ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ЗАРІЦЬКИЙ ОЛЕГ ВОЛОДИМИРОВИЧ

МЕТОДИ ТА ЗАСОБИ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ ТА ВПЛИВІВ

МЕТОДИЧНІ ВКАЗІВКИ
ДЛЯ ВИКОНАННЯ ПРАКТИЧНИХ, ЛАБОРАТОРНИХ ТА САМОСТІЙНИХ РОБІТ

Київ – 2026

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

Кафедра технологій управління

ЗАРІЦЬКИЙ ОЛЕГ ВОЛОДИМИРОВИЧ

«МЕТОДИ ТА ЗАСОБИ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ ТА ВПЛИВІВ»

Методичні вказівки
для виконання практичних, лабораторних та самостійних робіт

Київ - 2026

Рецензенти:

Ю.І. Мінаєва – к.т.н., доцент, доцент кафедри інтелектуальних технологій факультету інформаційних технологій Київського Національного Університету імені Тараса Шевченка.

Д.В. Широкорад – к.ф-м.н., доцент, доцент кафедри системного аналізу та обчислювальної математики Національного університету «Запорізька політехніка».

Рекомендовано до публікації кафедрою технологій управління, протокол №1 від «28» серпня 2025 р.

Рекомендовано до публікації Вченою радою факультету інформаційних технологій, протокол №8 від «19» січня 2026 р.

Заріцький Олег Володимирович

«Методи та засоби інтелектуального аналізу даних та впливів» [Електронний ресурс]: Методичні вказівки для виконання практичних, лабораторних та самостійних робіт з навчальної дисципліни / Заріцький О.В. – К. : КНУ імені Тараса Шевченка, 2026. – 83 с.

Дані методичні вказівки розроблені з метою забезпечення студентів детальною інформацією щодо змісту та методики виконання практичних, лабораторних та самостійних навчальних робіт у межах освітньо-наукової програми «Інформаційна аналітика та впливи» для магістрів. Методичні матеріали адресовані студентам денної та заочної форм навчання та розраховані на здобувачів вищої освіти в галузі знань 12 «Інформаційні технології», спеціальність 122 «Комп'ютерні науки». Метою цих методичних рекомендацій є сприяння ефективному засвоєнню навчального матеріалу, систематизації знань та формуванню практичних навичок у сфері «класичного» інтелектуального аналізу даних та впливів в рамках асоціативних правил, які можуть вказувати на координовану інформаційну кампанію або штучно створену асоціацію між подіями, а також джерел інформаційних впливів в соціальних мережах.

Публікується в авторській редакції.

ЗМІСТ

ПЕРЕЛІК ІЛЮСТРАЦІЙ

Вступ

Методичні вказівки з виконання практичних, лабораторних та самостійних робіт є важливим інструментом для студентів, які навчаються за освітньо-науковою програмою «Інформаційна аналітика та впливи». Ці рекомендації призначені забезпечити чітке розуміння завдань, які стоять перед студентами, а також надати необхідні інструкції для їх успішного виконання.

Практичні та лабораторні роботи є невід'ємною частиною навчального процесу, оскільки вони сприяють розвитку аналітичних навичок, критичного мислення та здатності до самостійної роботи. У даних рекомендаціях надано детальний опис завдань, порядок їх виконання, а також корисні поради щодо використання літератури та оформлення результатів досліджень. Ці матеріали допоможуть максимально ефективно засвоїти навчальний матеріал і досягти високих результатів у навчанні.

Лабораторне заняття (робота) – форма навчального заняття, при якій учень (студент) під керівництвом викладача особисто доводить натуральні або імітаційні експерименти чи досліди з метою практичного підтвердження окремих теоретичних положень даної навчальної дисципліни, набуває практичних навичок роботи з лабораторним обладнанням, обчислювальною технікою, вимірювальною апаратурою, методикою експериментальних досліджень у конкретній галузі.

Практичне заняття (робота) – форма навчального заняття, при якій викладач організовує детальний розгляд учнями (студентами) окремих теоретичних положень навчальної дисципліни та формує вміння і навички їх практичного застосування шляхом індивідуального виконання відповідно до сформованих завдань.

Лабораторні та практичні роботи мають велике значення в навчальному процесі з кількох причин (рис.1):

1. **Застосування теорії на практиці**: Вони дозволяють студентам застосовувати теоретичні знання на практиці, що сприяє глибшому розумінню матеріалу.
2. **Розвиток навичок**: Студенти набувають практичних навичок роботи з лабораторним обладнанням, приладами та технологіями, що є важливими для їхньої майбутньої професійної діяльності.
3. **Формування критичного мислення**: Під час проведення експериментів студенти вчяться аналізувати результати, робити висновки та ставити нові запитання, що сприяє розвитку критичного мислення.
4. **Підвищення мотивації**: Практична діяльність робить навчання більш цікавим і захоплюючим, що підвищує мотивацію студентів до навчання.
5. **Співпраця та командна робота**: Лабораторні роботи часто виконуються в групах, що сприяє розвитку навичок співпраці та комунікації.
6. **Розвиток дослідницьких навичок**: Студенти вчяться планувати та проводити дослідження, що є важливим для їхньої наукової діяльності в майбутньому.

Рисунок 1 Важливість лабораторних та практичних робіт.

Таким чином, лабораторні та практичні роботи є важливими для всебічного розвитку учнів, оскільки вони поєднують теорію з практикою, сприяють формуванню навичок і підвищують інтерес до навчання.

I. Рекомендації та завдання для виконання лабораторних робіт

Лабораторна робота сприяє встановленню міжпредметних зв'язків, реалізації принципу поєднання теорії з практикою та розвитку інтелектуальної активності студентів. Крім того, виконання лабораторної роботи забезпечує інтеграцію пізнавальної та практичної діяльності студентів під час вивчення основ навчальної дисципліни, що сприяє швидшому формуванню наукових знань і навичок використання методів науково-дослідної діяльності.

Лабораторна робота є невід'ємною складовою навчально-виховного процесу. Вона тісно пов'язана з навчальним експериментом, проведенням дослідів, виконанням домашніх експериментальних завдань та розв'язуванням задач на основі спостережень і експериментів. Під час лабораторних занять студенти здобувають навички роботи з приладами, лабораторним обладнанням, апаратурою, а також технічними засобами, включаючи комп'ютерну техніку та програмне забезпечення. Вони вчяться обробляти результати експериментів і вимірювань, а також узагальнювати та систематизувати явища і дані. Лабораторна робота сприяє розвитку у студентів впевненості у здатності пізнавати події та явища, а також умінь

виявляти причинно-наслідкові зв’язки та функціональні залежності між фактами, явищами, процесами і даними.

Лабораторна робота — це практичне завдання, яке передбачає проведення експерименту або дослідження для перевірки теоретичних знань, отримання нових даних або закріплення вивченого матеріалу. Вона проводиться під керівництвом викладача або самостійно в лабораторіях чи спеціально обладнаних класах залежно від навчальної програми.

Типи лабораторних занять [1].

1. Традиційні лабораторні заняття.
- Мета роботи визначена заздалегідь.
- Обладнання та інструменти підготовлені заздалегідь.
- Методика та послідовність дій чітко прописані в інструкціях.
- Результати роботи відомі студентам.
2. Дослідні лабораторні заняття.
- Тема і мета визначені, але студенти самостійно розробляють вимоги до виконання.
- Студенти самостійно вибирають інструменти з наявних у лабораторії.
- Методика роботи розробляється студентами на основі попереднього досвіду.
- Результати роботи є невідомими до виконання.

Мета лабораторних робіт:

- Перевірка теоретичних знань: студенти застосовують теорію на практиці.
- Розвиток експериментальних навичок: формування вмінь працювати з обладнанням, аналізувати результати, програмувати у відповідних середовищах і робити висновки.
- Інтеграція знань: зв'язок між теорією і практикою, розвиток міжпредметних компетенцій.
- Формування наукового мислення: розвиток логічного аналізу, уявного експериментування та творчих здібностей.

Основні етапи виконання лабораторної роботи [2]:

1. Підготовка:
- Вивчення теоретичного матеріалу.
- Визначення теми, мети та завдань роботи.
- Підготовка необхідного обладнання, матеріалів, програмного та апаратного забезпечення.
2. Проведення експерименту:
- Виконання запланованих дій згідно з методикою.
- Дотримання техніки безпеки [3].
3. Аналіз результатів:
- Обробка отриманих даних у вигляді таблиць, графіків чи діаграм.
- Формулювання висновків на основі отриманих результатів.
4. Оформлення звіту:

- Включає введення (мета, актуальність), опис методів, результати та висновки (Додаток 1).

Оформлення всіх лабораторних робіт здійснюється згідно із шаблоном (структурою) (Додаток 1), яка передбачає систематизацію отриманих даних та результатів і написання висновків.

Кожен змістовний розділ практичної та лабораторної роботи позначений відповідною іконкою (табл.1).

Таблиця 1. Позначення розділів лабораторних та практичних робіт

	**Мета роботи**. Розділ описує мету роботи та цілі у разі наявності окремих цілей. Мета роботи призначена для чіткого розуміння предмету дослідження та очікуваних результатів.
	**Теоретична інформація**. Розділ описує додатковий теоретичний матеріал, в деяких випадках надаючи більше деталей, необхідних для виконання конкретної роботи.
	**Вхідні дані**. У разі необхідності використання навчальних наборів даних, вказується посилання на розміщення такого набору в мережі, або назва набору у разі розміщення на ресурсі курсу.
	**Бібліотеки та пакети**. Розділ описує необхідні для виконання роботи бібліотеки та пакети, з посиланням на відповідні мови програмування.
	**План проведення заняття**. План проведення заняття, як правило, розробляється для практичної роботи та описує послідовність (алгоритм) виконання роботи для досягнення поставленої мети.
	**Питання вхідного контролю**. Розділ описує питання вхідного контролю, які розглядаються з викладачем на початку практичної роботи, для оцінювання готовності студентів до виконання роботи.
	**Хід проведення роботи**. Описується послідовність етапів, задач, які студент повинен виконати під час практичної чи лабораторної роботи, тобто що має бути зроблене і в якій послідовності.
	**Отримані результати, коди програм, функції**. Розділ заповнюється студентом у процесі виконання роботи інформацією, отриманою під час виконання конкретних дій: написання кодів, побудова графіків, моделювання, тобто результатами роботи. Хронологія опису результатів повинна співпадати з етапами ходу проведення роботи.

Висновки. Розділ має містити висновки щодо отриманих в ході проведення роботи результатів. Висновки не мають бути формальними і перераховувати етапи дослідження.
Висновки мають бути чіткими, логічними та узагальненими. Основні принципи:

1. **Стислість** – не повторюйте весь текст, лише ключові підсумки.
2. **Логічність** – висновки мають випливати з аналізу, а не бути новою інформацією.
3. **Узагальнення** – підсумуйте основні результати та їх значення.
4. **Практичне значення** – якщо доречно, вкажіть, де можна застосувати отримані результати.
5. **Перспективи** – коротко зазначте можливі подальші дослідження чи застосування.

Домашнє завдання. Деякі практичні роботи потребують виконання домашньої роботи під час самостійної підготовки з метою закріплення набутих навичок та знань під час виконання практичних завдань.

ЛАБОРАТОРНА РОБОТА №1. «Попередня підготовка даних».

Метою лабораторної роботи є формування професійних вмінь та навичок щодо використання інструментів та методів програмного середовища R в задачах виявлення та обробки викидів в наборах даних, вміння застосовувати отримані знання на практиці в практичних задачах інтелектуального аналізу даних.

Основний теоретичний матеріал висвітлений в лекційних матеріалах за тематикою «Попередня обробка даних у інтелектуальному аналізі даних. Методи первинної обробки даних. Стандарти інтелектуального аналізу даних».

Викид — це значення або спостереження, яке є далеким від інших спостережень, тобто точка даних, яка суттєво відрізняється від інших точок даних. Деякі автори розглядають викиди як значення, які настільки сильно відрізняються від інших спостережень, що можна припустити інший основний механізм вибірки.

У цій роботі досліджується кілька підходів для виявлення викидів у середовищі R, від простих методів, таких як описова статистика (включно з мінімумом, максимумом, гістограмою, boxplot діаграмою та процентилями), до більш формальних методів, таких як фільтр Гампеля, Граббса, Діксона та Тести Рознера на викиди [4].

Усі спостереження за межами наступного інтервалу вважатимуться потенційними викидами (1):

$$I = [Q_{0.25} - 1.5 * IQR; Q_{0.75} + 1.5 * IQR], \quad (1)$$

де IQR - interquartile range (внутрішній квартильний діапазон).

За допомогою методу процентилів усі спостереження, що знаходяться за межами інтервалу, утвореного 2,5 та 97,5 процентилями, розглядатимуться як потенційні викиди.

Якщо дані мають нормальний розподіл, можливо використовувати z-показники. Відповідно до цього методу будь-який z-показник:

< -2 або > 2 вважаються рідкісними.

< -3 або > 3 вважаються надзвичайно рідкісними.

Інший метод, відомий як фільтр Гампеля, полягає в розгляді як викидів значень за межами інтервалу (I), утвореного медіаною плюс-мінус 3 медіанних абсолютних відхилень (MAD) (2):

$$I = [median - 3 * MAD; median + 3 * MAD] \quad (2)$$

$$MAD = median(|x_i - \hat{X}|)$$

$\hat{X}$ – медіана.

Для цього методу спочатку встановлюють межі інтервалів завдяки функціям median() і mad().

У роботі буде розглянуто 3 перевірки гіпотез для виявлення викидів: тест Граббса, тест Діксона, тест Рознера (Grubbs's test, Dixon's test, Rosner's test).

Зазначені статистичні тести є частиною більш формальних методів виявлення викидів, оскільки всі вони включають обчислення тестової статистики, яка порівнюється з табличними критичними значеннями (які базуються на розмірі вибірки та бажаному рівні достовірності).

Треба зауважити, що зазначені тести підходять лише тоді, коли дані без будь-яких викидів розподіляються приблизно нормально. Рекомендується перевіряти нормальність візуально, наприклад, за допомогою графіка QQ-гістограми та/або boxplot графіка. Хоча нормальність також можна перевірити за допомогою формального тесту на нормальність (наприклад, тесту Шапіро-Вілка), наявність одного або кількох викидів може призвести до того, що тест нормальності відхилить нормальність, коли це насправді є розумним припущенням для застосування одного із трьох тестів на викиди, згаданих вище.

Тест Граббса дозволяє визначити, чи є найвище чи найнижче значення в наборі даних викидом. Тест Граббса виявляє один викид за раз (найвище або найнижче значення), тому нульова та альтернативна гіпотези є такими [5]:

H0 : найвище значення не є викидом;
H1: найвище значення є викидом;
якщо ми хочемо перевірити найвище значення, або:
H0: Найнижче значення не є викидом;
H1: Найнижче значення є викидом;
якщо ми хочемо перевірити найменше значення.

Як і для будь-якого статистичного тесту, якщо p-значення менше вибраного порогу значущості (зазвичай α=0,05), тоді нульова гіпотеза відхиляється, і ми робимо висновок, що найнижче/найвище значення є викидом.

Навпаки, якщо p-значення більше або дорівнює рівню значущості, нульова гіпотеза не відхиляється, і ми робимо висновок, що на основі даних ми не відхиляємо гіпотезу про те, що найнижче/найвище значення не є викид.

Тест Граббса не підходить для розміру вибірки 6 або менше (n≤6).

Подібно до тесту Граббса, тест Діксона використовується для перевірки того, чи є окреме маленьке або велике значення викидом. Таким чином, якщо підозрюється більше ніж один викид, тест потрібно проводити окремо для цих підозрюваних викидів. Тест Діксона найбільш корисний для невеликої вибірки (зазвичай n≤25).

Якщо потрібно знову виконати тест без найвищого чи найнижчого значення, це можна зробити, знайшовши номер рядка максимального чи мінімального значення, виключивши цей номер рядка з набору даних і нарешті застосувавши тест Діксона до цього нового набору даних.

Тест Рознера на викиди має наступні переваги:

- він використовується для виявлення кількох викидів одночасно (на відміну від тесту Ґраббса та Діксона, який потрібно виконувати ітераційно для відсіву кількох викидів), і
- його розроблено, щоб уникнути проблеми маскування, коли викид, близький за значенням до іншого викиду, може залишитися непоміченим.
- на відміну від тесту Діксона, тест Рознера є найбільш доцільним, коли розмір вибірки великий (n≥20).

Існує багато інших методів виявлення викидів:

1) у пакетах **{outliers},**
2) за допомогою функції **lofactor()** із пакету **{DMwR}:** Локальний фактор викиду (LOF) — це алгоритм, який використовується для визначення викидів шляхом порівняння локальної щільності точки з щільністю її сусідів,
3) **outlierTest()** із пакета **{car}** дає найбільш екстремальне спостереження на основі заданої моделі та дозволяє перевірити, чи є це викидом,
4) у пакеті **{OutlierDetection}** та
5) за допомогою функції **aq.plot()** із **{mvoutlier }** пакету.

ВХІДНІ ДАНІ. Набір даних mpg «Fuel economy data from 1999 to 2008 for 38 popular models of cars». Цей набір даних містить підмножину даних про паливну економічність, які EPA робить доступними на сайті ttps://fueleconomy.gov/. Він містить лише моделі, які випускалися щороку в період з 1999 по 2008 рік. Данні

було використано для визначення популярності автомобіля. Даний навчальний набір може бути використаний в навчальних цілях, оскільки має невеликий розмір, що робить його ідеальним кандидатом для моделювання і реалізації досліджень впливу гіперпараметрів моделі навіть на непотужних обчислювальних системах.

Для виконання роботи необхідно обрати інший навчальний набір даний дещо більшого розміру і описати його на початку роботи.

НЕОБХІДНІ БІБЛІОТЕКИ ТА ПАКЕТИ.

library(ggplot2), library(rstatix), library(car), library(outliers), library(EnvStats), library(mvoutlier).

ХІД РОБОТИ:

1. Завантажте dataset mpg або використовуйте інший датасет для досліджень та підвищення своєї оцінки. Датасет mpg наведено має невеликий розмір та використовується в навчальних цілях для пояснення роботи методів та функцій роботи з викидами.

2. Виконайте дослідження змінної **hwy** (highway miles per gallon) або змінної з вашого набору, вивівши описові статистики за допомогою функції **summary().** Знайдіть мінімальне, максимальне значення та діапазон з використанням відповідних функцій. Зробіть висновки про наявність викидів з аналізу базових статистик.

3. Іншим основним способом виявлення викидів є створення гістограми даних (рис.2). Створіть гістограму з використанням **функцій hist(), ggplot2().** Зробіть висновки.

4. Окрім гістограм, **boxplot()** діаграми також корисні для виявлення потенційних викидів. Створіть **boxplot()** діаграму та проаналізуйте її.

5. Опишіть та застосуйте функцію **boxplot.stats()$out**. Для пошуку викидів. Знайдіть індекси викидів в наборі даних (використовуйте функцію **which().** Виведіть спостереження (рядки) з викидами.

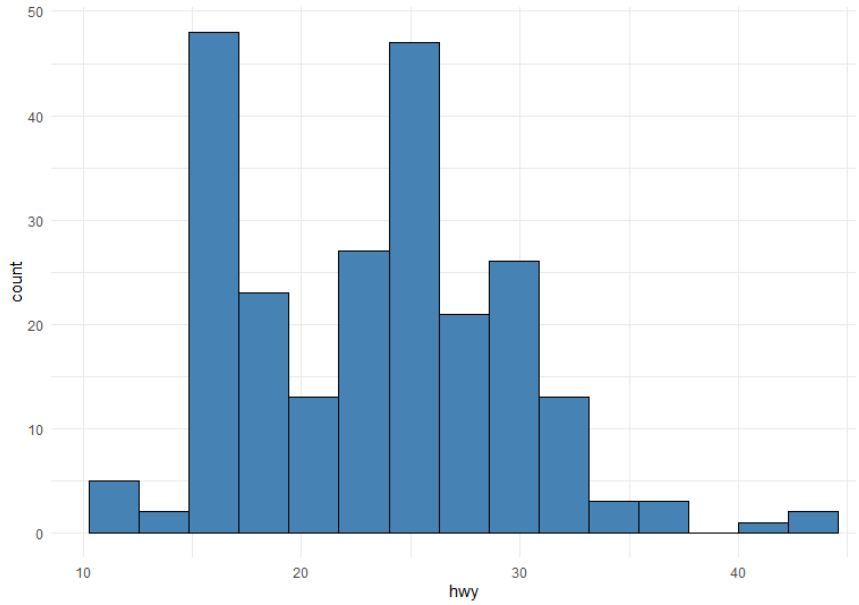


Рисунок 2. Приклад гістограми для змінної набору даних.

6. Ще один спосіб відображення цих конкретних рядків — функція **identify_outliers()** із пакета **{rstatix}**. Опишіть та використайте дану функцію.

7. Також можна надрукувати значення викидів безпосередньо на **boxplot()** діаграмі за допомогою функції **mtext().** Опишіть та використайте дану функцію (рис.3).

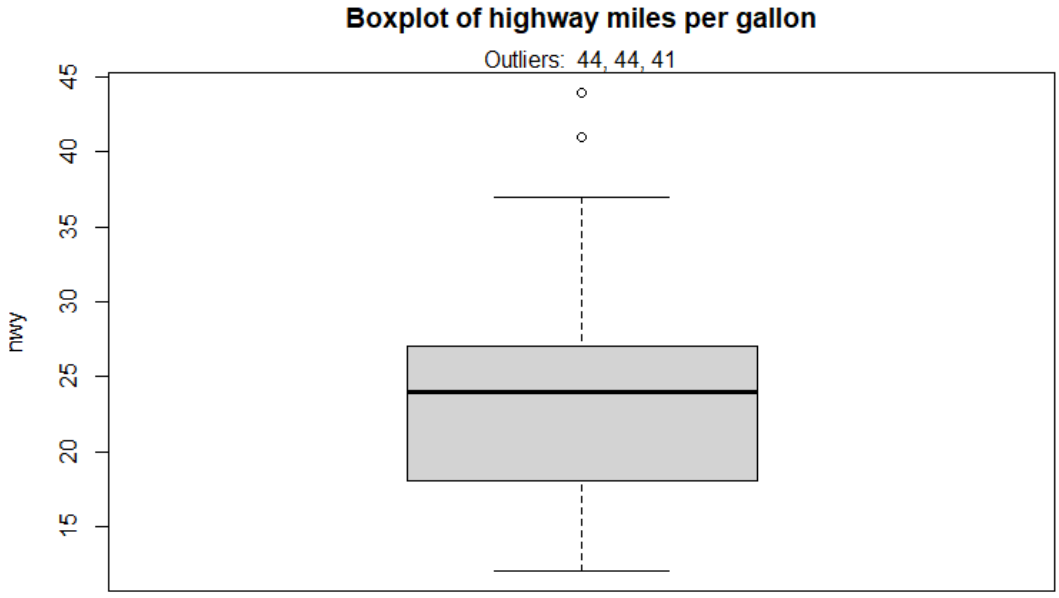


Рисунок 3. Приклад застосування функції mtext().

8. Ще один можливий метод виявлення викидів базується на процентилях, функція **quantile().** Побудуйте нижній та верхній процентилі для виявлення викидів.

9. Виконайте дослідження викидів за допомогою нормалізації даних **Z-score.**

10. Зробіть дослідження викидів за допомогою метода Гампеля (**Hampel filter**).

Статистичні тести.

11. Перевірте нормальність за допомогою **QQ-plot**() (рис.4).

