рівні частини випадковим чином. Виконати кластерний аналіз окремо для кожної половини. Порівняти кластерні центроїди двох підвибірок.
- 4. Дуже важливий критерій якості кластеризації - змістовна інтерпретація результатів

Різноманітність методів даного типу пов'язано з двома обставинами: по-перше, з тим, яка міра вважається відстанню між точками простору і, по-друге, за якими правилами визначаються відстані між кластерами, коли останні включають в себе два або більше об'єктів.

Правила визначення відстані між кластерами. На першому кроці, коли кожен об'єкт являє собою окремий кластер, відстані між цими об'єктами-кластерами визначаються обраної метрикою - мірою відстані або мірою подібності об'єктів у просторі змінних. Потім, якісь об'єкти об'єднуються в один кластер і з'являються кластери, в яких два і більше об'єкта. Виникає проблема: що вважати відстанню між такими кластерами? Тут є різні можливості:

1. Методи індивідуальних зв'язків:
a. Міжгрупових зв'язків. Відстань між кластерами розраховується шляхом усереднення всіляких відстаней від об'єкта одного кластера до об'єкта іншого
b. Внутрішньогрупових зв'язків. Відстань між кластерами обчислюється як середня відстань між всіма можливими парами об'єктів, що належать обом кластерам, у тому числі об'єктів, розташованих усередині одного і того ж кластера.
c. Найближчого сусіда. Відстанню між двома кластерами вважається відстань між двома найближчими точками з різних кластерів.
d. Далекого сусіда. Відстанню між двома кластерами вважається відстань між двома самими далекими точками з різних кластерів
2. Методи зв'язків між центрами кластерів:
a. Центроїдного кластеризація. На першому кроці кожен об'єкт утворює окремий кластер, координати цього об'єкту є центром (центроїдом) кластера. При злитті двох кластерів центроид нового кластера розраховується як зважене по числу об'єктів в кожному кластері середнє значення центроїдів вихідних кластерів. Таким чином, більше значення надається крупним кластерам. У підсумку на кожному кроці алгоритму центроид кожного кластера розташовується в точці з середніми по всіх об'єктах кластера значеннями координат.

b. Медіанна кластеризація. При злитті двох кластерів центроид нового кластера розраховується шляхом усереднення координат центроїдів двох зливаються кластерів. Число об'єктів, що входять в ці кластери, не враховується, тобто дрібні і великі кластери вважаються однаково важливими і враховуються з однаковими вагами.
3. Дисперсійний метод - метод Варда. Метод мінімізує середню суму квадратів евклідових відстаней від об'єктів кластерів до своїх кластерних центрів. На кожному кроці об'єднують такі два кластери, які дають найменший приріст внутрікластерной дисперсії нового кластеру.

2.3 НЕІЄРАРХІЧНІ МЕТОДИ

Неієрархічні методи мають за основу вже задану кількість кластерів (k-means, PAM кластеризація) або використовують складні алгоритми знаходження їх кількості (CLOPE, карти Кохонена).[5]

Серед неієрархічних методів кластеризації особливої уваги заслуговують ітеративні методи. Вони працюють за наступним алгоритмом (рис. 2.3.1):

1. вихідні дані розбиваються на певну кількість кластерів та обчислюються центри тяжіння цих кластерів;
2. кожна точка даних поміщується в кластер з найближчим центром тяжіння;
3. обчислюються нові центри тяжіння кластерів; кластери не замінюються на нові доти, поки не будуть повністю переглянуті всі дані;
4. кроки 2 і 3 повторюються доти, поки не перестануть змінюватись кластери.

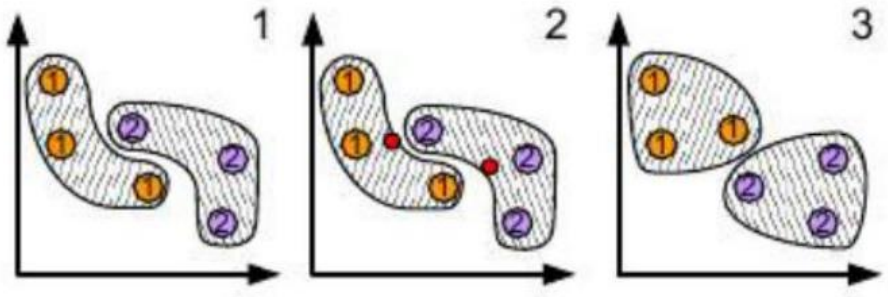


Рисунок 2.3.1 Загальна схема роботи ітеративних методів

На відміну від ієрархічних методів, які потребують обчислення і збереження матриці схожості між об’єктами розмірністю (N, N) , ітеративні методи працюють безпосередньо з первинними даними. Тому за їх допомогою можна обробляти доволі великі обсяги даних. Більш того, ітеративні методи виконують декілька переглядів даних і за рахунок цього компенсують наслідки невдалої вихідної розбивки даних. Ці методи породжують кластери одного рангу, які не є вкладеними, і тому не можуть бути частиною ієрархії. Більшість ітеративних методів не допускають перекриття кластерів. Зазвичай властивості ітеративних методів групування можуть бути описані за допомогою трьох основних чинників: вибір вихідної розбивки, тип ітерації і статистичного критерію. Ці чинники можуть різним чином поєднуватись, утворюючи алгоритми відбору даних при визначенні оптимальної розбивки. Різні комбінації ведуть до розробки методів, породжують різні результати при роботі з одними й тими ж даними.