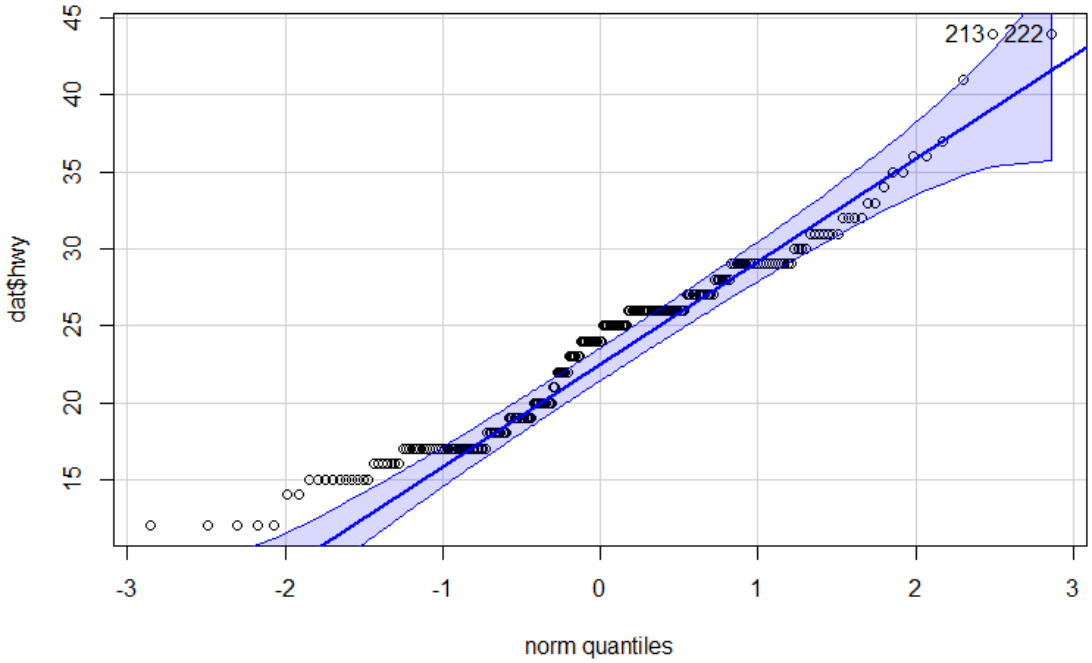


Рисунок 4. Приклад перевірки нормальності розподілення та викидів за допомогою QQ-plot()

12. Перевірте викиди за допомогою Grubbs's test (**grubbs.test()**,з пакету {**outliers**})

13. Перевірте викиди за допомогою Dixon test (**dixon.test()**,з пакету {**outliers**}) .

14. Хорошою практикою є завжди перевіряти результати статистичного тесту на викиди порівняно з графіком Boxplot, щоб переконатися, що ми протестували всі потенційні викиди. Використайте функцію mtext().

15. Перевірте викиди за допомогою Rosner's test (**rosnerTest()** з пакету {**EnvStats**}).

16. Для підвищення балу застосуйте один 5-ти останніх тестів виявлення викидів.

ОТРИМАНІ РЕЗУЛЬТАТИ:

Опишіть отримані в ході роботи результати, наведіть розроблені фрагменти кодів, програм, функцій для реалізації досліджень викидів.

ВИСНОВКИ:

Зробіть основні висновки по роботі. Починайте з короткого нагадування мети роботи, потім переходьте до основних результатів. Уникайте простого переказу того, що робили - зосередьтеся на тому, що дізналися та які закономірності виявили. Рекомендації що написання висновків ви можете знайти у вступній частині цих методичних вказівок.

ЛАБОРАТОРНА РОБОТА №2. «Задачі класифікації в середовищі аналізу даних R».

МЕТА РОБОТИ: Метою лабораторної роботи є формування професійних вмінь та навичок щодо використання інструментів та методів класифікації даних з використанням методу K-Nearest Neighbor (KNN) в середовищі мови обробки даних R, вміння застосовувати отримані знання на практиці в практичних задачах інтелектуального аналізу даних.

ТЕОРІЯ: K-Nearest Neighbor або KNN — це нелінійний алгоритм класифікації з вчителем. KNN у мові програмування R є непараметричним алгоритмом, тобто він не робить жодних припущень щодо базових даних або їх розподілу.

KNN в R є одним із найпростіших і найбільш широко використовуваних алгоритмів, який залежить від значення k (сусідів) і знаходить застосування в багатьох галузях, від фінансової індустрії до галузі охорони здоров'я тощо.

В алгоритмі KNN k визначає кількість сусідів, а його алгоритм виглядає наступним чином:

1) Виберіть число *k* сусідів.
2) Візьміть *k* найближчих сусідів до нової точки даних відповідно до визначеної відстані. *k* точок даних з найменшою відстанню до цільової точки є найближчими сусідами [6].
3) Серед k - сусідів підрахуйте кількість точок даних у кожній категорії.
4) Призначте нову (розглянуту) точку даних до категорії, з найбільшої кількістю сусідів або з найменшою відстанню у разі однакової кількості сусідів у кожній групі (рис.5). **У задачі класифікації** мітки класів *k*-найближчих сусідів визначаються шляхом голосування більшістю. Клас із найбільшою кількістю входжень серед сусідів стає прогнозованим класом для цільової точки даних. **У задачі регресії** мітка класу обчислюється шляхом усереднення цільових значень *k* найближчих сусідів. Розраховане середнє значення стає прогнозованим результатом для цільової точки даних.

У якості метрик для вимірювання відстані використовують або Евклідову відстань (Euclidean Distance) (3):

$$d_E(x_{ij}, x_{i+1,j}) = \sqrt{\sum_{j=1}^{k} [x_{ij} - x_{i+1,j}]^2} \qquad (3)$$

або Манхеттенську відстань (Manhattan Distance) (4):

$$d_M(x_{ij}, x_{i+1,j}) = \sqrt{\sum_{j=1}^{k} |x_{ij} - x_{i+1,j}|} \quad (4)$$

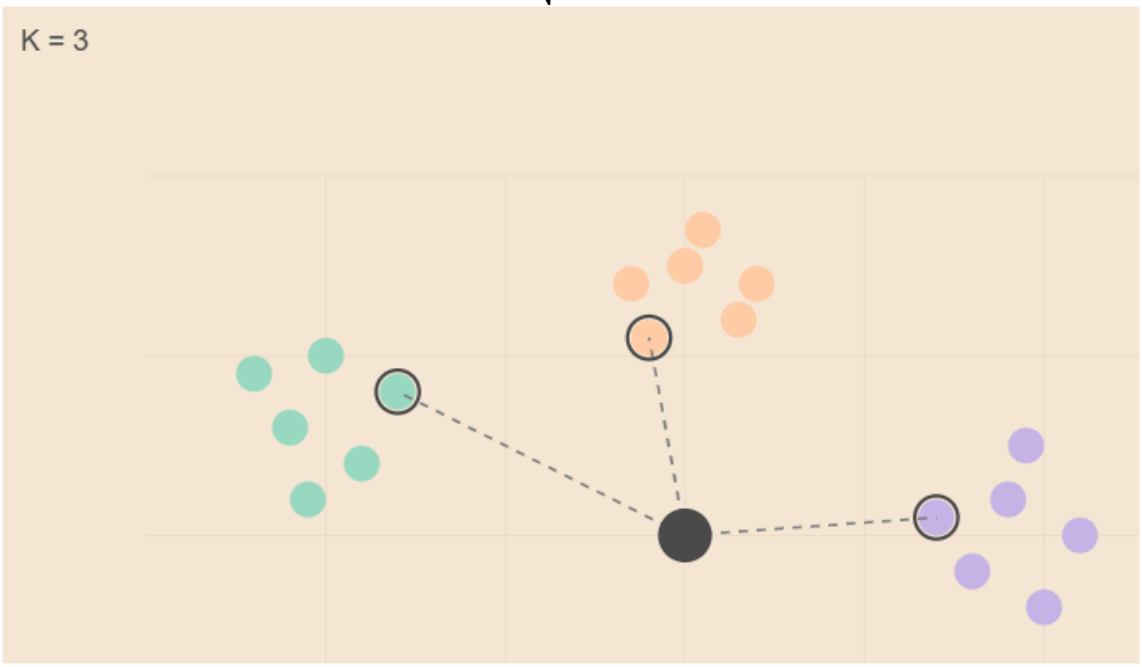


Рисунок 5. Вимірювання відстані до трьох (k=3) найближчих сусідів

Переваги алгоритму K-Nearest Neighbor:

- Легко реалізувати, оскільки складність алгоритму не така висока.
- Легко адаптується – згідно з роботою алгоритму KNN він зберігає всі дані в пам'яті, і, отже, щоразу, коли додається новий приклад або точка даних, алгоритм налаштовується відповідно до цього нового прикладу та також робить свій внесок у майбутні прогнози.
- Мало гіперпараметрів – єдиними параметрами, які потрібні для навчання алгоритму KNN, є значення *k* і вибір метрики відстані, яку ми хотіли б вибрати з нашої метрики оцінки.

Значення *k* дуже важливе в алгоритмі KNN для визначення кількості сусідів в алгоритмі. Значення *k* в алгоритмі *k*-найближчих сусідів слід вибирати на основі вхідних даних. Якщо вхідні дані мають багато викидів або шуму, більше значення *k* буде кращим. Рекомендується вибрати непарне значення для *k*.

Методи перехресної перевірки можуть допомогти у виборі найкращого значення *k* для даного набору даних.

Недоліки методу K-Nearest Neighbor:

▪ Не масштабується – оскільки алгоритм KNN також вважається «ледачим» алгоритмом. Основне значення цього терміну полягає в тому, що він потребує великої обчислювальної потужності, а також обсягів зберігання даних. Це робить цей алгоритм трудомістким і виснажливим.

▪ Прокляття розмірності – існує термін, відомий як явище піку, згідно з яким на алгоритм KNN впливає прокляття розмірності, що означає, що алгоритму важко правильно класифікувати точки даних, коли розмірність надто висока.

▪ Схильність до перенавчання – оскільки на алгоритм впливає прокляття розмірності, він також схильний до проблеми перенавчання. Тому для вирішення цієї проблеми зазвичай застосовуються методи вибору ознак, а також методи зменшення розмірності.

ВХІДНІ ДАНІ. Набір даних Iris (використовується для демонстрації методів та принципів роботи в ході лабораторної роботи) складається з 50 зразків кожного з 3 видів Iris (Iris setosa, Iris virginica, Iris versicolor) і багатовимірного набору даних, представленого британським статистиком і біологом Рональдом Фішером у його статті 1936 року «Використання кількох вимірювань у таксономічних проблемах». Для кожного зразка було виміряно чотири ознаки, тобто довжину та ширину чашолистків і пелюсток, і на основі комбінації цих чотирьох ознак Фішер розробив лінійну дискримінантну модель, щоб відрізнити види один від одного.

Даний навчальний набір може бути використаний в навчальних цілях, оскільки має невеликий розмір (лише 150 зразків), що робить його ідеальним кандидатом для моделювання і реалізації досліджень впливу гіперпараметрів моделі навіть на непотужних обчислювальних системах.

Для виконання роботи можливо обрати інший навчальний набір даних дещо більшого розміру, але детально його описати на початку роботи.

НЕОБХІДНІ БІБЛІОТЕКИ ТА ПАКЕТИ:

```
# Installing Packages: install.packages("e1071"),
install.packages("caTools"), install.packages("class")
  # Loading package: library(e1071), library(caTools), library(class)
library(ggplot2).
```

Пакет “**e1071**” є одним із найпопулярніших інструментів для статистичного моделювання та машинного навчання в мові R. Він був розроблений у Віденському університеті технологій і надає широкий спектр

функцій для виконання різноманітних завдань, включно з класифікацією, регресією та кластеризацією.

Ключові особливості пакета “e1071”:

1. Підтримка векторних машин (SVM): Пакет пропонує просту й ефективну реалізацію алгоритмів SVM, які можуть використовуватися як для класифікації, так і для регресії. Він підтримує різні ядра, включно з лінійними, поліноміальними та радіально-базисними функціями, що дає змогу обробляти як лінійно, так і нелінійно розділені дані.

2. Наївний байєсівський класифікатор: “e1071” також містить реалізацію наївного байєсівського класифікатора, що ґрунтується на теоремі Байєса і передбачає незалежність між ознаками. Цей метод особливо ефективний для завдань текстової класифікації.

3. Кластеризація: Пакет надає функції для виконання кластеризації, включно з алгоритмами k-середніх і нечіткою кластеризацією (fuzzy c-means), що дає змогу користувачам групувати дані за схожістю.

4. Додаткові утиліти: “e1071” включає функції для виконання короткочасного перетворення Фур'є та аналізу латентних класів, що розширює його функціональність для різних статистичних завдань.

Пакет “**caTools**” є універсальним інструментом для аналізу даних у мові R, що надає набір функцій, які спрощують виконання різноманітних математичних і статистичних операцій. Він широко використовується для опрацювання та аналізу даних, особливо в контексті часових рядів і маніпуляцій з даними.

Ключові функції та можливості пакета caTools:

1. Поділ даних: Пакет дає змогу легко розділяти дані на навчальні та тестові набори, що є важливим кроком у процесі побудови моделей машинного навчання.

2. Рухомі середні та фільтри: caTools надає функції для обчислення рухомих середніх і застосування різних фільтрів до часових рядів, що допомагає в аналізі трендів і сезонних коливань.

3. Базові статистичні функції: Пакет містить функції для виконання основних статистичних розрахунків, таких як кореляції та накопичувальні суми, що робить його корисним для попереднього аналізу даних.

4. Читання і запис файлів: caTools підтримує функції для читання і запису GIF і бінарних файлів ENVI, що розширює його застосування в галузі обробки зображень і роботи з геопросторовими даними.

Пакет “**class**” у мові R надає функції для реалізації методів класифікації, що робить його важливим інструментом для аналізу даних і машинного навчання. Він містить кілька алгоритмів, які дають змогу ефективно класифікувати об'єкти на основі їхніх характеристик.

Ключові функції та можливості пакета “class”:

1. Методи класифікації: Пакет містить різноманітні алгоритми, такі як метод k-найближчих сусідів (k-NN), який є одним із найпопулярніших методів

для розв'язання завдань класифікації. Цей метод ґрунтується на принципі, що об'єкти, які розташовані близько один до одного в просторі ознак, мають схожі класи.

2. Легкість використання: Функції в пакеті “class” мають простий та інтуїтивно зрозумілий інтерфейс, що дає змогу користувачам швидко застосовувати методи класифікації без необхідності глибокого розуміння алгоритмів.

3. Підтримка різних типів даних: Пакет може працювати з різними типами даних, включно з числовими та категоріальними змінними, що робить його універсальним інструментом для аналізу даних у різних галузях.

ХІД РОБОТИ:

1. Завантажте обраний вами невеликий навчальний набір даних та відповідні бібліотеки. Опишіть обраний вами набір даних кількісно і якісно (кількість спостережень, кількість ознак (змінних), опис ознак, типи даних тощо.

2. Розділіть дані на тренувальний та тестовий набори в співвідношенні (70% на 30%).

3. Нормалізуйте дані (це може бути Z-score).

4. Побудуйте K-model (knn()). Зробіть дослідження та опишіть параметри налаштування функції. Опишіть налаштування функції.

5. Побудуйте матрицю помилок (Confusiin Matrix) (рис.6). Зробіть висновки

	classifier_knn		
	setosa	versicolor	virginica
setosa	20	0	0
versicolor	0	19	1
virginica	0	1	19

Рисунок 6. Матриця помилок класифікації (k=3) для набора IRIS

6. Зробіть дослідження щодо точності прогнозування з різними числами *k (кількості сусідів)*.

7. Створіть графік точності прогнозування в залежності від *k* (рис.7).

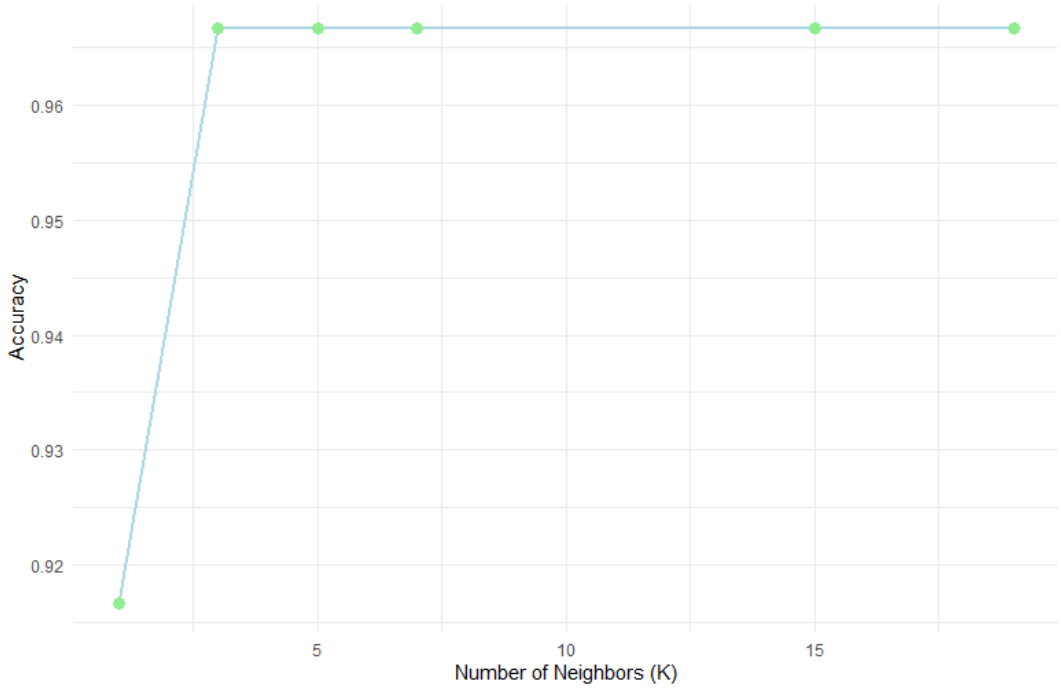


Рисунок 7. Точність класифікації в залежності від параметру - k (приклад для набору даних IRIS)

ОТРИМАНІ РЕЗУЛЬТАТИ.

Опишіть отримані в ході роботи результати, наведіть розроблені фрагменти кодів, програм, функцій для реалізації досліджень викидів.

ВИСНОВКИ.

Зробіть основні висновки по роботі. Порівняйте отримані дані з теоретичними значеннями або очікуваними результатами (припущеннями). Якщо є розбіжності, спробуйте їх пояснити – це може бути пов’язано з неякісними вхідними даними, особливостями коду або реальними фізичними явищами, які не враховує спрощена теорія. Рекомендації що написання висновків ви можете знайти у вступній частині цих методичних вказівок.

ЛАБОРАТОРНА РОБОТА №3. «Задача кластеризації в середовищі аналізу даних».

МЕТА РОБОТИ. Метою лабораторної роботи є формування професійних вмінь та навичок щодо використання інструментів та методів кластеризації даних з використанням методів кластеризації K-Means та ієрархічної кластеризації, вміння застосовувати отримані знання на практиці в практичних задачах аналізу даних клієнтів.

ТЕОРІЯ. **K-Means** — це ітераційна техніка жорсткої кластеризації, яка використовує алгоритм неконтрольованого навчання (рис.8). У цьому випадку загальна кількість кластерів попередньо визначається користувачем, і на основі подібності кожної точки даних відбувається їх групування. Цей алгоритм також визначає центроїд кластера.

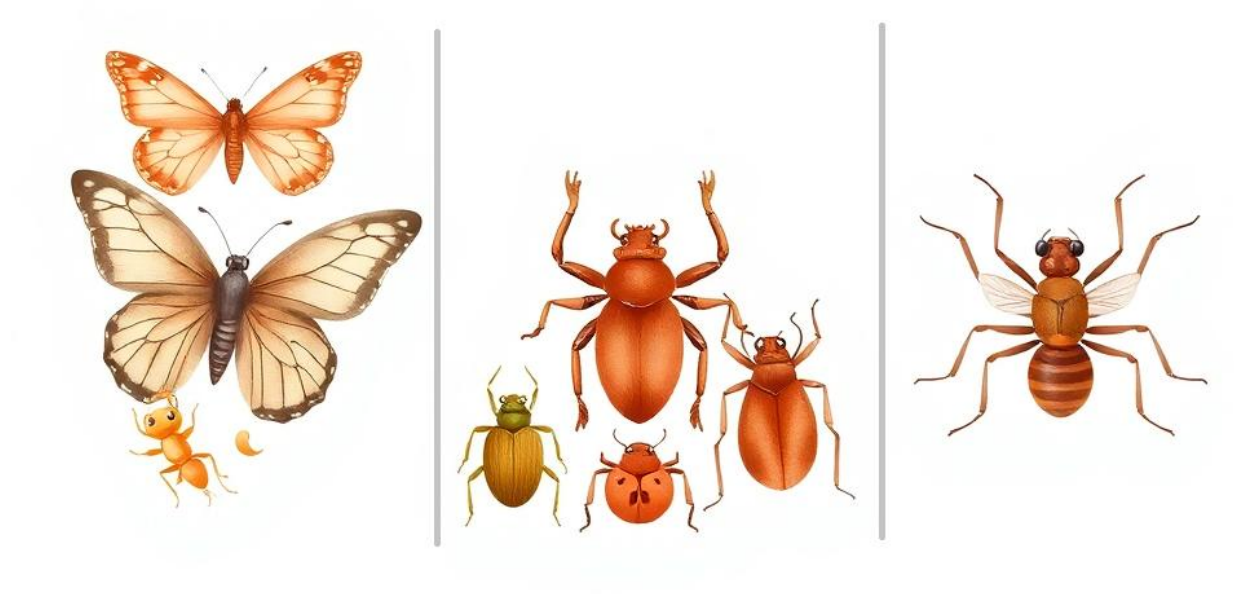


Рисунок 8. Кластеризація комах за будовою тіла

K-середніх кластеризація, призначає точки даних до одного з *k* кластерів залежно від їхньої відстані від центру кластерів. Починається з випадкового визначення центроїда кластерів у просторі. Потім кожна точка даних відноситься до одного з кластерів на основі її відстані від центроїда кластера. Після віднесення кожної точки до одного з кластерів, призначаються нові центроїди кластерів. Цей процес повторюється доти, доки не буде знайдено задовільне з погляду якості кластеризації рішення [7].

Алгоритм для K-Means:

1) Вкажіть кількість кластерів (*k*).
2) Випадково призначте кожну точку даних кластеру.

3) Обчислити центроїд кластера.

4) Перерозподіліть кожну точку даних до найближчого центроїда кластера.

5) Змінити центроїд кластера.

В аналізі припускаємо, що кількість кластерів задана наперед і ми повинні віднести точки до одного з них. Хоча існують аналітичні методи визначення кількості кластерів, наприклад метод ліктя [8].

У деяких випадках *k* не є чітко визначеним, і нам доводиться думати про оптимальну кількість *k*. Кластеризація за допомогою K-середніх найкраще працює, коли дані добре розділені. Коли точки даних перекриваються, ця кластеризація не підходить.

K-Means є швидшим порівняно з іншими методами кластеризації. Він забезпечує сильний зв'язок між точками даних. Кластери *K*-Means не надають чіткої інформації щодо якості кластерів. Різне початкове призначення центроїда кластера може призвести до різних форм кластерів. Крім того, алгоритм *K*-Means чутливий до шуму. Він може застрягти в локальних мінімумах.

Кожна точка даних класифікується за її найближчим середнім значенням (призначаються точки даних до найближчого кластерного центру). Використовуємо Евклідову відстань (2).

В ієрархічній кластеризації об’єкти класифікуються в ієрархію, подібну до деревоподібної структури, яка використовується для інтерпретації моделей ієрархічної кластеризації.

Метод починається з обробки кожної точки даних як окремого кластера, а потім ітеративно об’єднує найближчі кластери, доки не буде досягнуто критерій зупинки. Результатом ієрархічної кластеризації є деревоподібна структура (рис.9), яка називається **дендрограмою**, яка ілюструє ієрархічні зв’язки між кластерами [9].

В основному існує два типи ієрархічної кластеризації:

- Агломеративна кластеризація. Спочатку кожну точку даних розглядається як окремий кластер і на кожному кроці об’єднують найближчі пари кластерів. (Це метод «знизу вгору»).
- Роздільна кластеризація. У роздільній ієрархічній кластеризації ми беремо до уваги всі точки даних як один кластер і на кожній ітерації ми відокремлюємо точки даних від кластерів, які не можна порівняти.

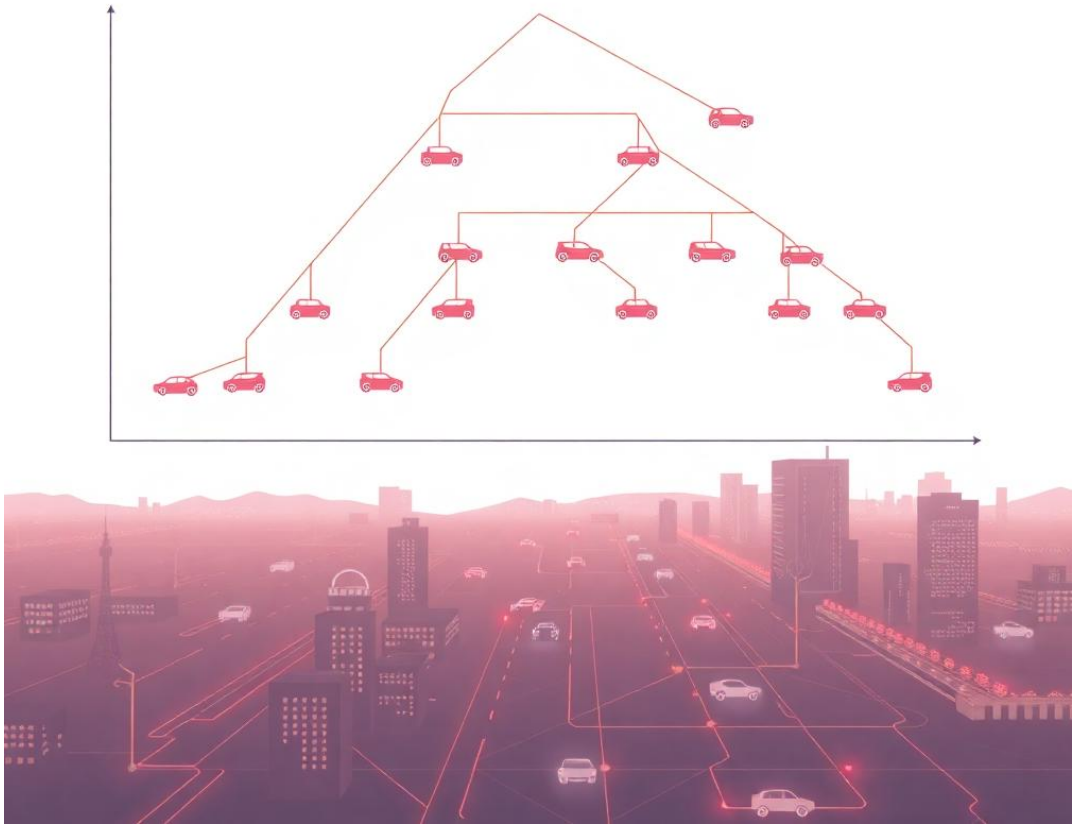


Рисунок 9. Дендрограма автомобілів (абстракція)

Основні етапи ієрархічної кластеризації:

1. Обчислення відстані між об'єктами – використовуються метрики, наприклад, Евклідова відстань.
2. Об'єднання найближчих об'єктів у малі кластери.
3. Поступове злиття кластерів – формування ієрархічної структури.
4. Побудова дендрограми (рис.9) – визначення оптимального рівня кластеризації (відсічення дерева).

Щоб почати ієрархічну кластеризацію, ми повинні спочатку вирішити три питання.

1. Як представити кластер?
2. Як вибрати два кластери для злиття?
3. Коли ми припинимо об'єднувати кластери?

Основні метрики для вимірювання якості кластеризації це радіус і діаметр кластеру. Радіус кластера — це максимальна відстань між центроїдом і точками цього кластера. Діаметр кластера - це максимальна відстань між усіма парами точок у цьому кластері. Використання діаметра кластера як фактора для об’єднання кластерів призводить до створення кластеру найменшого діаметру.

Коли алгоритм треба зупинити?

1. Алгоритм зупиняється, коли щільність найкращих об’єднаних кластерів стає меншою за деяке порогове значення. Щільність можна визначити відповідно до різних факторів, таких як діаметр або радіус. Наприклад, у двовимірному просторі можливо обчислити щільність кластера, поділивши кількість його вузлів на квадрат його радіуса, щоб продемонструвати кількість точок кластера в його одиничному об’ємі. Одиницю розміру також можна визначити як радіус у ступені розміру. Отже, якщо найкраще злиття викликає новий кластер з невеликою щільністю, об’єднані кластери були досить далеко один від одного.

2. Алгоритм зупиняється, якщо діаметр наступного найкращого об’єднаного кластера перевищує певну межу. Фактором може бути радіус або будь-який пов’язаний з ним варіант.

3. Алгоритм зупиняється, якщо комбінація кластерів створює поганий кластер. Наприклад, він може відстежувати середній діаметр кластерів. Цей коефіцієнт буде помірно зростати під час кластеризації. Але якщо спостерігається великий стрибок, це означає, що це точка для зупинки алгоритму. Треба зауважити, що в цьому правилі не можливо фіксувати порогове значення, необхідно враховувати тенденцію та зупинитися, коли трапляється щось незвичайне.

Ієрархічна кластеризація має кілька переваг перед іншими методами кластеризації:

- Здатність працювати з неопуклими кластерами та кластерами різного розміру та щільності.
- Можливість обробки відсутніх даних і даних із шумом.
- Здатність розкривати ієрархічну структуру даних, що може бути корисним для розуміння зв’язків між кластерами.

Недоліки ієрархічної кластеризації:

- Необхідність критерію зупинки процесу кластеризації та визначення остаточної кількості кластерів.
- Обчислювальна вартість і вимоги до пам’яті методу можуть бути високими, особливо для великих наборів даних.
- Результати можуть бути чутливими до початкових умов, критерію зв’язку та використовуваної метрики відстані.

ВХІДНІ ДАНІ.

В ході лабораторної роботи для вивчення параметрів функцій використовується навчальний набір mtcars (рис.10). Дані були взяті з журналу Motor Trend US за 1974 рік і включають споживання палива та 10 аспектів автомобільного дизайну та продуктивності для 32

автомобілів (1973–74 моделі). Даний навчальний набір може бути використаний в навчальних цілях, оскільки має невеликий розмір (32 об'єкта (спостереження) та 11 ознак (змінних)), що робить його ідеальним для моделювання і реалізації досліджень впливу гіперпараметрів моделі навіть на непотужних обчислювальних системах.

Для виконання роботи необхідно обрати інший навчальний набір даний дещо більшого розміру і описати його на початку роботи.

A data frame with 32 observations on 11 (numeric) variables.

[, 1] mpg	Miles/(US) gallon
[, 2] cyl	Number of cylinders
[, 3] disp	Displacement (cu.in.)
[, 4] hp	Gross horsepower
[, 5] drat	Rear axle ratio
[, 6] wt	Weight (1000 lbs)
[, 7] qsec	1/4 mile time
[, 8] vs	Engine (0 = V-shaped, 1 = straight)
[, 9] am	Transmission (0 = automatic, 1 = manual)
[,10] gear	Number of forward gears
[,11] carb	Number of carburetors

Рисунок 10. Набір даних mtcars.

НЕОБХІДНІ БІБЛІОТЕКИ ТА ПАКЕТИ:

#Installing Packages: install.packages("fpc"), install.packages("dplyr"), install.packages("factoextra")

Loading package: library(fpc), library(dplyr), library(factoextra)

ХІД РОБОТИ:

1. Завантажте навчальний набір даних (бажано обирати невеликі набори) та опишіть його параметри та змінні. Зробіть повний аналіз даних (тип, кількість тощо).

2. У разі необхідності здійсніть первинну обробку набору (знайдіть викиди, пропущені дані та очистіть набір). Можливо знадобиться нормалізація набору даних, скористайтеся відомими вам функціями нормалізації, наприклад Z-score.

3. Створіть модель kmeans(), опишіть параметри моделі, здійсніть дослідження якості кластеризації з різною кількістю кластерів.

4. Здійсніть візуалізацію отриманих результатів з використанням функції fviz_cluster з пакету factoextra (рис.11).

Рисунок 11. Приклад реалізації функції kmeans() для начального набору mtcars

5. Побудуйте матрицю відстаней (dist(data_set, method = 'euclidean')), зробіть аналіз (рис.12).

	Mazda RX4	Mazda RX4 Wag	Datsun 710	Hornet 4 Drive	Hornet
Mazda RX4 Wag	0.6153251				
Datsun 710	54.9086059	54.8915169			
Hornet 4 Drive	98.1125212	98.0958939	150.9935191		
Hornet Sportabout	210.3374396	210.3358546	265.0831615	121.0297564	
Valiant	65.4717710	65.4392224	117.7547018	33.5508692	
Duster 360	241.4076490	241.4088680	294.4790230	169.4299647	
Merc 240D	50.1532711	50.1146059	49.6584796	121.2739722	
Merc 230	25.4683117	25.3284509	33.1803843	118.2433145	
Merc 280	15.3641921	15.2956865	66.9363534	91.4224033	
Merc 280C	15.6724727	15.5837744	67.0261397	91.4612914	
Merc 450SE	135.4307018	135.4254826	189.1954941	72.4964325	
Merc 450SL	135.4014424	135.3960351	189.1631745	72.4313532	
Merc 450SLC	135.4794674	135.4723157	189.2345426	72.5718466	
Cadillac Fleetwood	326.3395903	326.3355070	381.0926242	234.4403876	

Рисунок 12. Приклад фрагменту матриці відстаней для навчального набору mtcars

6. Здійсніть кластеризацію обраного набору за допомогою ієрархічної кластеризації. Опишіть параметри налаштування функції hclust().

7. Побудуйте дендрограму кластеризації (дерево) (рис.13).

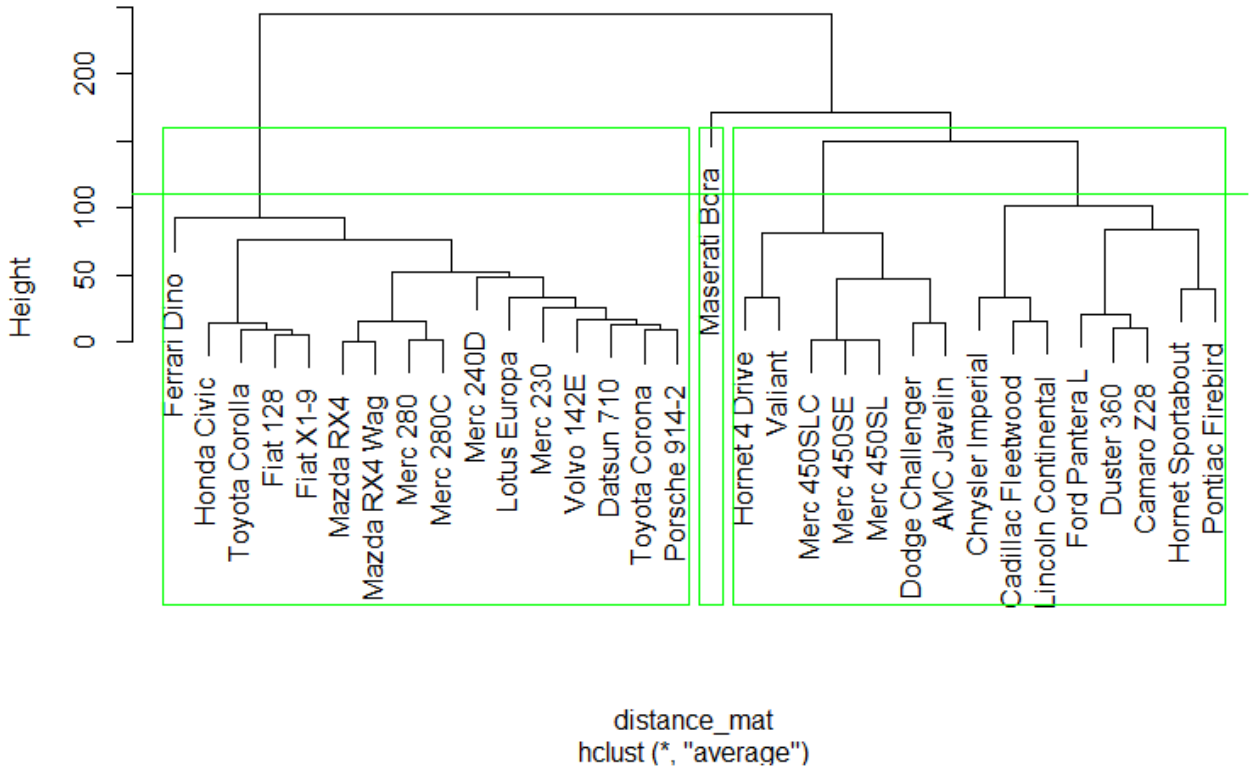


Рисунок 13. Приклад дендрограми для навчального набору mtcars

ОТРИМАНІ РЕЗУЛЬТАТИ. Опишіть отримані в ході роботи результати, наведіть розроблені фрагменти кодів, програм, функцій для реалізації досліджень викидів.

ВИСНОВКИ. Зробіть основні висновки по роботі. Починайте з короткого нагадування мети роботи, потім переходьте до основних результатів. Уникайте простого переказу того, що робили - зосередьтеся на тому, що дізналися та які закономірності виявили. Починайте з короткого нагадування мети роботи, потім переходьте до основних результатів. Уникайте простого переказу того, що робили - зосередьтеся на тому, що дізналися та які закономірності виявили.

ЛАБОРАТОРНА РОБОТА №4. «Прогнозування. Задача регресії в середовищі аналізу даних».

МЕТА РОБОТИ. Метою лабораторної роботи є формування професійних вмінь та навичок щодо використання інструментів та методів регресійного аналізу в задачах інтелектуального аналізу даних, вміння застосовувати отримані знання на практиці в практичних задачах аналізу даних.

ТЕОРІЯ. Проста лінійна регресія [10,11] використовується для прогнозування постійної змінної результату (y) на основі однієї єдиної змінної предиктора (x). Регресія з багатьма змінними (ознаками) може бути описана рівнянням (5):

$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n, \quad (5)$$

де

$\hat{y}$ - прогнозне значення (predicted value);

n – кількість ознак (number of features);

x_i - значення i-ї ознаки (value of i-th feature);

θ_j - j-та вага ознаки включаючи θ_0 член зміщення (bias term) або вільний член (intercept term);

Векторизована форма (6):

$$\hat{y} = h_\theta(x) = \theta^T \cdot \vec{x}, \quad (6)$$

де

θ – вектор параметрів моделі включаючи член зміщення θ_0 та ваги ознак θ_j

θ^T - транспонований вектор параметрів (вектор строка)

$\vec{x}$ - вектор ознак

$\theta^T \cdot \vec{x}$ - скалярний добуток

Типовим показником продуктивності для проблем регресії є квадратний корінь з середньоквадратичної похибки (RMSE) (7). Це дає уявлення про те, яку похибку система зазвичай робить у своїх передбаченнях (з вищою вагою для більших похибок).

$$RMSE(X,h) = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(h(\vec{x}^{(i)}) - y^{(i)})^2} \quad (7)$$

m – кількість зразків в наборі даних;

$\vec{x}^{(i)}$ - вектор всіх ознак (за виключенням міток) і-го зразка в наборі даних, $y^{(i)}$ мітка (бажане вихідне значення для даного зразка).

X – матриця, яка включає значення всіх ознак (за виключенням міток) всіх зразків набору даних. Кожен зразок в окремому рядку.

h - функція передбачення, або гіпотеза $\hat{y}^{(i)} = h(x^{(i)})$.

Припустимо, що є багато даних із викидами. У цій ситуації як правило, можна розглянути можливість використання середньої абсолютної похибки (MAE), яка також називається середнім абсолютним відхиленням (8).

$$MAE(X,h) = \frac{1}{m}\sum_{i=1}^{m}\left|h\left(\vec{x}^{(i)}\right) - y^{(i)}\right| \quad (8)$$

m – кількість зразків в наборі даних;

$\vec{x}^{(i)}$ - вектор всіх ознак (за виключенням міток) і-го зразка в наборі даних, $y^{(i)}$ мітка (бажане вихідне значення для даного зразка).

X – матриця, яка включає значення всіх ознак (за виключенням міток) всіх зразків набору даних. Кожен зразок в окремому рядку.

h - функція передбачення, або гіпотеза $\hat{y}^{(i)} = h(x^{(i)})$.

Метрики RMSE і MAE — це способи вимірювання відстані між двома векторами: прогнозним вектором і цільовим вектором. Існують різні міри відстані, або норми.

Перш ніж використовувати модель для прогнозів, необхідно оцінити статистичну значущість моделі. Це можна легко перевірити, відобразивши статистичний підсумок моделі за допомогою функції summary(), яка виводить наступні результати:

Call. Показує функцію, яка використовується для обчислення моделі регресії.

Залишки (Residuals). Надає швидкий перегляд розподілу залишків, які за визначенням мають нульове середнє значення. Тому медіана не повинна бути далекою від нуля, а мінімум і максимум повинні бути приблизно рівними за абсолютною величиною.

Коефіцієнти (Coefficients). Показує коефіцієнти регресії та їх статистичну значущість. Змінні предиктора, які суттєво пов'язані зі змінною результату, позначені зірочками.

Залишкова стандартна помилка (Residual standard error, RSE), R-квадрат (R2) і F-статистика – це показники, які використовуються для перевірки того, наскільки добре модель відповідає нашим даним.

Першим кроком у інтерпретації множинного регресійного аналізу є перевірка F-статистики та пов'язаного значення *p* у нижній частині підсумку моделі.

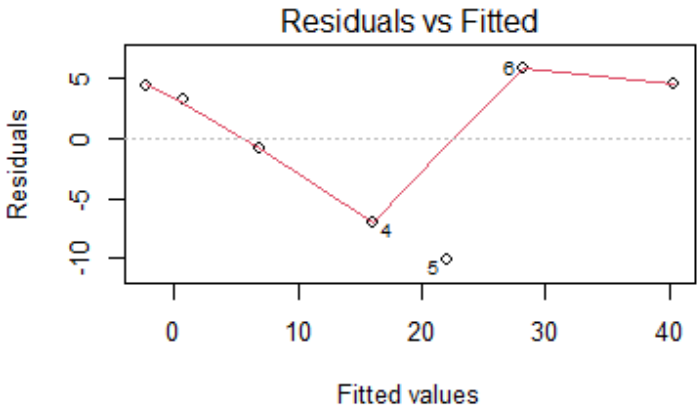
Значущість коефіцієнтів: Щоб побачити, які змінні прогнозу є значущими, ви можете переглянути таблицю коефіцієнтів, яка показує оцінку коефіцієнтів регресії та пов'язаних значень *t*-статистики *p*.

Слід зауважити, що лінійна регресія передбачає лінійний зв'язок між результатом і змінними предиктора. Це можна легко перевірити, створивши діаграму розсіювання змінної результату та змінної предиктора.

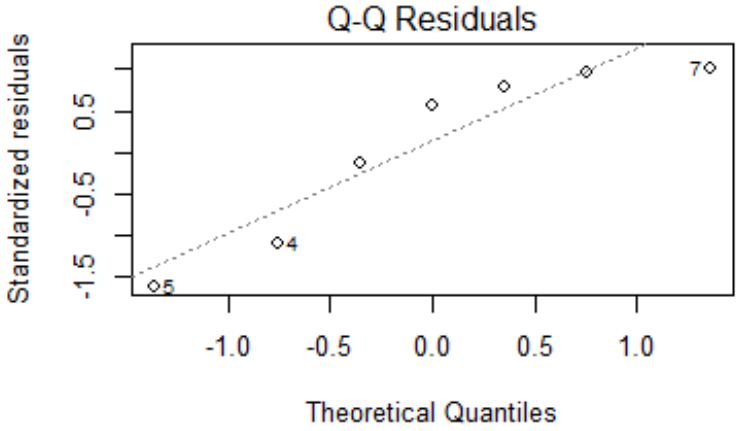
Діагностичні графіки: діаграми регресійної діагностики можна створити за допомогою базової функції R plot() або функції autoplot() [пакет ggfortify], яка створює графіку на основі ggplot2.

Діагностичні графіки виводять наступні результати:

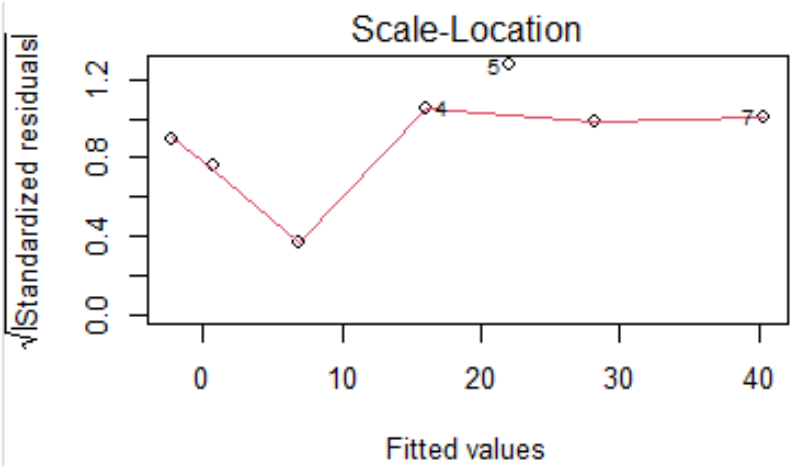
Залишки проти підігнаних (Residuals vs Fitted). Використовується для перевірки припущень про лінійну залежність. Горизонтальна лінія без чітких візерунків є показником лінійного зв'язку, що добре в дослідженнях.



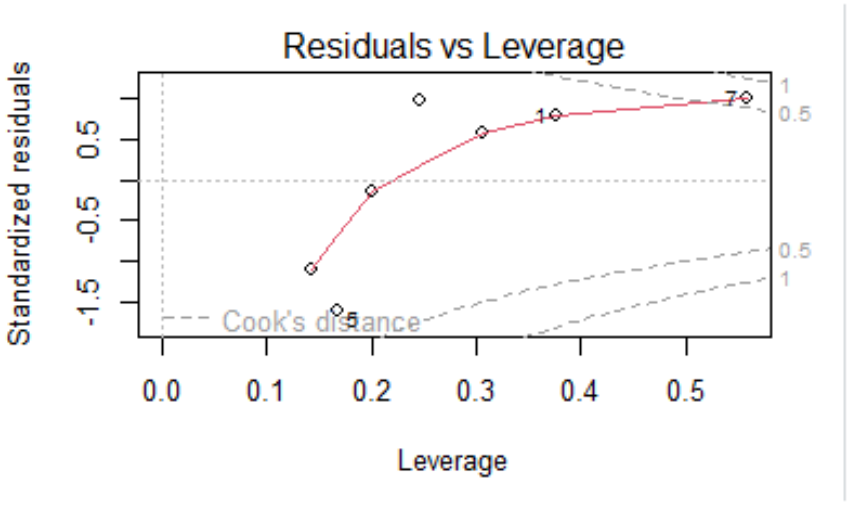
Нормальний Q-Q (Normal Q-Q). Використовується для перевірки нормального розподілу залишків [17]. Добре, якщо точки залишків розташовані вздовж прямої пунктирної лінії.



Масштаб-розташування (або розповсюдження-розташування) (Scale-Location (or Spread-Location)). Використовується для перевірки однорідності дисперсії залишків (гомоскедастичність). Горизонтальна лінія з рівномірно розподіленими точками є хорошим показником гомоскедастичності.