Ітерації за наведеним принципом полягають у приєднанні об’єктів до кластера з найближчим центром тяжіння. Кількість фінальних кластерів фіксована і задається до початку кластеризації. Перерахунок центру тяжіння кластера може здійснюватись як після кожної зміни його складу, так і після того, як буде завершено перегляд усіх даних. На сьогодні існує багато варіантів даного методу, що відрізняються особливостями роботи.

До найбільш простих і ефективних алгоритмів кластеризації відноситься k-means запропонований Г. Боллом і Д. Холлом у 1965 р. Конструктивно алгоритм – це ітераційна процедура, що складається з наступних кроків:

1. Задається кількість кластерів k, яка повинна бути сформована з об’єктів вхідної вибірки.
2. Випадковим чином обирається k записів, які будуть слугувати початковими центрами кластерів. Початкові точки, з яких потім виростають кластери, часто називають "насінням". Кожний такий запис являє собою "ембріон" кластера, що складається тільки з одного елемента.
3. Для кожного запису вхідної вибірки визначається найближчий до неї центр кластера.
4. Проводиться обчислення центроїдів – центрів тяжіння кластерів. Це робиться шляхом визначення середнього для значень кожної ознаки всіх записів у кластері. Наприклад, якщо в кластер увійшли три записи з наборами ознак $(x1, y1), (x2, y2), (x3, y3)$, то координати його центроїда будуть розраховуватися в такий спосіб:

$$(x, y) = \left(\frac{x_1 x_2 x_3}{3}, \frac{y_1 y_2 y_3}{3}\right)$$

Потім старий центр кластера зміщається в його центроїд. Таким чином, центроїди стають новими центрами кластерів для наступної ітерації алгоритму.

Кроки 3 і 4 повторюються доти, поки виконання алгоритму не буде перервано або поки не буде виконана умова відповідно до певного критерію збіжності.

Зупинка алгоритму проводиться, коли границі кластерів і розташування центроїдів перестають змінюватися, тобто на кожній ітерації в кожному кластері залишається той самий набір записів.

Алгоритм k-means звичайно знаходить набір стабільних кластерів за кілька десятків ітерацій.

2.4 ПОРІВНЯННЯ ІЄРАРХІЧНИХ ТА НЕІЄРАРХІЧНИХ МЕТОДІВ КЛАСТЕРИЗАЦІЇ

Обираючи між ієрархічними та неієрархічними методами кластерного аналізу, потрібно врахувати їх особливості.

Неієрархічний метод виявляє більш вищу стійкість відносно шумів та викидів, некоректного обрання метрики, введення незначних змінних до набору, що бере участь в кластерному аналізі. Ціною, що доводиться платити за ці переваги методу, є слово "апріорі". Аналітик повинен заздалегідь визначити кількість кластерів для розбиття, кількість ітерацій для обробки або правило закінчення аналізу, та деякі інші параметри для кластерного аналізу. Це особливо складно фахівцям-початківцям.

Якщо немає припущень щодо кількості кластерів, рекомендують використати ієрархічні алгоритми. Однак якщо обсяг вибірки не дозволяє це зробити, можливий шлях – провести низку експериментів з різною кількістю кластерів, наприклад, почати розбивку сукупності даних з використання двох груп і порівнювати результати , поступово збільшуючи кількість груп. За рахунок такого порівняння результатів досягається доволі велика гнучкість кластеризації.

Ієрархічні методи, на відміну від неієрархічних, не визначають на початку кількість кластерів, а будують повне дерево з вкладених кластерів. Складності ієрархічних методів кластеризації:

- обмеження обсягу набору даних,
- вибір міри для визначення близькості об’єктів,
- негнучкість отриманих класифікацій кластерів.

Перевага даної групи методів в порівнянні з неієрархічними – є їх наочність та можливість одержати детальне подання структури даних.

Використовуючи ієрархічні методи, можливо доволі легко ідентифікувати викиди в наборі даних й, у результаті, підвищити якість даних. Ця процедура є

основою двокрокового алгоритму кластеризації. Такий набір даних надалі можна використати для проведення неієрархічної кластеризації.

Існує ще один аспект, про який слід згадати. Це питання кластерного аналізу всієї сукупності даних або вибірки з них. Названий аспект важливий для обох розглянутих груп методів, однак він критичний більше для ієрархічних методів. Вони не можуть працювати з великими об’ємами даних, тому використання деяких вибірок, тобто частин даних з загального обсягу даних, можливо б дозволило застосовувати ці методи.

Результати кластерного аналізу можливо не матимуть достатнього статистичного обґрунтування. З іншого боку, під час розв’язання задач з кластеризації є допустимою нестатистична інтерпретація отриманих результатів, а також велика розмаїтість варіантів отримання поняття кластера. Ця нестатистична інтерпретація дозволяє аналітикові одержати результати кластеризації, які задовольняють його, що у разі використання інших методів часто доволі складно.

Ієрархічні алгоритми забезпечують порівняно високу якість кластеризації при високій наочності. Більшість з них мають складність $O(f(n^2))$. Тому ієрархічні методи кластерного аналізу зазвичай використовуються при невеликих об’ємах наборів даних.