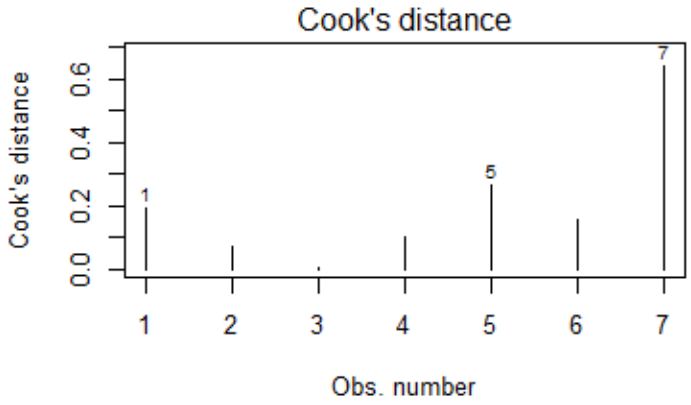
Residuals vs Leverage. Використовується для виявлення впливових випадків, тобто екстремальних значень, які можуть вплинути на результати регресії, якщо їх включити або виключити з аналізу.



Статистики розробили метрику під назвою «відстань Кука», щоб визначити вплив значення. Цей показник визначає вплив як комбінацію левериджу та залишкового розміру.

Графік «Залишки проти левериджу» може допомогти знайти впливові спостереження, якщо такі є. На цьому графіку значення, що виходять за межі, зазвичай розташовані у верхньому правому куті або в нижньому правому куті. Ці точки є місцями, де точки даних можуть впливати на лінію регресії. Отже, є лише один елемент (№7), який є впливовим (знаходиться за лініями Кука). Реконструкція після видалення цього елемента може дати краще рішення.

Відстані Кука (plot(model, 4)). Емпіричне правило полягає в тому, що спостереження має великий вплив, якщо відстань Кука перевищує 4/(n - p - 1), де *n* — кількість спостережень, а *p* — кількість змінних предиктора.



ВХІДНІ ДАНІ. Довільний фрейм даних з невеликою кількістю спостережень (100 - 150) та з декількома незалежними змінними (до 3), наприклад, витрати на онлайн рекламу та результати продажів за звітний період. Або невеликий датасет (кількість спостережень до 500 з декількома змінними) для реалізації простого регресійного аналізу.

НЕОБХІДНІ БІБЛІОТЕКИ ТА ПАКЕТИ:

Використовуються стандартні бібліотеки регресійного аналізу, які включають функцію lm().

ХІД РОБОТИ.

1. Створіть фрейм даних з невеликою кількістю спостережень (100 - 150) та з декількома незалежними змінними (до 3), наприклад, витрати на онлайн рекламу та результати продажів за звітний період. Або завантажте невеликий датасет для реалізації простого регресійного аналізу.

2. Створіть регресійну модель з використанням функції lm(). Опишіть параметри налаштування моделі. Виведіть отриману модель за допомогою функції summary(). Запишіть отримане рівняння (рис.14).

```
Call:
lm(formula = sales ~ youtube, data = marketing)

Residuals:
    Min      1Q  Median      3Q     Max
-8.712 -4.913 -0.675   5.151 10.542

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.09301    3.43953   0.027    0.979
youtube      2.29099    0.18934  12.100 7.17e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.984 on 9 degrees of freedom
Multiple R-squared:  0.9421,    Adjusted R-squared:  0.9357
F-statistic: 146.4 on 1 and 9 DF,  p-value: 7.175e-07
```

Рисунок 14. Приклад опису лінійної регресії

3. Розрахуйте прогнозне значення доходів від продажів при певному рівні вкладень у рекламу, створіть новий набір даних для рекламного бюджету та отримайте прогноз з використанням розробленої моделі.

4. Оцініть статистичну значущість моделі. Зробіть висновки по отриманим оцінкам наскільки добре модель відповідає вашим даним.

5. Створіть діаграму розсіювання змінної результату (y) та змінної предиктора (x) (рис.15), використовуючи функцію ggplot().

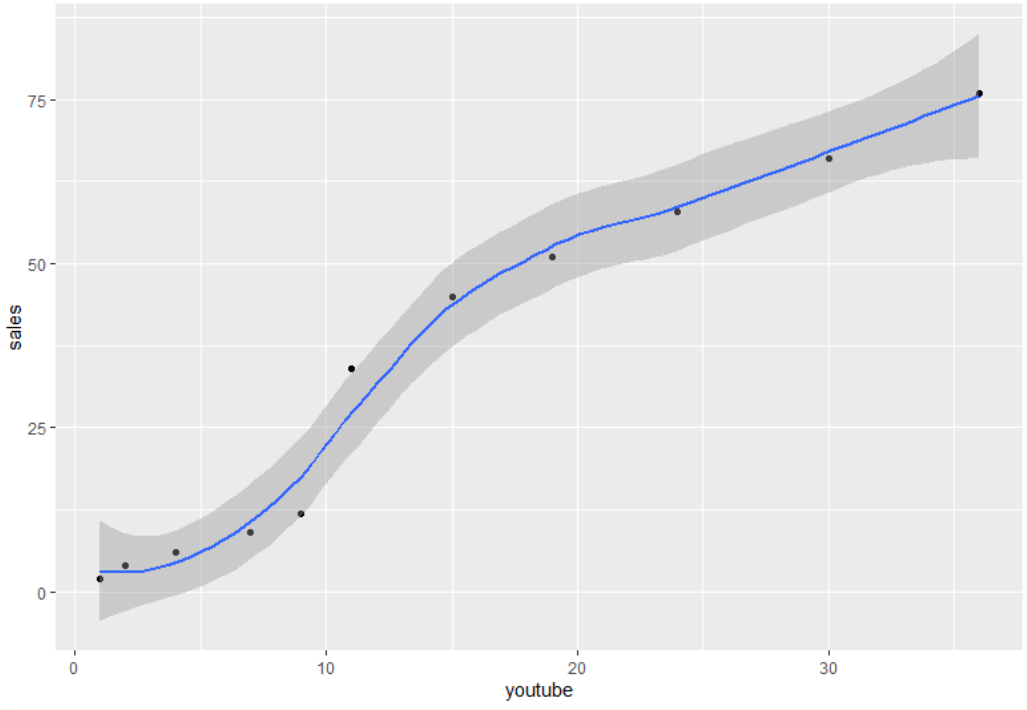


Рисунок 15. Приклад візуалізації графіка лінійної регресії

6. Створіть діагностичні графіки за допомогою базової функції R plot(model). Поясніть отримані результати (рис.16).

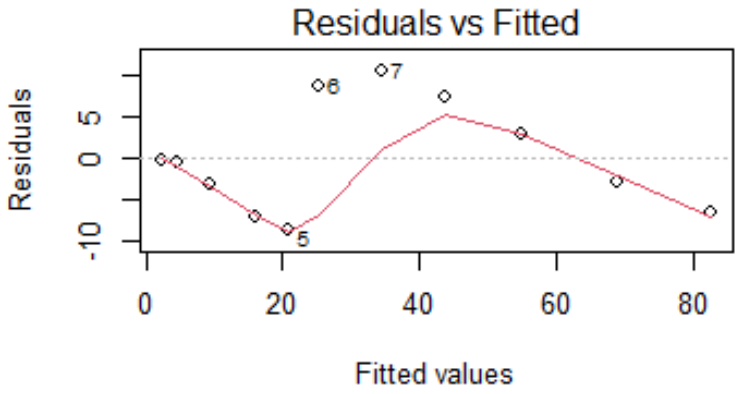


Рисунок 16. Приклад графіку Residuals vs Fitted

7. Побудуйте та візуалізуйте відстані Кука (рис.17).

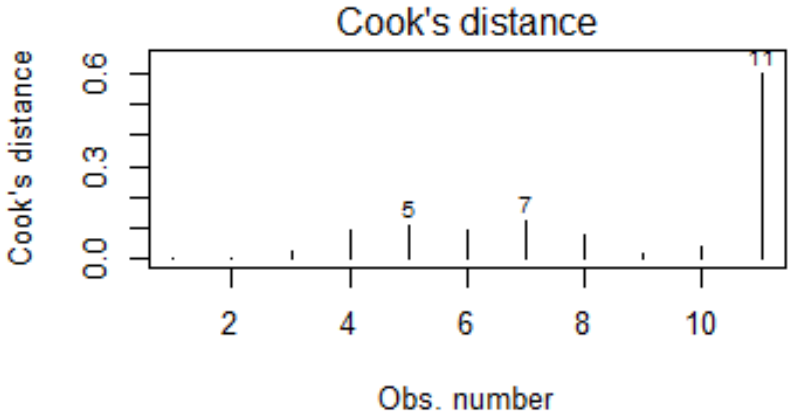


Рисунок 17. Відстані Кука (приклад)

ОТРИМАНІ РЕЗУЛЬТАТИ. Опишіть отримані в ході роботи результати, наведіть розроблені фрагменти кодів, програм, функцій для реалізації досліджень викидів.

ВИСНОВКИ. Зробіть основні висновки по роботі. Починайте з короткого нагадування мети роботи, потім переходьте до основних результатів. Уникайте простого переказу того, що робили - зосередьтеся на тому, що дізналися та які закономірності виявили. Завершіть висновки пропозиціями щодо того, як можна було б покращити методику експерименту або отримати більш точні результати.

Пам'ятайте: висновки мають демонструвати ваше критичне мислення та розуміння матеріалу, а не просто констатувати факти.

ЛАБОРАТОРНА РОБОТА №5. «Пошук асоціативних правил в середовищі аналізу даних».

МЕТА РОБОТИ. Метою лабораторної роботи є формування професійних вмінь та навичок щодо використання апріорного алгоритму (A-priori) для обчислення частих наборів елементів (товарів) аналізу інформаційних впливів, вміння застосовувати отримані знання на практиці в практичних задачах аналізу великих даних.

ТЕОРІЯ. Теоретичні питання щодо інструментів та методів побудов асоціативних правил детально розглянуті в лекційних матеріалах за темою «Пошук асоціативних правил».

Пошук правил асоціації – це метод машинного навчання для виявлення цікавих зв’язків між змінними у великих базах даних та аналізу впливів. Він призначений для виявлення сильних правил у базі даних на основі деяких показників, наприклад, однакових наборів покупок, колективів авторів, впливів одних даних на інші тощо [12].

Для будь-якої заданої транзакції з кількома елементами правила асоціації спрямовані на отримання правил, які визначають, як і чому певні елементи пов’язані. Правила асоціації в основному використовуються для аналізу та прогнозування поведінки клієнтів.

Асоціативні правила також можуть бути потужним інструментом для виявлення прихованих патернів у розповсюдженні та взаємодії інформаційних впливів. Застосовуючи алгоритми на кшталт Apriori або FP-Growth до даних соціальних мереж, новинних стрічок чи месенджерів, аналітики можуть виявляти, які теми, наративи або меседжі часто з'являються разом у певних часових проміжках або інформаційних потоках. Наприклад, правило "якщо з'являється тема A, то з ймовірністю 85% з'явиться й тема B протягом наступних 24 годин" може вказувати на координовану інформаційну кампанію або штучно створену асоціацію між подіями.

Особливо цінним є застосування асоціативних правил для розпізнавання шаблонів дезінформації та маніпулятивних технік. Аналізуючи великі масиви текстових даних, можна виявити стійкі асоціації між емоційно забарвленими словами, певними джерелами інформації та конкретними наративами. Якщо система виявляє, що певні фейкові новини систематично супроводжуються специфічними риторичними прийомами або з'являються в певних комбінаціях медіа-каналів, це дозволяє швидше ідентифікувати та протидіяти майбутнім інформаційним атакам. Метрики підтримки (support) та достовірності (confidence) правил допомагають оцінити масштаб і систематичність інформаційного впливу.

Крім того, асоціативні правила дають змогу відстежувати еволюцію інформаційних кампаній у часі. Порівнюючи набори правил, згенерованих для різних часових періодів, дослідники можуть побачити, як змінюються тактики впливу, які нові асоціації створюються пропагандистами, та які теми поступово "прив'язуються" одна до одної у свідомості аудиторії. Це особливо актуально для аналізу гібридних загроз, де інформаційний вплив є довготривалим і багаторівневим процесом. Такий підхід дозволяє не лише реагувати на поточні загрози, а й прогнозувати майбутні напрямки інформаційних операцій.

Вивчення правил асоціації є однією з дуже важливих концепцій машинного навчання, і воно використовується для аналізу ринкового кошика, аналізу використання Інтернету, безперервного виробництва тощо. У цьому випадку аналіз ринкового кошика є технікою, яку використовують різні великі роздрібні торговці для виявлення асоціацій між товарами [13].

Вивчення правила асоціації працює на основі концепції операторів If і Else, наприклад, **if A then B.**Тут елемент If називається антецедентом, а оператор **then** називається консиквентом (Consequent). Ці типи зв’язків, за допомогою яких ми можемо виявити певний зв’язок або відношення між двома елементами, відомі як одна кардинальність.

Вся справа в створенні правил, і якщо кількість елементів збільшується, відповідно збільшується і їх кількість. Отже, для вимірювання зв’язків між тисячами елементів даних існує кілька показників. Ці показники наведено нижче [14]:

- Підтримка
- Впевненість
- Ліфт

Підтримка — це частота або те, як часто елемент X з’являється в наборі даних. Вона визначається як частка транзакції T_X, яка містить набір елементів X в загальній кількості транзакцій T. Якщо є X наборів даних, то для транзакцій T це можна записати так (9):

$$Supp(X) = \frac{T_X}{T} \tag{9}$$

Достовірність (впевненість). Достовірність показує, як часто правило виявляється істинним. Або як часто елементи X і Y зустрічаються разом у наборі даних, коли X уже вказано. Це відношення транзакції, яка містить X і Y, до кількості записів, які містять X (10).

$$\text{Confidence} = \frac{T_{X,Y}}{T_X} \tag{10}$$

Lift. Це сила будь-якого правила, яке можна визначити як співвідношення спостережуваної міри підтримки та очікуваної підтримки, якщо X і Y незалежні один від одного (11).

$$\text{Lift} = \frac{Supp(X,Y)}{Supp(X) * Supp(Y)} \quad (11)$$

Ліфт – це міра, яка враховує статистичні залежності. Наприклад, якщо у вас є правило асоціації $S_i \Rightarrow S_j$, **lift** вимірює кореляцію між наборами елементів S_iі S_j, тобто наскільки тісно пов'язані два набори в загальній кількості N товарів (12).

$$lift = \frac{P(S_j|S_i)}{P(S_j)},\ P(S_j) = \frac{S_j}{N} \quad (12)$$

Показник має три можливі значення:

Якщо Lift = 1: ймовірність появи попереднього та наступного предмету не залежить одне від одного.

Lift >1: визначає ступінь залежності двох наборів елементів один від одного (позитивна кореляція).

Lift<1: це говорить нам, що один предмет замінює інші предмети, що означає, що один предмет негативно впливає на інший.

Існуючі алгоритми:

1. Апріорний алгоритм: Цей алгоритм використовує часті набори даних для створення правил асоціації. Він призначений для роботи з базами даних, які містять транзакції. Цей алгоритм використовує пошук у ширину та хеш-дерево для ефективного обчислення набору елементів. Він в основному використовується для аналізу ринкового кошика та допомагає зрозуміти продукти, які можна купити разом. Його також можна використовувати в галузі охорони здоров'я для визначення реакції пацієнтів на ліки [15].
2. Алгоритм Eclat: Алгоритм Eclat розшифровується як перетворення класу еквівалентності (Equivalence Class Transformation). Цей алгоритм використовує метод пошуку в глибину для пошуку частих наборів елементів у базі даних транзакцій. Він виконується швидше, ніж алгоритм Apriori [16].
3. Алгоритм зростання F-P: Алгоритм зростання F-P розшифровується як Frequent Pattern і є вдосконаленою версією алгоритму Apriori. Він представляє базу даних у формі деревовидної структури, яка відома як шаблон частоти або дерево. Метою цього дерева частот є виділення найбільш частого шаблону.

Apriori — це алгоритм для пошуку частих наборів елементів і вивчення правил асоціації в реляційних базах даних.

Він працює шляхом ідентифікації частих окремих елементів у базі даних і розширення їх до більших і більших наборів елементів, якщо ці набори елементів досить часто з'являються в базі даних. Часова та просторова складність цього алгоритму дуже висока: $O(2\hat{}d)$.

Ключовою концепцією в апріорному алгоритмі є наступне:

1. По-перше, якщо набір елементів є частим, то всі його підмножини також мають бути частими.

2. По-друге, підтримка набору елементів ніколи не перевищує підтримку його підмножини.

3. Також досить цікаво, що наш типовий поріг підтримки s, становитиме 1% від кошика.

ВХІДНІ ДАНІ.

Набір даних Market Basket складається з 15010 спостережень із датою, часом, функцією транзакції та елементом або стовпцями. Змінні дані в стовпцях варіюються від 30/10/2016 до 09/04/2017. Час — це категоріальна змінна, яка вказує час. Транзакція — це кількісна змінна, яка допомагає диференціювати транзакції. Товар — це категоріальна змінна, яка пов'язана з продуктом.

Можна використовувати набір даних "Groceries" («Продовольчі товари»), який містить приблизно 9835 транзакцій, які включають «n» кількість товарів, які були куплені разом у магазині. Набори завантажені у відповідний електроний ресурс навчальної дисципліни.

Також додатково можна розглянути набори даних рекламних кампанії або зібрати декілька наративів пропагандистського характеру з точки зору аналізу інформаційних впливів. Аналізуючи великі масиви текстових даних, можна виявити стійкі асоціації між емоційно забарвленими словами, певними джерелами інформації та конкретними наративами.

НЕОБХІДНІ БІБЛІОТЕКИ ТА ПАКЕТИ.

install.packages("arules"), library(arules), install.packages("arulesViz"), library(arulesViz), library(RColorBrewer), data("Groceries")

ХІД РОБОТИ:

1. Завантажте необхідні бібліотеки
2. Завантажте набір даних '**Market_Basket_Optimisation.csv**' (навчальний) або "Groceries". Опишіть цей набір даних: кількість записів, які дані в ньому збережені тощо.

dataset = read.transactions('Market_Basket_Optimisation.csv', sep = ',', rm.duplicates = TRUE).

У разі аналізу соціальних мереж зточки зору інформаційних впливів, детально опишіть джерела інформації, їх періодичність, обсяг, вашу оцінкумети публікації.

3. Використайте функцію **apriori()**. Опишіть параметри функції та можливі налаштування.

4. Застосуйте функцію **inspect()**. Опишіть параметри функції та можливі налаштування. Опишіть отримані асоціації (рис.18).

```
     lhs                                    rhs             support     confidence coverage   lift     count
[1]  {light cream}                       => {chicken}       0.004532729 0.2905983  0.01559792 4.843951 34
[2]  {pasta}                             => {escalope}      0.005865885 0.3728814  0.01573124 4.700812 44
[3]  {pasta}                             => {shrimp}        0.005065991 0.3220339  0.01573124 4.506672 38
[4]  {eggs, ground beef}                 => {herb & pepper} 0.004132782 0.2066667  0.01999733 4.178455 31
[5]  {whole wheat pasta}                 => {olive oil}     0.007998933 0.2714932  0.02946274 4.122410 60
[6]  {herb & pepper, spaghetti}          => {ground beef}   0.006399147 0.3934426  0.01626450 4.004360 48
[7]  {herb & pepper, mineral water}      => {ground beef}   0.006665778 0.3906250  0.01706439 3.975683 50
[8]  {tomato sauce}                      => {ground beef}   0.005332622 0.3773585  0.01413145 3.840659 40
[9]  {mushroom cream sauce}              => {escalope}      0.005732569 0.3006993  0.01906412 3.790833 43
[10] {frozen vegetables, mineral water, spaghetti} => {ground beef}   0.004399413 0.3666667  0.01199840 3.731841 33
```

Рисунок 18. Приклад застосування функції **inspect()**.

5. Зробіть візуалізацію отриманих результатів щодо підтримки різних продуктів, використовуючи функцію itemFrequencyPlot(). Опишіть параметри налаштування функції (рис.19).

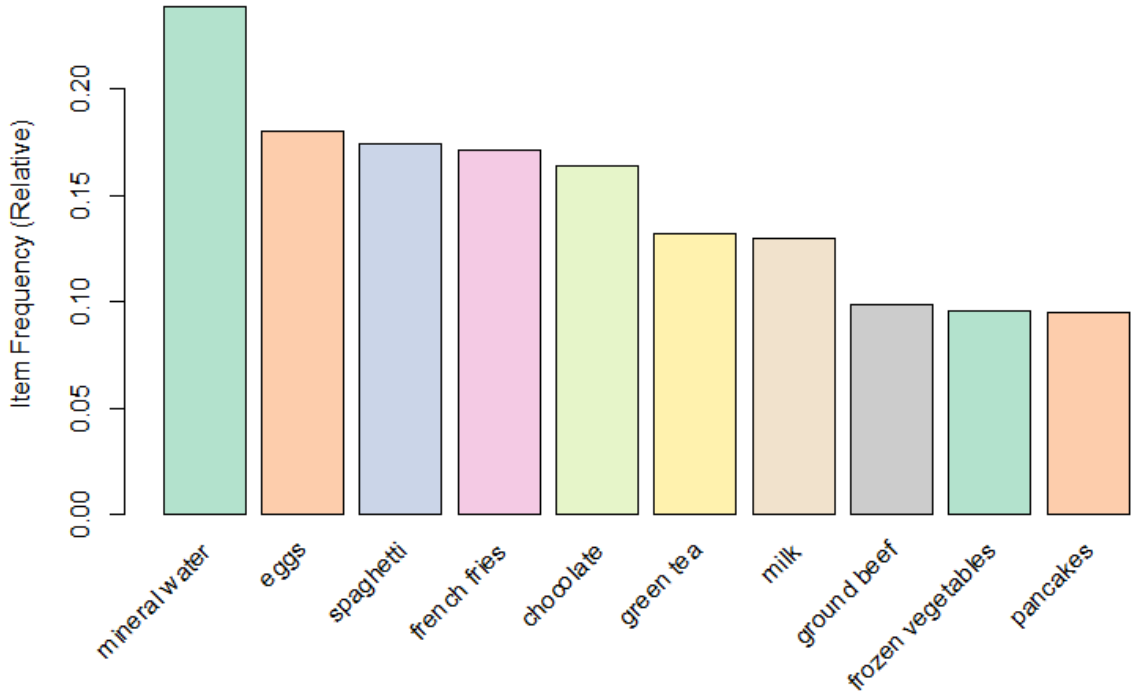


Рисунок 19. Приклад візуалізації частоти покупок (набір Market_Basket)

6. Побудуйте візуалізацію правил за допомогою функції (рис.20):

```
plot(associa_rules, method = "graph",
    measure = "confidence", shading = "lift")
```

Рисунок 20. Приклад візуалізації правил.

7. Якщо ви аналізуєте інформаційні впливи, оцініть чи певні фейкові новини систематично супроводжуються специфічними риторичними прийомами або з'являються в певних комбінаціях медіа-каналів, це дозволить швидше ідентифікувати та протидіяти майбутнім інформаційним атакам. Детально проаналізуйте метрики підтримки (support) та достовірності (confidence) отриманих правил та оцініть масштаб і систематичність інформаційного впливу.

ОТРИМАНІ РЕЗУЛЬТАТИ. Опишіть отримані в ході роботи результати, наведіть розроблені фрагменти кодів, програм, функцій для реалізації досліджень викидів.

ВИСНОВКИ. Зробіть основні висновки по роботі. Починайте з короткого нагадування мети роботи, потім переходьте до основних результатів. Уникайте простого переказу того, що робили - зосередьтеся на тому, що дізналися та які закономірності виявили. Завершіть висновки пропозиціями щодо того, як можна було б покращити методику експерименту або отримати більш точні результати.

Пам'ятайте: висновки мають демонструвати ваше критичне мислення та розуміння матеріалу, а не просто констатувати факти.