РОЗДІЛ 2. МЕТОДИ ТА АЛГОРИТМ КЛАСТЕРІЗАЦІЇ ТЕКСТОВИХ ДАНИХ

2.1 АЛГОРИТМ C-MEANS

Нечітка кластеризація (м'яка кластеризація або м'яким k-середній) - це форма кластеризації, в якій кожна точка даних може належати більш ніж одному кластеру.[6]

Найбільш популярним алгоритмом нечіткого кластерного аналізу даних є алгоритм c-середніх або c-means. Він є модифікацією методу k-середніх. Алгоритм кластеризації Fuzzy C-Means (FCM) був запропонований Дж. Данном в 1973 і доопрацьований Дж. Бездеком в 1981. На відміну від більшості існуючих алгоритмів кластеризації, даний алгоритм є нечітким – кожен з об'єктів не входить однозначно до будь-якого кластеру, а належить усім кластерам з різними ступенями належності. Це дає переваги як розбиття у випадках, коли кластери знаходяться близько один до одного, і велика кількість точок знаходиться на їх межах. Однак ціною такої нечіткості служать більші обчислювальні витрати, ніж у таких точних алгоритмів, як Hard C-Means і K-Means, при збереженні таких недоліків, як апріорне визначення числа кластерів і відсутність гарантії глобальної оптимальності результату. Кроки роботи алгоритму:

1. Вибрати початкове нечітке розбиття об'єктів n на k кластерів шляхом вибору матриці приналежності U розміру $n\ x\ k$.
2. Використовуючи матрицю U, знайти значення критерію нечіткої помилки: $E^2(X,U) = \sum_{i=1}^{N}\sum_{k=1}^{K} U_{ik} \left\| x_i^{(k)} - c_k \right\|^2$, де c_k – це «центр маси» нечіткого кластеру k: $c_k = \sum_{i=1}^{N} U_k\ x_i /$
3. Перегрупувати об'єкти з метою зменшення значення критерію нечіткої помилки.
4. Повертатися до пункту 2, поки зміни матриці U не стануть незначними.

2.2 МОВА ПРОГРАМУВАННЯ PYTHON

Python— це об'єктно-орієнтована інтерпретована мова програмування високого рівня зі строгою динамічною типізацією. Вона була розроблена в 1990 році Гвідо ван Россумом. Використання структурних даних високого рівня разом із використанням динамічної семантики та динамічного зв'язування дає можливість для швидкої розробки програм та як засобу для поєднування наявних компонентів. Мова прогрування Python сприяє модульності та повторному використанню коду, завдяки підтримці модулів та пакети модулів. Доступ до інтерпретатору Python та стандартних бібліотек, як у скомпільованій, так і у вихідній формі, є на всіх основних платформах. В Python підтримується декілька парадигм програмування: об'єктно-орієнтованість, процедурність, функціональність та аспектно-орієнтованість.[7]

Серед основних переваг цієї мови можна назвати такі[8]:

- чистий синтаксис (мірою визначення вкладеності коду використовуються відступи);
- переносність програм;
- стандартна версія дистрибутиву має велику кількість корисних модулів для застосування (наявний модуль для розробки графічного інтерфейсу);
- можливість використання в діалоговому режимі (корисне для експериментів та розв'язання простих задач);
- в базовій комплектації має вбудоване потужне середовище розробки, яке має назву IDLE та написане мовою програмування Python;
- зручна для розв'язання математичних питань (є засоби роботи з комплексними числами, можна проводити операції з цілими числами довільної величини, у діалоговому режимі можливе використання як потужного калькулятору);
- відкритий код (велика кількість модулів, створених самими користувачами).

Недоліки:

- Низька швидкість. Python не найшвидший серед мов. Мова є інтерпретованою, тому швидкість виконання програм може бути нижчою.

- Проблеми в інтерпретації.
- Великі відмінності в синтаксисі в порівнянні з іншими мовами програмування.
- Помилки виконання. Типи змінних визначаються автоматично (“динамічна типізація”), тому під час виконання можуть бути помилки, які розробник не передбачив.
- Мобільна технологія. Мова не застосовується для розробки мобільних програм.

Python велику кількість бібліотек, які застосовуються на різних ступенях аналізу даних:

1. Пошук даних. Програмісти використовують Scrapy та BeautifulSoup для пошуку даних за допомогою Python. За допомогою Scrapy можна створювати програми, які збирають структуровані дані в мережі. Також його можна використовувати для збирання даних з API. BeautifulSoup застосовується там, де отримати дані з API не виходить; він збирає дані та розставляє їх у певному форматі.
2. Обробка та моделювання даних. На цьому етапі часто використовуються бібліотеки NumPy і Pandas. NumPy (Numerical Python) використовується для сортування великих наборів даних. Він спрощує математичні операції та їхню векторизацію на масивах. Pandas пропонує дві структури даних: Series (список елементів) та Data Frames (таблиця з декількома колонками). Ця бібліотека конвертує дані в Data Frame, дозволяючи видаляти та додавати нові колонки, а також виконувати різні операції.
3. Візуалізація даних. Matplotlib та Seaborn широко використовуються для візуалізації даних. Вони допомагають конвертувати величезні списки чисел у зручні графіки, гістограми, діаграми, теплові карти тощо.

Крім перелічених бібліотек існують і інші більш спеціалізовані бібліотеки.

2.3 БІБЛІОТЕКА BEATIFULSOUP

Beautiful soup, це гарний інструмент для веб-парсингу за допомогою Python через його основні функції. Він може допомогти програмісту швидко отримати дані з певної веб-сторінки. Ця бібліотека допоможе отримати дані з файлів HTML і XML. Але проблема з Beautiful Soup в тому, що вона не може виконати всю роботу самостійно. ця бібліотека потребує певних модулів для роботи.[9]

Залежності Beautiful soup:

- Необхідна бібліотека для надсилання запиту на веб-сайт, тому що сам Beautiful soup не може зробити запит на конкретний сервер. Для подолання цієї проблеми потрібна допомога однієї з найпопулярніших бібліотек Requests або urlib2. Ці бібліотеки допоможуть нам зробити запит на сервер.
- Після завантаження даних HTML або XML на наш комп'ютер Beautiful Soup потрібен зовнішній аналізатор для аналізу завантажених даних. Найбільш відомі парсери – це XML-parser lxml, HTML-parser lxml, HTML5lib, html.parser.

Переваги:

- Бібліотека легка у навчанні та її легко освоїти.
- Вона має хорошу всеосяжну документацію, яка допомагає в освоєнні бібліотеки.
- Вона має гарну підтримку спільноти, яка дозволяє швидко вирішувати проблеми під час роботи з цією бібліотекою.

Веб-парсинг — перетворення у структуровані дані інформації з веб-сторінок, які призначені для перегляду людиною за допомогою браузера.

Веб-парсинг охоплює завантаження та вилучення. Спочатку завантажується сторінка, після цього можна здійснювати добувати потрібну інформацію. Зміст сторінки може бути проаналізовано, переформатовано, його дані скопійовані в електронну таблицю тощо. Веб-парсери, як правило, беруть якусь інформацію зі сторінки для використати її в інших цілях. На приклад, це може бути пошук і копіювання імен та телефонних номерів або компаній та їх URL-адрес до списку (контактне сканування).[10]

Веб-сторінки побудовані за допомогою текстових мов розмітки (HTML та XHTML) і часто містять велику кількість корисних даних у текстовій формі. Однак більшість веб-сторінок призначені для кінцевих користувачів, а не для зручності автоматичного використання. Через це були створені набори інструментів, які «збирають» веб-вміст. Веб-парсери — це прикладний програмний інтерфейс для вилучення даних з веб-сайту.

2.4. МОВА ПРОГРАМУВАННЯ R

R — мова програмування і програмне середовище для статистичних обчислень, аналізу та зображення даних в графічному вигляді. Розробка R відбувалась під істотним впливом двох наявних мов програмування: мови програмування S з семантикою успадкованою від Scheme. R названа за першою літерою імен її засновників Роса Іхаки та Роберта Джентлмена (працівників Оклендського Університету в Новій Зеландії. Незважаючи на деякі принципові відмінності, більшість програм, написаних мовою програмування S запускаються в середовищі R.[11]

R має значні можливості для здійснення статистичних аналізів, включаючи лінійну і нелінійну регресію, класичні статистичні тести, аналіз часових рядів (серій), кластерний аналіз і багато іншого. R легко розбудовується завдяки використанню додаткових функцій і пакетів доступних на сайті Comprehensive R Archive Network (CRAN). Більша частина стандартних функцій R, написана мовою R, однак існує можливість підключати код написаний C, C++, або Фортраном. Також за допомогою програмного коду на C або Java можна безпосередньо маніпулювати R об'єктами.

Функції, які можливо виконати за допомогою мови програмування R:

- Обробити, очистити та перетворити дані для дослідження.
- Провести статистичні випробування.
- Виконати розвідувальний аналіз.
- Працювати з таблицями різних форматів.

- Намалювати інтерактивний графік.
- Створити інтерактивну програму.
- Аналізувати регресійні моделі.

Переваги мови програмування R:

- Необмежений набір функцій для аналізу даних завдяки підключенню бібліотек.
- Можливість роботи з великими таблицями та базами даних, які складно обробити за допомогою інших засобів.
- Просунуті налаштування інтерфейсу.
- Повністю безкоштовна екосистема – компоненти розповсюджуються безкоштовно під ліцензією GNU.
- Доступний для більшості операційних систем: Windows, MacOS, FreeBSD, Solaris, різних версій Unix і Linux.
- Багаті можливості візуалізації: є можливим створювати програми, будувати графіки різних типів, зокрема інтерактивні, і навіть редагувати їх елементи. Наприклад, графік на основі відомого датасета iris, що показує щільність пелюсток та чашолистків залежно від виду ірисів(Додаток Б).
- Обширна та зрозуміла документація.

Головним недоліком цієї мови програмування є вузька сфера застосування: він є ідеальним для аналізу даних, але для розробки програм не підходить.