ІІ. Рекомендації та завдання для виконання практичних робіт

Практичне заняття — це форма навчального процесу, що передбачає детальний розгляд теоретичних аспектів навчальної дисципліни та формування вмінь і навичок їх практичного застосування шляхом виконання спеціально сформульованих завдань.

Головною метою практичного заняття є:

- Опанування студентами навчальної дисципліни.
- Забезпечення глибокого аналізу та обговорення основних проблем курсу.
- Навчання елементам творчого застосування отриманих знань на практиці.

Практичні заняття виконують кілька важливих функцій [18]:

1. Навчальна: поглиблення та систематизація знань, засвоєних під час лекцій.
2. Розвивальна: розвиток логічного мислення, умінь працювати з літературними джерелами.
3. Виховна: виховання відповідальності, культури спілкування та інтересу до дисципліни.
4. Діагностично-корекційна: контроль за якістю засвоєння матеріалу та виявлення прогалин у знаннях.

Структура практичного заняття:

1. Підготовка:

- Визначення теми і мети заняття.
- Ознайомлення студентів з теоретичним матеріалом.
- Підготовка необхідних ресурсів і обладнання.

2. Виконання завдання:

- Відповіді на питання вхідного контролю.
- Проведення практичної роботи в індивідуальному або груповому форматі.
- Дотримання техніки безпеки.

3. Аналіз результатів:

- Оформлення результатів роботи.
- Обговорення отриманих даних та формулювання висновків.

Практичні заняття є невід'ємною частиною навчального процесу, що сприяє розвитку професійних навичок і підготовці студентів до майбутньої діяльності.

Структура всіх практичних робіт побудована за шаблоном (Додаток 2), який передбачає систематизацію отриманих даних та результатів і написання висновків.

ПРАКТИЧНА РОБОТА №1. «Класифікаційні правила та дерева рішень».

МЕТА. Метою практичної роботи є формування професійних вмінь та навичок щодо використання інструментів (методів, бібліотек, пакетів програмного забезпечення R) для побудови дерев рішень (класифікації), більш глибоке дослідження питань класифікації, розглянутий протягом лекції, вміння застосовувати отримані знання на практиці в практичних задачах аналізу даних.

ТЕОРІЯ. Дерево рішень - це граф, який представляє варіанти та результати рішень (подій) у вигляді дерева, або дозволяє здійснити класифікацію сутностей, які досліджуються. Вузли на графу представляють подію чи вибір, а гілки графа представляють правила чи умови прийняття рішення. В основному використовується у додатках машинного навчання та інтелектуального аналізу даних в задачах прикладної аналітики [19].

Дерева рішень — це універсальний алгоритм машинного навчання, який може виконувати завдання класифікації, і регресії. Це дуже потужні алгоритми, здатні проаналізувати дуже складні набори даних. Крім того, дерева рішень є фундаментальними компонентами випадкових лісів, які є одними із найпотужніших інструментів машинного навчання.

Термінологія дерев рішень [20].

Кореневий вузол (Root Node**)**: кореневий вузол дерева рішень, який представляє початковий вибір або функцію, від якої дерево розгалужується, є найвищим вузлом.

Внутрішні вузли (вузли прийняття рішень) (**Internal Nodes (Decision Nodes**)): вузли в дереві, вибір яких визначається значеннями конкретних атрибутів. На цих вузлах є гілки, які йдуть до інших вузлів.

Листові вузли (кінцеві вузли) (**Leaf Nodes (Terminal Nodes)**): кінцеві точки гілок, коли приймаються рішення або прогнози. Відгалужень на вузлах листя більше немає.

Гілки (грані) (**Edges**): зв’язки між вузлами, які показують, як приймаються рішення у відповідь на певні обставини.

Поділ (Splitting): процес поділу вузла на два або більше вузлів на основі критерію прийняття рішення. Він передбачає вибір функції та порогового значення для створення підмножин даних.

Батьківський вузол (Parent Node): вузол, який розділений на дочірні вузли. Вихідний вузол, з якого походить поділ.

Дочірній вузол (Child Node): вузли, створені в результаті відокремлення від батьківського вузла.

Критерій прийняття рішення (Decision Criterion): правило або умова, що використовується для визначення того, як дані повинні бути розділені на вузлі прийняття рішення. Він передбачає порівняння значень ознак із порогом.

Відсікання (Pruning): процес видалення гілок або вузлів із дерева рішень для покращення його узагальнення та запобігання переобладнанню.

Дерево рішень (рис.21) використовує подання дерева для вирішення проблеми (пошуку рішення), у якій кожен листовий вузол відповідає мітці класу, а атрибути представлені у внутрішньому вузлі дерева. Можливо представити будь-яку булеву функцію на дискретних атрибутах за допомогою дерева рішень.

На початку вважаємо весь навчальний набір коренем. Бажано, щоб значення ознак були категоріальними. Якщо значення неперервні, то вони дискретизуються перед побудовою моделі.

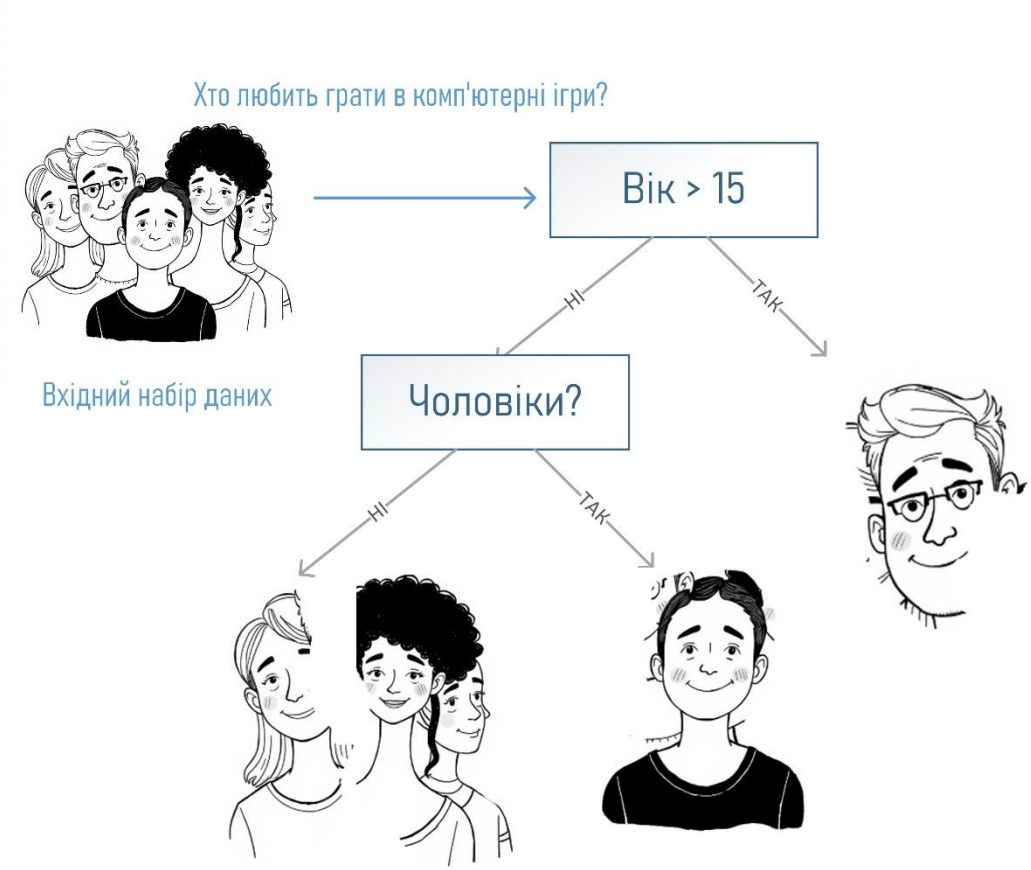


Рисунок 21. Дерево рішень

На основі значень атрибутів записи розподіляються рекурсивно. Використовуємо статистичні методи для впорядкування атрибутів як кореневого або внутрішнього вузла.

Методи вибору атрибутів.

1. Приріст інформації (**Information Gain**).

Коли використовуємо вузол у дереві рішень для поділу екземплярів навчання на менші підмножини, ентропія змінюється. Приріст інформації є мірою цієї зміни ентропії (13).

$$Entropy(S) = -\sum_{c \in C} p(c) \log_2 p(c) \tag{13}$$

S представляє набір даних, для якого обчислюється ентропія;

c представляє класи в наборі S;

p(c) представляє частку точок даних, які належать до класу c, до загальної кількості точок даних у наборі, S

2. Домішка Джіні (Gini Index).

Домішка Джіні — це ймовірність неправильної класифікації випадкових точок даних у наборі даних, якщо їх було позначено на основі розподілу класів у наборі даних. Подібно до ентропії, якщо встановлено, що C є чистим, тобто належить до одного класу, то його домішка дорівнює нулю. Це позначається такою формулою (14):

$$G_i = 1 - \sum_{k=1}^{n} p_{i,c}^2 \tag{14}$$

$p_{i,c}$ - частка вибірок класу ***c*** серед навчальних вибірок в і-му вузлі.

Деякі додаткові функції та характеристики індексу Джині:

1) Він обчислюється шляхом підсумовування квадратів ймовірностей кожного результату в розподілі та віднімання результату від 1.
2) Нижчий індекс Джині вказує на більш однорідний або чистий розподіл, тоді як вищий індекс Джині вказує на більш неоднорідний або нечистий розподіл.
3) У деревах рішень індекс Джині використовується для оцінки якості розбиття шляхом вимірювання різниці між домішкою батьківського вузла та зваженою домішкою дочірніх вузлів.

4) Порівняно з іншими мірами домішок, такими як ентропія, індекс Джині обчислюється швидше та є більш чутливим до змін у ймовірностях класу.

5) Одним із недоліків індексу Джині є те, що він надає перевагу розбиттям, які створюють дочірні вузли однакового розміру, навіть якщо вони не є оптимальними для точності класифікації.

6) На практиці вибір між використанням індексу Джині чи інших мір домішок залежить від конкретної проблеми та набору даних і часто вимагає експериментів і налаштування.

ВХІДНІ ДАНІ.

В роботі використовується вбудований набір даних R з іменем readingSkills для створення дерева рішень. Він описує оцінку чиєїсь навички читання, якщо ми знаємо змінні «вік», «розмір взуття», «оцінка» і те, чи є людина носієм мови чи ні. Набір не є реальним дослідженням, а лише абстрактний набір даних для легкого засвоєння методики побудови дерев рішень.

Для виконання практичної роботи необхідно обрати інший навчальний набір даний дещо більшого розміру і описати його на початку роботи.

НЕОБХІДНІ БІБЛІОТЕКИ ТА ПАКЕТИ:

library(dplyr), library(rparty), library(rparty.plot).

Основний синтаксис для створення дерева рішень у R: ctree(formula, data), де formula - це формула, що описує предиктор та змінні відповіді; Data - ім'я набору даних, що використовується.

ПЛАН ПРОВЕДЕННЯ ЗАНЯТТЯ:

1. Відповіді на вхідні питання.
2. Проведення досліджень гіперпараметрів функції побудови дерева рішень в середовищі R. Отримання більш глибоких знань щодо методу побудови дерева рішень.
3. Підготовка звіту за результатами дослідження.
4. Підготовка домашнього завдання.
5. Захист практичної роботи.

ПИТАННЯ ДЛЯ ВХІДНОГО КОНТРОЛЮ:

1. Яка основна мета побудови дерев рішень?
2. Що таке дерево рішень і в яких задачах воно застосовується?

3. Які основні компоненти дерева рішень?
4. Що таке кореневий вузол, і яку роль він відіграє?
5. Чим відрізняються внутрішні вузли від листових?
6. Що таке гілки (грані) у дереві рішень?
7. Які функції виконує батьківський та дочірній вузли?
8. Що таке поділ (Splitting) у дереві рішень?
9. Як працює критерій прийняття рішення?
10. Що таке відсікання і навіщо воно потрібне?
11. Чому важливо дискретизувати неперервні атрибути перед побудовою дерева?
12. Що таке приріст інформації (Information Gain) і як він розраховується?
13. Що таке ентропія, і яку роль вона відіграє у визначенні приросту інформації?
14. Як розраховується домішка Джині (Gini Index)?
15. Чому індекс Джині може бути більш чутливим до змін у ймовірностях класу?
16. Які переваги та недоліки використання індексу Джині?

ХІД ПРОВЕДЕННЯ ЗАНЯТТЯ:

1. Інсталюйте необхідні для роботи з деревами пакети та бібліотеки. Ви також повинні встановити залежні пакети, якщо такі є.

install.packages("party"), library(party)

Пакет «party» має функцію ctree(), яка використовується для створення та аналізу дерева рішень.

1. Завантажте бібліотеку «party». Прочитайте заголовки змінних у наборі даних readingSkills, визначте і опишіть їх формат. За допомогою довідки опишіть набір даних, що означають змінні, скільки досліджень включено до набору. Чи є це реальний набір, чи тільки навчальний (рис.22).

```
'data.frame':   200 obs. of  4 variables:
 $ nativeSpeaker: Factor w/ 2 levels "no","yes": 2 2 1 2 2 2 1 2 2 1
 $ age          : int  5 6 11 7 11 10 7 11 5 7 ...
 $ shoeSize     : num  24.8 26 30.4 28.7 31.9 ...
 $ score        : num  32.3 36.6 49.6 40.3 55.5 ...
```

Рисунок 22. Характеристики набору даних readingSkills

Опис змінних та їх формат:

Опишіть всі змінні набору, їх типи, ваші рекомендації щодо їх попередньої обробки.

2. За допомогою функції (help ___ або ?___) виведіть довідку по всі параметрам функції ctree(). Здійсніть дослідження параметрів та опишіть їх

властивості. Функція ctree() використовується для побудови умовного дерева рішень. Ключові параметри:

controls: об'єкт класу TreeControl, створений за допомогою функції ctree_control. Визначає параметри, що впливають на процес побудови дерева.

Опис функції ctree_control ():

3. Побудуйте дерево за допомогою функції ctree() (рис.23).

```
model_1 <- ctree()
plot(model_1)
```

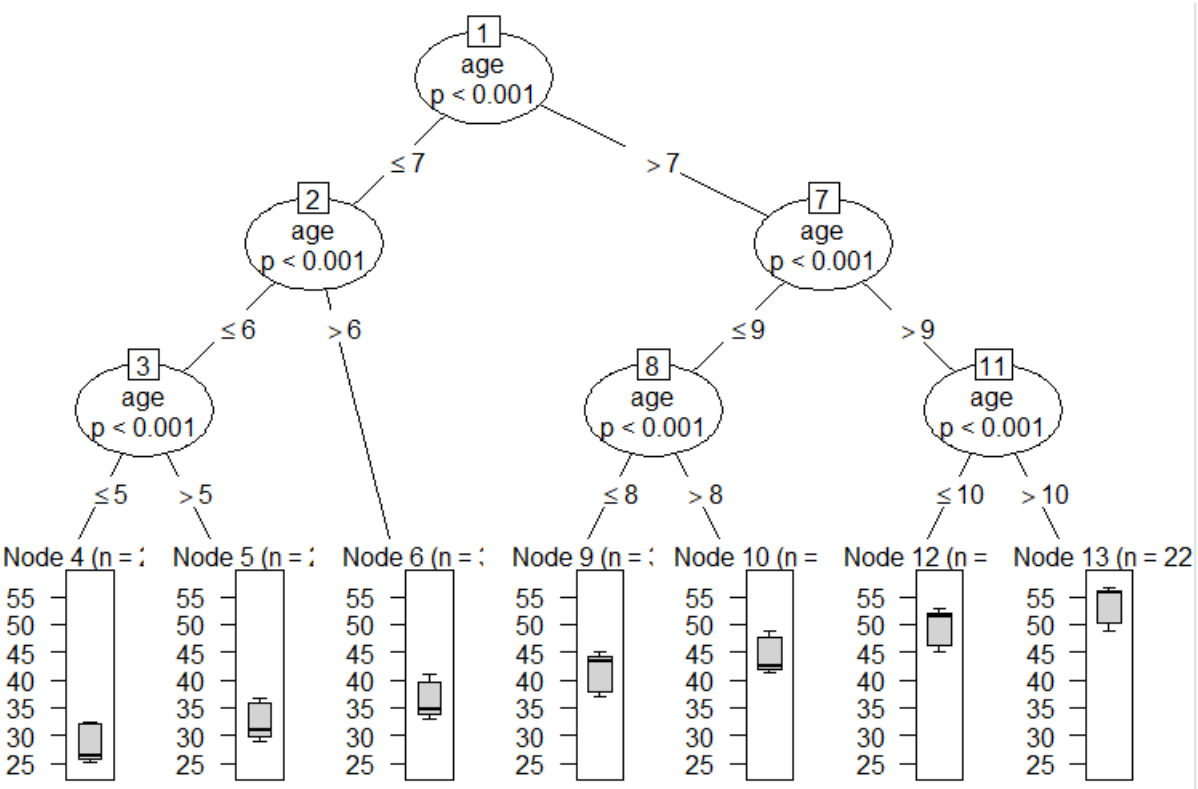


Рисунок 23. Приклад дерева рішень для навчального набору

4. Змініть параметри функції ctree() в частині опису формули та додаткових параметрів, опишіть отримані зміни.

```
model_2 <- ctree( ……., control = ctree_control(          ))
model_3 <- ctree( ……., control = ctree_control(          ))
```

ВИСНОВКИ. Зробіть висновки за отриманими результатами моделювання, які фактори є значущими для отримання максимальної кількості балів з мовного тесту. Чи знайшли ви якісь цікаві залежності. Опишіть, які практичні навички ви набули або закріпили. Це можуть бути технічні вміння (робота з програмним забезпеченням, обладнанням), методологічні підходи або аналітичні здібності.

ДОМАШНЄ ЗАВДАННЯ. Ознайомитися з додатковою літературою за тематикою практичного завдання з метою розширення знань з питань побудови дерев рішень. Знайдіть відповіді на питання вхідного контролю, перевіривши свої знання та підготуйте звіт з роботи.

ПРАКТИЧНА РОБОТА №2. «Огляд наукових публікацій з методів інтелектуального аналізу даних та впливів».

МЕТА. Метою практичної роботи є розширення знань щодо методів та інструментів інтелектуального аналізу даних, із застосовуванням сучасних сервісів штучного інтелекту для пошуку нових сучасних наукових методів та підходів в галузі Data Mining та інформаційних впливів, оброблення отриманих даних та підготовку висновків та звітів.

ТЕОРІЯ:

Наукові дослідження відіграють ключову роль у розвитку суспільства, оскільки сприяють відкриттю нових знань, удосконаленню технологій та розв'язанню глобальних проблем. Завдяки науці ми отримуємо розуміння природи, медицини, економіки та багатьох інших сфер, що безпосередньо впливає на якість життя людей. Однак із розвитком науки обсяг інформації стрімко зростає, що створює виклик: як швидко та ефективно знаходити необхідні дані серед величезного масиву знань.

Саме тут на допомогу приходять інструменти штучного інтелекту, які здатні аналізувати великі обсяги інформації, знаходити закономірності та допомагати у формуванні висновків. Вони дозволяють автоматизувати рутинні процеси, прискорюють пошук наукових публікацій і навіть можуть прогнозувати нові відкриття. Використання таких технологій стає необхідною навичкою для дослідників, студентів і всіх, хто прагне ефективно працювати з інформацією. Інтеграція штучного інтелекту у наукову діяльність не лише підвищує продуктивність, а й відкриває нові горизонти для досліджень, роблячи науку доступнішою та більш динамічною. Особливої уваги потребують дослідження методів та інструментів аналізу інформаційних впливів, які стали потужним інструментом в галузі формування суспільної думки.

Розглянемо найбільш цікаві ресурси, які допоможуть вам у ваших дослідженнях. Інформація з питання використання різноманітних інструментів ШІ в сфері наукових досліджень детально розглянуті та представлена на каналі [21].

Існує багато інструментів штучного інтелекту (ШІ), які допомагають у пошуку, аналізі та управлінні науковими статтями. Ось основні категорії та приклади таких інструментів:

1. Пошук наукових статей (AI-powered search engines).

Ці інструменти використовують ШІ для знаходження релевантних публікацій на основі семантичного аналізу (рис.24):

- **Semantic Scholar** – використовує NLP (Natural Language Processing) для пошуку статей, оцінює цитованість і значущість робіт.
- **Google Scholar** – базовий, але ефективний інструмент для пошуку наукових статей.
- **Connected Papers** – будує граф зв'язаних статей, допомагаючи знайти нові дослідження за ключовими роботами.
- **Elicit** – ШІ-асистент для наукових досліджень, який може відповідати на питання, аналізуючи статті.
- **Scite.ai** – показує, як стаття була процитована (підтвердження, спростування, аналіз).

---

2. Аналіз та систематизація наукових статей.

Інструменти для збору, організації та аналізу інформації (рис.25):

- **Zotero** – безкоштовний менеджер бібліографії з можливістю інтеграції з браузерами.
- **Mendeley** – керування бібліографією, пошук статей та інструменти для спільної роботи.
- **Paperpile** – менеджер бібліографії для Google Docs із можливістю роботи з PDF.
- **ResearchRabbit** – створює взаємопов'язані графи статей, допомагаючи дослідникам бачити зв'язки між роботами.

---

3. Генерація огляду літератури (Literature Review AI Tools)

Автоматизація збору ключові ідеї з літератури (рис.26):

- **Litmaps** – створює карти літератури, які допомагають бачити, як статті пов'язані між собою.
- **Iris.ai** – використовує ШІ для аналізу наукових статей і створення оглядів літератури.
- **R Discovery** – ШІ-інструмент, що аналізує нові наукові статті та допомагає дослідникам залишатися в курсі.

---

4. Генерація та перевірка гіпотез.

ШІ може допомагати у формулюванні та перевірці гіпотез:

- **Elicit** – допомагає знаходити підтримку чи спростування гіпотез у наукових статтях.
- **Consensus** – відповідає на запитання, базуючись на наукових статтях.

---

5. Написання та редагування наукових статей.

Покращити якість тексту або перевірити наукову англійську можливо з:

- **Grammarly** – перевіряє граматику та стиль.
- **Hemingway Editor** – допомагає зробити науковий текст чіткішим.
- **SciSpace Copilot** – пояснює складні терміни в наукових статтях.