Реалізації алгоритму c-means буде відбуватися за допомогою пакету R «e1071», який вміщує в собі різні методи кластерного аналізу даних.[12]

РОЗДІЛ 3. РЕАЛІЗАЦІЯ КЛАСТЕРНОГО АНАЛІЗУ ТЕКСТОВИХ ДАНИХ WEB-ФОРУМУ

3.1 WEB-ПАРСЕР

Форум — це інструмент для спілкування на сайті. Щоб отримати можливість використовувати його з метою обміну повідомленнями, достатньо за допомогою будь-якого браузера завантажити потрібну сторінку сайту або просто заповнити на ньому відповідну форму. Повідомлення, якими обмінюються учасники форуму, деякою мірою схожі на поштові — в кожного з них є автор, тема і вміст. Зберігаються вони на одному сервері і ніколи не розповсюджуються.

Структура форуму. Ви можете надіслати до форуму власне повідомлення або дати відповідь на повідомлення, що вже існують. Всі повідомлення у форумі об’єднуються в потоки. Коли користувач відповідає на форумі на будь-яке повідомлення, його відповідь прив’язується до вихідного повідомлення. Послідовність таких відповідей (відповідей на відповіді) створює потік. Потік зображується у вигляді дерева, яке розгалужується в напрямку зліва направо. Корінь дерева — не повідомлення, що починає потік. З нього «виростають» відповіді, з яких, в свою чергу, «виростають» відповіді на відповіді і тому подібне. «Господар» форуму визначає правила поведінки у ньому і в разі необхідності вилучає повідомлення, які не відповідають тематиці обговорення (модерує форум).

Кожен форум є сховищем великої кількості різноманітної інформації по різноманітнішим темам: розваги, політика, спілкування, технічні підтримка, фахові середовища(лікарі, програмісти, вчителя), тощо. На даний час, існують форуми кількість постів на яких перевищує трильйон.

Для отримання необхідних для кластерного аналізу текстових даних було обрано форум 4PDA.to. Основною темою цього форуму є мобільні пристрої. За своєю будовою даний форум є доволі простим в плані веб-парсингу.

Веб-парсер було побудовано за допомогою мови програмування Python з використанням бібліотек Beautiful Soup, requests, csv. Алгоритм роботи веб-парсеру (Додаток В):

1. Підключення необхідних для роботи бібліотек (Beautiful Soup, requests, csv).
2. Оголошення перемінної dataPost, в якій буде зберігатися отриманні дані.
3. Оголошення циклу для переходів по сторінкам (для переходу на іншу сторінку необхідно змінити числове значення відображення групи постів)
4. Отримання за допомогою методу бібліотеки requests HTML сторінки.
5. Передача HTML сторінки до парсеру LXML з бібліотеки Beautiful Soup.
6. Оголошення циклу для обходу HTML сторінки з записом постів до перемінної dataPost.
7. Повторення пункт 6 до закінчення циклу.
8. Повернення до пункту 3 до закінчення постів в темі обговорення, яка обходиться за допомогою веб-парсеру.
9. Запис отриманих даних з перемінної dataPost до csv файлу.

Отриманий веб-парсер можна використовувати для обходу і інших тем для обговорення на даному форумі. Для цього необхідно лише: адаптувати цикл для обходу сторінок, замінити url адрес на адрес необхідної теми обговорення, змінити назву csv файлу.

3.2 РЕАЛІЗАЦІЯ АЛГОРИТМУ C-MEANS

Кластерний аналіз текстових даних форуму має деякі особливості:

- Через велику кількість постів, які необхідно обробити, ієрархічні кластерні алгоритми матимуть низьку ефективність.
- На відміну від більш розширеного аналізу великих текстів (наприклад email-листувань) серед яких можна виділити загальну тему відносно якої є можливим проведення жорсткої кластеризації (k-means), пости форуму за розміром можуть дуже сильно розрізнятися та не завжди можна в них виділити якусь одну тему.

Кластерний аналіз було вирішено проводити методом нечіткої кластеризації, а саме алгоритму c-means. Алгоритм реалізований за допомогою пакету «e1071» в мові програмування R. Приклад застосування даного пакету зображена в додатку Г.[13]

Алгоритм кластерного аналізу текстових даних форуму:

1. Отримання підготовлених даних до системи (відкриття csv файлу, отриманого за допомогою веб-парсеру, та запис його до таблиці)
2. Виклик функції cmeans() з передаванням необхідних для роботи атрибутів (таблиця постів, кількість необхідних кластерів, ступінь схожості текстів для входження в кластер). Данна функція проводить аналіз таблиці за допомогою алгоритму c-means:

a. Отримання атрибутів необхідних для аналізу.
b. Обирається початкове нечітке розбиття об'єктів n на k кластерів шляхом обрання матриці приналежності U розміру $n\ x\ k$.
c. Використовуючи матрицю U, знаходження значення критерію нечіткої помилки: $E^2(X,U) = \sum_{i=1}^{N}\sum_{k=1}^{K} U_{ik} \left\| x_i^{(k)} - c_k \right\|^2$
d. Перегрупування об'єктів з метою зменшення значення критерію нечіткої помилки.
e. Повернення до пункту c, поки зміни матриці U не стануть незначними.
f. Повернення кластерізованих даних.

3. Робота з векторами кластерів (виведення необхідних даних, графічне зображення кластерів, і інші операції)
4. В разі отримання задовільного результату запис отриманих векторів кластерів до окремих таблиць, якщо ні – повернення до пункту 2 з зміною атрибутів(кількість кластерів та ступені схожості)
5. Створення відповідних файлів (необхідного для подальшої роботи формату) в яких зберігаються отримані кластери.

3.3 UNSUPERVISED LEARNING

Машинне навчання (МО, Machine Learning, ML)[14] – це великий підрозділ в розробці штучного інтелекту, який вивчає методи побудови алгоритмів, здатних до навчання; це «розділ II, який досліджує методи, що дозволяють комп'ютерам покращувати свої характеристики на основі отриманого досвіду».

Першу програма на основі алгоритму, здатного до самонавчання, було розроблено Артуром Самуелем (Arthur Samuel) ще в 1952 році. ЇЇ призначенням була гра в шашки. Самуель також вперше визначив термін «машинне навчання», що позначав область досліджень з розробки машин, які заздалегідь не є запрограмованими. Набагато пізніше Т. М. Мітчелл дав точніше визначення цьому термінові[15]: «Кажуть, що комп'ютерна програма навчається на основі досвіду E стосовно деякого класу задач T і заходи якості P, якщо якість вирішення завдань з T, виміряний на основі P, поліпшується з набуттям досвіду E». Зараз йде розробка різноманітних систем машинного навчання, які призначенні для використання в таких технологіях, як Інтернет Речі, Промислові Інтернет Речі, в концепції «розумне» місто, при створенні безпілотників, тощо.

Навчання без вчителя (англ. Unsupervised learning) — один зі способів машинного навчання, для вирішення якого випробовувана система, без втручання з боку експериментатора, спонтанно опановує навички виконування поставлених завданнь. З точки зору кібернетики, є одним з видів кібернетичного експерименту. Як правило, це використовується тільки для задач, з вже відомим опис множини об'єктів (навчальною вибіркою), серед яких і необхідно виявити наявні в середин взаємозв'язки, залежності, закономірності, які існують між об'єктами.

Навчання без вчителя часто протиставляється навчанню з учителем, коли для кожного об'єкта, що навчається, примусово задається «правильна відповідь», і потрібно знайти залежність між стимулами та реакціями системи.

Для побудови теорії та відходу від кібернетичного експерименту у різноманітних теоріях експеримент з навчанням без вчителя намагаються формалізувати математично. Існує багато різних підвидів постановки та визначення даної формалізації. Одна з яких відображена у теорії розпізнавання образів.

Такий відхід від експерименту та побудова теорії пов'язані з різними поглядами спеціалістів. Відмінності, зокрема, проявляються при відповіді на питання: «Чи можливі єдині принципи адекватного описання образів різноманітної природи, або ж таке описане кожен раз є задача для спеціалістів конкретних знань?».

В першому випадку постановка повинна бути направлена на виявлення загальних принципів використання апріорної інформації при складанні адекватного опису образів. Важливо, що тут апріорні відомості про образи різноманітної природи різні, а принцип їх обліку один й той самий. У другому випадку проблема отримання опису виноситься за межі загальної постановки, і теорія навчання машин розпізнаванню образів з точки зору статистичної теорії навчання розпізнаванню образів може бути зведена до проблеми мінімізації середнього ризику в спеціальному класі вирішальних правил.

В теорії розпізнавання образів розрізняють в основному три підходи до даної проблеми:

- Евристичні методи;
- Математичні методи;
- Лінгвістичні (синтаксичні) методи.

Експеримент навчання без вчителя при розв'язуванні задачі розпізнавання образів можна сформулювати як задачу кластерного аналізу. Вибірка об'єктів розбивається на підмножини, що не перетинаються (вони називаються кластерами), це відбувається для того, щоб кожен кластер складався зі схожих між собою об'єктів, а об'єкти в кожному з різних кластерів суттєво відрізнялись. Початкова інформація представляється у вигляді матриці відстаней.

Таким чином отриманні за допомогою кластерного аналізу текстових даних форуму кластери в подальшому можна використати для навчання нейронної мережі на основі вже існуючого кластеру даних для отримання спеціалізованої нейронної мережі з аналізу даних або створення текстів по необхідним темам.

ВИСНОВКИ

У даній роботі було проведено дослідження алгоритмів кластеризації та їх використання для проведення кластерного аналізу текстових даних форуму.

Було сформовано підхід з отримання інформації для аналізу та подальшого кластерного аналізу з використанням технології web scraping, реалізація була виконана мовою програмування Python.

Розглянуто реалізацію нечіткого алгоритму кластерного аналізу на основі алгоритму c-means, реалізація мовою програмування R.

Також було розглянуто один із шляхів подальшого використання отриманих даних – навчання нейромережі без вчителя.

Як подальший розвиток теми можна розрахувати інформаційну систему автоматичної обробки текстових даних форумі з використанням вже розглянутого алгоритму c-means.