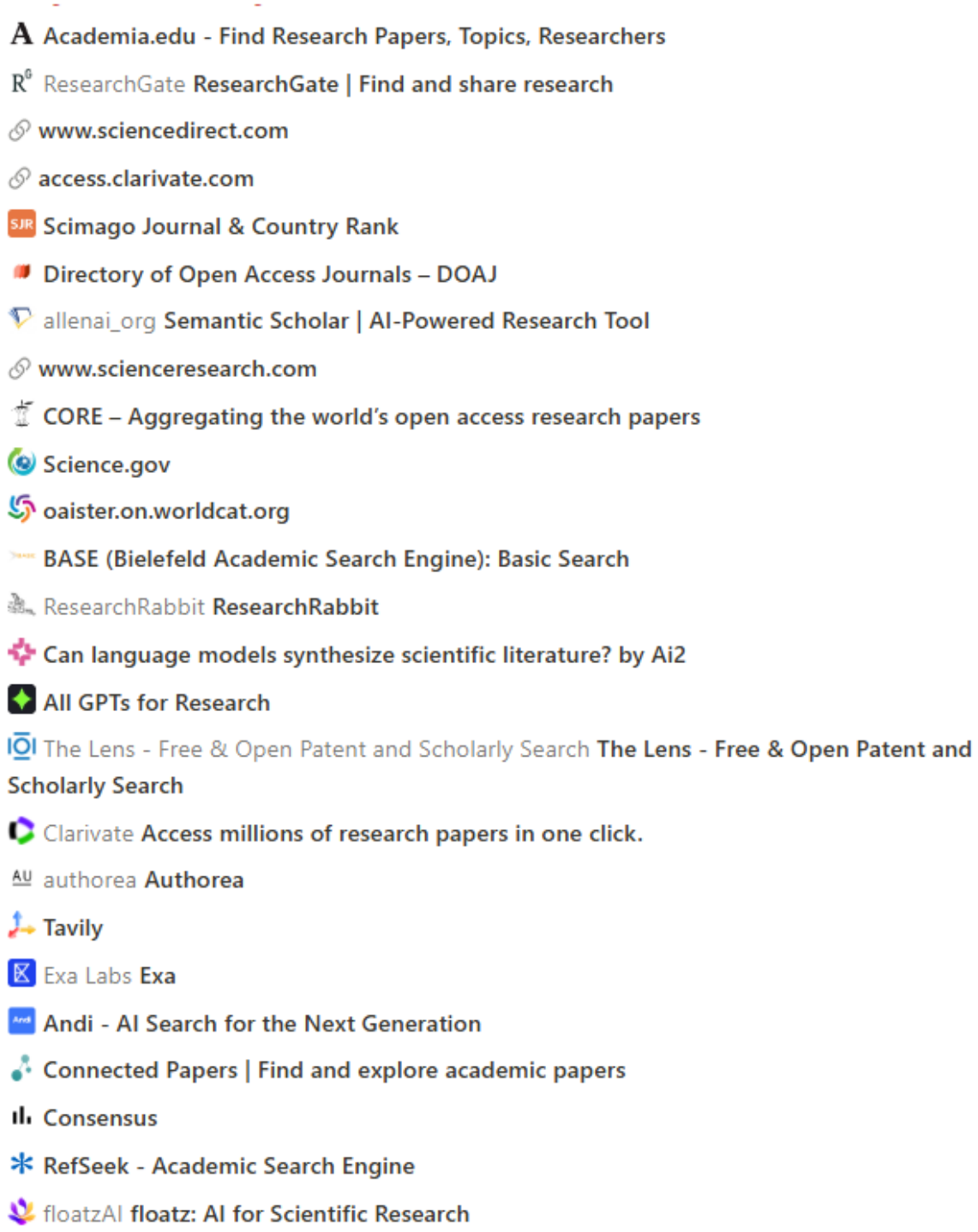


Рисунок 24. Ресурси ШІ для пошуку наукових публікацій та роботи з ними

6. Пошук відкритого доступу до статей.

Якщо стаття платна, можна перевірити, чи є вона у відкритому доступі:

- **Unpaywall** – знаходить безкоштовні версії статей.
- **Open Access Button** – шукає відкритий доступ до статей або допомагає запитати їх у авторів.

Якщо потрібен швидкий пошук **наукових статей (рис.25)**, спробуйте **Semantic Scholar, Google Scholar, Elicit**. Якщо потрібно **керувати бібліографією**, використовуйте **Zotero або Mendeley**. Якщо потрібно **генерувати огляд літератури**, скористайтеся **Litmaps або Iris.ai**.

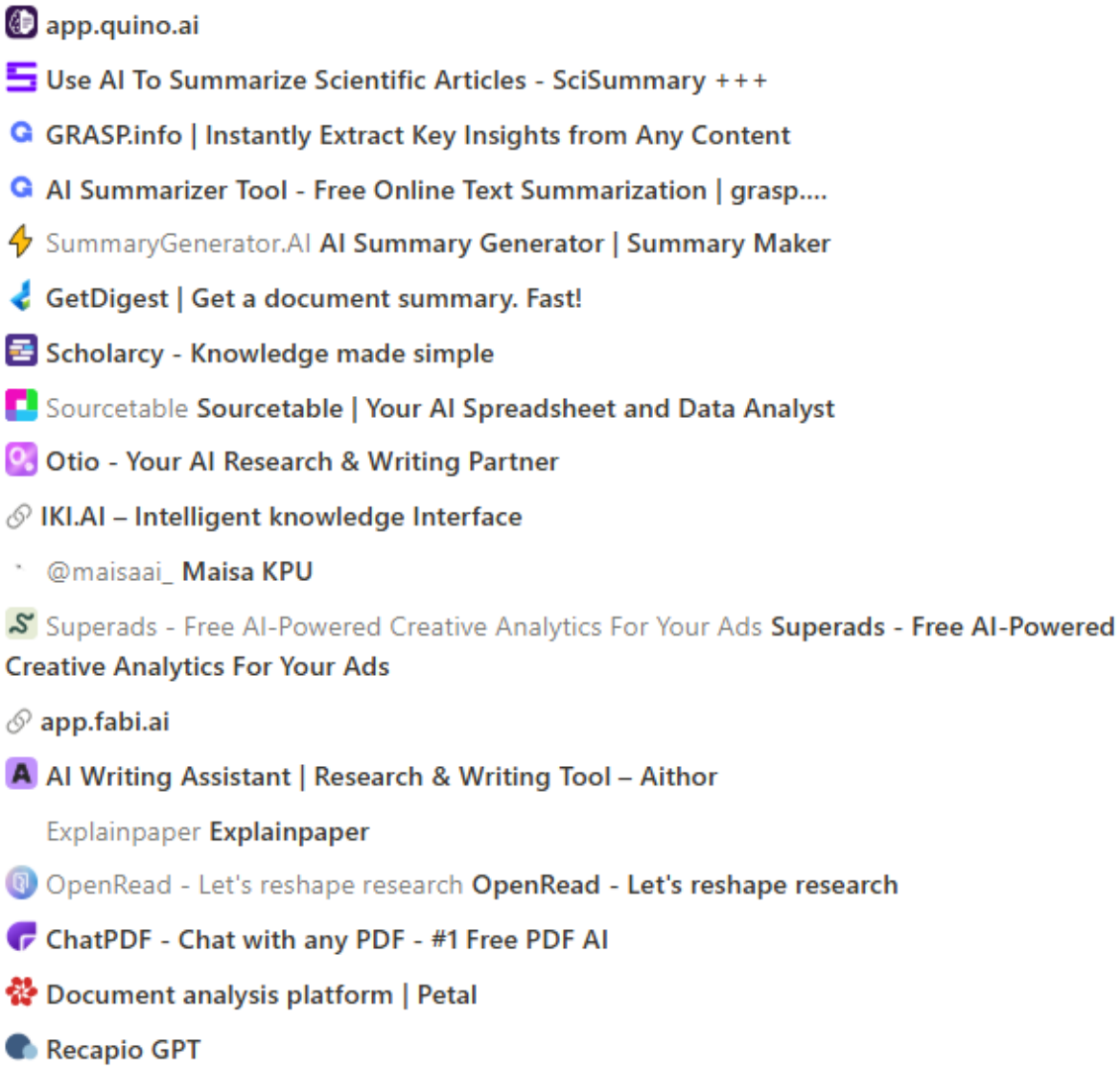


Рисунок 25. Помічник ШІ в аналізі документів

Якщо потрібні ШІ-інструменти для написання наукових статей, найкращі варіанти за категоріями можуть бути представлені:

1. Генерація та структуризація наукового тексту.

Ці ШІ-інструменти допомагають формулювати тези, абзаци, секції статті:

- **SciSpace (раніше Typeset.io)** – допомагає писати статті у форматах журналів, автоматично форматуючи текст відповідно до вимог.
- **Jenni AI** – асистент для написання академічних текстів, який пропонує підказки та завершує речення.
- **Trinka AI** – спеціалізований редактор для академічного письма з перевіркою стилю, граматики та формату.
- **Scholarcy** – аналізує наукові тексти, створює короткі резюме, допомагає у структуризації статті.
- **QuillBot** – допомагає перефразовувати, узагальнювати та покращувати текст.

---

2. Перевірка граматики та стилю академічного письма

Для перевірки мови, стилю та точності тексту:

- **Grammarly** – покращує граматику, стиль і чіткість тексту.
- **Hemingway Editor** – допомагає зробити текст більш зрозумілим і академічним.
- **Writefull** – спеціально створений для академічного письма, перевіряє формулювання та термінологію.
- **Trinka AI** – спеціалізована для академічних текстів, виправляє термінологічні помилки та стиль.

---

3. Автоматичне створення бібліографії та цитувань

Для швидкого форматування списку літератури:

- **Zotero** – безкоштовний менеджер бібліографії.
- **Mendeley** – керування бібліографією та автоматичне створення цитувань.
- **EndNote** – потужний інструмент для бібліографії (платний).
- **Paperpile** – інтегрується з Google Docs для автоматичного цитування.

---

4. Генерація наукових резюме та аналіз статей

Якщо потрібно швидко отримати короткий виклад статті:

- **Elicit** – AI-асистент для пошуку та аналізу наукових статей.
- **Scholarcy** – створює автоматичні реферати та аналізує структуру статті.
- **Litmaps** – допомагає створювати карти літератури для огляду.

---

5. Переклад та спрощення складних наукових текстів.

Якщо потрібно пояснити складні терміни чи адаптувати текст:

- **DeepL** – один із найкращих інструментів для точного перекладу академічних текстів.
- **SciSpace Copilot** – пояснює складні терміни в наукових статтях.
- **Explainpaper** – розшифровує складні статті простими словами.

---

6. Перевірка наукових статей на плагіат

Для перевірки унікальності тексту перед поданням у журнал:

- **Turnitin** – стандарт у перевірці плагіату для академічних текстів.
- **Grammarly Plagiarism Checker** – перевіряє текст на схожість із іншими публікаціями.
- **Plagscan** – перевірка унікальності академічних документів.

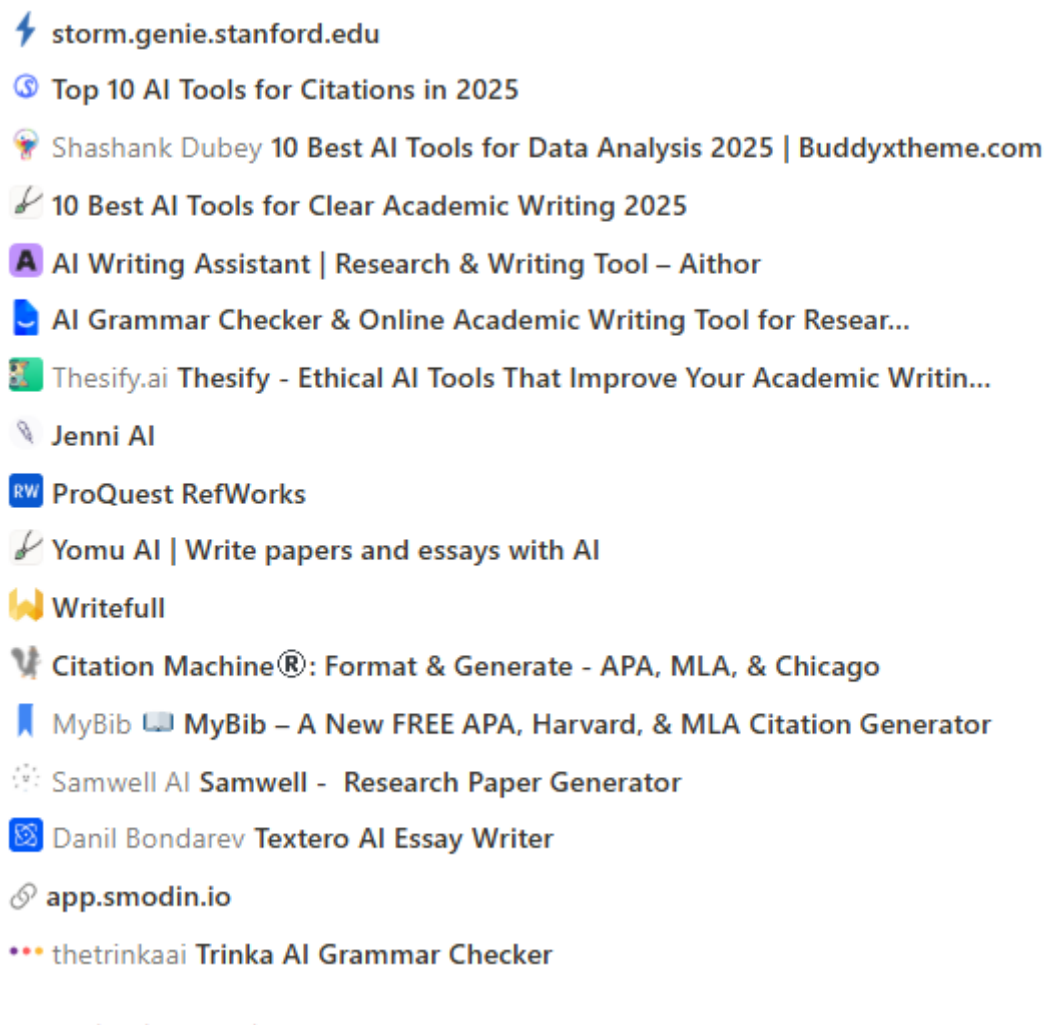


Рисунок 26. Ресурси ШІ для написання статей

Який ШІ-інструмент вибрати?

- Потрібно писати науковий текст? → SciSpace, Jenni AI, Trinka AI.

- Перевірка стилю та граматики? → Grammarly, Writefull, Trinka AI.
- Бібліографія та цитування? → Zotero, Mendeley, Paperpile.
- Резюме статей? → Scholarcy, Elicit.
- Переклад складних термінів? → DeepL, SciSpace Copilot.
- Перевірка плагіату? → Turnitin, Grammarly Plagiarism Checker.

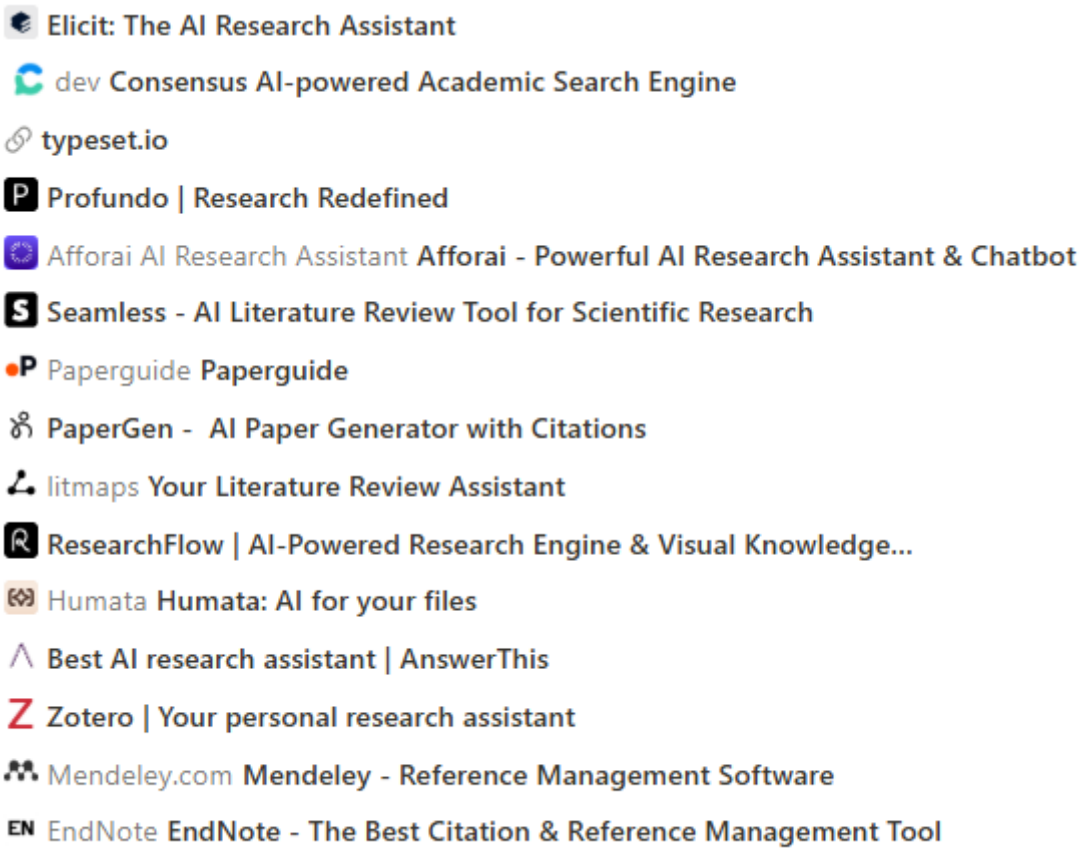


Рисунок 27. Спеціалізовані ресурси ШІ для організації досліджень

ПЛАН ПРОВЕДЕННЯ ЗАНЯТТЯ:

1.Відповіді на вхідні питання.
2. Проведення досліджень сучасних наукових публікацій з методів інтелектуального аналізу даних з використанням інструментів та ресурсів ШІ, широко представлених у теоретичній частині практичної роботи.
3. Підготовка звіту за результатами дослідження.
4. Презентація отриманих результатів, обговорення сучасних трендів в інтелектуальному аналізі даних.

ПИТАННЯ ДЛЯ ВХІДНОГО КОНТРОЛЮ:

1. Що таке аналіз даних?

2. Основні етапи data mining.
3. Прикладні галузі застосування data mining.
4. Технології data mining.
5. Програмне забезпечення та інструменти інтелектуального аналізу даних.
6. Моделі методології CRISP-DM.

ХІД ПРОВЕДЕННЯ ЗАНЯТТЯ:

Опрацювати сучасні наукові публікації за тематикою методів інтелектуального аналізу даних з використанням існуючих та доступних засобів штучного інтелекту, виявити основні тренди розвитку методів та інформаційних систем в галузі інтелектуального аналізу даних.

ВИСНОВКИ

Зробіть висновка щодо трендів розвитку як теорії так і програмного забезпечення в галузі інтелектуального аналізу даних. Обґрунтуйте, де в майбутній професійній діяльності можуть знадобитися отримані навички та знання. Це допомагає усвідомити цінність виконаної роботи.

ДОМАШНЄ ЗАВДАННЯ.

Зробіть додаткове дослідження ресурси, які дозволяють аналізувати інформаційні впливи, фейки в соціальних мережах та медіа. Підготуйте презентацію та підготуйтеся до захисту результатів роботи. Презентація повинна включати результати досліджень основних трендів в інтелектуальному аналізі даних та обов’язково опис дослідженого вами під час самостійної роботи ресурсу ШІ, який призначений допомогти науковцям в їх дослідженнях (формування гіпотез, аналіз статей за тематикою дослідження тощо), а також у виявлені інформаційних впливів.

ПРАКТИЧНА РОБОТА №3. «Базові засади аналізу текстів та систем рекомендацій контенту».

МЕТА. Метою практичної роботи є формування професійних вмінь та навичок щодо базових засад інтелектуального аналізу текстів [22], вміння застосовувати отримані знання на практиці в практичних задачах інтелектуального аналізу даних та інформаційних впливів.

ТЕОРІЯ.

Інтелектуальний аналіз текстів (text mining) став незамінним інструментом для виявлення та дослідження інформаційних впливів у сучасному цифровому середовищі. Використовуючи методи обробки природної мови (NLP), аналітики можуть автоматично обробляти величезні обсяги текстової інформації з соціальних мереж, новинних сайтів, форумів та месенджерів, виявляючи ключові наративи, тональність повідомлень та емоційне забарвлення контенту. Технології sentiment analysis дозволяють відстежувати, як змінюється суспільна думка під впливом цілеспрямованих інформаційних кампаній, виявляючи сплески негативу чи позитиву навколо певних тем, що часто свідчить про координовані дії з формування громадської думки.

Особливо ефективним є застосування методів topic modeling, таких як LDA (Latent Dirichlet Allocation) або BERTopic, для виявлення прихованих тематичних структур у великих текстових корпусах. Ці методи дозволяють автоматично ідентифікувати основні теми дискусій, відстежувати їх динаміку та виявляти моменти, коли певні наративи штучно впроваджуються або посилюються. Аналіз поширення специфічних лінгвістичних конструкцій, фразеологізмів та ключових слів допомагає встановити джерела інформаційного впливу та картографувати шляхи розповсюдження меседжів через різні канали комунікації. Це особливо важливо для виявлення ботів та тролів, які часто використовують однотипні мовні патерни та шаблонні фрази.

Сучасні технології глибокого навчання та трансформерні моделі (BERT, GPT та їх варіанти) відкривають нові можливості для детекції складних форм маніпуляції, включаючи виявлення deepfake-текстів, автоматично згенерованого контенту та тонких форм пропаганди. Named Entity Recognition (NER) дозволяє відстежувати, як згадуються конкретні персони, організації чи події в різних контекстах, виявляючи спроби дискредитації або героїзації. Аналіз семантичних мереж та word embeddings показує, які асоціації створюються між поняттями у публічному дискурсі, що є критично важливим для розуміння механізмів когнітивного впливу. Інтеграція цих методів з

часовим аналізом дає змогу не лише фіксувати поточний стан інформаційного простору, а й прогнозувати розвиток майбутніх інформаційних кампаній, забезпечуючи проактивний підхід до протидії інформаційним загрозам.

Протягом практичної роботи будуть розглянуті базові поняття та визначення інтелектуального аналізу даних. Оброблені (tidy) «акуратні» дані мають певну структуру:

- ❖ Кожна змінна є стовпцем.
- ❖ Кожне спостереження є рядком.
- ❖ Кожен вид одиниці спостереження являє собою таблицю.

Таким чином, ми визначаємо оброблений текстовий формат як таблицю з одним маркером (токеном) на рядок (**a table with one-token-per-row**).

Токен — це значуща одиниця тексту, наприклад слово, яку ми хочемо використовувати для аналізу, а токенізація — це процес поділу тексту на токени. Ця структура «один маркер на рядок» залежить від того, як текст часто зберігається в поточному аналізі, можливо, у вигляді рядків або в матриці термінів документа.

Для акуратного видобутку (mining) тексту токен, який зберігається в кожному рядку, найчастіше є окремим словом, але також може бути n-грамою, реченням або абзацом. У пакеті «tidytext» ми реалізовуємо функціональність для токенізації часто використовуваних одиниць тексту, і перетворюємо у формат «один терм на рядок».

Оброблені (tidy) набори даних дозволяють маніпулювати стандартним набором «tidy» інструментів, включаючи такі популярні пакети, як dplyr (Wickham and Francois 2016), tidyr (Wickham 2016), ggplot2 (Wickham 2009) і broom (Robinson 2017).

Структурування текстових даних таким чином означає, що вони відповідають принципам акуратності даних і ними можна маніпулювати за допомогою набору узгоджених інструментів.

❖ **Рядок (String)**: Текст можна, звичайно, зберігати як рядки, тобто вектори символів, у R, і часто текстові дані спочатку зчитуються в пам’ять у цій формі.

❖ **Корпус (Corpus)**: ці типи об’єктів зазвичай містять необроблені рядки, анотовані додатковими метаданими та деталями.

❖ **Матриця документ-термін** (**Document-term matrix**): це розріджена матриця, що описує набір (тобто корпус) документів з одним рядком для кожного документа та одним стовпцем для кожного терміна. Значення в матриці зазвичай є кількістю слів або tf-idf.

Важливість слова є фактором при інтелектуальному аналізі (data mining) колекції документів.

Перша концепція розглядає слова, які найкраще характеризують документ у колекції - Inverse Document Frequency, IDF (15).

$$IDF_w = \log_2\left(\frac{N}{n_w}\right) \tag{15}$$

w – слово;

N – загальна кількість документів у колекції;

n_w - кількість документів, в яких зустрічаємо слово w.

Друга концепція розглядає важливість слова у документі, а потім і в колекції - **term frequency** of word w in documentary D - TF (16).

$$TF_{\{w,D\}} = \frac{f\{w,D\}}{f\{D,\ max\}} \tag{16}$$

$f\{w,D\}$ - кількість разів зустрічі слова w в документі D;

$f\{D,\ max\}$ - максимальна кількість разів зустрічі будь-якого слова в документі D.