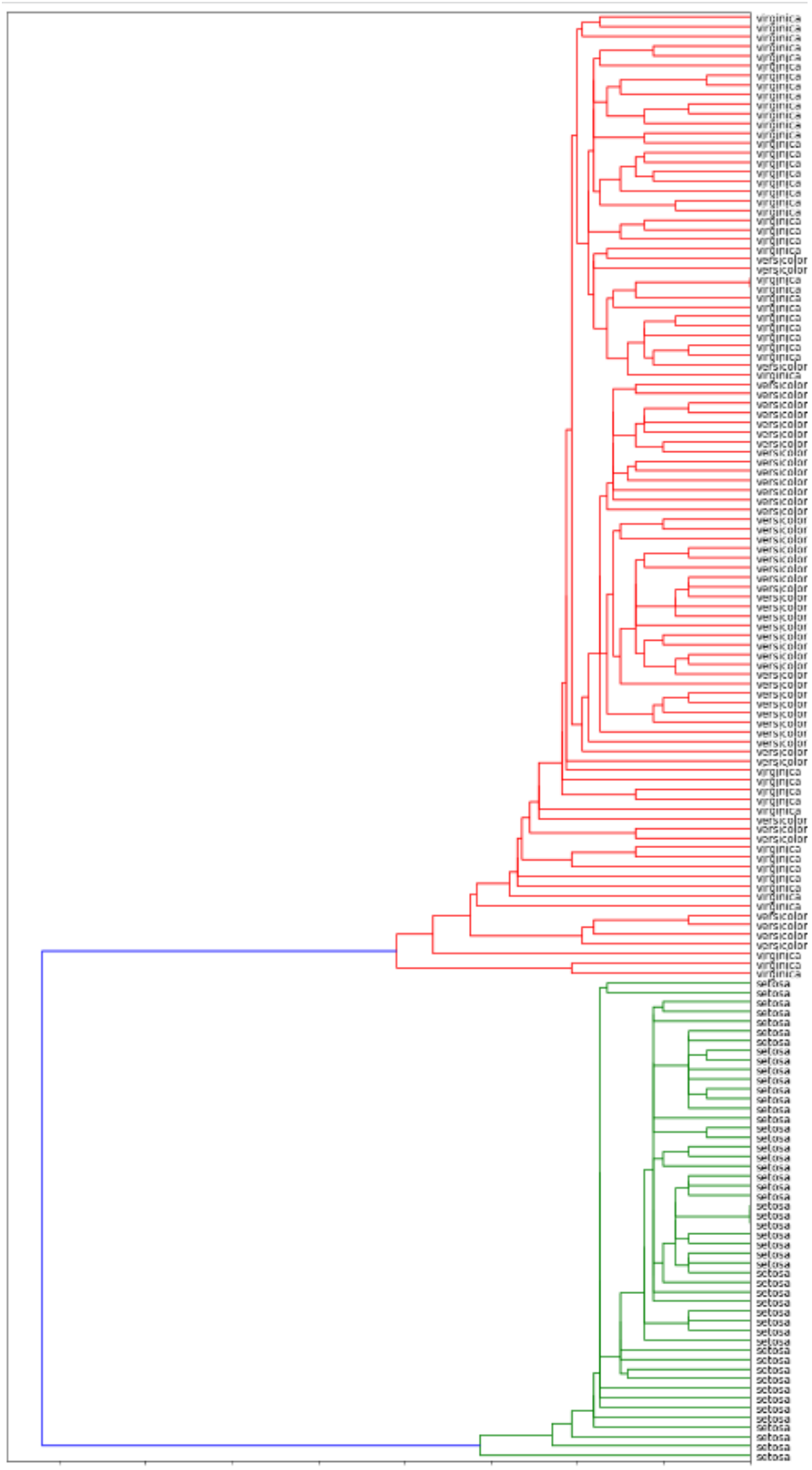
ПЕРЕЛІК ПОСИЛАНЬ

1. Data Mining: Practical Machine Learning Tools and Techniques [Електронний ресурс] Доступно: https://www.cs.waikato.ac.nz/~ml/weka/book.html
2. A. K. Jain. Data clustering: a review / A. K. Jain, M. N. Murty, P. J. Flynn.– ACM Comput. Surv.–1999. – №31. - 60 с.
3. Моделі та методи прийняття рішень : навч. посіб. для студ. вищ. навч. закл. / О.Ф. Волошин, С.О. Мащенко. – 2-ге вид., перероб. та допов. – К. :Видавничо-поліграфічний центр «Київський університет», 2010. – 336 с.
4. Методи кластеризації [Електронний ресурс] Доступно: http://pzs.dstu.dp.ua/DataMining/cluster/index.html
5. Charu C. Aggarwal Data Mining. The Textbook. / C.A.Charu; — Springer, 2015 .— 746 p.
6. Pattern Recognition and Machine Intelligence / Sergei O. Kuznetsov Deba P. Mandal Malay K. Kundu Sankar K. Pal (Eds.); 4th International Conference, PReMI 2011 Moscow, Russia, June 27 – July 1, 2011 Proceedings .— Springer, 2011 .— 495 p.
7. Python Documentation [Електронний ресурс] Доступно: https://www.python.org/doc/
8. Основи програмування. Python. Частина 1[Електронний ресурс]: підручник для студ. спеціальності 122 «Комп’ютерні науки», спеціалізаії «Інформаційні технології в біології та медицині» / А.В. Яковенко; КПІ ім. Ігоря Сікорського. – Електронні текстові данні (1 файл: 1,59 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2018. – 195 с.
9. BeautifulSoup Documentation [Електронний ресурс] Доступно: https://www.crummy.com/software/BeautifulSoup/bs4/doc.ru/bs4ru.html
10. Boeing, G.; Waddell, P. New Insights into Rental Housing Markets across the United States: Web Scraping and Analyzing Craigslist Rental Listings // Journal of Planning Education and Research. — 2016.
11. R. Книжка рецептів: Перевірені рецепти для статистики, аналізу та візуалізації даних/ Дж. Д. Лонг, Пол Титор. переклад с англ. Д. О. Белікова. – М.: ДМК Пресс, 2020. – 510 с.: іл.