Щоб пов'язати важливість слова в документі D і загальної колекції об'єднують два показники шляхом їх множення (17):

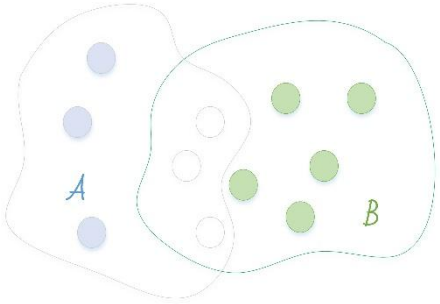
$$TF.IDF_{\{w,D\}} = TF_{\{w,D\}} \cdot IDF_w \tag{17}$$

Подібність Жаккара множин A та B визначається таким рівнянням (18) [23]:

$$SIM(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{18}$$

і її можна позначити SIM(A,B).

Подібність Жаккара описує відношення розміру перетинів множин A та B до розміру їх об'єднання.



Подібність між двома документами можна виміряти, обчисливши косинус подібності між їхніми векторними представленнями (19).

$$CosSim(A,B) = \frac{\sum_{i=1}^{n} A_i \cdot B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \tag{19}$$

Документи можна представити як набір векторів у векторному просторі, де кожному терміну відповідає одна вісь. Для кожного слова в документі використовується хеш-функція, яка зіставляє його з цілочисельним індексом. Після цього документ можна перетворити на набір векторів, що представляють слова в документі. Векторне представлення документів не містить порядку інформації - воно відображає, скільки разів слово з'являється

в документі. При цьому втрачається порядок слів. У більшості випадків векторного представлення достатньо, щоб передати зміст документа, незважаючи на те, що порядок інформації втрачено.

Якщо отримано косинусоїдальну подібність +1, два досліджувані документи можна вважати однаковими. Однак, зверніть увагу, що ці два документи не є фактично однаковими, але схожими на основі обраних термінів. Косинусоїдальна подібність -1 вказує на два вектори, які є діаметрально протилежними, тобто документи за абсолютно різною тематикою.

ПЛАН ПРОВЕДЕННЯ ЗАНЯТТЯ:

1.Відповіді на питання вхідного контролю.

2. Проведення досліджень невеликих текстів українських класиків або інших творів на вибір. Виявлення схожості текстів, підрахунок частоти використання слів.
3. Підготовка звіту за результатами дослідження.
4. Підготовка домашнього завдання.
5. Захист домашнього завдання.

ПИТАННЯ ДЛЯ ВХІДНОГО КОНТРОЛЮ:

1. Важливість слова. Показники Inverse Document Frequency, IDF та term frequency.
2. Хеш-функції. Хеш-функції.
3. Індекси.
4. Що таке схожі сутності і як їх ідентифікувати?
5. Подібність документів.
6. Представлення документів у вигляді наборів символів.
7. Побудова shingle наборів на основі стоп слів.
8. Матричне представлення множин.
9. MinHash та сігнатурна матриці.
10. Матриця рейтингів (вподобань).
11. Системи Спільної фільтрації.
12. Системи рекомендацій на основі змісту.
13. Ефективність рекомендаційних систем.
14. Інтелектуальний аналіз текстів у задачах виявлення інформаційних впливів.

НЕОБХІДНІ БІБЛІОТЕКИ ТА ПАКЕТИ:

library(tidytext), library(dplyr), library(ggplot2)

ХІД ПРОВЕДЕННЯ ЗАНЯТТЯ:

1. Перший варіант. Завантажте в середовище аналізу два невеликих твори українських письменників, наприклад, Л.Українка та Т.Шевченко. Можна виконувати завантаження з ресурсів, а також можливо створити окрему змінну, в яку завантажити рядки наприклад віршів, на більше 500 строк.

```
# To be or not to be?.. Л.Українка
Text_1 <- c("Стій, серце, стій! не бийся так шалено.",
            "Вгамуйся, думко, не літай так буйно!",
            "Не бий крильми в порожньому просторі.",
            " ……")
```

Другий варіант. Завантажте корпус пропагандистських текстів з метою виявлення найбільш частих патернів (слів, виразів). Опишіть детально джерела та зміст, на що направлений інформаційний вплив на вашу думку.

2. Завантажте необхідні бібліотеки.
3. Для перетворення вихідного тексту в акуратний (tidy) набір даних (data set), його необхідно помістити у фрейм даних (data frame) за допомогою функції tibble().
4. Здійсніть токенізацію завантажених текстів за допомогою функції unnest_tokens().
5. Проаналізуйте завантажені тексти та створіть власний список стоп слів (рис.28), наприклад,

```
mystopwords <- tibble(word = c("не", "і", "з", "в", "де", "на", "ті", "тільки",
                               "а", "що", "он", "лиш", "нема", "їх", "все", "так",
                               "геть", "який", "о", "туди", "й", "до", "усе", "коли",
                               "куди", "як", "для", "чи", "щоб", "у", "сі", "та",
                               "по", "бо", "то"))
```

Рисунок 28. Власний словник стоп слів

6. Очистіть завантажені та оброблені (токенізовані) тексти від стоп слів з використанням функції anti_join().
7. Порахуйте частоту вживання кожного слова та побудуйте діаграми (рис.29).
8. Порахуйте показник **term frequency** кожного слова (токену) (рис.30).

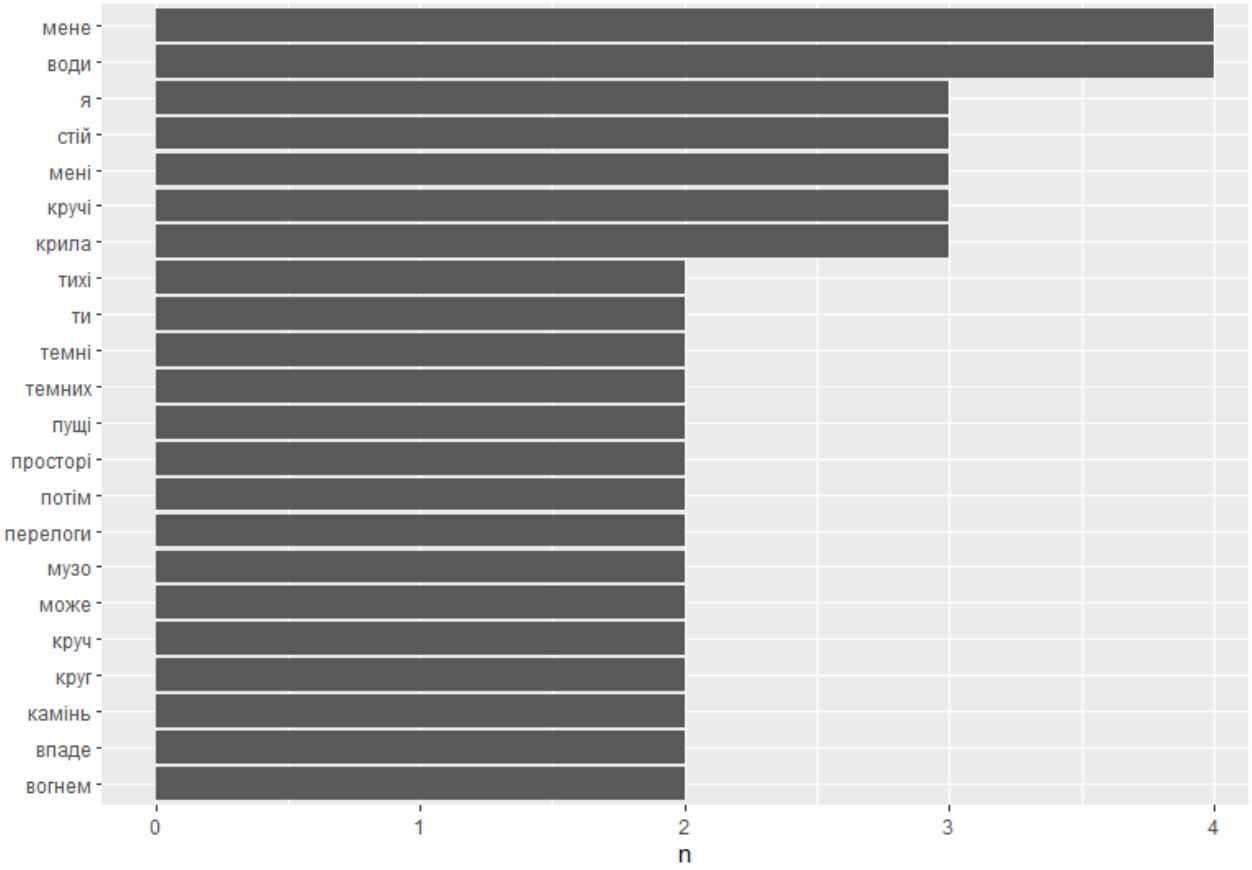


Рисунок 29. Підрахунок кількості слів у тексті

	word	n	tf	fr
26	безсмертних	1	0.25	0.004629630
27	бий	1	0.25	0.004629630
28	бийся	1	0.25	0.004629630
29	блискавицю	1	0.25	0.004629630
30	борозну	1	0.25	0.004629630
31	буде	1	0.25	0.004629630
32	буйно	1	0.25	0.004629630
33	була	1	0.25	0.004629630
34	вгамуйся	1	0.25	0.004629630
35	великий	1	0.25	0.004629630

Рисунок 30. Приклад розрахунку показника **term frequency**

9. Побудуйте функцію для розрахунку подібності Жаккара двох текстів.
10. Порівняйте схожість завантажених вами текстів. У разі аналізу інформаційного впливу визначте ключовий характерний для даних текстів патерн.

ВИСНОВКИ. Коротко перерахуйте основні завдання, які були виконані під час практичної роботи. Не переписуйте весь хід роботи, а виділіть ключові етапи та досягнуті результати. Опишіть, які практичні навички ви набули або закріпили. Це можуть бути технічні вміння (робота з програмним забезпеченням, обладнанням), методологічні підходи або аналітичні здібності.

Якщо під час роботи ви помітили певні закономірності, тенденції або особливості, обов'язково їх опишіть. Це демонструє ваше аналітичне мислення.

Обґрунтуйте, де в майбутній професійній діяльності можуть знадобитися отримані навички та знання. Це допомагає усвідомити цінність виконаної роботи.

ДОМАШНЄ ЗАВДАННЯ.

Оскільки протягом теоретичної частини за темою «Базові засади аналізу текстів та системи рекомендацій контенту» розглядаються лише основи та фундаментальні поняття інтелектуального аналізу текстів, необхідно протягом самостійної роботи більш детально ознайомитися з лінгвістичним підходом щодо оброблення природної мови та основними етапами Text mining за літературою з відкритих джерел. Підготувати посилання на 3-4 сучасних джерела із зазначеної тематики.

Оформлення звіту та підготовка до захисту.

ПРАКТИЧНА РОБОТА №4. «Google web search. Аналіз соціальних мереж».

МЕТА. Метою практичної роботи є формування професійних вмінь та навичок щодо використання інструментів та методів аналізу моделей соціальних мереж з використанням теорії графів, концепцію жорстко пов’язаних компонентів мережі, основи web пошуку та ранжування сторінок PageRank, параметри мертвих вузлів та пасток у web пошуку та вміння застосовувати отримані знання на практиці в практичних задачах аналізу великих даних.

ТЕОРІЯ.

Аналіз соціальних мереж (Social Network Analysis, SNA) надає унікальні можливості для розуміння структури та динаміки розповсюдження інформаційних впливів через людські зв'язки та комунікаційні канали. Побудова графів взаємодій між користувачами, групами та каналами дозволяє виявляти ключових акторів інформаційного впливу - тих, хто ініціює та розповсюджує певні наративи. Метрики центральності (**degree centrality, betweenness centrality, eigenvector centrality**) допомагають ідентифікувати найвпливовіших користувачів, хабові вузли та "брокерів інформації", які з'єднують різні спільноти. Це особливо важливо для розуміння того, як інформація перетікає з маргінальних груп у мейнстрімні медіа, або як іноземні актори можуть впливати на внутрішній інформаційний простір через мережу локальних "агентів впливу".

Виявлення спільнот (community detection) та кластерний аналіз дозволяють картографувати інформаційні екосистеми, визначаючи ізольовані "ехо-камери" та "фільтрувальні бульбашки", де користувачі споживають однобічну інформацію. Аналіз каскадів поширення контенту показує, як швидко та широко розповсюджуються певні повідомлення, виявляючи аномально швидкі або масштабні кампанії, які можуть свідчити про використання ботів або координованої неавтентичної поведінки. Темпоральний аналіз мережевої динаміки розкриває, як формуються нові зв'язки навколо певних тем, як активізуються раніше неактивні акаунти під час інформаційних операцій, та як змінюється топологія мережі під впливом зовнішніх подій. Інтегруючи SNA з контент-аналізом, дослідники можуть не лише бачити, хто і з ким спілкується, а й розуміти, які саме меседжі циркулюють у різних сегментах мережі, що дає комплексне уявлення про механізми та масштаби інформаційного впливу в цифровому просторі.

Звичайні способи моделювання Інтернету або соціальної мережі включають побудову [24, 25]:

- випадкових графів (**random networks**);
- безмасштабних графів (**scale-free network**);
- графу малого світу (**small world graph**) .

Випадковий граф.

Припускаємо, що в графі є n вузлів, які з'єднані ребром з імовірністю p. Ступінь графу відповідає розподілу Пуассона (20).

$$P(x = k) = \frac{\lambda^k}{k!} e^{-\lambda}, k \in N \quad (20)$$

1. Події відбуваються незалежно.
2. Ймовірність того, що певна подія відбудеться протягом фіксованого проміжку часу, залишається незмінною протягом часу.

За цими умовами зв’язки у випадковому графі додаються незалежно та випадковим чином (рис.31). Це демонструє ключові властивості випадкових мереж:

- Випадковий характер з'єднань.
- Відсутність чіткої ієрархії.
- Приблизно однакова кількість зв'язків для кожного вузла.
- Можливість досягнути будь-якого вузла через певну кількість кроків.

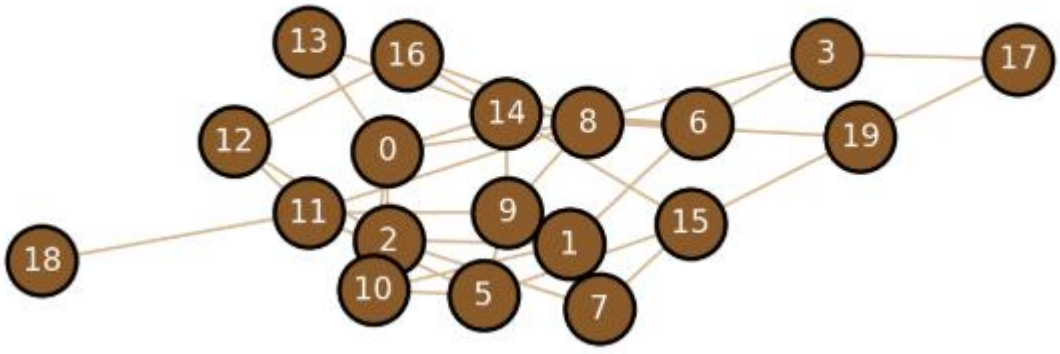


Рисунок 31. Приклад випадкової мережі (графу).

Як формується безмасштабна мережа? Розглянемо мережу, де нові дуги приєднуються до вузлів з імовірністю, пов’язаною з кількістю зв’язків, які він вже має (рис.32). Це ймовірність того, що новий вузол з’єднається з даним існуючим вузлом, пропорційна k, тобто ступеню вузла.

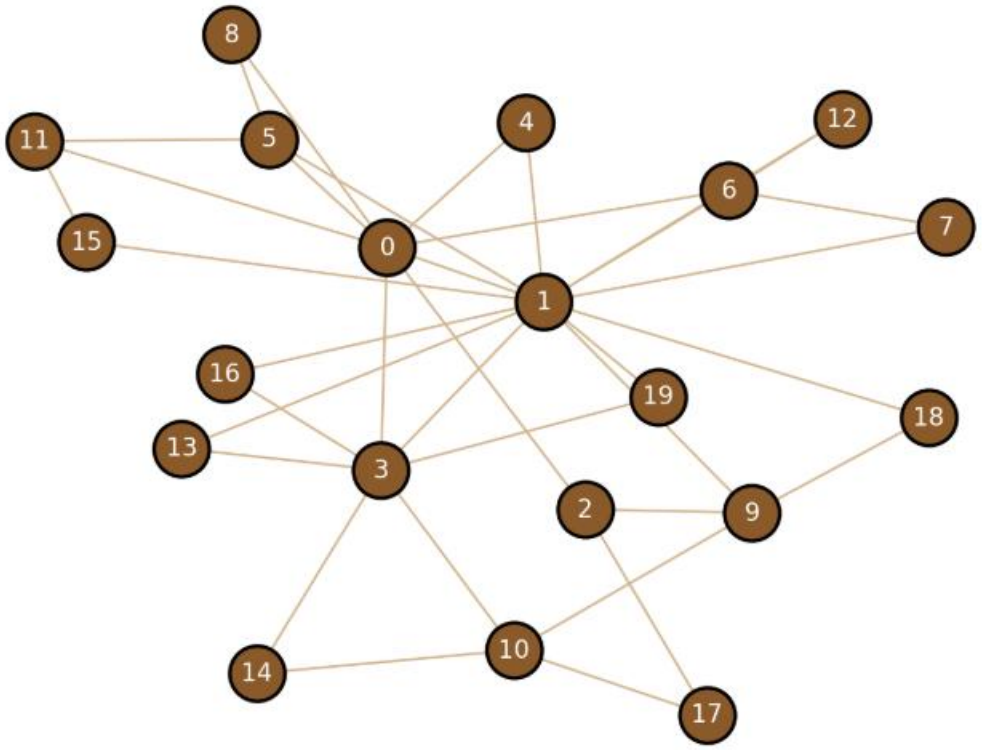


Рисунок 32. Приклад безмасштабної мережі.

Така структура відображає ключову властивість безмасштабних мереж - степеневий розподіл ступенів вершин, де небагато вузлів мають багато зв'язків, а більшість вузлів мають мало зв'язків. Граф безмасштабної мережі (модель Барабаші-Альберт) відрізняється тим, що деякі вузли мають набагато більше зв’язків, ніж інші, формуючи "вузли-хаби". Це часто зустрічається в реальних мережах, таких як Інтернет або соціальні мережі.

У цій моделі мережа стає безмасштабною. Цей тип мережі стійкий до мережевих збоїв, оскільки ймовірність того, що вузол із високим ступенем буде видалено, низька.

Графи малого світу мають такі властивості (рис.33):

1) Вони сильно кластеризовані.
2) Розподіл ступенів слідує розподілу Пуассона, як випадковий граф.

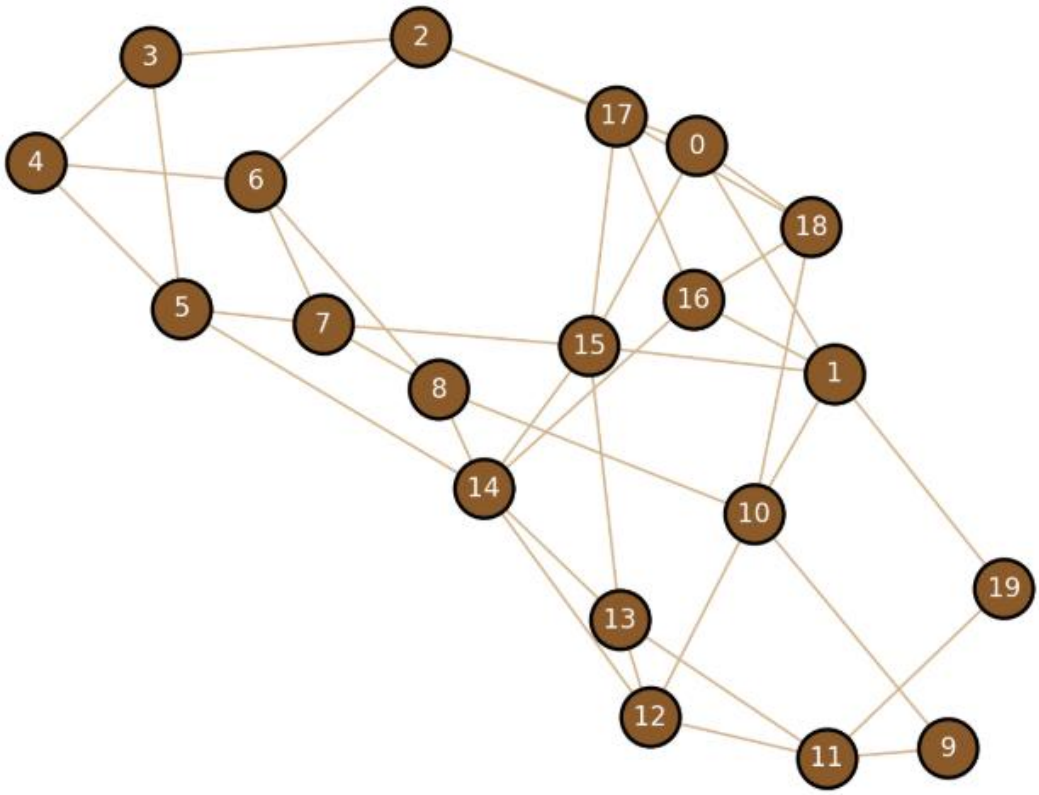


Рисунок 33. Приклад графа small world graph.

Граф демонструє високу кластеризацію та короткі шляхи між вузлами — ключові властивості моделей Уоттса-Строгаца.

Під час веб-сканування пошуковий сканер або павук використовується для створення списків слів, знайдених на веб-сторінках. Щоб створити такі списки, веб-сканер повинен переглядати багато сторінок, і список потрібно постійно оновлювати, оскільки веб-сайти постійно змінюються.

Інформація, зібрана веб-сканерами, обробляється пошуковою системою для індексування завантажених сторінок для ефективного пошуку в Інтернеті.

Веб-сканер дотримується набору визначених політик [26].

Політика відбору. Через величезний розмір пошукові системи охоплюють не всю мережу. Навіть велика пошукова система проіндексує лише приблизно 70% веб-сайтів, які можна індексувати. Необхідна політика вибору, яка визначає, які сторінки завантажувати, оскільки найбільш релевантні сторінки для веб-пошуку представляють більший інтерес.

Правила повторного відвідування. Мережа має динамічний характер, оскільки вона постійно змінюється. Веб-сканування – це трудомісткий процес, який може зайняти тижні або місяці залежно від ряду факторів, у тому числі вашої політики вибору. Під час процесу сканування ви можете очікувати, що

відбулися зміни. Зазвичай вартість оцінюється за невиявленням зміни. Найпоширенішими функціями витрат є «свіжість» і вік. Свіжість описує точність локальної копії. Вік визначає, наскільки застаріла локальна копія.

Політика «ввічливості». Сканери мають високу продуктивність у отриманні даних із веб-сайтів. Таким чином, вони можуть мати великий вплив на продуктивність веб-сайту. Використання веб-сканерів може призвести до перевантаження сервера, на якому розміщено веб-сайт, збою сервера/маршрутизатора або збою в роботі мережі/сервера. Необхідно визначити, який розділ Інтернету можна сканувати, щоб гарантувати, що це не вплине на продуктивність веб-сайту.

Політика розпаралелювання. Деякі сканери можуть запускати кілька процесів паралельно. Тому для кожного сканера слід встановити правила, щоб уникнути багаторазового завантаження одного й того самого вмісту.

PageRank можна розглядати як функцію, яка призначає дійсне число кожній сторінці (вірогідність опинитися на цій сторінці).

Процес починається в одному із станів s_i і послідовно переходить від одного стану до іншого.

Ймовірність переходу від стану s_i до s_j визначається як p_{ij} і вона не залежить від стану, в якому перебуває ланцюжок (21):

$$M_T = \begin{bmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{n1} & \cdots & p_{nn} \end{bmatrix}$$
$$V_k = M_T \times V_{k-1} \qquad (21)$$
$$V_0 = \left[\frac{1}{n}, \frac{1}{n}, \ldots \frac{1}{n}\right]^T$$

Ймовірність p_{ij} називається **ймовірністю переходу**.