12. Advanced Clustering [Електронний ресурс] Доступно: https://www.datanovia.com/en/courses/advanced-clustering/
13. R Package «e1071» Documentation [Електронний ресурс] Доступно: https://cran.r-project.org/web/packages/e1071/index.html
14. Machine Learning - Машинне навчання. [Електронний ресурс]. Доступно: https://www.it.ua/knowledge-base/technology-innovation/machine-learning
15. Mitchell, T. (1997). Machine Learning. McGraw Hill. p. 2. ISBN 978-0-07-042807-2

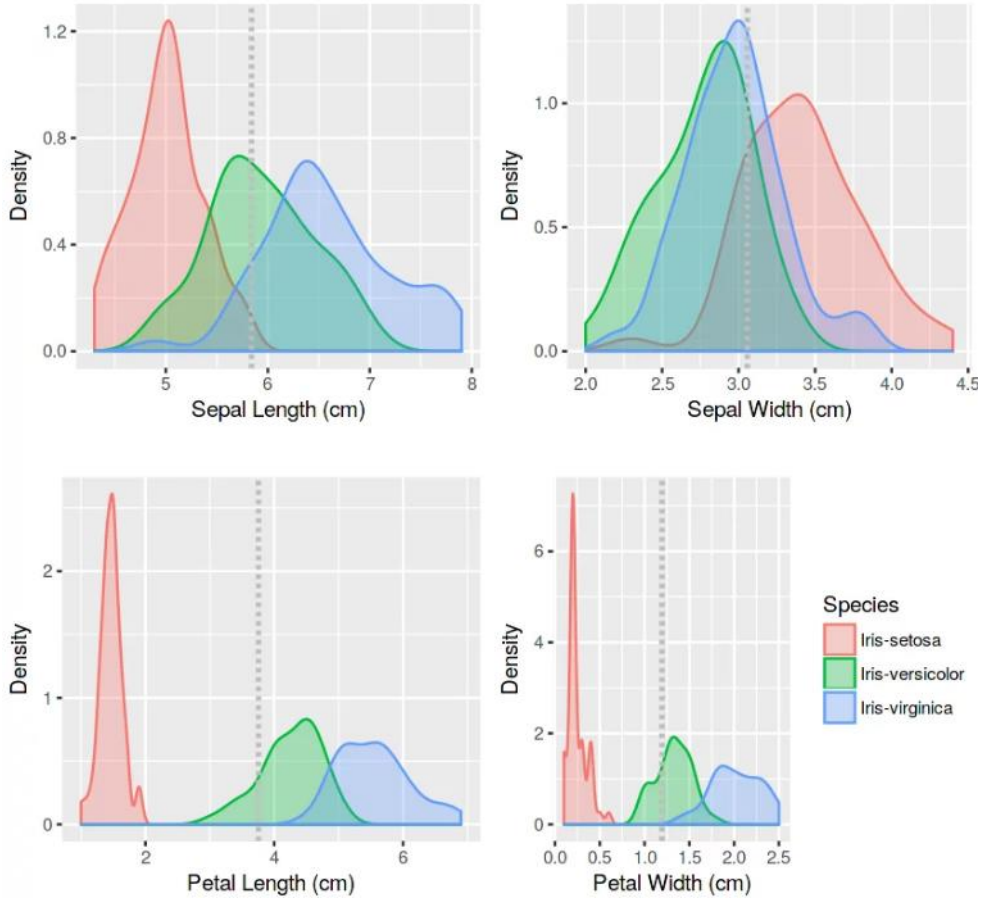
ДОДАТОК А

Дендрограма кластеризації ірисів Фішера. Метод одиночного зв'язку.



ДОДАТОК Б

Приклад візуалізації в R на прикладі датасету iris.



ДОДАТОК В

Листинг коду веб-парсеру.

```
import requests
from bs4 import BeautifulSoup
import csv

dataPost = []
for i in range(132):
    i=i+1
    p=20*i
    url=f"https://4pda.to/forum/index.php?showtopic=27115={p}"
    html = requests.get(url).text
    soup = BeautifulSoup(html, 'lxml')
    for j in range(21):
        dpost = soup.findAll('div', class_='post_body')[j].text
        dataPost.append(dpost)
        j=j+1
print(dataPost)

with open('datapost.csv', 'w') as f:
    f.write("\n".join(dataPost))
```

ДОДАТОК Г

Приклад використання пакету «e1071»

```
set.seed(123)
data("USArrests")
ss <- sample(1:50, 20)
df <- scale(USArrests[ss,])
library(e1071)
cm <- cmeans(df, 4)
cm
head(cm$membership)
library(corrplot)
corrplot(cm$membership, is.corr = FALSE)
cm$cluster
library(factoextra)
fviz_cluster(list(data = df, cluster=cm$cluster),
             ellipse.type = "norm",
             ellipse.level = 0.68,
             palette= "jco",
             ggtheme = theme_minimal())


```