Головний власний вектор V_k можна обчислити, починаючи з початкового вектора V_0 та множачи на матрицю переходів T_G кілька разів, доки V_k не покаже незначних змін на кожному етапі. На практиці достатньо 50-75 ітерацій для web. Сталі значення V_k і будуть характеризувати вірогідність опинитися на певній сторінці.

ПЛАН ПРОВЕДЕННЯ ЗАНЯТТЯ:

1.Відповіді на питання вхідного контролю знань.

2. Проведення досліджень невеликих моделей соціальних мереж. Виявлення особливості мережі (наявність пасток, мертвих вузлів).
3. Підготовка звіту за результатами дослідження.
4. Захист отриманих результатів досліджень.

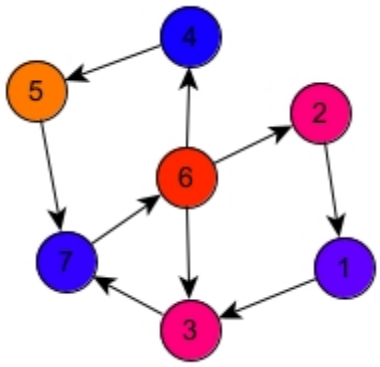
ПИТАННЯ ДЛЯ ВХІДНОГО КОНТРОЛЮ:

1. Соціальна мережа як граф.
2. Графи з кількома типами вузлів.
3. Small-world Effect. Small world graph.
4. Кластеризація соціальних мереж.
5. Нерівномірний (перекошений) розподіл ступенів.
6. Моделювання зв’язків між об’єктами в соціальній мережі .
7. Random networks. Scale-free network.
8. Степеневий закон (закон масштабування).
9. Community detection in social networks. Node Betweenness. Edge betweenness.
10. Виявлення спільноти в соціальних мережах.
11. Структура інтернету. Strongly Connected Components (SCC).
12. Introduction to web search.
13. PageRank. Мертві вузли (Dead nodes). Пастка павука (Spider traps).
14. Ідея показників degree centrality, betweenness centrality, eigenvector centrality.

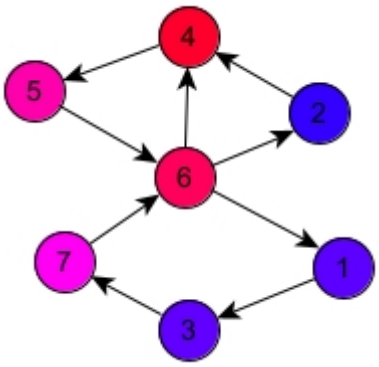
ХІД ПРОВЕДЕННЯ ЗАНЯТТЯ.

1. Проаналізуйте граф соціальної мережі (згідно з рис.34 відповідно до свого номеру у списку). В результаті необхідно вияснити чи є мертві вузли та пастки в графі мережі та визначитися з методами розрахунку показника PageRank.

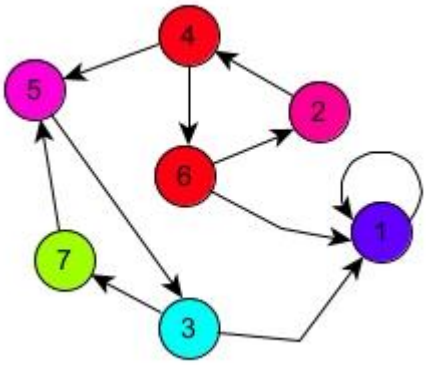
1



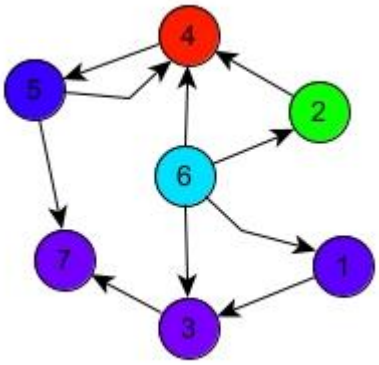
2



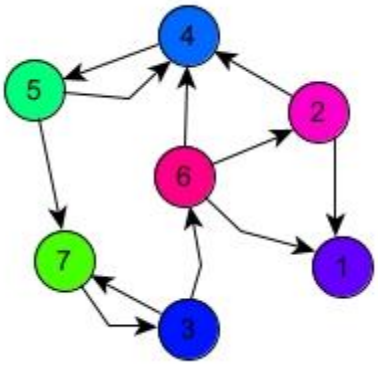
3


4
5
2
6
1
7
3

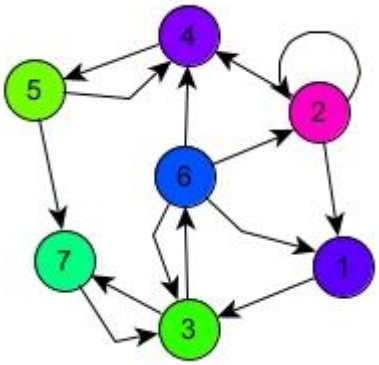
4


4
5
2
6
7
1
3

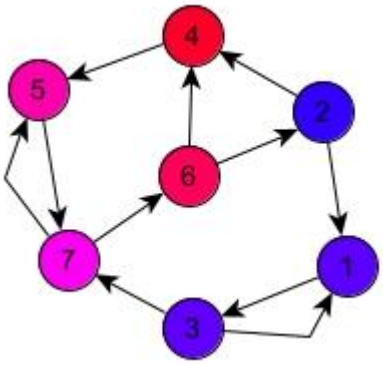
5


4
5
2
6
7
1
3

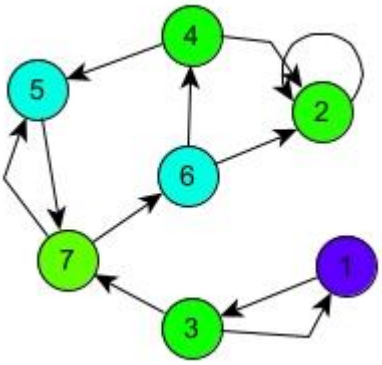
6


4
5
2
6
7
1
3

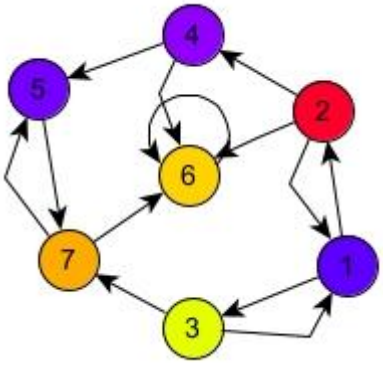
7


4
5
2
6
7
1
3

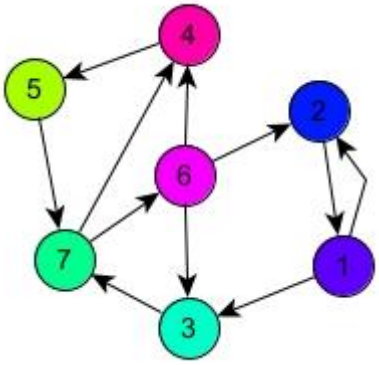
8

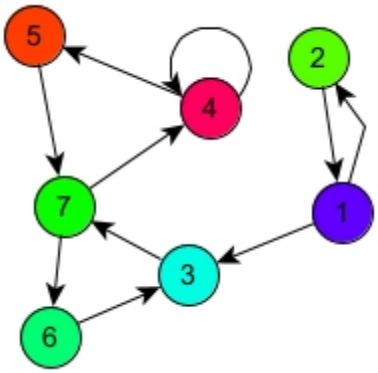

4
5
2
6
7
1
3

9


4
5
2
6
7
1
3

10


4
5
2
6
7
1
3

11 

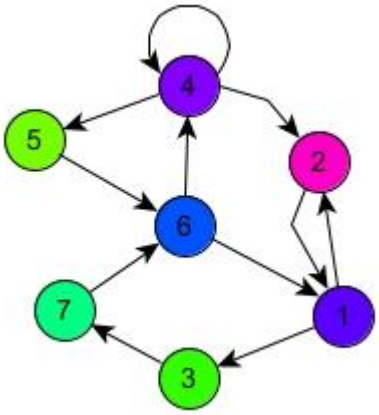
12 

Рисунок 34. Графи соціальних мереж для аналізу

2. Створіть матрицю суміжності (рис.3).

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]
[1,]	0	0	1	0	0	0	0
[2,]	1	0	0	0	0	0	0
[3,]	0	0	0	0	0	0	1
[4,]	0	0	0	0	1	0	0
[5,]	0	0	0	0	0	0	1
[6,]	0	1	1	1	0	0	0
[7,]	0	0	0	0	0	1	0

Рисунок 35. Приклад матриці суміжності

3. Створіть матрицю переходів.

4. Розрахуйте початковий вектор.

5. Розрахуйте значення показника PageRank для кожного вузла.

6. У разі наявності мертвих вузлів або пасток, використовуйте відповідні методи, розглянуті в лекційному матеріалі.

7. Оформіть розрахунки пп.3-5 у вигляді функцій та циклів для уніфікації та можливості моделювання різних матриць.

8. Розрахуйте показники degree centrality, betweenness centrality, eigenvector centrality з метою виявлення найвпливовіших користувачів, хабові вузли та "брокерів інформації" з точки зору аналізу інформаційних впливів.

ВИСНОВКИ. Коротко перерахуйте основні завдання, які були виконані під час практичної роботи. Не переписуйте весь хід роботи, а виділіть ключові етапи та досягнуті результати. Якщо під час роботи ви помітили певні закономірності, тенденції або особливості, обов'язково їх опишіть. Це демонструє ваше аналітичне мислення. Обґрунтуйте, де в майбутній професійній діяльності можуть знадобитися отримані навички та знання. Це допомагає усвідомити цінність виконаної роботи. Запропонуйте, як можна покращити методику практичної роботи або зробити її більш ефективною для навчання.

ДОМАШНЄ ЗАВДАННЯ. Підготувати звіт по роботі, ознайомитися з питаннями вхідного контролю, підготуватися до захисту роботи.

Під час виконання самостійної роботи та підготовки до захисту практичної роботи необхідно ознайомитися з базовими поняттями теорії графів та платформ з відкритим доступом, призначених для аналізу соціальних мереж. Розглянути можливість виконання практичної роботи з використанням такого класу програмного забезпечення або ресурсів.

III. Питання до модульних контрольних робіт та екзамену

Підсумкове оцінювання. Студенти складають письмовий екзамен, який складається з теоретичної та практичної частини. На екзамен виносяться питання, які дають можливість перевірити програмі результати навчання.

Студент не допускається до екзамену, якщо під час вивчення дисципліни він не виконав та не захистив всі практичні та лабораторні роботи. Максимальна оцінка за екзамен складає 40 балів. Оцінка за екзамен не може бути меншою **24 балів** для отримання загальної позитивної оцінки за курс.

Протягом семестру кожний студент виконує 2 модульні контрольні роботи (МКР) після проходження відповідних змістовних модулів. На МКР виносяться такі ж питання за змістом, як і на письмовий екзамен, з метою проміжної перевірки знань студентів та їх підготовки до складання екзамену.

Основні питання:

1. Що таке аналіз даних? Основні етапи data mining.
2. Прикладні галузі застосування data mining.
3. Технології data mining.
4. Програмне забезпечення та інструменти інтелектуального аналізу даних.
5. Моделі методології CRISP-DM.
6. Етапи попередньої обробки даних. Сукупність та вибірка, типи даних.
7. Очищення даних. Етапи очищення даних.
8. Трансформація даних.
9. Скорочення (масштабування) даних.
10. Окремі методи попередньої обробки даних. Відсутні значення (Missing Value).
11. Методи виявлення відсутніх даних. Ефективні стратегії обробки відсутніх значень.
12. Нормалізація даних. Бінгування в інтелектуальному аналізі даних (binning).
13. Що таке викид? Види викидів. Методи виявлення викидів: статистичні методи, методи на основі відстані та кластеризації.
14. Методи обробки викидів. Важливість виявлення викидів у машинному навчанні.
15. Кластерний аналіз. Властивості кластеризації.
16. k-means clustering. Що таке кластеризація K-середніх? Алгоритм K-середніх.
17. Hierarchical clustering. Ієрархічна кластеризація. Агломеративна кластеризація. Роздільна кластеризація.

18. Кластеризація на основі щільності. Density-Based Spatial Clustering Of Applications With Noise (DBSCAN). Параметри алгоритму DBSCAN. Реалізація в Python. Використання DBSCAN замість K-Means у кластерному аналізі.

19. Особливості використання DBSCAN замість K-Means у кластерному аналізі.

20. Призначення дерев класифікації. Визначення та основна ідея.

21. Термінології дерева рішень. Ідея методу дерев рішень.

22. Методи вибору атрибутів. Приріст інформації та ентропія.

23. Інформаційна ентропія. Приклад розрахунку.

24. Додаткові функції та характеристики індексу Джіні.

25. «Наївний» класифікатор Байєса. Умовна ймовірність. Види Байєсовських класифікаторів. Мережа Байєса.

26. Алгоритм K-Nearest Neighbors (KNN). Показники відстані, які використовуються в алгоритмі KNN. Робота алгоритму. Переваги та недоліки.

27. Лінійна регресійна модель. Нормальне рівняння

28. Показники продуктивності моделі.

29. Градієнтний спуск – GD. Пакетний градієнтний спуск. Міні Пакетний градієнтний спуск. Вплив швидкості навчання.

30. Стохастичний градієнтний спуск- SGD. Шляхи алгоритмів градієнтного спуску у просторі параметрів.

31. Поліноміальна регресія. Криві навчання.

32. Регуляризовані лінійні моделі. Гребенева регресія. Лассо-регресія. Еластична мережа.

33. Логістична регресія. Навчання та функція вартості.

34. Softmax regression.

35. Концепція правил асоціації. Показники правил. Алгоритми.

36. Додатки для реалізації апріорних алгоритмів. Види асоціативної класифікації.

37. Набори товарів, які купуються разом. Модель ринкового кошика. Приклади розрахунку показників.

38. A-priori алгоритм. Аналіз інформаційних впливів з використанням A-priori.

39. Алгоритм зростання частотного шаблону.

40. Online Analytical Processing (OLAP). Концепція OLAP. Побудова та типи OLAP.

41. Словник OLAP. Операції в OLAP.

42. Серверні архітектури в OLAP.

43. Data Warehouse vs DBMS. Приклади застосування сховищ даних.

44. Програмне забезпечення для керування метаданими.

45. Кроки, необхідні для створення будь-якого сховища даних.

46. Метадані та їх типи в сховищах даних.

47. Кроки створення моделі розмірних даних.

48. Архітектура сховища даних.

49. Важливість слова. Показники Inverse Document Frequency, IDF та term frequency. Аналіз інформаційних впливів за допомогою інтелектуального аналізу текстів.

50. Хеш-функції. Хеш-функції.

51. Індекси.

52. Що таке схожі сутності і як їх ідентифікувати?

53. Подібність документів.

54. Представлення документів у вигляді наборів символів.

55. Побудова shingle наборів на основі стоп слів.

56. Матричне представлення множин.

57. MinHash та сігнатурна матриці.

58. Матриця рейтингів (вподобань).

59. Системи Спільної фільтрації.

60. Системи рекомендацій на основі змісту.

61. Ефективність рекомендаційних систем.

62. Соціальна мережа як граф.

63. Графи з кількома типами вузлів.

64. Small-world Effect. Small world graph.

65. Кластеризація соціальних мереж.

66. Нерівномірний (перекошений) розподіл ступенів.

67. Моделювання зв’язків між об’єктами в соціальній мережі .

68. Random networks. Scale-free network.

69. Степеневий закон (закон масштабування).

70. Community detection in social networks. Node Betweenness. Edge betweenness.

71. Виявлення спільноти в соціальних мережах. Ідентифікація найвпливовіших користувачів, хабових вузлів та "брокерів інформації", які з'єднують різні спільноти.

72. Структура інтернету. Strongly Connected Components (SCC).

73. Introduction to web search.

74. PageRank. Мертві вузли (Dead nodes). Пастка павука (Spider traps).

75. Показники degree centrality, betweenness centrality, eigenvector centrality.

IV. Додатки

Додаток 1. Шаблон оформлення звіту з лабораторної роботи

МІНІСТЕРСТВО НАУКИ І ОСВІТИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ № ____ ЗА ТЕМОЮ:

_____

_____

Група _____

Курс _____

Студент (ка) _____

Дата оформлення _____

Перевірив _____

Дата _____

МЕТА РОБОТИ:

ТЕОРІЯ:

ХІД РОБОТИ:

ОТРИМАНІ РЕЗУЛЬТАТИ:

ВИСНОВКИ:

ВХІДНІ ДАНІ:

МІНІСТЕРСТВО НАУКИ І ОСВІТИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗВІТ З ПРАКТИЧНОЇ РОБОТИ № ____ ЗА ТЕМОЮ:

_____

_____

Група _____

Курс _____

Студент (ка) _____

Дата оформлення _____

Перевірив _____

Дата _____

МЕТА:

ТЕОРІЯ:

ПЛАН ПРОВЕДЕННЯ ПРАКТИЧНОГО ЗАНЯТТЯ:

ПИТАННЯ ДЛЯ ВХІДНОГО КОНТРОЛЮ СТУДЕНТІВ:

ХІД ПРОВЕДЕННЯ ПРАКТИЧНОГО ЗАНЯТТЯ:

ВИСНОВКИ

ДОМАШНЄ ЗАВДАННЯ:

V. Література

[1] “Лабораторна робота та її аналіз,” *Освіта.UA*, May 15, 2008. https://osvita.ua/school/method/technol/724/ (accessed Aug. 08, 2025).

[2] Н.О. Безсонова, І.С. Латунов,О.О. Герасимова. Методичні рекомендації «Організація та проведення лабораторних, практичних та семінарських занять». МР А.2.2-36-046. Національний фармацевтичний університет. Харків, 2025ю – 39 с.

[3] Теорія і практика впровадження інноваційних технологій навчання у професійну підготовку кваліфікованих робітників: монографія / [Лузан П. Г., Манько В. М., Нестерова Л. В, Романова Г. М.]; за заг. ред. Г. М. Романової. – К. : ТОВ «НВП Поліграфсервіс», 2014. – 216 с.

[4] Hodge, V. and Austin, J. (2004). A Survey of Outlier Detection Methodologies. *Artificial Intelligence Review*, [online] 22(2), pp.85–126. doi:https://doi.org/10.1023/b:aire.0000045502.10941.a9.

[5] Grubbs, F.E. (1950). Sample Criteria for Testing Outlying Observations. *The Annals of Mathematical Statistics*, 21(1), pp.27–58. doi:https://doi.org/10.1214/aoms/1177729885..

[6] T. M. COVER, and P. E. HART, “Nearest Neighbor Pattern Classification,” IEEE TRANSACTIONS, 1967. Accessed: Dec. 27, 2025. [Online]. Available: https://isl.stanford.edu/~cover/papers/transIT/0021cove.pdf.

[7] B. Artley, “Unsupervised Learning: K-Means Clustering | Towards Data Science,” *Towards Data Science*, Jun. 27, 2022. https://towardsdatascience.com/unsupervised-learning-k-means-clustering-27416b95af27/

[8] E. Schubert, “Stop using the elbow criterion for k-means and how to choose the number of clusters instead,” vol. 25, no. 1, pp. 36–42, Jun. 2023, doi: https://doi.org/10.1145/3606274.3606278.

[9] A. Christopher, “Hierarchical Clustering and Density-Based Spatial Clustering of Applications with Noise (DBSCAN),” *Medium*, Feb. 04, 2021. https://antoblog.medium.com/hierarchical-clustering-and-density-based-spatial-clustering-of-applications-with-noise-dbscan-b8d903095532

[10] Roustaei, N. (2024). Application and interpretation of linear-regression analysis. *Medical Hypothesis Discovery & Innovation in Ophthalmology*, [online] 13(3), pp.151–159. Available at: https://pmc.ncbi.nlm.nih.gov/articles/PMC11537238/.

[11] Google, “Linear regression,” *Google for Developers*, 2024. https://developers.google.com/machine-learning/crash-course/linear-regression

[12] G. Surraco, “Fast Algorithms for Mining Association Rules - Agrawal - Srikant - APRIORI.pdf,” *Academia.edu*, Jul. 13, 2015. https://www.academia.edu/14010751/Fast_Algorithms_for_Mining_Association_Rules_Agrawal_Srikant_APRIORI_pdf (accessed Dec. 27, 2025).

[13] R. Agrawal, T. Imieliński, and A. Swami, “Mining association rules between sets of items in large databases,” *Proceedings of the 1993 ACM SIGMOD international conference on Management of data - SIGMOD ’93*, 1993, doi: https://doi.org/10.1145/170035.170072.

[14] C. Borgelt, “Frequent item set mining,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 2, no. 6, pp. 437–456, Oct. 2012, doi: https://doi.org/10.1002/widm.1074.

[15] C. Borgelt and R. Kruse, “Induction of Association Rules: Apriori Implementation,” pp. 395–400, Jan. 2002, doi: https://doi.org/10.1007/978-3-642-57489-4_59.

[16] C. Borgelt, “Efficient implementations of apriori and eclat,” *Academia.edu*, 2003. https://www.academia.edu/77973047/Efficient_implementations_of_apriori_and_eclat

[17] Weine, E., Mary Sara McPeek and Abney, M. (2023). Application of Equal Local Levels to Improve Q-Q Plot Testing Bands with *R* Package **qqconf**. *Journal of Statistical Software*, [online] 106(10). doi:https://doi.org/10.18637/jss.v106.i10.

[18] “Практичне заняття та його аналіз,” *Освіта.UA*, Apr. 08, 2008. https://osvita.ua/school/method/technol/725/ (accessed Aug. 08, 2025).

[19] Breiman, Leo. “Random Forests.” *Machine Learning*, vol. 45, no. 1, Oct. 2001, pp. 5–32.

[20] Neri Van Otten and Neri Van Otten, “Decision Trees In ML Complete Guide [How To Tutorial, Examples, 5 Types & Alternatives],” *Spot Intelligence*, May 22, 2024. https://spotintelligence.com/2024/05/22/decision-trees-in-ml/

[21] Viktor Kauk, “Навчання та дослідження,” *YouTube*, Jan. 31, 2025. https://www.youtube.com/watch?v=3LnkeKWD7hs (accessed Aug. 08, 2025).

[22] М.В. МОГИЛЬНА and В.І. ДУБРОВІН, “ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ТЕКСТУ: ЗАСТОСУВАННЯ ТА БЕЗКОШТОВНІ ПРОГРАМНІ ЗАСОБИ,” *Prikladnì pitannâ matematičnogo modelûvannâ*, vol. 5, no. 2, pp. 41–49, Jun. 2023, doi: https://doi.org/10.32782/mathematical-modelling/2022-5-2-5.

[23] S. Shtovba, “Jaccard index-Based Assessing the Similarity of Research Fields in Dimensions,” 2019. Available: https://ceur-ws.org/Vol-2533/paper11.pdf

[24] R. Zafarani, M. Ali Abbasi , and H. Liu, “Social Media Mining,” *Google Books*, 2025. https://books.google.com.ua/books?id=fVhzAwAAQBAJ&redir_esc=y (accessed Dec. 27, 2025).

[25] W. Campbell, C. Dagli, and C. Weinstein, “ Social Network Analysis with Content and Graphs,” *62 LINCOLN LABORATORY JOURNAL, VOLUME*, vol. 20, 2013, Available: https://pzs.dstu.dp.ua/DataMining/social/bibl/20_1_5_Campbell.pdf

[26] Веб-сканування: Вичерпний посібник, “Веб-сканування: Вичерпний посібник,” *Ranktracker.com*, Jul. 09, 2024. https://www.ranktracker.com/uk/blog/web-crawling-a-comprehensive-guide/